IQ™4 Configuration Manual

Applies to version 4.34 firmware

TREND

Author: Trend Technical Publications

Issue: 18

Date: 15-Dec-2023

Part Number: TE201263

Copyright: © 2023 Honeywell Products and Solutions SARL, Connected Building Division. All rights reserved.

This manual contains proprietary information that is protected by copyright. No part of this manual may be reproduced, transcribed, stored in a retrieval system, translated into any language or computer language, or transmitted in any form whatsoever without the prior consent of the publisher.

Manufactured for and on behalf of the Connected Building Division of Honeywell Products and Solutions SARL, La Pièce, 16, 1180 Rolle, Switzerland by its Authorized Representative, Trend Control Systems Limited. For information contact:

Trend Control Systems Limited
St. Marks Court
North Street
Horsham
West Sussex
RH12 1BW

NOTICE: Trend Control Systems Limited makes no representations or warranties of any kind whatsoever with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Trend Control Systems Limited shall not be liable for any errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material. Trend Control Systems Limited reserves the right to revise this publication from time to time and make changes in the content hereof without obligation to notify any person of such revisions or changes.

Please send any comments on this or any other Trend technical publication to techpubs@trendcontrols.com

TABLE OF CONTENTS

ABOUT THIS MANUAL 17
Conventions Used in this Manual 17
Contacting Trend 17

PART 1 19

1 WHAT IS IQ4? 21
1.1 IQ411 and IQ412 (IQ41x Series Controllers) 21
1.2 IQ422 21
1.3 IQ4NC 21
1.4 IQ4E 22
1.5 XNC Variants 22
1.6 INT Variants 22
1.7 IQ4/IO Expansion Modules 22

2 FIRMWARE 23
2.1 Strategy Modules 23
2.1.1 Module Limits 23
2.1.2 Module Parameters 23
2.2 Firmware Upgrade 24

3 STRATEGY 25
3.1 Adding Modules 25
3.2 Connecting Modules 25
3.2.1 Connection to I/O Channels 26
3.3 Configuring Modules 26
3.4 Edit Module Parameters 26
3.5 Module Servicing 26
3.5.1 Sequenced Servicing 27
3.5.2 Event Driven Servicing 28
3.5.3 Maximum Number of Modules Serviced 29
3.5.4 Disabling Modules 29
3.5.5 Multiplexing 30
3.6 Module Overrides 31
3.7 Display and Directory Modules 31
3.7.1 Web Client Access 31
3.7.2 Supervisor/Tool Access 32
3.7.3 User Dependent Displays 33
3.8 Saving Critical Data 34
3.9 IQ4 Files 34
3.9.1 Strategy File (.IQ4) 34
3.9.2 Backdrop Files 34
3.9.3 XNC File (/XNC variants only) 34
3.9.4 Writing Changes to Flash 34
3.10 Restore to Factory Defaults 35

4 HARDWARE 37

5 TREND COMMUNICATIONS 39
5.1 Ethernet Communications 40
5.1.1 Building LANs on Ethernet 40
5.1.2 Building Internetworks on Ethernet 40
5.1.3 Stand Alone IQ4 42
5.1.4 Connection to an IQ4's vCNC 43
5.1.5 Connection by a Web Client 48
5.1.6 Connection of IQVIEW SCD 48
5.1.7 Email Alarms from IQ4 48
5.1.8 IP Alarms from IQ4 49
5.1.9 Upload/Download of Configuration Files 49

5.1.10 Communication with Modbus IP Devices 49
5.1.11 Using Routers to Provide Multiple Sites 49
5.1.12 IP Addressing 50
5.1.13 UDP 51
5.2 RS232 Communications 52
5.2.1 Connection of Displays/Supervisors/Tools for Communication Only with the IQ4 52
5.2.2 Network connection of Displays/Supervisors/Tools 52
5.2.3 Upload/Download of Strategy Files 53
5.2.4 Communication with Modbus Serial Devices 53
5.2.5 Communication with M-bus Devices 53
5.3 USB Communications 53
5.3.1 Network Connection of IQSET 54
5.3.2 Upload/Download of Strategy Files 54
5.4 Wallbus Communications 54
5.4.1 Connection of Wallbus Devices for Communication Only with the IQ4 55
5.5 RS485 Communications 55
5.6 Trend Current Loop Communications 55
5.6.1 Building LANs on a Trend Current Loop Network 56
5.6.2 Upload/Download of Configuration Files 57
5.6.3 Building an Internetwork on a Trend Current Loop Network 57
5.7 MS/TP Trunk Communications 58
5.7.1 Building LANs on MS/TP Trunk 58
5.7.2 Connection of MS/TP LAN to Internetwork 59
5.8 IQ4NC Communication Modes 59
5.8.1 Ethernet to MS/TP Mode 59
5.8.2 Trend Internetwork to MS/TP Mode 60
5.8.3 Ethernet to Trend LAN Mode 60
5.8.4 Ethernet to Trend Internetwork Mode 61
5.8.5 Manual Communication Mode 61

6 BACNET COMMUNICATIONS 63
6.1 Compliance with BACnet Testing Laboratory Requirements 63
6.2 BACnet IP Communications 63
6.2.1 BACnet COV Reporting and Receiving 64
6.2.2 BACnet Alarms 65
6.2.3 Trend Log Object 66
6.2.4 BACnet Schedule Object 66
6.2.5 Analogue and Binary BACnet Objects 66
6.2.6 Backup and Restore 66
6.2.7 BACnet User Alarm Acknowledgments 67
6.2.8 Writes over BACnet 68
6.2.9 Create/Delete Service 68
6.2.10 BACnet with Automatic IP Addressing 68
6.2.11 BACnet Properties Supported 69
6.3 BACnet over MS/TP 69

7 ALARMS 71
7.1 Alarm Types 71
7.1.1 Network Alarms 71
7.1.2 General Alarms 76
7.1.3 Item Alarms 77
7.2 Alarm Handling Strategy 79
7.2.1 Delivery Failure 80
7.2.2 IQ Lan Alarm Destinations 80
7.2.3 IP Address Alarm Destinations 81
7.2.4 Email Alarm Destinations 81
7.2.5 BACnet Alarm Destinations (/BACnet variants and IQ4NC only) 81
7.2.6 Format of Alarms Sent Using the Alarm Handling Strategy 82
7.2.7 Alarm Format Validation (IQ4NC only) 85
7.2.8 Alarm Handling Strategy Examples 86
7.2.9 Autodialled Destination with Routine Transmission of Low Priority Alarms 90

7.3 INC Alarm Forwarding (IQ4NC only) 91
7.3.1 Alarm Forwarding Examples 92

8 WEB SERVER 95
8.1 Functionality 95
8.1.1 Graphical Display Pages 96
8.2 Clients 96
8.3 Web Server Security 97

9 SECURITY 99
9.1 Securing IQ4 99
9.1.1 Security Checklist 99
9.1.2 Developing a Security Program 99
9.1.3 Disaster Recovery Planning 99
9.1.4 Physical and Environmental Consideration 99
9.1.5 Security Updates and Service Packs 99
9.1.6 Virus Protection 99
9.1.7 Network Planning and Security 100
9.1.8 Virtual Environments 100
9.1.9 Securing Wireless Devices 101
9.1.10 System Monitoring 101
9.1.11 Windows Domains 101
9.1.12 Securing Access to the Operating System 101
9.1.13 Security Features 101
9.2 IQ4 Security System 101
9.2.1 Changes Using Text Communications 102
9.2.2 Strategy Upload and Download 102
9.2.3 Access to Web Pages 102
9.2.4 Access to XML Web Services 104
9.2.5 Network Use (v3.30 or greater) 104
9.2.6 Connection to vCNCs (v3.60 or greater) 104

10 XML WEB SERVICES 105

11 XNC FACILITY 107
11.1 Strategy 107
11.2 Type 1 (XNC) Interface Module 107
11.3 Trend Custom Language 108
11.3.1 TCL Application Configuration 108
11.4 Compatibility with IQ3 TCL 108

12 IQ INTERFACING 109
12.1 MODBUS 109
12.2 M-bus 109
12.3 Configuring IQ Interfacing 110

PART 2 111

1 INSTALLING AN IQ4 113
1.1 Installation Procedure 113
1.1.1 Mount 113
1.1.2 Connect Power 113
1.1.3 Connect Ethernet 113
1.1.4 Connect Inputs 113
1.1.5 Connect Outputs 113
1.1.6 Connect Auxiliary Supply 114
1.1.7 Connect Trend Current Loop Network (IQ4NC and /LAN only) 114
1.1.8 Connect MS/TP Trunk (IQ4NC only) 114
1.1.9 Connect Wallbus (Not IQ422/00/XNC/.... or IQ4NC/00/... variants) 114
1.1.10 Connect to XNC 3rd Party System (/XNC only) 114
1.1.11 Install I/O Modules (IQ4E only) 114

1.1.12 Connect Serial MODBUS (/INT only) 115
1.1.13 Connect M-Bus (/INT only) 115
1.2 Cutting the Earth Link Track (/230 variants only) 115
1.3 Calculate the Power Used by the Combined Supply 116

2 CREATING A STRATEGY 117
2.1 Plan the System 117
2.1.1 Creating a BTL Compliant Strategy 117
2.2 Create a IQSET Project and Add Controllers 118
2.3 Configure the Strategy 118
2.3.1 Set up the Address Module 118
2.3.2 Set up the Type 1 (IQ LAN) Network Module (/LAN variants and IQ4NC only) 119
2.3.3 Set up the Type 4 (Ethernet IP) Network Module 119
2.3.4 Set up the Type 12 (BACnet MSTP) Network Module (IQ4NC only) 120
2.3.5 Set up the Type 5 (BACnet IP) Network Module (/BAC variants and IQ4NC only) 120
2.3.6 Set up the Type 13 (BACnet Application) Network Module (/BAC variants and IQ4NC only) 120
2.3.7 Set up the Sensor Type Modules 121
2.3.8 Add States Category Modules 121
2.3.9 Write TCL Application (/XNC variants only) 121
2.3.10 Set up the Type 17 (XNC Serial) Network Module (/XNC variants only) 121
2.3.11 Set up the Type 1 (XNC) Interface Module (/XNC variants only) 121
2.3.12 Add and Link Modules 121
2.3.13 Set up the Alarm Handling Strategy 121
2.3.14 Add I/O Modules (IQ4E only) 121
2.3.15 Assign I/O Channels 122
2.3.16 Set up Strategy for Use with Wallbus Devices (v3.30 or greater) 122
2.3.17 Edit Module Parameters 122
2.3.18 Set up the Sequence Table 122
2.3.19 Set up Event Driven Strategy 122
2.3.20 Set up Daily Profile 122
2.3.21 Set up Calendar Modules (v3.50 or greater) 123
2.3.22 Set up Exceptions 123
2.3.23 Set up IC Communications 123
2.3.24 Set up Security 123
2.3.25 Set up CNC User Modules (v3.60 or greater) 123
2.3.26 Set up Virtual CNC Modules (v3.60 or greater) 124
2.3.27 Configure Web Servers 124
2.3.28 Set up Display and Directory Modules 124
2.4 Test the Strategy 124
2.5 Create the Strategy File 125

3 INSTALLING I/O MODULES 127
3.1 IQ4/IO Module Installation Procedure 127
3.1.1 Mount 127
3.1.2 Connect I/O Bus 127
3.1.3 Connect Power 127
3.1.4 Terminate I/O Bus 127
3.1.5 Connect Inputs 128
3.1.6 Connect Outputs 128
3.2 XCITE I/O Module Installation Procedure 128
3.2.1 Mount 128
3.2.2 Connect I/O Bus 128
3.2.3 Connect Power 128
3.2.4 Terminate I/O Bus 129
3.2.5 Connect Inputs 129
3.2.6 Connect Outputs 129
3.3 Calculate the Amount of Power the IQ4 can Supply to the I/O Bus 129
3.4 Calculating I/O Module Power Requirements 129
3.5 Using a Additional 24 Vdc Supply 129
3.6 Mixing IQ4/IO and XCITE I/O Modules 129
3.7 Replacing an XCITE I/O Module with an IQ4/IO Module 130

4 CONFIGURING AN IQ4 131
4.1 Prepare Ethernet Network 131
4.1.1 Configure Firewalls 131
4.2 Configure I/O Modules (IQ4E only) 132
4.3 Connect PC/Laptop 132
4.4 Check Watchdog/Power Indicators 132
4.5 Check Ethernet Indicators 132
4.6 Connect to IQ4 Using IP Tool 132
4.7 Set up IP Addressing Details 132
4.8 Set up Virtual CNCs (if required) 133
4.8.1 Adding the vCNC modules and CNC User Modules to the Strategy (v3.60 or greater) 133
4.8.2 From IQSET's System View (v3.60 or greater) 133
4.8.3 Using IP Tool to Enable vCNCs and Connecting to the vCNC Using IQSET 133
4.9 Set up IQ4NC Communication Mode (IQ4NC only) 134
4.10 Check Trend Current Loop Network Indicators (/LAN or IQ4NC only) 135
4.11 Check MS/TP Trunk Indicators (IQ4NC only) 135
4.12 Set up Ethernet Access Across Routers 135
4.13 Set up Trend Addressing 135
4.14 Set up Current Loop Baud Rate (/LAN or IQ4NC only) 135
4.15 Set up RS232/USB Port Addressing 135
4.16 Check LAN OK Indicator (/LAN only) 136
4.17 Download Strategy 136
4.17.1 Parameter Download Exclusions 136
4.18 Check Input & Output Operation 136
4.19 Connect Supervisors/Tools/Display Panels (if required) 136
4.20 Configure SSL Certificates (if required) 136
4.20.1 Create the SSL Certificates 137
4.20.2 Upload the Controller Certificate to the IQ4 155
4.20.3 Test Access 157
4.21 Backup the Strategy 157

5 CONFIGURE I/O MODULES 159
5.1 Select Universal Input Channel Type (XCITE/UI type modules only) 159
5.2 Cut Screen Earth (Ground) Link(s) (XCITE/UI type modules only) 159
5.3 Set I/O Module Address (XCITE I/O modules only) 159
5.4 Check the Power and Watchdog Indicators 159
5.5 Check Supply Voltage (IQ4/IO modules only) 159
5.6 Check the I/O Bus Indicator 159
5.6.1 IQ4/IO Modules 159
5.6.2 XCITE/IO Modules 160
5.7 Set Physical I/O Module Addresses (if required) 160
5.7.1 IQ4/IO Modules 160
5.7.2 XCITE/IO Modules 161
5.8 Resolve Physical I/O Module Duplicate Addresses (if required) 161
5.8.1 IQ4/IO Modules 161
5.8.2 XCITE/IO Modules 161
5.9 Check Physical I/O Module Addressing 161
5.9.1 IQ4/IO Modules 161
5.9.2 XCITE/IO Modules 162
5.10 Check Strategy I/O Modules are Set Correctly 162

PART 3 163

1 IQ4 MODULES 165
1.1 Common IQ4 Module Parameters 165

2 ADDRESS MODULE 167
2.1 Functionality 167
2.2 Address Module Parameters 167
2.3 Address Module Connectives 172

3 ALARM DESTINATION MODULES 173
3.1 Functionality 173
3.1.1 Delivery failure 173
3.2 Alarm Destination Module Types 173
3.2.1 Type 1 (IQ Lan) 174
3.2.2 Type 2 (IP Address) 174
3.2.3 Type 3 (Email) 175
3.2.4 Type 4 (BACnet) 176
3.3 Alarm Destination Module Common Parameters 177
3.4 Alarm Destination Module Connectives 178

4 ALARM GROUP MODULES 179
4.1 Functionality 179
4.2 Alarm Group Module Parameters 179
4.3 Alarm Group Module Connectives 179

5 ALARM LOG MODULES 181
5.1 Functionality 181
5.1.1 Add/Removing Alarms 181
5.1.2 High Water Mark 181
5.1.3 Alarm Log Format 182
5.2 Alarm Log Module Parameters 182
5.3 Alarm Log Module Connectives 183

6 ALARM ROUTE MODULES 185
6.1 Functionality 185
6.2 Alarm Route Module Parameters 185
6.3 Alarm Route Module Connectives 185

7 ANALOGUE NODE MODULES 187
7.1 Functionality 187
7.2 Analogue Node Module Parameters 187
7.3 Analogue Node Module Connectives 188

8 CALENDAR MODULES 189
8.1 Functionality 189
8.1.1 Calendar Entries 189
8.1.2 Link to Time Schedule Modules 191
8.1.3 Link to Time Schedule Module Exceptions 191
8.2 Calendar Module Parameters 191
8.2.1 Module Properties 191
8.2.2 Calendar Entry Sub-Module Parameters 192

9 DIGITAL BYTE MODULES 193
9.1 Functionality 193
9.2 Digital Byte Module Parameters 193
9.3 Digital Byte Module Connectives 193

10 DIGITAL INPUT MODULES 195
10.1 Functionality 195
10.1.1 Number of Starts, Hours Run and State 195
10.1.2 Alarms 195
10.1.3 Module Override 196
10.1.4 Use with Type 6 (Hours Run) Logic Module 196
10.1.5 Module Disable 197
10.1.6 COV Operation 197
10.2 Digital Input Module Types 197
10.2.1 Type 1 (Internal) 197
10.2.2 Type 2 (External) 198
10.3 Digital Input Module Common Parameters 199
10.4 Digital Input Module Connectives 200

11 DIRECTORY MODULES 201
11.1 Functionality 201
11.2 Directory Module Parameters 201
11.3 Directory Module Connectives 202

12 DISPLAY MODULES 203
12.1 Functionality 203
12.2 Display Module Types 203
12.2.1 Static Text 203
12.2.2 Static Text Display Module Parameters 203
12.2.3 Dynamic Text 204
12.2.4 Dynamic Text Display Module Parameters 204
12.3 Display Module Common Parameters 205
12.4 Display Module Connectives 206

13 DRIVER MODULES 207
13.1 Functionality 207
13.1.1 Alarms 207
13.1.2 Module Disable 209
13.1.3 Module Override 210
13.1.4 Hours Run 210
13.1.5 Driver Invert 210
13.1.6 Hand Off Auto (HOA) Operation 211
13.1.7 COV Operation 212
13.2 Driver Module Types 212
13.2.1 Type 1 (Digital) 213
13.2.2 Type 2 (Analogue) 214
13.2.3 Type 3 (Time Proportional) 216
13.2.4 Type 4 (Raise Lower with End Stop) 220
13.2.5 Type 5 (Binary Hysteresis) 224
13.2.6 Type 6 (Time Proportional + OR) 226
13.2.7 Type 7 (Raise Lower Continuous) 228
13.2.8 Type 8 (Multistage Digital) 230
13.3 Driver Module Common Parameters 234
13.4 Driver Module Connectives 240

14 FUNCTION MODULES 241
14.1 Functionality 241
14.1.1 Module Disable 241
14.2 Function Module Alarms 241
14.3 Function Module Types 242
14.3.1 Type 1 (Filter) 243
14.3.2 Type 2 (Rescale From) 244
14.3.3 Type 3 (Rescale To) 245
14.3.4 Type 4 (Limit At) 247
14.3.5 Type 5 (Limit To) 248
14.3.6 Type 6 (Log) 249
14.3.7 Type 7 (Square Root) 249
14.3.8 Type 8 (Adder/Scaler) 250
14.3.9 Type 9 (Multiplier) 251
14.3.10 Type 10 (Divider) 252
14.3.11 Type 11 (Minimum) 252
14.3.12 Type 12 (Maximum) 253
14.3.13 Type 13 (Average) 254
14.3.14 Type 14 (Analogue Gate) 254
14.3.15 Type 15 (Comparator) 255
14.3.16 Type 16 (Enthalpy) 255
14.3.17 Type 17 (Hysteresis Band) 256
14.3.18 Type 18 (Analogue to Digital Converter) 257
14.3.19 Type 19 (Lookup) 261
14.3.20 Type 20 (Adder) 261

14.3.21 Type 21 (Hysteresis Limits) 262
14.3.22 Type 22 (Proximity) 263
14.3.23 Type 23 (Power) 263
14.3.24 Type 24 (Absolute) 264
14.3.25 Type 25 (Truncate) 265
14.3.26 Type 26 (Trigonometric) 265
14.3.27 Type 27 (Heat Meter) 266
14.3.28 Type 28 (Integrator) 267
14.3.29 Type 29 (Change of Value) 270
14.3.30 Type 30 (Sample and Hold) 270
14.3.31 Type 31 (Psychrometrics) 271
14.3.32 Type 32 (Sliding Window) 273
14.3.33 Type 33 (Differencer) 274
14.3.34 Type 34 (Analogue Input Selector) 275
14.4 Function Module Common Parameters 277
14.5 Function Module Connectives 278

15 IC COMMS MODULES 279
15.1 Functionality 279
15.1.1 Receiving IC comms from Other Controllers 279
15.1.2 Initiating IC comms to Other Devices 280
15.1.3 Module Disable 280
15.1.4 Security on IC comms Writes 280
15.1.5 IC Comms/COV Subscription Limitations 281
15.1.6 Retransmission Interval and Significant Change 281
15.1.7 COV Service 282
15.1.8 Retries and Failed Bit 282
15.1.9 Generating a Digital Byte IC Comms 282
15.1.10 IC Comms Writes 283
15.1.11 Power On Delay 283
15.2 BACnet IC Comms 284
15.3 Controller Selection 284
15.3.1 Absolute Controller Selection 284
15.3.2 Label Matching Controller Selection 284
15.4 Item Selection 286
15.4.1 Label Matching Item Selection 286
15.4.2 Absolute Item Selection 287
15.5 IC Comms Module Types 288
15.5.1 Type 1 (To) 288
15.5.2 Type 2 (Global To) 292
15.5.3 Type 3 (Min) 296
15.5.4 Type 4 (Max) 296
15.5.5 Type 5 (Sum) 298
15.5.6 Type 6 (Average) 299
15.5.7 Type 7 (From Analogue) 301
15.5.8 Type 8 (From Byte) 302
15.5.9 Type 9 (From Bit) 304
15.6 IC Comms Modules Common Parameters 305
15.7 Differences Between IQ4 and IQ2 IC Comms 307
15.8 IC Comms Module Connectives 307

16 INTERFACE MODULES 309
16.1 Functionality 309
16.2 Interface Module Types 309
16.2.1 Type 1 (XNC) 309
16.2.2 Type 2 (Wallbus) 312
16.2.3 Type 3 (Modbus) 317
16.2.4 Type 4 (M-bus) 326
16.3 Interface Module Common Parameters 328
16.4 Interface Module Connectives 328

17 I/O MODULES 329
17.1 I/O Module Types 329
17.1.1 Type 0 (Trend I/O) 329
17.2 I/O Module Connectives 332

18 KNOB MODULES 333
18.1 Functionality 333
18.1.1 Reset Operation 333
18.1.2 COV Operation 333
18.1.3 Knob Adjustment Mode 333
18.1.4 Security 333
18.1.5 Range 333
18.1.6 Knob Sequencing 334
18.2 Knob Module Parameters 334
18.3 Knob Module Connectives 335

19 LOGIC MODULES 337
19.1 Functionality 337
19.2 Module Disable 337
19.3 Logic Module Types 337
19.3.1 Type 1 (Combination) 338
19.3.2 Type 2 (Timer) 341
19.3.3 Type 3 (Readback) 342
19.3.4 Type 4 (Counter) 345
19.3.5 Type 5 (Delay) 347
19.3.6 Type 6 (Hours Run) 348
19.3.7 Type 7 (Fan Out) 350
19.3.8 Type 8 (D to A) 350
19.3.9 Type 9 (Change of State) 352
19.3.10 Type 10 (Flip Flop) 354
19.3.11 Type 11 (Latch) 355
19.3.12 Type 12 (Digital Input Selector) 356
19.4 Logic Module Common Parameters 357
19.5 Logic Module Connectives 358

20 LOOP MODULES 359
20.1 Functionality 359
20.1.1 Loop Module Schematic 360
20.1.2 Closed Loop Control 361
20.1.3 Setpoint Selection 368
20.1.4 Module Disable 368
20.1.5 Manual Override 368
20.1.6 Deadband 368
20.1.7 Alarms 369
20.2 Loop Tuning 371
20.2.1 The Stability Limit Method 371
20.2.2 The Step Method 372
20.2.3 Tuning an Integral Only Loop 372
20.3 Loop Examples 373
20.3.1 On/Off Pump Control 374
20.3.2 Time Proportional Control of Pump 374
20.3.3 Weather Compensated Control 375
20.3.4 Cascade Control 377
20.3.5 Flow Temperature Cascade Control 378
20.3.6 Feed Forward Control 379
20.3.7 Cascade Control with Outside Air Temperature Compensation 379
20.3.8 Heating and Cooling 381
20.4 Loop Module Parameters 383
20.5 Loop Module Connectives 386

21 NON-TREND DEVICE MODULES 387

21.1 Functionality 387
21.2 NTD Module Parameters 387
21.3 NTD Module Connectives 388

22 NETWORK MODULES 389
22.1 Functionality 389
22.1.1 IQ4NC Communication Mode Selection 389
22.2 Network Module Types 390
22.2.1 Type 1 (IQ LAN) 390
22.2.2 Type 4 (Ethernet IP) 392
22.2.3 Type 5 (BACnet IP) 397
22.2.4 Type 8 (Modbus Serial) 399
22.2.5 Type 12 (BACnet MSTP) 400
22.2.6 Type 13 (BACnet Application) 401
22.2.7 Type 14 (Wallbus) 405
22.2.8 Type 15 (M-bus) 406
22.2.9 Type 16 (Trend Client) 407
22.2.10 Type 17 (XNC Serial) 408
22.2.11 Type 19 (Modbus IP) 409
22.2.12 Network Module Common Parameters 409

23 OPTIMUM START STOP MODULES 411
23.1 Functionality 411
23.1.1 Physical Algorithm 411
23.1.2 Statistical Algorithm 413
23.1.3 Target Temperatures 415
23.1.4 Adapted Data 416
23.1.5 Heating and Cooling Loops in the Zone 419
23.1.6 OSS State Outputs 419
23.1.7 Time Errors 420
23.1.8 OSS Should be in Control 421
23.1.9 Connection to Time Schedule Module 422
23.2 Differences Between IQ4 and IQ2 OSS Modules 422
23.3 OSS Module Parameters 423
23.4 Optimum Start Stop Connectives 426
23.5 OSS Examples 426
23.5.1 Example 1 Zone 1 426
23.5.2 Example 2 Zone 2 (heat/cool) 427
23.5.3 Example 3 Electric Heating 428

24 OPTIONS MODULE 429
24.1 Functionality 429
24.2 Options Module Parameters 429

25 PERFORMANCE MODULE 431
25.1 Functionality 431
25.1.1 IQSET Device/Diagnostic Plots Feature 431
25.2 Performance Module Parameters 431
25.3 Performance Module Connectives 432

26 PAGE MODULES 433
26.1 Functionality 433
26.2 Page Module Parameters 433

27 PLOT MODULES 435
27.1 Functionality 435
27.1.1 BACnet Compliance 435
27.1.2 Alarms 435
27.1.3 Stop When Full 435
27.1.4 Threshold Notification 436
27.1.5 Module Disable 436

27.1.6 Double Precision (v3.30 or greater) 436
27.2 Plot Module Types 436
27.2.1 Type 1 (Synchronised) 436
27.2.2 Type 2 (Triggered) 438
27.2.3 Type 3 (Periodic) 440
27.2.4 Type 4 (COV) 441
27.2.5 Double Precision 441
27.3 Plot Module Common Parameters 442
27.4 Access to Plots 443
27.4.1 New Compact Protocols 443
27.4.2 New Text Comms Protocols 444
27.4.3 IQ4 Graph Webpages 444
27.4.4 Graphs 444
27.5 Plot Memory 444
27.6 Plot Module Limits 445
27.7 Effect of Time Synchronisation 445
27.7.1 Effect of Time Synchronisation on Type 1 Plot Modules 445
27.7.2 Effect of Time Synchronisation on Type 2, Type 3 and Type 4 Plot Modules 445
27.8 Effect of Daylight Saving 445
27.9 Plot Records 446
27.9.1 Plot Record Parameters 446
27.10 Plot Module Connectives 447

28 PROGRAM MODULE 449
28.1 Functionality 449
28.1.1 User Module Merging (v3.60 or greater) 449
28.2 Program Module Parameters 449

29 SCHEDULE OFFSET MODULES (SCHEDULE MODULES) 451
29.1 Functionality 451
29.1.1 Manual Override 451
29.1.2 Alarm Bit Handling 451
29.1.3 Service Timing 451
29.2 Schedule Offset Module Types 451
29.2.1 Type 1 (OSS) 451
29.2.2 Type 2 (OTP) 452
29.3 Schedule Offset Module Common Parameters 453
29.4 Schedule Offset Module Connectives 453

30 SENSOR MODULES 455
30.1 Functionality 455
30.1.1 Alarms 455
30.1.2 COV Operation 457
30.1.3 Module Disable 457
30.1.4 Module Override 458
30.1.5 Logging Sensor Modules 458
30.1.6 Double Precision (v3.30 or greater) 458
30.2 Sensor Module Types 458
30.2.1 Type 0 (Analogue) 458
30.2.2 Type 1 (Internal Analog) 462
30.2.3 Type 2 (Digital) 465
30.2.4 Type 3 (Internal Digital) 466
30.3 Sensor Module Common Parameters 466
30.4 Sensor Module Connectives 468

31 SENSOR TYPE MODULES 469
31.1 Functionality 469
31.1.1 Sensor Type Library 469
31.1.2 Input Type 469
31.1.3 Scaling 469
31.1.4 Use with Type 19 (Lookup) Function Modules 477

31.2 Sensor Type Module Parameters 478

32 SEQUENCE TABLE MODULE 479
32.1 Functionality 479
32.1.1 Multiple Entries 479
32.1.2 Disabled Modules 480
32.1.3 Servicing Non-Sequence Table Modules 480
32.2 Sequence Table Module Parameters 480

33 STATES CATEGORY MODULES 481
33.1 Functionality 481
33.2 States Category Module Parameters 481
33.2.1 Main States Category Module Parameters 481
33.2.2 States Sub-module Parameters 481

34 SWITCH MODULES 483
34.1 Functionality 483
34.1.1 Reset Operation 483
34.1.2 Self Reset Operation 483
34.1.3 Toggle Switch Operation 483
34.1.4 COV Operation 484
34.1.5 Security 484
34.1.6 Switch Module Parameters 484
34.2 Switch Module Connectives 485

35 TIME MODULE 487
35.1 Functionality 487
35.1.1 Power Failure 487
35.1.2 Timing Pulses 487
35.1.3 Time Synchronisation 487
35.1.4 Daylight Saving 489
35.1.5 UTC Time 490
35.1.6 Differences between IQ2 and IQ4 Time Modules 490
35.2 Time Module Parameters 490
35.3 Time Module Connectives 493

36 TIME SCHEDULE MODULES (TIME ZONE MODULES) 495
36.1 Functionality 495
36.1.1 When the Time Schedule Applies (Effective Period) 495
36.1.2 Calculating the Output 496
36.1.3 Use with OSS and Schedule Offset Modules 503
36.1.4 Security 503
36.1.5 Module Disable 503
36.1.6 Legacy Configuration 503
36.1.7 Time Schedule Module Types 505
36.1.8 Type 0 (Digital) 505
36.1.9 Type 1 (Multistate) 505
36.1.10 Type 2 (Analogue) 506
36.2 Time Schedule Module Parameters 506
36.2.1 Main Time Schedule Modules Parameters 506
36.2.2 Day of Week Sub-module Parameters 508
36.2.3 Day of Week Time value Sub-module Parameters 508
36.2.4 Exception Sub-module Parameters 508
36.2.5 Exception Time value Sub-module Parameters 510
36.3 Time Schedule Module Connectives 510

37 USER MODULES 511
37.1 Functionality 511
37.2 User Module Types 511
37.2.1 Type 0 User Modules 511
37.2.2 Type 1 CNC User Modules (v3.60 or greater) 515

37.3 User Module Common Parameters....516

38 VIRTUAL CNC MODULES....517
38.1 Functionality....517
38.1.1 Secure Communications (v3.60 or greater)....517
38.1.2 Alarms....518
38.2 Virtual CNC Module Parameters....518

APPENDICES....519

A1 BACNET OBJECT/PROPERTY CODES AND UNITS....519
A1.1 BACnet Object Codes....519
A1.2 BACnet Property Codes....519
A1.3 BACnet Units....523

A2 XML SYNTAX....527
A2.1 Request Structure....527
A2.2 Response Structure....528
A2.2.1 Invalid Request Response....528
A2.2.2 Response to a Read....529
A2.2.3 Response to a Write....530
A2.3 Tsite File Schema....531
A2.3.1 Site Element....531
A2.3.2 Internetwork Element....531
A2.3.3 LAN Element....531
A2.3.4 Device Element....532
A2.3.5 Module Element....532
A2.3.6 Parameter Element....532
A2.4 System Compatibility....533
A2.4.1 Logging....534
A2.5 Performance....534
A2.5.1 Size of Response....534
A2.6 Using IQ4 Web Services - Trend JavaScript Libraries....535
A2.6.1 Overview....535
A2.6.2 Basic XMLWebservices requests using TrendControls.XMLWebservices....535
A2.6.3 Creating Active Data Points....538
A2.6.4 Displaying Graphs Using TrendControls.Flot.....539
A2.6.5 Browser Support....540
A2.6.6 Response Codes....540
A2.7 Authentication Security....540
A2.7.1 Authentication Types....540
A2.7.2 Hashing....541

A3 TEXT COMMUNICATIONS....543
A3.1 Addressing and Item Selection....543
A3.1.1 Absolute Addressing/Item Selection....543
A3.1.2 Attribute Addressing....543
A3.1.3 Label Matching Item Selection....543
A3.2 Text Communications Message Types....544
A3.2.1 Targeted Messages....544
A3.2.2 Global Messages....544
A3.2.3 Visitor Messages....544
A3.3 Addressing a Text Communications Message....545
A3.3.1 Specific Device....545
A3.3.2 All Devices on a LAN....546
A3.3.3 All Devices on all LANs....546
A3.4 Message Structure....546
A3.4.1 Traditional Text Communications (v comms)....546
A3.4.2 Structured Text Communications (t comms)....552
A3.4.3 Wildcards....553
A3.5 Examples of Text Communications....553

A3.6 Differences Between IQ2 and IQ4 Text Communications 556

A4 BACNET PROPERTIES SUPPORTED 557
A4.1 BACnet Analogue Input Object Type A 558
A4.2 BACnet Analogue Input Object Type B 560
A4.3 BACnet Binary Input Object 561
A4.4 BACnet Analogue Output Object 563
A4.5 BACnet Binary Output Object 564
A4.6 BACnet Analogue Value Object 565
A4.7 BACnet Binary Value Object 565
A4.8 BACnet Device Object 566
A4.9 BACnet Notification Class Object 568
A4.10 BACnet Schedule Object 569
A4.11 BACnet Calendar Object 570
A4.12 BACnet Trend Log Object 571
A4.13 BACnet over MS/TP 573

A5 UPGRADE THE LICENCE 575

INDEX 577

ABOUT THIS MANUAL

This manual describes the method of configuring an IQ4.

It is arranged both for instruction and reference. In some areas it is best to read through once in order to obtain an overview, and then again in order to understand specific details.

- **Part 1** - Provides a general overview of the IQ4 series of controllers, and the basic principles and concepts.
- **Part 2** - Describes how to install and engineer an IQ4.
- **Part 3** - Describes the IQ4 strategy modules in alphabetical order.

Note: This manual is specific to the IQ4 range of controllers; the IQ1xx and IQ2xx controllers are covered by the IQ Configuration Reference Manual (90-1533) and its addendum (TD200118), the IQ3 is covered by the IQ3 Configuration Reference Manual (TE200768), and the IQeco is covered by the IQeco Configuration Manual (TE201089).

Conventions Used in this Manual

The conventions below are designed to make it both quick and easy to find and understand the information.

- Menu commands are in **bold** type.
- Buttons, and options in dialogue boxes that you need to select are in **bold** type.
- The names of text boxes and dialogue boxes are in **bold** type.
- Key combinations that you should press appear in normal type. If joined with a plus sign (+), press and hold the first key while you press the remaining one(s). For example CTRL+P indicates holding down the control key while pressing P.
- Text you should enter is in *Italic* type.

Contacting Trend

Head Office

Trend Control Systems Limited
St. Marks Court
North Street
Horsham
West Sussex
RH12 1BW

Tel: +44 (0) 1403 211888
Fax: +44 (0) 1403 241608

Details of regional offices can be found on our web site.

Internet

Our company web site (**www.trendcontrols.com**) provides information about our products and us, or our support web site (**https://partners.trendcontrols.com**).

Technical Support

Our support department provides technical support during normal office hours. Before contacting our support department ensure that you have your Technical Support PIN number available, without this we will be unable to provide you with any support.

Tel: +44 (0) 1403 226600
Email: **trendts@trendcontrols.com**
Fax: +44 (0) 1403 226310

Technical Publications

Please send any comments on this or any other Trend technical publication to **techpubs@trendcontrols.com**.

PART 1

This part of the manual provides a general overview of the IQ4 series of controllers, and the basic principles and concepts. It is divided into the following sections:

What is IQ4?
Hardware
Firmware
Strategy
Trend Communications
BACnet Communications
Alarms
Security
Web Server
XML Web Services
XNC Facility

1 WHAT IS IQ4?

The IQ4 range is a set of small, highly configurable, intelligent controllers designed for all building control applications. Operating standalone or as part of a network they provide the solution for virtually any type and size of building. A small IQ4 may be fitted to a single item of plant (unitary controller) or a larger IQ4 may be used to control multiple items of plant in a building or plant room.

Any small system may be expanded as the need arises, by networking. In a networked system, the IQ4 can transfer information between controllers via inter-controller communications thus providing a fully integrated system and retaining the considerable advantages of distributed control. Using Wide Area Networks (WANs) a building can become one of many controlled from a central office.

Local displays can be connected to all IQ4s to enable local monitoring and adjustment by a user. A supervisor or networked display panel can be used to monitor and adjust the IQ4s over the network. All IQ4s provide full compatibility with other Trend IQ controllers.

There are five main variants in the range:

- IQ411
- IQ412
- IQ422
- IQ4NC
- IQ4E

The following sections describe the functionality in more detail:

Hardware
Firmware
Strategy
Trend Communications
BACnet Communications
Alarms
Security
Web Server
XML Web Services
XNC Facility

The range also includes the IQ4/IO Expansion modules.

1.1 IQ411 and IQ412 (IQ41x Series Controllers)

Designed for use in unitary applications. The IQ411 has 6 solid state digital outputs, and the IQ412 has 4 solid state digital outputs and two 0 to 10V outputs.

For full details of IQ411 and IQ412 see the IQ41x Series Controllers Data Sheet (TA201247).

1.2 IQ422

Designed for use in unitary and small plant applications. It has 6 universal inputs and 6 analogue voltage outputs.

For full details of the IQ422 see the IQ422 Controllers Data Sheet (TA201260).

1.3 IQ4NC

A combined controller and router, the IQ4NC is designed to join Trend networks on different media (either Ethernet, Trend current loop, or MS/TP) in various configurations. It is available in different versions that enable it to act as a network router, and as a controller.

For full details of the IQ4NC see the IQ4NC Controllers Data Sheet (TA201285).

1.4 IQ4E

Designed for use in main plant control applications. The IQ4E has 10 universal inputs and 6 analogue voltage outputs which can be expanded up to 192 points by adding DIN rail mounting I/O modules.

For full details of the IQ4E see the IQ4E Controllers Data Sheet (TA201340).

1.5 XNC Variants

The IQ4 XNC variants provide a flexible way of interfacing the Trend system with 3rd party systems. They utilise both standard IQ configuration modules, and the Trend Custom Language (TCL) to present information from other systems as though it were from an IQ controller.

The functionally of the /XNC variants is described in the IQ4/.../XNC/... Interface Data Sheet (TA201346).

1.6 INT Variants

The IQ4 INT variants enable the IQ interfacing feature. IQ interfacing enables easy integration of Serial (RS485) MODBUS, IP MODBUS and M-bus devices (e.g. meters, sensors/chillers/VRF Systems) to the IQ4. This enables the IQ4 to read and write values directly. If required an IQ4E can be connected to both MODBUS

and M-bus. flexible way of interfacing the Trend system with 3rd party systems. They utilise both standard IQ configuration modules, and the Trend Custom Language (TCL) to present information from other systems as though it were from an IQ controller.

The functionally of the /XNC variants is described in the IQ4/.../XNC/... Interface Data Sheet (TA201346).

1.7 IQ4/IO Expansion Modules

IQ4/IO modules provide a range of DIN rail mounted I/O expansion modules for use with the IQ4E, offering additional input and output channel connection points. Modules are available offering different arrangements of universal inputs, analogue and relay outputs. For details of the full range see the IQ4/IO Modules Data Sheet (TA201341).

2 FIRMWARE

Each IQ4 has a program (firmware) in a memory chip which provides the basic building blocks (strategy modules) that are configured to produce a control strategy (strategy).

2.1 Strategy Modules

Each strategy module (referred to as modules in this manual) has defined functionality. The modules available are dependent on the IQ4 type - see the appropriate IQ4 data sheet for details.

Modules that perform similar functions are grouped together in module types (e.g. modules that perform mathematical functions, such as addition or subtraction are grouped together as Function modules). Each module type is identified by a letter 'module identifier' (e.g. F is used for Function modules). A specific module of a particular module type is identified by a number (e.g. F1 is Function module 1). Each individual module is therefore identified by its module identifier, and a number that identifies the module within the group of modules.

For a full description of each module type see **'Part 3'** of this manual.

2.1.1 Module Limits

The number of modules that can be included in a strategy is limited by the amount of memory available in the IQ4.

The memory available for modules is measured in brIQs. The capacity is dependent on IQ4 type - see the appropriate IQ4 data sheet. Once the limit is reached it is not possible to add more modules. Each type of module has a memory requirement in brIQs which is listed in **'Part 3'** of this manual.

In addition to the brIQ limit IQ4 has an absolute limit of 4000 modules. The IQ4 also limits the number of Alarm Destination modules to eight. IQSET limits the number of each type of module to prevent this limit from being exceeded.

2.1.2 Module Parameters

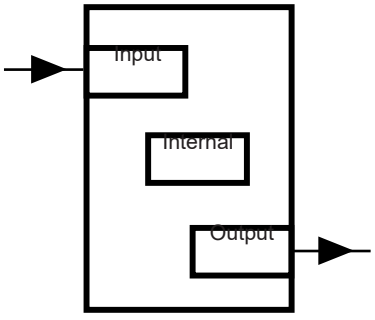
Each module has a number of parameters that control the way the module performs its function.

Each parameter is identified by a letter 'text communications identifier' (e.g. $ is used for the 'Label' parameter) which is used in conjunction with the module identifier and module number to identify a specific parameter within a specific module (e.g. S1($) specifies the 'Label' parameter of Sensor module 1).

The way a module works is defined by the firmware and cannot be changed. However a module's operation can be tailored to suit the requirements by setting up its parameters.

There are three types of parameter:

Inputs (I)
Outputs (O)
Internal (N)



Parameters are classified by data types.

Data Type	*Description*
+Full analogue	As for full analogue, but these parameters can only have zero and positive values.
Digital	Binary digits having two states (0 or 1). If they are connectable parameters they can only be connected to digital inputs or outputs as appropriate.
Double Precision	v3.30 firmware or greater: Double precision numeric parameters. If they are connectable parameters they can only be connected to analogue inputs or outputs as appropriate. A double precision analogue value has a negative excursion -10^{20} to 0 and positive excursion 0 to $+10^{20}$. However the resolution is 16 decimal digits.
Enum	A special data type that is limited to a set of predefined constants. The variable must be equal to one of the values that have been predefined for it.
Full Analogue	Floating point numeric parameters. If they are connectable parameters they can only be connected to analogue inputs or outputs as appropriate. A full analogue value has a negative excursion -10^{20} to 0 and positive excursion 0 to $+10^{20}$. However the resolution is 7.5 decimal digits (the actual number is 16777215), so any further digits are represented by zeros. Thus the number 98765432100 would be represented by 98765430000. Other analogue parameters operate over a restricted range.
Integer	Any whole number.
Strings	Alphanumeric strings. Any ASCII character except ( ){ }; , :
Time	The value is a time in the format hh:mm.

2.1.2.1 Inputs (I)

Input parameters provide an input to the module. They can be linked to another module's output, or set to a specific value. If linked to another module their value is changed by that module. If set to a specific value it will remain at that value.

In both cases the value can be changed by a supervisor or tool, or another controller using an IC communications message. If the input is connected to another module it will remain at the new value until it is updated by the connected module or another change is made. If set to a specific value it will remain at the new value until the value is changed.

2.1.2.2 Outputs (O)

Output parameters are outputs from the module. Most are available in the strategy and can be connected to the input of other modules. Some are only available to text communications and BACnet communications.

2.1.2.3 Internal (N)

Internal parameters control the way the module functions, they cannot be connected to another module. They will remain at the set value unless they are changed by a supervisor or tool, or another controller using an IC communications message. Some internal parameters are updated by the module as part of its operation.

2.2 Firmware Upgrade

The IQ4's firmware can be upgraded using the IQTool Firmware Upgrade Applet - see the IQTool Firmware Upgrade Applet Manual (TE201297). Upgrading provides access to the latest features and performance enhancements.

The firmware upgrade process will upgrade the IQ4's firmware, and then, for controllers connected to I/O modules with a lower firmware version it will upgrade the I/O module's firmware. This upgrade can take up to 7.5 minutes for 30 modules during which time the I/O modules and controller will not be able to control connected plant. The I/O modules indicate they are being upgraded by either flashing the watchdog and I/O Bus LED, or by keeping the Watchdog LED illuminated until the upgrade is complete, depending on the module's firmware version.

3 STRATEGY

In order to operate as a controller, the modules provided by the firmware must be configured to define the way the connected equipment is controlled. This configuration is known as the strategy. Strategies are configured using the System Engineering Tool (IQ™SET) by:

Adding Modules
Connecting Modules
Configuring Modules

This produces a strategy file that can be downloaded to the IQ4 to define its operation. This file consists of all the strategy modules, their parameters, and links. When this is downloaded it is stored in the IQ4 and then run using the firmware.

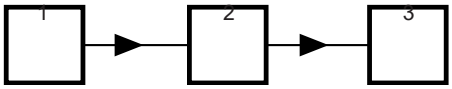
For details of using IQSET see the IQSET Manual (TE200147).

3.1 Adding Modules

Modules can only be added to the strategy using IQSET; it is not possible to add modules in any other way.

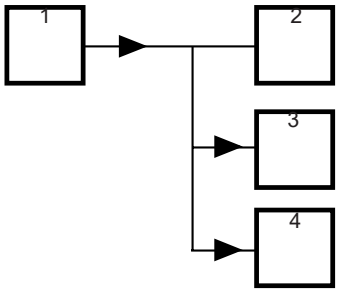
3.2 Connecting Modules

Modules can be connected in virtually limitless combinations by connecting a output from one module to an input of another, to achieve the required control strategies.



In the diagram above a signal is passed from module 1 to module 2 and similarly another signal is passed from module 2 to module 3.

Inputs to more than one module can be sourced from a single output. In the diagram below the output of module 1 is connected to the inputs of modules 2, 3, and 4.



Normally the input to a module will be sourced from one output, but it is possible to connect more than one output to an input; this is used in multiplexing - see **"Multiplexing" on page 30**.

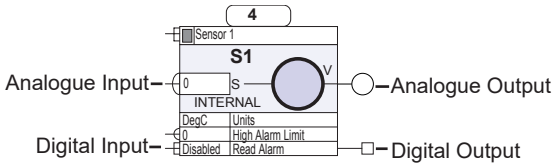
Note: An output cannot be connected to another output.

Connection Types: Connections between modules can be one of two data types:

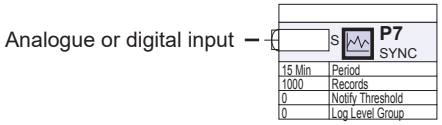
Analogue
Digital

An analogue connection is used to pass a value, whereas a digital connection is used to pass an ON/OFF status (i.e. 1 or 0).

On strategy diagrams analogue outputs are represented by circles and digital outputs by squares. Analogue inputs are represented by half circles and digital inputs by half squares.



Normally an analogue output cannot be connected to a digital input, nor can a digital output be connected to an analogue input. An exception is the Plot module's 'Source' parameter which can be connected to either an analogue or digital output and it has a special symbol:



3.2.1 Connection to I/O Channels

Driver, Type 0 (Analogue) Sensor, Type 2 (Digital) Sensor, and Type 2 (External) Digital Input modules must be connected to the hardware input/output channel (I/O channel) to which the corresponding I/O device is wired. This is done by specifying the hardware module of the input/output, and its channel number.

The 'Hardware Module' parameter is used to specify the I/O module on which the I/O channel is located. Hardware module '0' specifies inputs/outputs on the IQ4E, numbers greater than '0' refer to inputs/outputs on external I/O modules.

Note: Only IQ4Es and IQ4NC/XNCs support external I/O modules therefore for all other types of IQ4 the 'Hardware Module' parameter must be set to '0'.

The 'Hardware Channel' parameter is used by Type 0 (Analogue) Sensor, Type 2 (Digital) Sensor, and Type 2 (External) Digital Input modules to specify the I/O channel. Driver modules can use two output channels, an in-phase channel and an anti-phase channel. The channel used by the in-phase channel is specified by the 'In-Phase Hardware Channel' parameter, and the channel used by the anti-phase channel is specified by the 'Anti-phase Hardware Channel' parameter.

Note: A Driver module's in-phase and anti-phase channels must be located on the same hardware I/O module.

3.3 Configuring Modules

Modules are configured by:

Editing their parameters to the required values to define the module's operation - see **Part 3** of this manual for details of individual modules and their parameters.
Setting how and when the module is serviced - see **"Module Servicing" on page 26**.

3.4 Edit Module Parameters

Module parameters can be edited using IQSET (recommended), or the IQ4's web pages.

3.5 Module Servicing

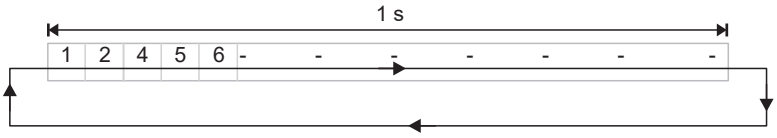
The input and internal parameters of the modules in the strategy are processed and the internal and output parameter updated (serviced) in one of two ways:

- **Sequenced Servicing**
- **Event Driven Servicing**

3.5.1 Sequenced Servicing

Sequenced servicing ensures that the modules are serviced at regular intervals and that the order of execution is fixed. Most modules use sequenced servicing to determine when they are processed. When using sequenced servicing the servicing of modules is controlled by the sequence table which is executed once per second (cycle time). The sequence table has a set number of steps, dependent on IQ4 variant - see IQ4 data sheet for details.

Each module that is to use sequenced servicing is allocated a sequence step. The IQ4 works its way through the sequence table servicing each module in turn (the module at step 1 first then the module at step 2 etc).



The following module types may be allocated a sequence step:

Sensor
Function
Logic
Loop
Schedule Offset (pass through value)
Internal Digital Input
Knob
Switch

Note: Knob and Switch modules default to event driven servicing but should be entered into the sequence table if any of their inputs are connected.

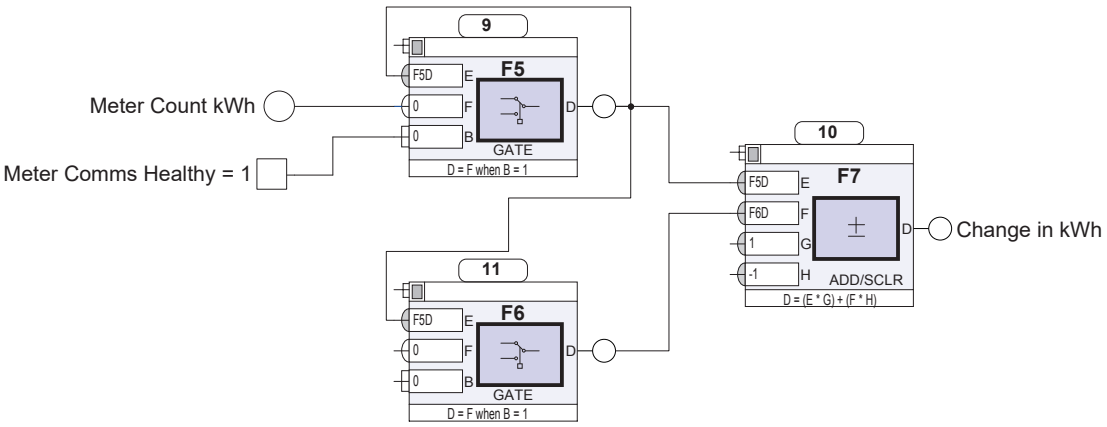
Some modules are not in the sequence table, but are serviced regularly. Type 1 (Synchronised) Plot modules are checked to see if they require to be updated at the beginning of the sequence table cycle (i.e. once per second). Time Schedule, OSS, Schedule Offset modules are checked every minute.

3.5.1.1 Sequence Table Order

The order of the modules in the sequence table determines the order in which they are serviced; some strategies will only operate correctly if serviced in the correct order. In general terms, modules are entered in the order of information flow in the control strategy:

- Sensors modules before Loop modules
- Function, Logic, Schedule and Loop modules before Driver modules

Examples of where sequence table order is important, are duty/standby selection (the status of plant device is required before a decision to select is made), and the example below where the change in meter output is being calculated.



In the example, the meter count in kWh is connected through F5 as long as the meter comms are healthy. In F7 the old value is subtracted so the output of F7 is the change since the last meter reading. F6 takes the old value and connects it through F6 to input F of F7. It is important that F6 is serviced after F5 and F7 so that it retains the old value ready for the next sequence cycle.

When the strategy is first started the value of F7's input F will be '0', so that the output of F7 will be the full current count of the meter. If the meter comms goes unhealthy, the F5 will recycle the old meter count so that F7's output will be '0', and when the comms returns, the output of F7 will include the change over the missed cycles.

3.5.1.2 Multi-sequencing

A module may be serviced more frequently than once per cycle by allocating more than one sequence step.

Note: It is recommended that, Type 4 (Counter) Logic modules and Loop modules are not multi-sequenced. A Loop module's service time is specified by the 'Reschedule Time' parameter and this cannot be less that the sequence table cycle time; inserting more than once alters the meaning of the 'Integral Time' parameter and hence the integral term.

A whole sequence of modules (i.e. a sub-strategy) may be put in the sequence table more than once to economise on modules at the expense of sequence table entries. This is called Multiplexing - see **"Multiplexing" on page 30**.

3.5.2 Event Driven Servicing

Event driven servicing enables a module to be serviced when a change to one of its input values (an event) occurs. A set of event driven modules is generally used for fast processing e.g. to switch a light on when the switch is pressed.

The following modules are always event driven:

- External Digital Input
- Analogue Node
- Digital Byte

Knob and Switch modules can also be event driven but can be sequenced if required. They default to event driven servicing but should be entered into the sequence table if any of their inputs are connected.

Sensor, Driver, Function, Logic, Schedule, Type 1 (Internal) Digital Input, Knob, and Switch modules can be set to be event driven by taking them out of the sequence table.

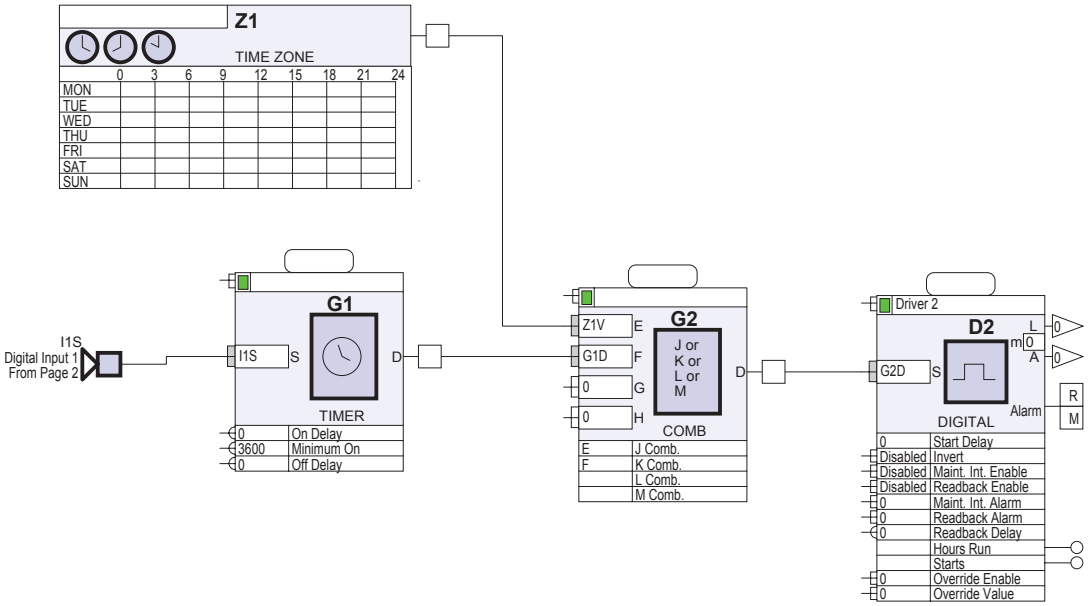
Note: It is recommended that an event driven sequence does not include Loop, Type 1 (Filter) Function, or Type 6 (Hours Run) Logic modules.

Note: The number of modules that can be event driven is limited to 300.

If a module is event driven its 'Sequence Table' parameter is set to '0'.

3.5.2.1 Service Order

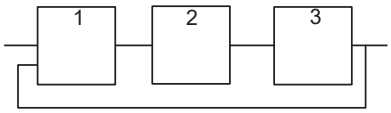
An event driven module is not serviced until the input changes. The advantage is that the module will respond immediately to the input change. Event driven modules connected together form a fast sequence; the modules are serviced in the order they are connected as the event 'travels' through them.



The above modules (G1, G2, D2) are event driven as they do not have sequence steps. When Digital Input module 1's output is set to '1', G1 'immediately' starts a 1 hour pulse (3600s) which will switch the Driver module ON during non-occupation; during occupation the Driver module is ON anyway.

3.5.2.2 Feedback

Event driven strategies cannot include feedback (i.e. a connection from an output back into an input of a preceding module) - see below:



In the strategy above the output of module 3 is linked to the input of module 1 (feedback). This will cause an infinite loop because when the output of module 3 changes it causes an event for module 1 which in turn will change the output of module 3. The IQ4 protects against an event driven feedback loop consuming resources by limiting the number of times a module can be serviced when event driven to 15 per sequence table cycle.

Note: Since Analogue Node, Knob, and Switch modules are always event driven, their outputs should never be connected back to their inputs.

3.5.3 Maximum Number of Modules Serviced

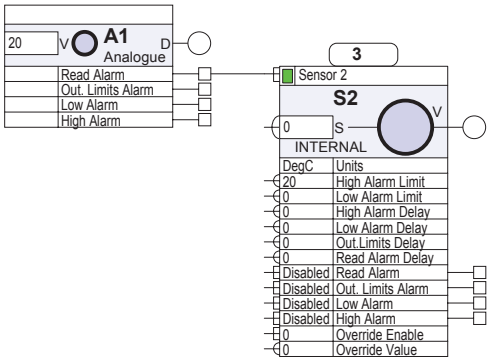
The total number of modules the IQ4 can service in 1s is limited according to the formula below:

number of sequence steps available in the IQ4+maximum number of event driven modules

3.5.4 Disabling Modules

Some types of modules can be disabled by setting the 'Disable Module' parameter to '1'. A disabled module is no longer serviced and its output(s) and internals are maintained at their values prior to disabling. Disabling modules can be used for strategy fault finding or in event driven sequences to prevent 'run away' operation.

The ‘Disable Module’ parameter can either be sourced from another module in the strategy or can be set to a specific value. In the strategy below the Sensor module is disabled when the Analogue Node module is in READ alarm.

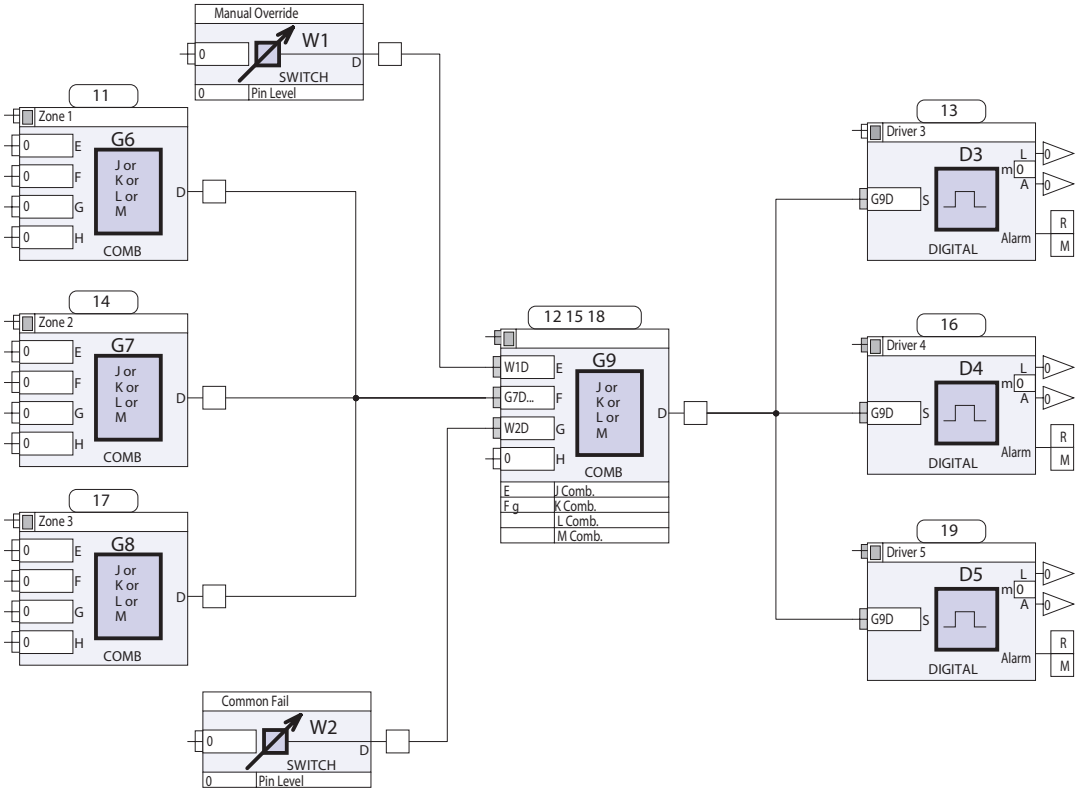


3.5.5 Multiplexing

Multiplexing is not recommended for IQ4 and should not be necessary as the IQ4s have plenty of capacity. It is described in this manual so that strategies imported from earlier IQs may be understood.

The general principle of multiplexing is to have a common sub-strategy with input and output points and to enter this many times in the sequence table preceded by its source modules and succeeded by its destination modules.

Example - Strategy using multiplexing



The strategy above is a simple example to demonstrate the principle. There are three zones producing a demand calling for a zone fan for each zone and all the fans are to switched ON using one manual override signal and all switched OFF during normal running if there is a common fail condition.

Instead of requiring three Logic modules G9 can be used three times by multiplexing it between the input and output modules. The modules must be put in the sequence table as shown:

Sequence Step	*Module*
11	G6

Sequence Step	*Module*
12	G9
13	D3
14	G7
15	G9
16	D4
17	G8
18	G9
19	D5

This process saves two Logic modules but, fault finding a strategy like this is difficult, and the technique should not be used unless it is necessary to save modules.

Note: Modules whose current output depends on current input and previous output levels such as Loop modules (with Integral), Type 1 (Filter) Function modules, Type 4 (Counter) modules Logic, or Type 6 (Hours Run) modules should not be multiplexed.

3.6 Module Overrides

Driver, Sensor, and Digital Input modules can be overridden by setting the module's 'Override Output' parameter to '1'. It is normally for use by IQSET's Live Edit feature, or by an engineer using a web page during commissioning, but can be sourced from another module in the strategy.

When overridden the value of the module's 'Override Value' or 'Override Status' parameter is applied to the input to the module's algorithm and will be used along with the module's other input and internal parameters to calculate the output.

Note: Whether the module has an 'Override Value', or 'Status' parameter depends on its type, e.g. a Type 2 (Analogue) Driver module will have an 'Override Value' parameter, whereas a Type 1 (Digital) Driver module will have an 'Override Status' parameter.

Because the override value is applied to the module's algorithm the module will generate any alarms in the normal way. This allows the module's alarms to be tested by adjusting a module's 'Override Value' or 'Override Status' parameter while it is overridden (e.g. raising a Sensor module's 'Override Value' parameter while it is overridden can be used to test its HIGH alarm operation).

Automatic clearing of overridden modules: In order to prevent modules being accidently left in override, when a module is overridden the Address module's override timer is started and when it reaches '0' any overridden modules will be forced out of override and returned to normal operation.

Note: Use of the override facility should not be used for safety during maintenance work.

3.7 Display and Directory Modules

Display and Directory modules are used to configure the structure and content of a set of pages for display on user interfaces (e.g. IQView8, or IQ4 GraphIQs) within an IQ4.

A web client can access the Display and Directory modules in the form of graphic pages known as GraphIQs. Supervisor and Tools access the Display and Directory modules in a hierarchy.

For details of Display module parameters see **"Display Modules" on page 203** and for Directory modules see **"Directory Modules" on page 201**.

3.7.1 Web Client Access

Web clients present Display and Directory modules as graphical pages 'GraphIQs' that enable defined values to be displayed against a graphical backdrop.

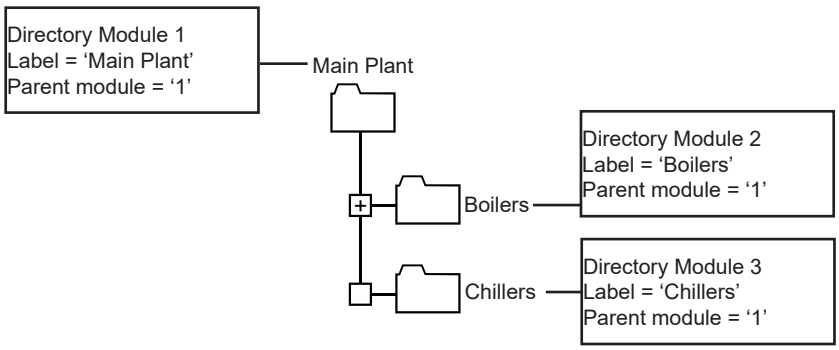
They are created from the Display and Directory modules' structure:

- Each Directory module is a separate GraphIQ page and the Display modules define what is displayed on the GraphIQ page. Directory modules can have a graphical backdrop onto which the Display modules' text is be placed.
- Display modules can be set to either static or dynamic. The Dynamic Display modules reference values from the IQ4 and are automatically refreshed at a defined rate. Static Display modules can be used for labels (in GraphIQs pages only). On GraphIQs pages, Static Display modules can have hyperlinks to other web pages, and Dynamic Display modules can have hyperlinks to the appropriate modules' details web pages. They have 'Xco-ordinate' and 'Yco-ordinate' parameters which enable them to be located at appropriate points in front of the backdrop. The Directory module also has a 'Backcolour' parameter which defines a colour to be used as background if there is no backdrop or the browser cannot access the backdrop file.

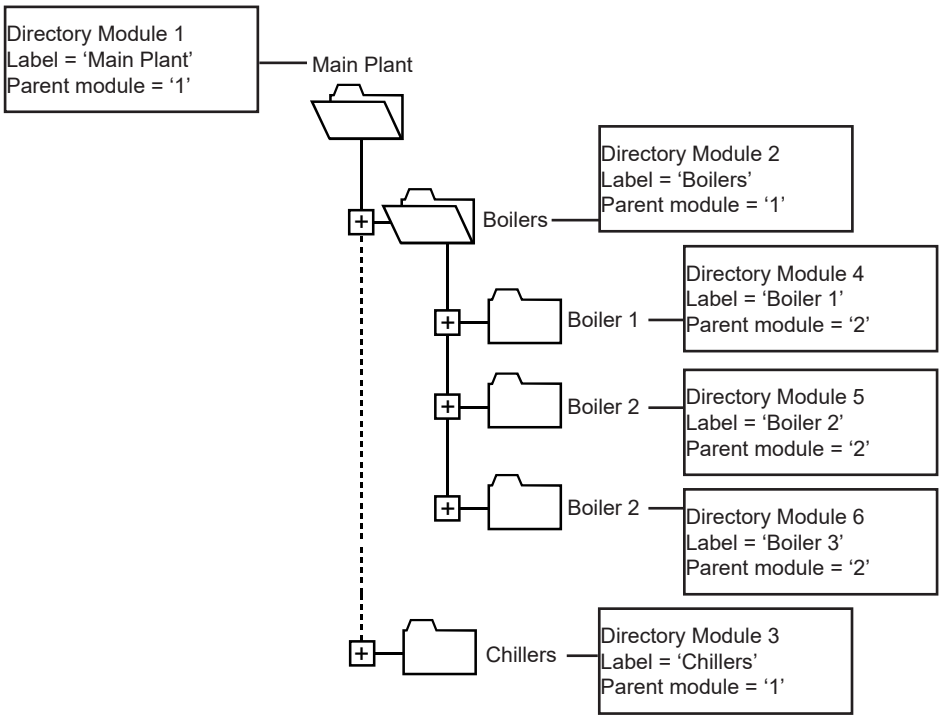
3.7.2 Supervisor/Tool Access

Supervisors and tools present Display and Directory modules in a hierarchy that enables the user to navigate to the required information. This gives users the ability to view the equipment and the associated items (e.g. Sensor, and Knob modules) in a structured manner. In addition, the actual displays can be user dependent by setting up appropriate PIN levels.

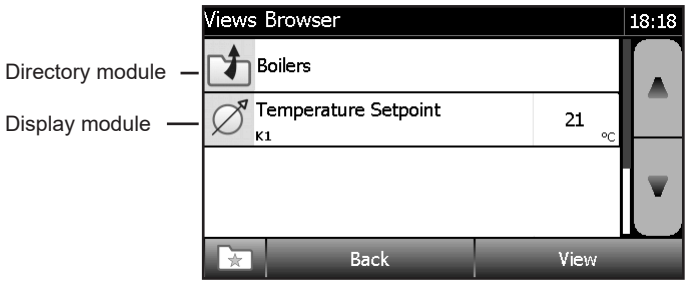
The Directory modules define the hierarchy which enables the user to navigate the system. Each module has a 'Parent' parameter which refers to the Directory module above it in the hierarchy. Directory module 1 is always the root of the structure (and its parent should be set to '1'). For example: the diagram below shows three Directory modules. Directory module 1 is at the top of the hierarchy (root). Directory modules 2 and 3, have Directory module 1 as their parent so form the second level of the hierarchy.



This structure can be expanded by having more Directory modules using Directory module 2 as their parent.



A supervisor or display (e.g. IQVIEW4) can request the Display and Directory modules and display them in a display similar to that shown below.



Display modules define what information is displayed when a particular Directory module is selected in the hierarchy Each display module has a 'Parent' parameter which enables it to be attached to a Directory module.

In the above display the Directory module 'Boilers' has been selected. The Display module shown has the 'Boilers' Directory module as its parent. The display is able to read the item reference to be displayed from the Display module, and request the relevant parameters' values (e.g for a Sensor module this would be the 'Label', 'Value', and 'Units' parameters).

The item within a Dynamic Display module may be selected from the following module types (they are known as presentation modules).

Sensor
Digital Input
Knob
Switch
Driver
Time Schedule

The following Directory module parameters are ignored when displayed by a supervisor/tool:

Backdrop
Background Colour
Width
Height
Page Size

Static Display modules are not shown by a supervisor/tool, and the following Dynamic Display module parameters are ignored:

Link Reference
Foreground colour
InAlarm colour
Background colour
Font Type
Font Size

Display modules which reference modules other than Sensor, Digital Input, Knob, Switch, Driver, or Time Schedule modules are ignored.

Note: IQView8 will read these parameters and use them to display the GraphIQ page.

3.7.3 User Dependent Displays

User dependent displays are achieved by using PIN levels. Each Directory module is assigned a PIN level. For the Directory module to be visible in the hierarchy to a particular user they must be logged on to the supervisor/display with a password with associated a PIN that matches one in one of the IQ4's Type 0 User Modules that is greater than or equal to PIN level of the Directory module. This means that the user will only see the information appropriate to their level.

3.8 Saving Critical Data

When an IQ4 powers down (deliberately or due to a power failure) or it detects a fault condition such as a program exception (i.e. the program stops running), or watchdog failure, it stores the values of module input and output parameters to flash memory according to specific criteria to allow the strategy to run correctly on the first sequence cycle after power is restored. The values stored are called critical data and are selected according to the following rules:

- Unconnected input parameters. These are critical because they cannot be restored by recalculating from earlier modules in the sequence.
- Connected input parameters unless they are connected to a module that is earlier in the sequence table. These are critical because they cannot be restored by recalculating from earlier modules in the sequence. Input parameters of event driven modules are excluded from this because they will be generated correctly on the event driven first pass.
- Internal parameter values
- Special exceptions

Values of module output parameters are non-critical because they will always be recalculated.

3.9 IQ4 Files

The IQ4 has the following files that are downloaded to define its operation:

Strategy File (.IQ4)
Backdrop Files
XNC File (/XNC variants only)

The IQ4 stores these files in non-volatile memory.

3.9.1 Strategy File (.IQ4)

The main strategy file for the IQ4 is the '.IQ4' file which contains the modules in the strategy, the definition of how they are linked together, and the values of their parameters.

3.9.2 Backdrop Files

Backdrop files are the files used as a backdrops in the IQ4's GraphIQs. They can be .jpg or .gif files. When a backdrop is used it must be downloaded to the IQ4.

Note: It is recommended that a backdrop file is not greater than 100 kbytes. It is not allowed to be greater that 200 kbytes. The maximum memory available in the IQ4 to hold backdrops is 2 Mbytes.

3.9.3 XNC File (/XNC variants only)

/XNC variants can run a TCL application (XNC file) created in TCLTool2. The XNC file can only be downloaded to /XNC variants. It can be downloaded using TCLTool2 or IQSET as part of the strategy download.

3.9.4 Writing Changes to Flash

Alarm Log or Plot module parameter values are immediately written to flash, but any strategy changes (e.g. via web pages) are saved up and written periodically. This normally occurs within 15s, but if changes are being repeatedly written back, this time will be extended to preserve flash life. This process is referred to as 'archiving' and its current status is shown by the Program module's 'Archive State' parameter.

3.10 Restore to Factory Defaults

The IQ4 can be restored to factory defaults by using the service button - see the appropriate Installation Instructions - Configuring for details. Restoring to factory defaults will clear down the strategy files, and set the following parameters to their default values:

Address module
- Local Lan
- Local Address
- Identifier

Type 4 (Ethernet IP) Network module
- IP Addressing Mode
- IP Address
- Subnet Mask
- UDP Port
- Router 1
- Remove Device IP Addresses and subnet masks

Virtual CNC modules
- Set to default settings

User modules
- All are deleted

Alarm logs
- All are deleted

Plot modules
- All are deleted, and all logged data is lost

TLS Certificates
- All X.509 certificates used for the HTTPS web server are removed
- The sVCNC certificate is removed

Note: The time and date are not changed.

After the reset the IQ4 should be re-configured from scratch and the strategy downloaded.

4 HARDWARE

The hardware of all variants in the range are designed to keep a family feel while accommodating the individual requirements. All variants provide the following features:

DIN compatible to fit within standard electrical enclosures - TS35 Din rail (EN500022)
230Vac and 24Vac versions.
Ethernet connection
Trend current loop option
Wallbus connection
RS232 and USB local supervisor/engineering ports.

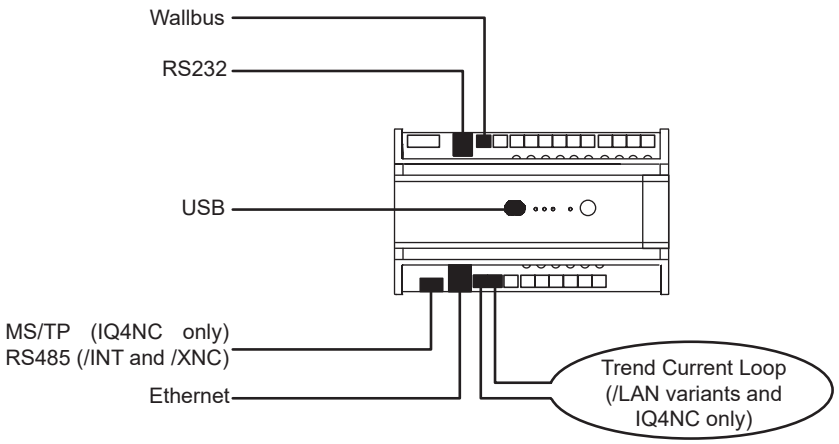
The IQ4E hardware also provides an I/O bus for connection of Trend I/O modules. The IQ4NC hardware also provides connection to an MS/TP trunk. /XNC variants provide the ability to connect to an RS232 or RS485 network, and /INT variants provide the ability to connect to MODBUS or M-bus.

For full details of the hardware see the appropriate IQ4 data sheet.

5 TREND COMMUNICATIONS

This section describes communications using Trend communication protocol. Communications using the BACnet protocol are described in **"BACnet Communications" on page 63**. To understand this section it is necessary to have knowledge of the principles of Trend networks - see the IQ System Manual (TE200896).

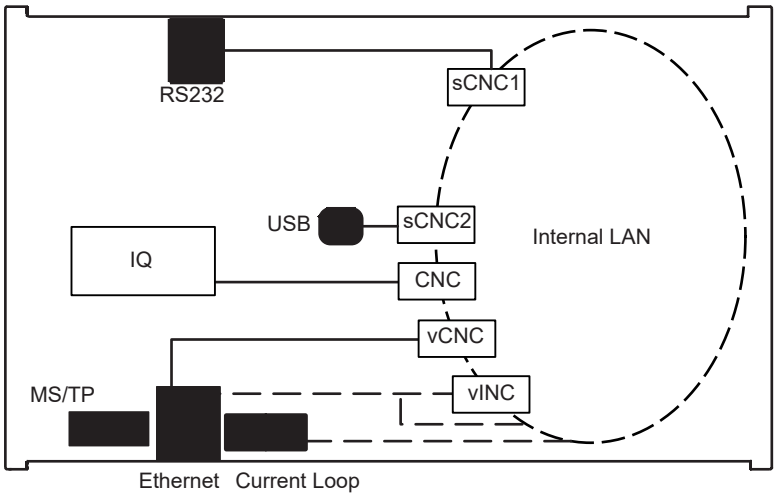
IQ4s have ports for Ethernet, RS232, USB Local Engineering and Wallbus. In addition /LAN variants have a Trend current loop, and IQ4NCs have both Trend current loop and MS/TP.



The IQ4 will create its own internal LAN which can include:

a node for its connection to the Trend network (CNC)
a sCNC for its RS232 Local Supervisor Port (sCNC1)
a sCNC for its USB Local Engineering Port (sCNC2)
a vCNC to allow connection of a supervisor/tool/display to the Trend network (vCNC)
a virtual INC for connection to other LANs i.e. build an internetwork (vINC).

The diagram below shows an IQ41x connected via it's own CNC (CNC) to a internal network including its sCNC for the RS232 port (sCNC1), its sCNC for the USB port (sCNC2), its vCNC (vCNC), and its vINC (vINC) which is enabled when the IQ4 is acting as the INC for a LAN.



IQ422s, IQ4NCs and IQ4Es have the same internal LAN but have additional vCNCs. The IQ4NCs vINC is always enabled.

Note: /LAN variants with the Trend current loop network enabled will not have the vINC. /LAN variants are unable to perform INC functions on the Trend current loop LAN.

Note: A vCNC has to be enabled by setting up its 'Device Address' and 'Port Number' parameters.

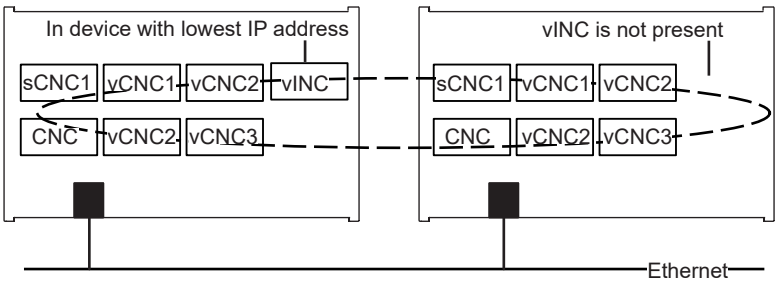
5.1 Ethernet Communications

The IQ4's Ethernet communications operate via the Ethernet Port which allows the IQ4 to be connected to an Ethernet network. Connection to an Ethernet network enables:

Building LANs on Ethernet
Building Internetworks on Ethernet
Stand Alone IQ4
Connection to an IQ4's vCNC
Connection by a Web Client
Connection of IQVIEW SCD
Email Alarms from IQ4
IP Alarms from IQ4
Upload/Download of Configuration Files

5.1.1 Building LANs on Ethernet

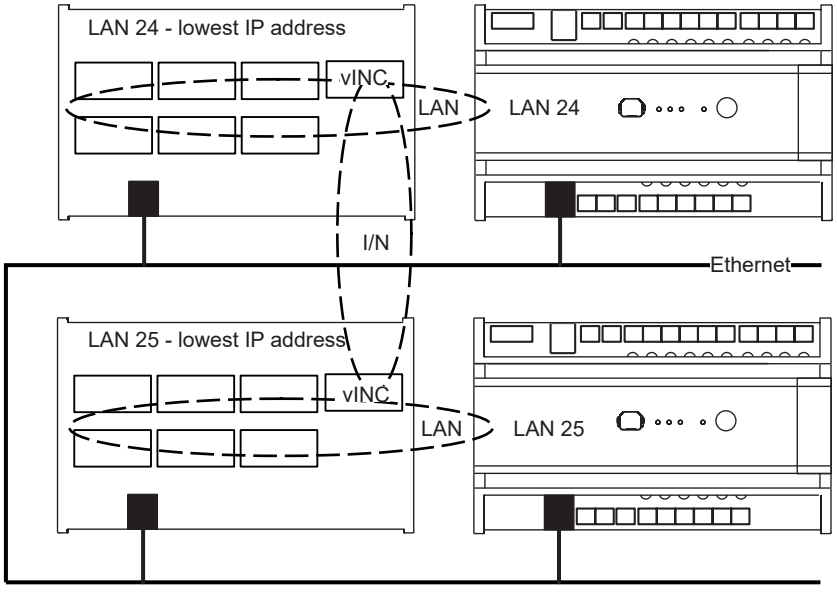
If multiple Trend devices with the same LAN number are connected to the Ethernet network, then a Trend LAN on Ethernet will form to include these devices and their internal nodes. They must be on the same Ethernet segment, in the same subnet and use the same UDP port. The device with the lowest IP address assumes the INC functionality and is responsible for maintaining the LAN (the vINCs in the other devices disappear).



Note: An IQ4NC always has a vINC present on its internal LAN. The vINC always has a LAN to connect to the internetwork so the IQ4NC cannot join another Trend LAN on Ethernet, and hence there cannot be another Trend Ethernet device (e.g. XTEND) with the same LAN number.

5.1.2 Building Internetworks on Ethernet

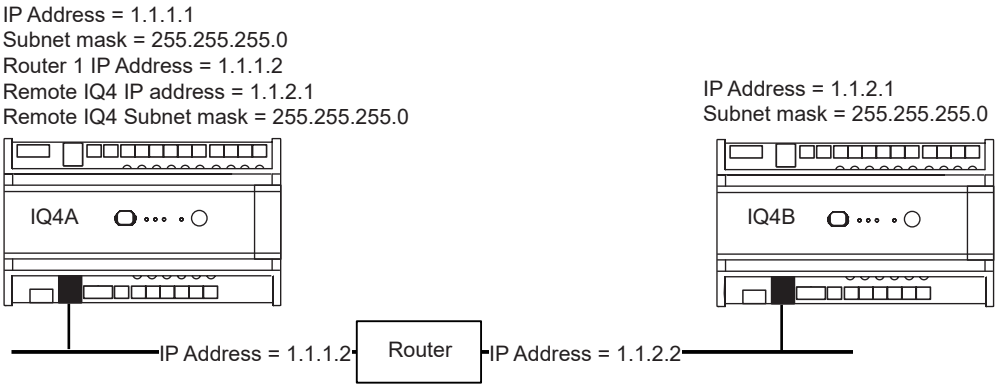
If Trend Ethernet devices have different LAN numbers they connect together via an internetwork on Ethernet. The devices with the lowest IP address on each LAN retain their vINCs and provide the INC functionality for the LAN, while any other devices lose theirs. The vINCs are responsible for constructing the internetwork.



5.1.2.1 Building an Internetwork on Ethernet Across Routers

In order for the internetwork to be built across a router, the device connecting the LAN to the internetwork (device with INC functionality) must have details of at least one device the other side of the router (remote devices). A remote device's details consist of its IP address or host name and subnet mask. They are stored in the Type 4 (Ethernet IP) Network module (up to 20 remote devices can be set up).

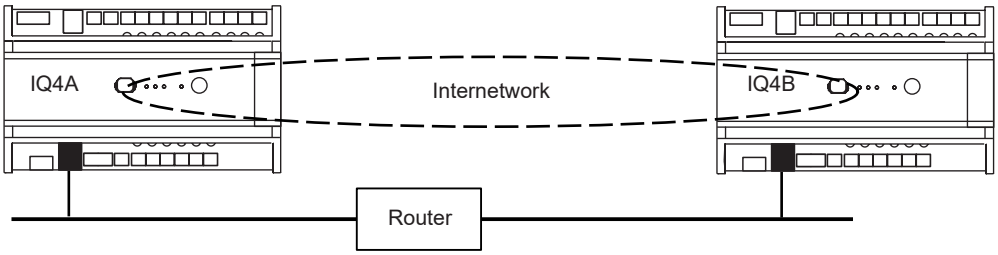
In the diagram below the two IQ4s are separated by a router.



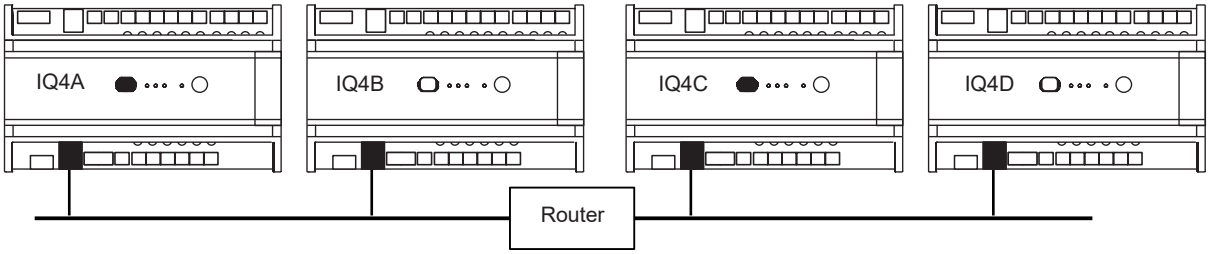
To build an internetwork across the router, IQ4A's Type 4 (Ethernet IP) Network module must be set up as follows:

Parameter	*Description*
Router 1	Set to IP address or host name of local router (e.g. 1.1.1.2)
Remote Trend Device n: Remote Device	IP address or host name of IQ4B (e.g. 1.1.2.1)
Remote Device Subnet n	Subnet mask of IQ4B (e.g. 255.255.255.0)

This information enables IQ4A to construct an internetwork with IQ4B across the router.



If several Trend Ethernet devices are connected one side of a router, the device with the lowest IP address (cross-router master) will attempt to construct the network across the router if it has a remote device set up.



In the diagram, if the cross router master is IQ4A, it should be set up with the information for an IQ4 the other side, e.g. IQ4C. IQ4A will then construct the internetwork across the router. If IQ4C were to be switched off, the message from IQ4A will no longer reach the remaining IQ4 on the far subnet (IQ4D).

To prevent this, IQ4D's details should also be entered in the network module of IQ4A so that it can connect to IQ4D if IQ4C is switched off.

To reduce the risk of losing communications due to the loss of power to one device, the two devices with the lowest IP addresses from each subnet should be set up in the two devices with the lowest IP addresses on the local subnet.

Because IP addresses may not be fixed it may not be possible to specify which devices have the lowest IP addresses, so the details should be set up in every device on the subnet so that any device may be able to become the cross router master.

If there is more than one router on the system the details of at least one Trend Ethernet device from every subnet must be set up in the cross router master. The cross router masters on the other subnets will need similar details.

It is recommended that the details of two devices in the network from each subnet be set up in every device on the local subnet. For increased reliability, details of additional devices should also be set up.

Note: The lowest IP address on each LAN will assume INC responsibility and join the internetworks, while the lowest IP address on the subnet will become the cross router master (if remote devices are set up) and attempt to cross the router. Thus the cross router master will also have INC responsibility for its own LAN.

Crossing Routers if DHCP is Operating

In the DHCP regime, if the internetwork is to cross a router(s), the Remote Devices (1 to 20) 'IP Address' parameters should be set up as host names. This will enable the IP addresses to be obtained from the WINS servers.

Note: If any communication using a host name crosses a router(s), then a WINS server address must be set up.

5.1.2.1.1 When Not to Build Internetworks Across Routers

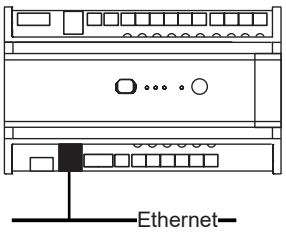
Trend Ethernet devices should not be configured to build internetworks across routers where the routers are being used to connect the Ethernet network via Wide Area Networks (WANs). This is because the quality, availability and security of the connection cannot be guaranteed.

Building an internetwork across routers is intended for use in intranet applications where the quality of the service between nodes on the network is permanently good. For wide area applications, or where there may be poor or intermittent connectivity then multi-site networking using temporary vCNC connections into the sites is recommended.

Note: If IC communications are to be used between controllers separated by a WAN then an internetwork has to be constructed crossing the routers; this can only be done if there is a good IP connection.

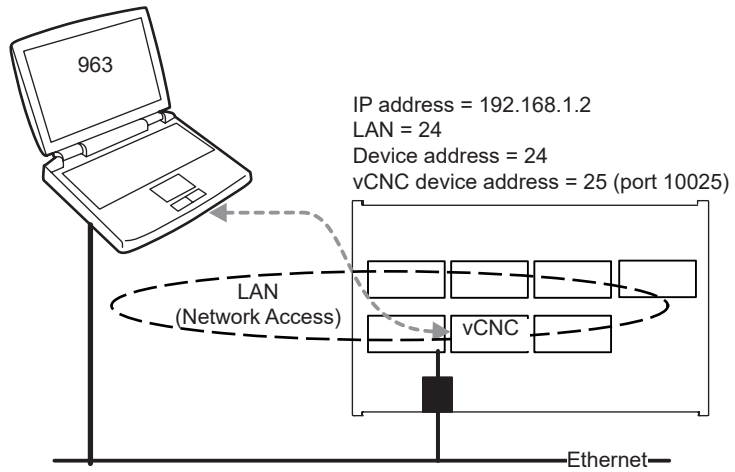
5.1.3 Stand Alone IQ4

Stand alone mode can be enabled in the Type 4 (Ethernet IP) Network module. Enabling stand alone mode stops the IQ4 from attempting to build LANs or internetworks with other Trend Ethernet devices. It will still communicate with supervisors, web clients, and local supervisors/displays. An IQ4 should be set to stand alone mode to reduce Ethernet network traffic (i.e. to disable polling messages trying to find other Trend devices) when there is no requirements for it to build LANs and internetworks on Ethernet.



5.1.4 Connection to an IQ4's vCNC

A supervisor/tool or display can use a vCNC in the IQ4 to connect to the Trend network. The connection is made over Ethernet by specifying the IP address or host name of the IQ4 and the port number of the vCNC. In the example below the 963 connects to the Trend network via the vCNC in the IQ4. The 963 is configured to connect to IP address: 192.168.1.2 port 10025. Once connected communications intended for the 963 are addressed to LAN 24 device address 25 (the vCNCs address.



In order to connect to the vCNC, an IP connection must be possible between the connecting device and the device containing the vCNC.

Note: There may be a firewall(s) between the PC and the IQ4, either at the PC end, the IQ4 end or both. In this case ensure that the port used by the IQ4's vCNC is open in both directions (send and receive) on the firewall(s).

Consult with an IT specialist in order to ensure that the IP connection is possible and does not cause a security risk.

Note: If the IQ4 has a large strategy, the IQ4's vCNC should not be used as the main supervisor connection to a large system; this should be done using an XTEND.

5.1.4.1 Permanent/Temporary Connections

The connection to the vCNC can be either permanent or temporary. A permanent connection is one where the software is permanently connected to a vCNC. A temporary connection is one where the software connects to the vCNC when necessary, and then closes that connection when it is no longer required (e.g. on a multi-site system).

Note: Not all Trend software is capable of making temporary connections.

If the connection to the vCNC is permanent, then the IQ4 can send Trend LAN type alarms (traditional Trend system alarms using LAN and device addressing) to connected Trend software via to the vCNC. However, if the connection is temporary, then there are three choices:

- Send alarms as IP address type (recommended).
- Send alarms to a vCNC in an XTEND that is operating in alarm mode.
- Use an IQ4NC's alarm forwarding feature - see **"INC Alarm Forwarding (IQ4NC only)" on page 91**.

5.1.4.2 Secure Connections (v3.60 or greater)

From v3.60 the vCNC can be configured to use secure communications between the IQ4 and the connected device - see **"Secure Mode (v3.60 or greater)" on page 517**. It is recommended that where possible secure communications are used.

5.1.4.3 Connecting to an IQ4's vCNC

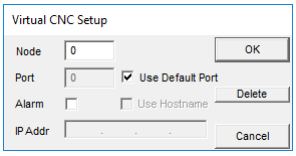
Note: All firewalls between the client and the IQ4 must be configured to allow the connection - see **"4.1.1 Configure Firewalls" on page 131**.

5.1.4.3.1 Connecting to an Unconfigured IQ4's vCNC

This procedure applies to an IQ4 that is unconfigured, or does not have any CNC User modules set up using IQSET.

To connect to an unconfigured IQ4:

1. Enable the required vCNCs using IPTool - for detail of using IPTool see the IPTool Manual (TE200638).
   - Connect to the Ethernet network.
   - Run IPTool.
   - In the **Devices List**, click the IQ4 containing the vCNC that is to be enabled.
   - In the list of vCNC's double click the vCNC that is to be enabled. The **Virtual CNC Setup** dialogue box is displayed.

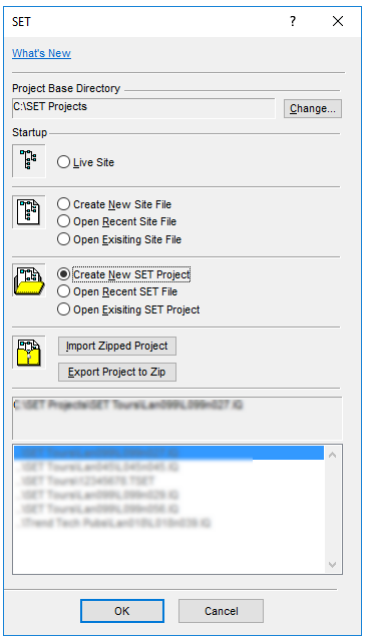


   - In the **Node** box enter the network address of the vCNC, range (1 to 119 excluding 2, 3, and 10). 0 will disable the vCNC.
   - Specify the TCP port used by virtual CNC.
     - To force the default port number to be used select the **Use Default Port** check box this will cause the port number to be set to 10000 plus the network address.
     - To specify a port manually clear the **Use Default Port** check box and in the **Port** box enter the required port number.

     Take note of this port number because it will be required when connecting to the vCNC.

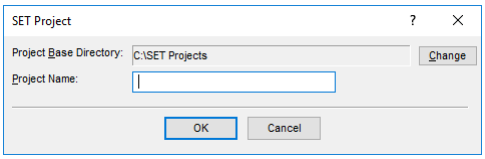
   - Click **OK**.
   - Click **WRITE DATA**.

2. Add a connection the vCNC in IQSET - for details of using IQSET see the IQSET Manual (TE200147).
   When the vCNC is enabled it will operate in secure mode there for the initial connection must be made by IQSET which will enable the IQ4's vCNCs and a CNC User module to be configured.
   - Run IQSET. The **SET** dialogue box is displayed.

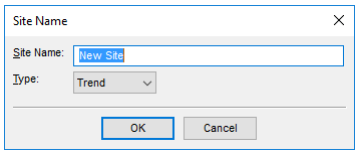


   - Click **Create New Site File**.

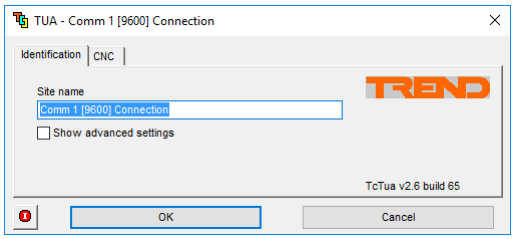
- Click **OK**. The **SET Project** dialogue box is displayed.



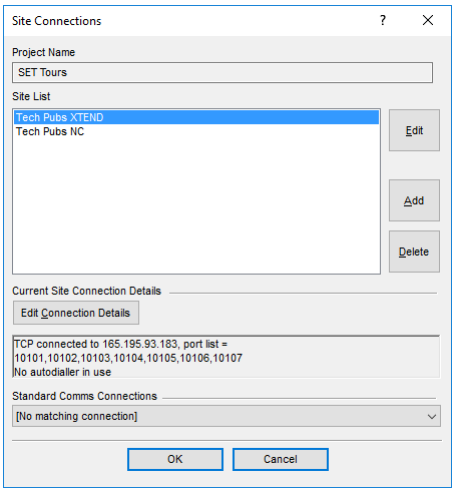
- In the **Project Name** box enter a name for the project.
- Click **OK**. The **Site Name** dialogue box is displayed.



- In the **Site Name** box enter a name for the site.
- Select **Trend** in the **Type** box.
- Click **OK**. The **Connection Wizard** is displayed.
- Click **Launching TUA Editor**.
- Click **Next**. The **TUA Editor** is displayed.



- Select the **CNC** tab.
- Select the **TCP/IP** option.
- In the **Device address** box enter the IP address or host name of the IQ4.
- Specify the port number of the vCNC by adding them to the **Remote IP port list**. To add a port to the list, enter the required number into the box above **Add** and click **Add**.
- Click **OK**. The **Site Connections** dialogue box is displayed site will be added to the System View and a site file created.

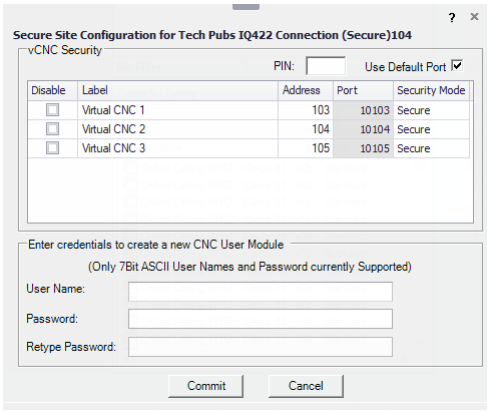


- Click **OK**. The site will be added to the **System View** and a site file created. When a site is first added to the **System View** it does not automatically connect.

3. Connect the vCNC in IQSET - for details of using IQSET see the IQSET Manual (TE200147).
   - Right click the site and select **Refresh**. The **Service Button Validation** dialogue box is displayed.



   - Go to the device containing the vCNC and press its service button within 10 minutes to prove you have physical access to the device. The **Secure Site Configuration** dialogue box is displayed which allows the device's vCNC connections to be configured.



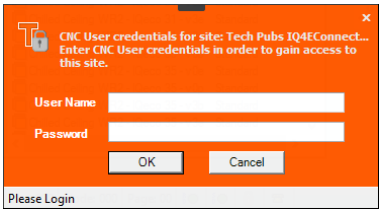
To continue with the just the existing vCNC in secure mode you MUST add a CNC User module, alternatively you can configure the change the IQ4's vCNCs.

To add a CNC User module:
- Enter the CNC User module's name in the **Name** box.
- Enter the CNC User module's password in the **Password** box
- Retype the password in the **Retype Password** box.

To change the IQ4's vCNCs:
- Specify whether the vCNC port number is to be manually specified or use the default value (10000 + Address) by selecting/clearing the **Use Default Port** box.

  Note: The 'Use Default Port' setting applies to all vCNCs in the controller.

- Enable/disable the required vCNCs by clearing/selecting the **Disable** check boxes, (selected = disabled) for click OK.
- Specify the label for the vCNC in the **Label** box.
- Specify the network address of the vCNC in the **Address** box (range 1, 4 to 119 excluding 10).
- If the default port is NOT being used specify the port for the vCNC in the **Port** box.
- Specify the security mode in the **Security Mode** box (Secure recommended)
- If any of the vCNCs are operating in 'Secure' mode and you have not already added a CNC User module you **MUST** add one.
- Click **Commit**. A message is displayed once the changes have been successfully sent to the controller.
- Click **X** to continue.
- If the CNC connection is still secure the **CNC User credentials** dialogue box is displayed.



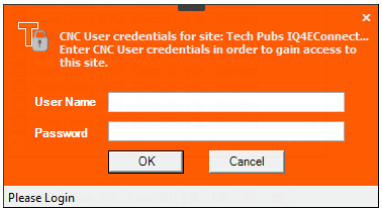
- Enter the CNC User module's name in the **Name** box (user name for the CNC User module that has just been created).
- Enter the CNC User module's password in the **Password** box (password for the CNC User module that has just been created).
- Click **OK**. The connection will be made.

5.1.4.3.2 Connecting to a Configured IQ4's vCNC in Secure Mode

This procedure applies to connecting to an IQ4s vCNC in secure mode that is configured with CNC User modules using IQSET.

To connect to a configured IQ4's vCNC in Secure Mode:

1. Add a connection the vCNC in IQSET - see page 44.
2. Connect the vCNC in IQSET - for details of using IQSET see the IQSET Manual (TE200147).
   - Right click the site and select **Refresh**. The **CNC User credentials** dialogue box is displayed.



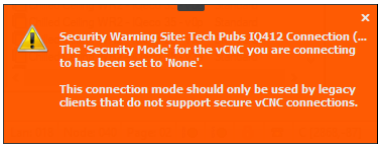
- Enter the CNC User module's name in the **Name** box (user name of any CNC User module the IQ4).
- Enter the CNC User module's password in the **Password** box (password for the CNC User module specified above).
- Click **OK**. The connection will be made.

5.1.4.3.3 Connecting to a Configured IQ4's vCNC in Non Secure Mode (Not Recommended)

This procedure applies to connecting to an IQ4s vCNC in operating in non secure mode using IQSET.

To connect to an IQ4's vCNC in Non Secure Mode:

1. Add a connection the vCNC in IQSET - see page 44.
2. Connect the vCNC in IQSET - for details of using IQSET see the IQSET Manual (TE200147).
   - Right click the site and select **Refresh**. The **Trend Security Warning** dialogue box is displayed.

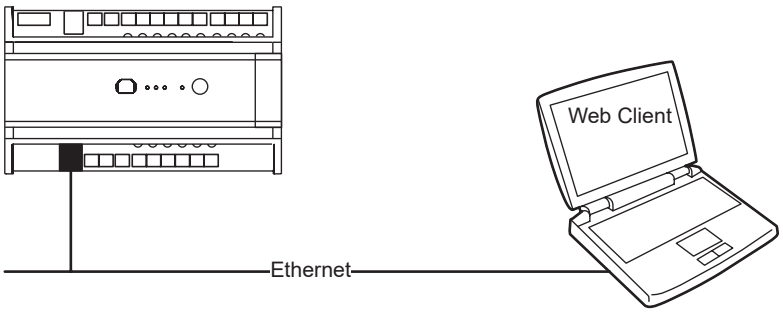


- Click **X** to continue IQSET will connect securely (encrypted) without the need to specify a user name and password.

This type of connection should only be used by software that does not support secure connections, e.g 963. When connection using software that does not support secure connections a warning is not displayed and the connection will be made without encryption. If the software does support a secure connection the connection will be encrypted but not authenticated.

5.1.5 Connection by a Web Client

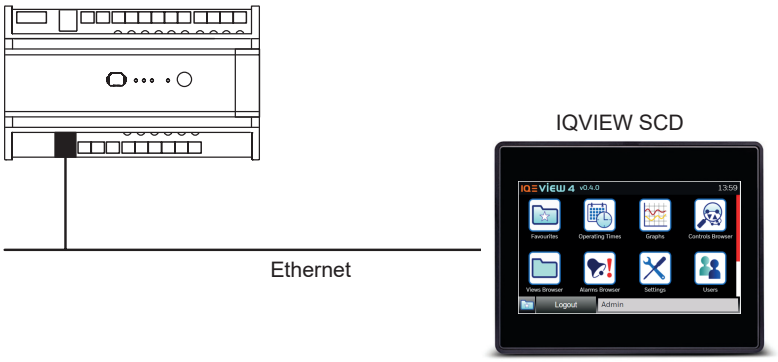
When the IQ4 is connected to Ethernet, a web client can connect to the IQ4's web server providing an IP connection is possible between the two - see **"Web Server" on page 95**.



Note: All firewalls between the client and the IQ4 must be configured to allow the connection - see **"4.1.1 Configure Firewalls" on page 131**.

5.1.6 Connection of IQVIEW SCD

From v3.34 firmware of IQ4 an IQVIEW Single Controller Display (IQVIEW SCD) can be be connected to the IQ4 to provide a local display.



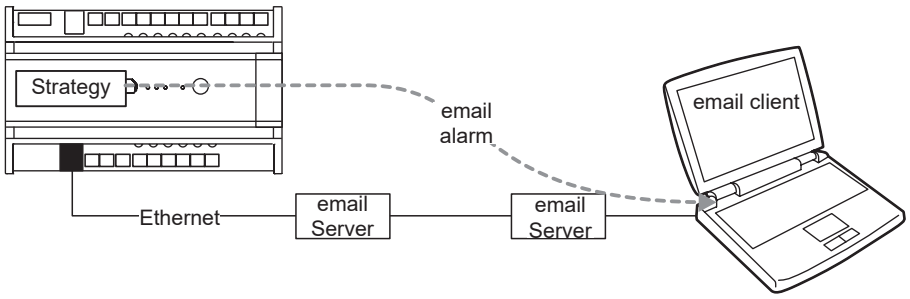
The display will be connected over Ethernet to the IQVIEW SCD application which is hosted in the IQ4 and provides all the functionality.

The application must be downloaded to the controller using IQSET and then configured.

For details of installing the application see the IQVIEW SCD for IQ4 Manual (TE201476).

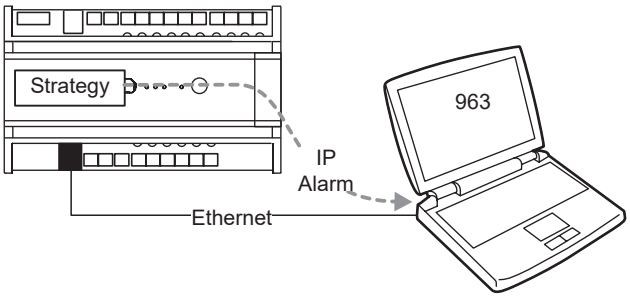
5.1.7 Email Alarms from IQ4

When the IQ4 is connected to Ethernet, the IQ4 is able to send alarms to an email address.



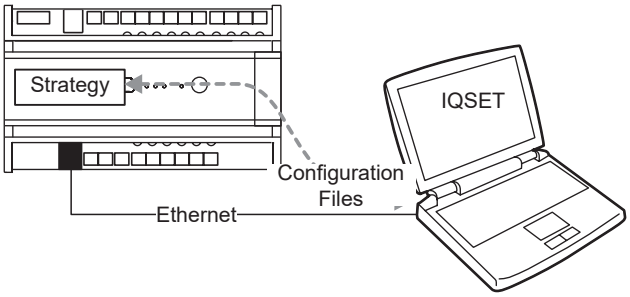
5.1.8 IP Alarms from IQ4

When the IQ4 is connected to Ethernet, the IQ4 is able to send alarms to a specified port number on a specified IP address. The device at this address must be capable of listening for IP alarms.



5.1.9 Upload/Download of Configuration Files

The strategy file (IQ4 file), XNC files and any backdrop files required for GraphIQ pages can be uploaded/downloaded over the Ethernet network.



5.1.10 Communication with Modbus IP Devices

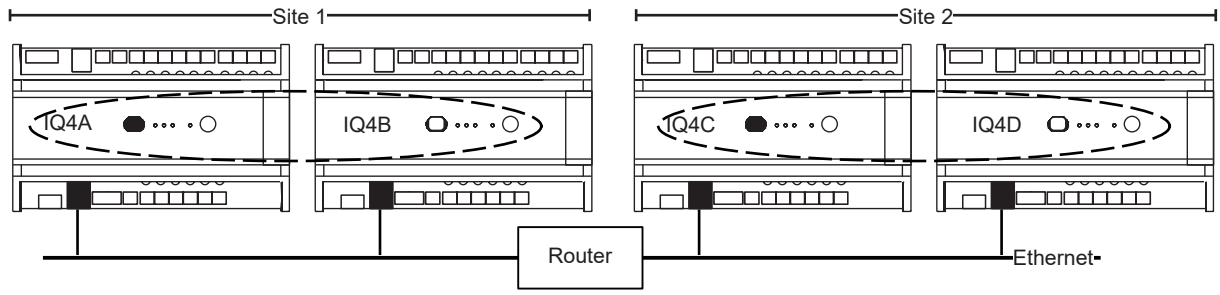
On /INT variants the Ethernet port can be used to communicate with IP Modbus devices.

5.1.11 Using Routers to Provide Multiple Sites

If the IQ4s are not configured to cross the router, they will construct two separate networks (sites) one each side of the router.

Only one internetwork is allowed on a site, but can be treated as separate sites by a supervisor and each site may have an internetwork.

In the diagram below there are two IQ4s either side of a router.



Note: There cannot be any IC communications between the sites.

Note: IQ4 alarms can be sent to the supervisor directly as IP address type alarms.

5.1.12 IP Addressing

The IQ4's IP addressing details can be set up automatically (**Automatic IP Addressing**), or manually (**Static IP Addressing**).

5.1.12.1 Automatic IP Addressing

The IQ4 is able to operate in a system where the IP addresses are automatically allocated by a Dynamic Host Configuration Protocol (DHCP) server (i.e. the IP addresses are not fixed). To operate with DHCP the IQ4's Type 4 (Ethernet IP) Network module's 'IP Addressing Mode' parameter must be set to 'Obtain Automatically', this is the default.

When set to use automatic addressing the IQ4 obtains its IP addressing details from the DHCP server - see **"DHCP Operation" on page 50**. If the DHCP server fails to respond, the IQ4 will enter 'link/local' mode where it will negotiate its IP address with the other devices on its Ethernet segment - see **"Link/Local Operation" on page 51**.

Host name

Because the IP address is not fixed when using automatic IP addressing, the IQ4 has an addressing parameter 'Hostname' which does not change when the IP address changes. The 'Hostname' parameter defaults to a function of the IQ4's MAC address ('TREND' plus the last six digits of the MAC address e.g. TREND_00_14_D0).

When the IQ4 powers up it sends its host name to a Windows Internet Naming Service (WINS) server. If a device wishes to communicate with the IQ4 it will send the IQ4's host name to the WINS server which will return the associated IP address allowing the device to address the IQ4.

If a WINS server does not exist, the host name can be used over the local segment only (i.e. not across routers). The host name can also be used to communicate with the IQ4 if static IP addressing is being used.

Email Server Address

Because the email server's IP address is not fixed on a DHCP system the IQ4s Type 4 (Ethernet IP) Network module's 'Email Server Address' parameter must be specified using a host name or an Internet domain name. When an email alarm is to be sent the Email Server IP Address is resolved by either a WINS server (if a host name is used), or by Domain Name System (DNS) server (if a domain name is used).

Servers used in Auto-addressing

DHCP server: If the IQ4's IP address settings are to be supplied by a DHCP server, a DHCP server must be installed on the same segment as the IQ4. The DHCP server must be capable of downloading the WINS server address, and the DNS server address.

Note: On a multiple segment system a single DHCP server may be used providing it has connections to each segment i.e. multiple connections.

DNS server: If email alarms are being sent, and the email server address is identified by Internet domain name, then a DNS server must be installed somewhere on the system.

WINS server: If host names are being used for IP addressing across a router, then a WINS server must be installed on any segment.

5.1.12.1.1 DHCP Operation

If the IQ4's 'IP Addressing Mode' parameter is set to 'Obtain Automatically', when the IQ4 powers up it will inform the DHCP server which then provides it with:

IP Address
Subnet Mask
Router 1 (default router)
WINS Servers (1 to 4)
Time Server Address
DNS Servers (1 to 4)

It will then use these parameters to connect to the Ethernet network and it will be able to communicate across routers if correctly configured.

Fixing an IQ4's Address on a DHCP Controlled System

It is possible for an IQ4 to operate in a DHCP regime with a fixed IP address by setting up the DHCP server so that it always gives that particular device the same IP address. An alternative is to set the IQ4's Type 4 (Ethernet IP) Network module's 'IP Addressing Mode' parameter to 'Enter manually', and set its IP address outside DHCP server's range.

5.1.12.1.2 Link/Local Operation

If the IQ4's 'IP addressing Mode' parameter is set to 'Obtain Automatically' the IQ4 will enter link/local mode if it powers up and the DHCP server fails to respond, or if it has been running successfully with a DHCP server and the server fails to renew its lease on its IP address.

When in link local mode the IQ4 will negotiate its IP address with other devices on the same Ethernet segment; IP addresses start at 169.254.0.0. Its subnet mask is set to 255.255.0.0, and the 'Router 1', 'WINS Server', and 'DNS Server' parameters will remain at their last settings.

Note: There may be a delay between the DHCP server failure and the IQ4 entering link/local mode as it will only prompt the DHCP server after its lease has expired which may be a long time. The first device to go into link/local will cause the other devices on its segment to go into link/local. The IQ4 will generate a 'Changed to DHCP Link/Local' network alarm - see **"DHCP Alarms" on page 75**.

Any devices wanting to communicate with an IQ4 in link/local mode using IP addressing must be on the same Ethernet segment and must be operating in link/local mode.

Note: A PC (running supervisor or tool software) should be set up for auto-addressing. On DHCP failure it may take a very long time to enter link/local mode, and there will appear to be communication failure. This can be avoided by power cycling the PC which will cause it to enter link/local mode on power up.

If the DHCP server operation recovers, the IQ4 will return to automatic DHCP operation, and will generate a 'Changed to DHCP automatic' network alarm - see **"DHCP Alarms" on page 75**.

Link/local Default Operation

Trend Ethernet devices are set to use automatic IP addressing by default. If a group of these devices are connected together on an Ethernet segment (without DHCP, WINS servers) they will power up in link/local and negotiate their IP addresses. If they have been set up with device addresses and LAN numbers they will construct a Trend network. A supervisor or system tool running on a PC on the same segment will be able to communicate with them using host names (if the PC is set up for auto-addressing). Such a system cannot form an internetwork across a router.

Note: Link/local mode can only be used as normal operation on a single segment system, (not across a router). Other device communicating with the IQ4 (e.g. a supervisor PC) must also be in link/local mode.

5.1.12.2 Static IP Addressing

The IQ4 is able to operate in a system where the IP addresses are static. To operate with static IP addressing the IQ4's 'IP Addressing Mode' parameter must be set to 'Enter Manually'.

When set to use static IP addressing the IQ4s IP addressing parameters ('IP Address', 'Subnet Mask', 'Router 1', 'WINS Server' and 'DNS Server') must be specified manually.

5.1.13 UDP

User Datagram Protocol (UDP) is used by Trend Ethernet devices to build LANs and internetworks, to cross routers, and for IC comms. To construct a network the Trend Ethernet devices must use the same UDP port. By default all Trend Ethernet devices use 57612.

The port number should only be changed:

- If the IT system already uses UDP port 57612.
- To have separate sites on one Ethernet subnet.

5.2 RS232 Communications

The IQ4's RS232 communications operate via the RS232 Local Supervisor Port which is provided for the optional connection of local display devices, sensors or supervisory tools. It enables:

Connection of Displays/Supervisors/Tools for Communication Only with the IQ4
Network connection of Displays/Supervisors/Tools
Upload/Download of Strategy Files

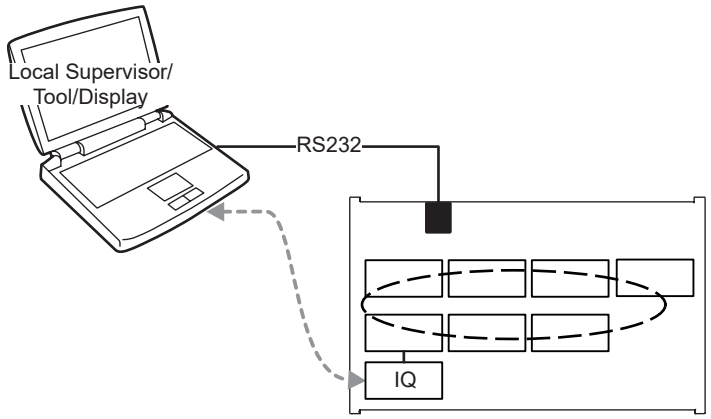
Adapter cables are available to suit different devices. The port can supply power to SDU and RD, but not XW/R/IQ, IQView, IQView4, or IQView8. For details of the power available see the appropriate IQ4 data sheet. Only one device may be connected.

Note: On standard IQ4s the connector can be used for connection of displays/supervisors/tools for communication only with the IQ4, network connection of displays/supervisors/tools, and upload/download of strategy files but will not be able to be used to connect to a 3rd party system.

On /XNC variants it can be used to communicate with devices with an RS485 connections. On /INT variants it is used to communicate with Serial Modbus and M-bus devices.

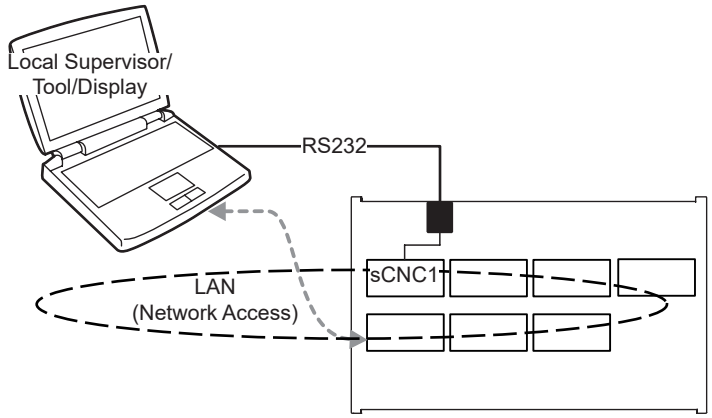
5.2.1 Connection of Displays/Supervisors/Tools for Communication Only with the IQ4

The connected device can be restricted to communicate only with the IQ4 by setting the Address module's 'Supervisor Port' parameter to '0'.



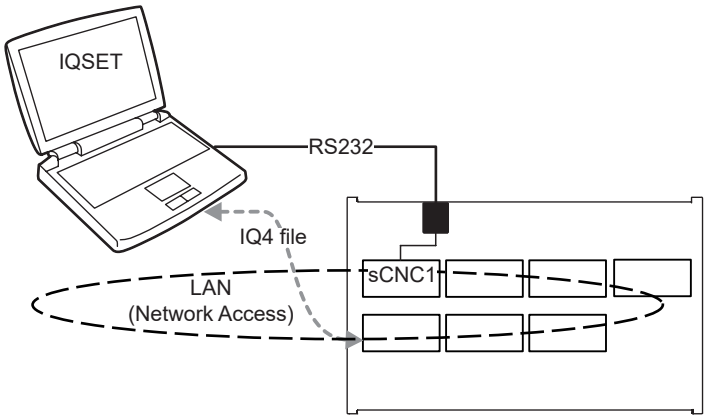
5.2.2 Network connection of Displays/Supervisors/Tools

The connected device can be allowed to communicate with the entire Trend network by setting the 'Supervisor Port' parameter to a value other than '0' which enables the IQ4's sCNC.



5.2.3 Upload/Download of Strategy Files

The strategy file (IQ4 file), but no other strategy files may be uploaded/downloaded over the RS232 connection to any IQ4 on the Trend network.



Note: If the Address module's 'Supervisor Port' parameter is set to '0' it will only be possible to upload/download strategy files to the local IQ4 not other controllers on the network.

5.2.4 Communication with Modbus Serial Devices

On /INT variants the RS232 port can be used in conjunction with an externally powered RS232 to RS485 converter to communicate with serial (RS485) Modbus devices.

5.2.5 Communication with M-bus Devices

On /INT variants (v4.30 or greater) the RS232 and RS485 port can be used in conjunction with an RS232 to M-bus converter or RS485 to M-bus converter to communicate with M-bus devices.

5.3 USB Communications

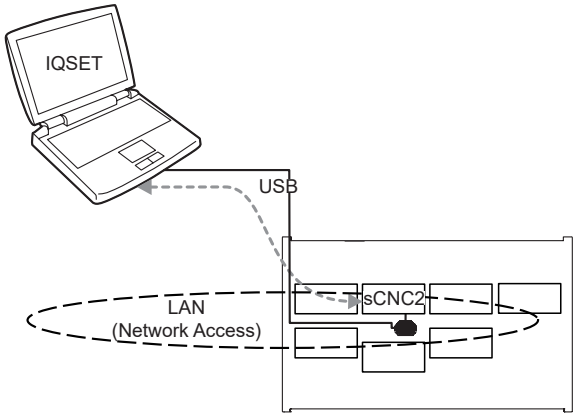
The IQ4's USB communications operate via the USB Local Engineering Port which is provided for the optional connection of IQSET. It enables:

Network Connection of IQSET
Upload/Download of Strategy Files

Connection is made using a standard USB cable. Only one device may be connected at one time.

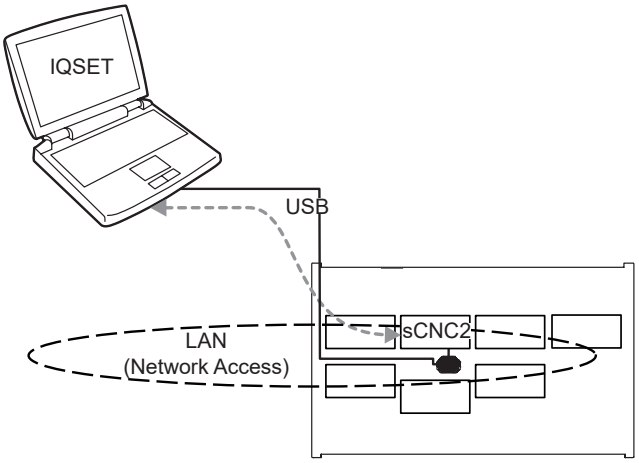
5.3.1 Network Connection of IQSET

If the Address module's 'USB Supervisor Port' parameter is set to '0' (default) and IQSET is connected to the port a supervisory CNC (sCNC2) at address 125 is dynamically created while IQSET is connected. When IQSET is removed the sCNC times out and no longer exists on the network. If the 'USB Supervisor Port' parameter is set to non-zero, the sCNC2 remains on the network when IQSET is removed.



5.3.2 Upload/Download of Strategy Files

The strategy file (IQ4 file not backdrop files) may be uploaded/downloaded over the USB connection to any IQ4 on the Trend network.



5.4 Wallbus Communications

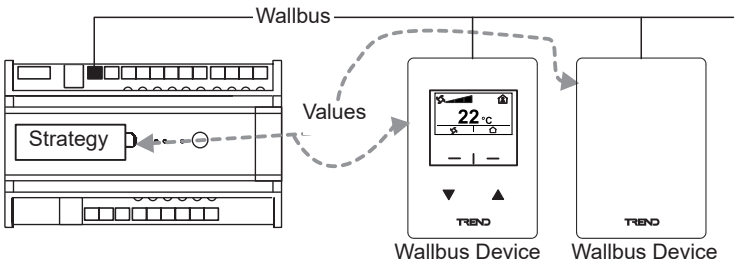
The IQ4's Wallbus communications are only available on IQ4 with v3.30 firmware or greater. They operate via the Wallbus connection. It enables:

Connection of Wallbus Devices for Communication Only with the IQ4

Note: Wallbus communications are not available on IQ422/00/XNC/.... and IQ4NC/00 variants.

5.4.1 Connection of Wallbus Devices for Communication Only with the IQ4

IQ4's Wallbus communications support connection of Trend's range of Wallbus devices. Connected devices can only communicate with the IQ4 not other devices on the Trend network. More than one device can be connected providing the Wallbus is able to provide enough power for each device.



For details of the values that can be shared between the IQ4 and the display see the Wallbus device's documentation.

5.5 RS485 Communications

The IQ4's RS485 communications are only available on /XNC, and /INT variants.

On / XNC variants it can be used to communicate with devices with an RS485 connections. On /INT variants it is used to communicate with Serial Modbus and M-bus devices.

Note: IQ4NC /INT variants have an RS485 port, which is normally used for the MSTP Trunk - see **"5.7 MS/TP Trunk Communications" on page 58***. However, if the RS485 port is not used for the MSTP trunk it can be used for Modbus.*

5.6 Trend Current Loop Communications

The IQ4's Trend current loop communications operate via the current loop connector. The Trend current loop network connectivity is only available on /LAN variants and IQ4NCs. Connection to a Trend current loop network enables:

Building LANs on a Trend Current Loop Network
Upload/Download of Configuration Files

If IQ4 has its Trend current loop network enabled, communications using a vCNC or sCNC will only operate if the network is intact. If there is no Trend current loop network connected, connect a loop back - see the installation instructions provided with the IQ4.

Communications using the Trend current loop can be disabled using the Type 1 (IQ LAN) Network module's 'Disable Module' parameter.

Address/Baud rate: The IQ4's address on the network is configured in the Address module. The 'Local Address' and 'Local LAN' parameters are set up in the factory on a rolling basis, so in a batch of IQ4s, each will have a different default address (printed on the unit's label).

The baud rate can be auto-detected in the range 1k2, 4k8, 9k6, 19k2, 38k4 (where 4k8 = 4.8 kilobaud). The Type 1 (IQ LAN) Network module's 'Autobaud' parameter enables the IQ4 automatic baud rate setting feature if set to 'Autobaud' (default). If the 'Autobaud' parameter is set to 'Manual Baud', the baud rate must be entered manually. Normally autobaud is suitable, but it may preferable to set the baud rate manually if:

- The network consists of all IQ4s, (then one IQ4 should be set to the baud rate master and have its baud rate fixed by setting it manually).
- The IQ4 has problems detecting the baud rate (LAN OK green LED flashing for longer than 30 s).

In addition to building LANs on a Trend current loop network the IQ4NC can build internetworks on a Trend current loop network.

Building an Internetwork on a Trend Current Loop Network

In order for the Trend current loop to operate it must be enabled in the Type 1 (IQ LAN) Network module (it is enabled by default).

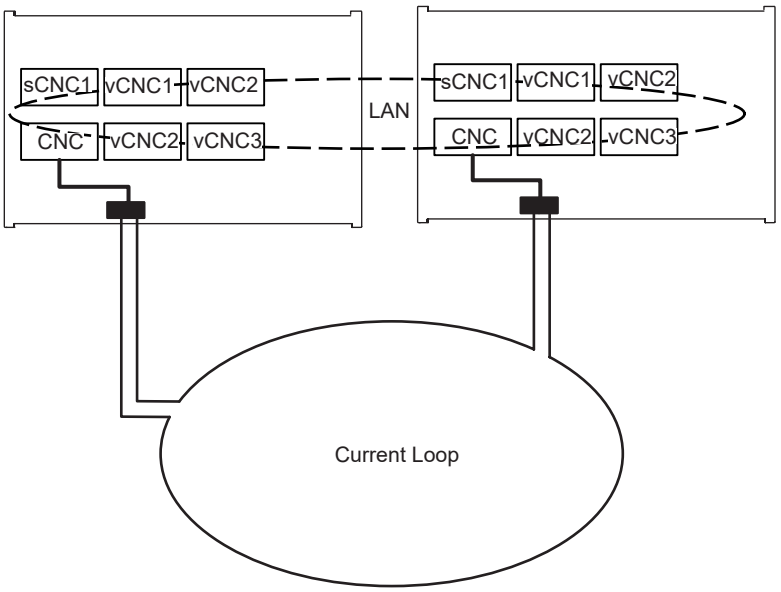
The Trend current loop will operate on IQ4NC when in Current Loop Internetwork to MS/TP LAN, Ethernet Internetwork to Current loop LAN or Ethernet Internetwork to Current loop Internetwork modes.

The /LAN variants can join a Trend current loop LAN, but can no longer join a Trend LAN on Ethernet. The IQ4NC is able to join both Trend current loop and Ethernet networks.

5.6.1 Building LANs on a Trend Current Loop Network

/LAN variants and IQ4NCs will build a LAN with other Trend devices connected to the same Trend current loop network, providing the baud rate is the same. This is not an issue with IQ4s because they will automatically detect the baud rate.

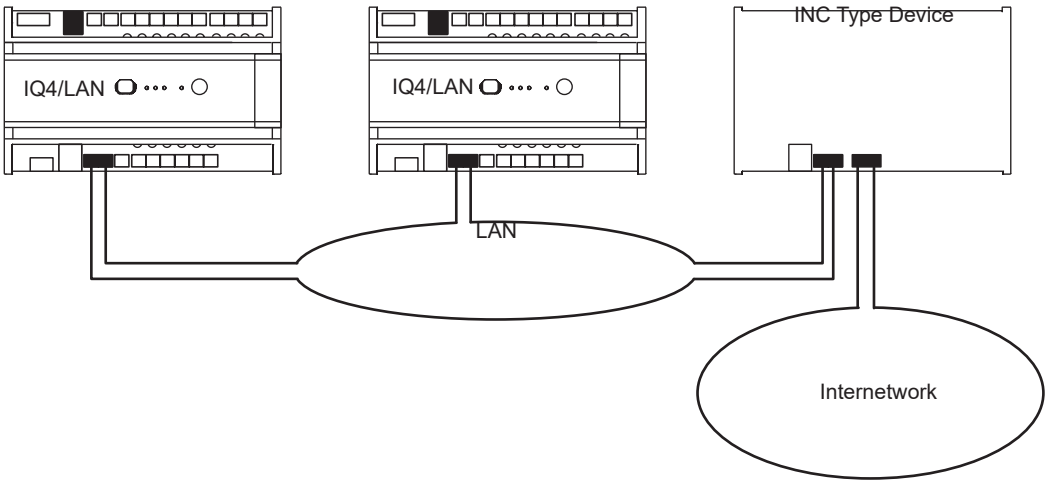
Note: For IQ4NCs their communications mode must be configured so that the current loop connector is functioning as a LAN not an internetwork - see **"IQ4NC Communication Modes" on page 59**.



The IQ4's LAN number is determined by the INC type device on the LAN, and its address on the LAN by its 'Local Address' parameter. It is recommended that the IQ4s 'Local LAN' parameter is set up for use in alarm messages.

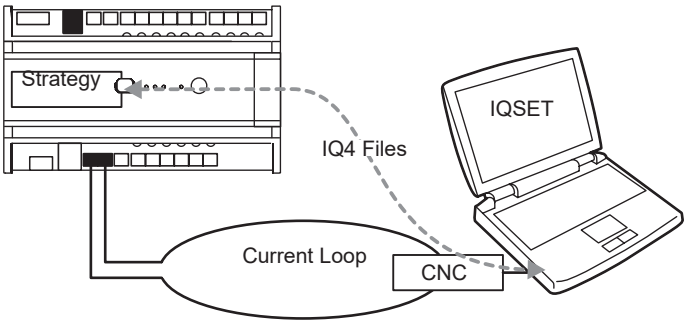
Note: The IQ4/LAN with Trend current loop enabled cannot be part of a Trend LAN or internetwork on Ethernet, and cannot communicate over Ethernet using IC comms; but all other types of Ethernet communication operate. If the IQ4/LAN has the Trend current loop disabled then the current loop connection will not function, and the IQ4 can then be part of a Trend LAN or internetwork over Ethernet, and can communicate over Ethernet using IC communications.

The LAN can be connected to an internetwork (e.g. on Ethernet or Trend current loop) using an INC type device.



5.6.2 Upload/Download of Configuration Files

The strategy file (IQ4 file), but no other strategy files may be uploaded/downloaded over the Trend current loop network (only if baud rate > = 9k6).



5.6.3 Building an Internetwork on a Trend Current Loop Network

IQ4NCs will build an internetwork with other INC type devices connected to the same Trend current loop network, providing their communications mode is configured so that the current loop connector is functioning as an internetwork not a LAN - see **“IQ4NC Communication Modes” on page 59**, and the baud rate is the same. This should not be an issue with IQ4NCs because they will automatically detect the baud rate.

The IQ4NC’s ‘Local Lan’ parameter determines the LAN number of the LAN it is connecting to the internetwork, its vINC address on the LAN is always 126.

5.7 MS/TP Trunk Communications

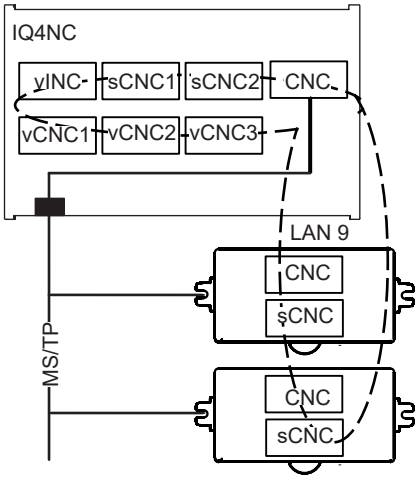
The MS/TP communications are only available on IQ4NCs. The IQ4NC's MS/TP communications operate via the MS/TP connector which is provided for the optional connection to an MS/TP trunk. Connection to an MS/TP trunk enables:

Building LANs on MS/TP Trunk
Connection of MS/TP LAN to Internetwork

Note: Other non-Trend MS/TP devices can be added to the trunk, but their presence may compromise the physical network loading and bandwidth. It is recommended that there can be the IQ4NC with up to 64 IQeco controllers or other BACnet devices on the BACnet trunk. Care must be taken to avoid a BACnet MAC address clash. Master devices are required to use BACnet addresses in the range 0 to 127, and slave devices are required to use BACnet MAC addresses 128 to 254 (IQ4NC and IQecos are master devices). The IQ4NC's internal nodes (CNC, vCNCs, sCNC) should always use addresses in range 1, 4 to 9.

5.7.1 Building LANs on MS/TP Trunk

If an IQ4NC is connected to an MS/TP trunk that contains other Trend MS/TP devices (e.g. IQecos), then a Trend LAN on MS/TP will form to include these devices and their internal nodes. The IQ4NC provides the INC functionality for the LAN and its 'Local Lan' parameter determines the LAN number of this LAN. By default the IQ4NC has its 'Local Lan' parameter set to '9'. The IQ4NC vINC is always at address 126 on the LAN.

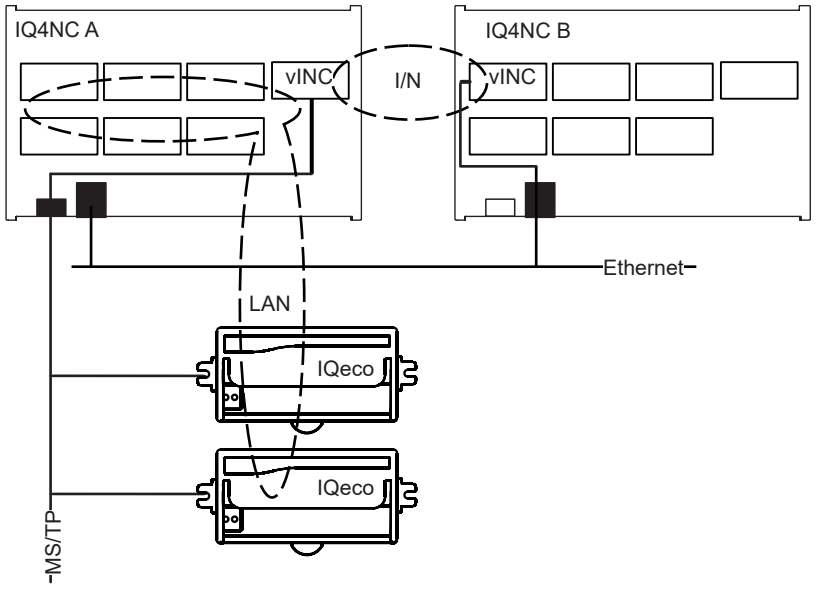


The diagram above shows the IQ4NC building a LAN on the MS/TP trunk. The IQ4NC is using default address settings and therefore the LAN number is '9'. The IQ4NCs address on the LAN is 126.

Note: There can only be one LAN on the MS/TP trunk; there cannot be another INC type device on the MS/TP trunk. The communications mode must be configured so that the MS/TP connector is functioning as a LAN - see **"IQ4NC Communication Modes" on page 59**.

5.7.2 Connection of MS/TP LAN to Internetwork

Once an IQ4NC has built a LAN with other Trend MS/TP devices it enables that LAN to be connected to an internetwork on either an Ethernet, or Trend current loop network. Its vINC joins the LAN to the internetwork.



The diagram above shows the IQ4NC connecting an Ethernet internetwork to an MS/TP LAN. The IQ4NCs form an internetwork on Ethernet connected via their vINCs. IQ4NC A is configured to operate in Ethernet to MS/TP Mode mode and connects to the MS/TP trunk and it forms a LAN with the IQeco's. It connects that LAN to the internetwork via its vINC enabling communication with devices on other LANs.

Note: If communicating with a Trend MS/TP trunked device through a BACnet router, communication with the Trend MS/TP trunked device is only possible using BACnet protocol. Trend communications will not work unless the router is an IQ4NC.

5.8 IQ4NC Communication Modes

The IQ4NC enables Trend network on Ethernet, Trend current loop, or MS/TP trunks to be joined. The IQ4NCs Network module web page facilitates the configuration of the IQ4NC into one of four standard communication modes:

Ethernet to MS/TP Mode
Trend Internetwork to MS/TP Mode
Ethernet to Trend LAN Mode
Ethernet to Trend Internetwork Mode

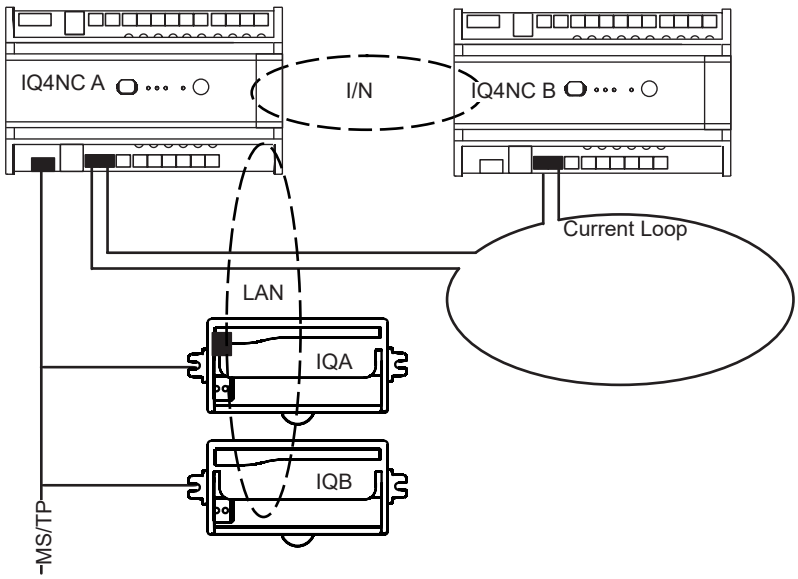
It is also possible to manually configure the communication mode to one other than the four standard ones - see **"Manual Communication Mode" on page 61**.

5.8.1 Ethernet to MS/TP Mode

In Ethernet to MS/TP mode the IQ4NC acts as an interface between an internetwork on Ethernet and a LAN on the MS/TP trunk. Its permanent INC functionality enables it to join an internetwork on Ethernet and extend its own LAN onto the MS/TP trunk, an example is shown on **page 59**.

5.8.2 Trend Internetwork to MS/TP Mode

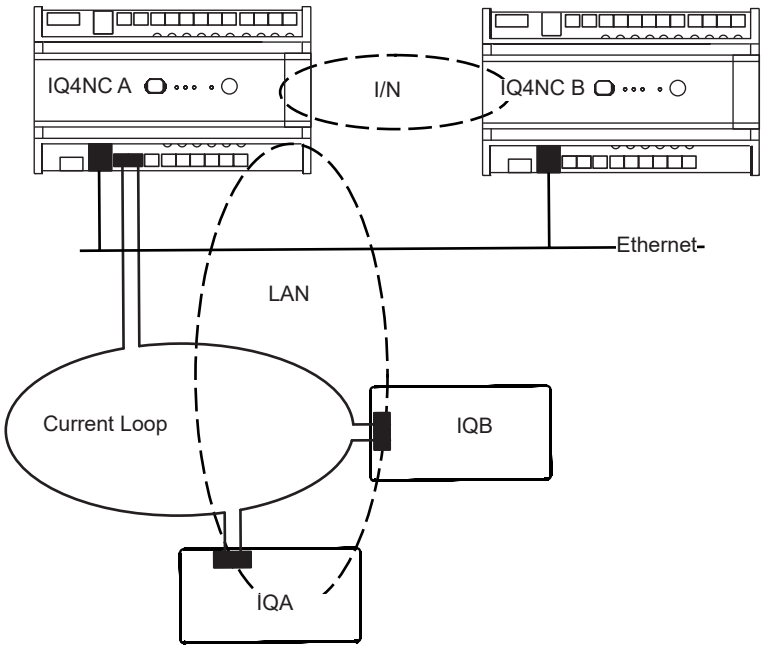
In Trend Internetwork to MS/TP mode the IQ4NC acts as an interface between an internetwork on a Trend current loop network and a LAN on the MS/TP trunk. Its permanent INC functionality enables it to join an internetwork on the Trend current loop network and extend its own LAN onto the MS/TP trunk.



In the above diagram the IQ4NCs form an internetwork on the Trend current loop network. IQ4NC A forms a LAN on the MS/TP trunk with IQA and IQB and connects it to the internetwork. The IQ4NC's Local LAN is the MS/TP LAN.

5.8.3 Ethernet to Trend LAN Mode

In Ethernet to Trend LAN mode the IQ4NC acts as an interface between an internetwork on Ethernet, and a LAN on the Trend current loop network. Its own LAN is extended onto the Trend current loop network.

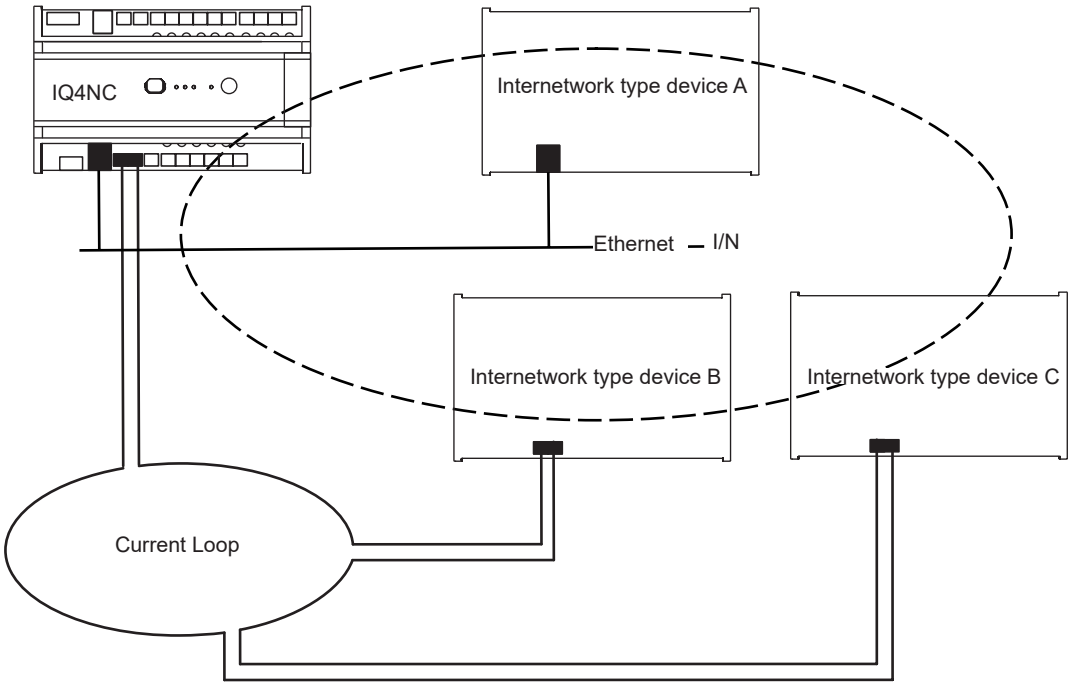


In the above diagram IQ4NC A and IQ4NC B have built an internetwork on Ethernet. IQA and IQB and the IQ4NC A form a LAN on the Trend current loop network. The IQ4NC's Local LAN is the Trend current loop LAN. The IQ4NC connects the Ethernet internetwork to the Trend current loop LAN.

Note: That when operating in this mode it is not possible to have another internetworking device (e.g. XTEND) on the Trend current loop LAN.

5.8.4 Ethernet to Trend Internetwork Mode

In Ethernet to Trend Internetwork mode the IQ4NC acts as an interface between an internetwork on Ethernet, and an internetwork on the Trend current loop network.



In the above diagram the IQ4NC and internetwork type device A are Trend Ethernet networked devices and internetwork type devices B and C are Trend current loop networked devices. The IQ4NC builds an internetwork on the Ethernet network with internetwork type device A. It also builds an internetwork on the Trend current loop network with internetwork type devices B and C thus building an internetwork on both the Ethernet network and Trend current loop network.

5.8.5 Manual Communication Mode

When in manual communication mode the IQ4NC can be set to comminution modes other than the four standard ones.

Caution should be taken when using this mode as although the interface will prevent invalid combinations being configured its is easy to make an error and incorrectly configure the IQ4NC.

An example of a communications mode that can be configured in this way is Ethernet to Trend current loop Internetwork with an MS/TP LAN.

6 BACNET COMMUNICATIONS

All versions of IQ4 can be purchased with or without BACnet interoperability, IQ4NCs and /XNC variants always include it, although it can be disabled.

All IQ4s that support BACnet are capable of **BACnet IP Communications** (i.e. BACnet communications using IP addressing on Ethernet). The IQ4NC is also capable of BACnet MS/TP communications (i.e. BACnet communications over an MS/TP trunk) - see **"BACnet over MS/TP" on page 69**.

IQ4s that support BACnet are certified as BACnet Building Controllers (B-BC). For full details of the objects, properties, and BIBBS supported by the /BAC variants see the IQ4 BACnet Protocol Implementation Conformance Statement (TP201248) and for IQ4NCs see the IQ4NC/.../ BACnet Protocol Implementation Conformance Statement (TP201286).

Important

BACnet is an insecure protocol therefore in the interests of security Trend has initially disabled the following BACnet services:

Write property, Write property multiple
Reinitialise device
Backup/Restore

This means that by default the controller is not BTL compliant.

If these services are required the individual service should be enabled as necessary. Trend recommends that the Reinitialise Device and Backup/Restore services are disabled again after use.

Write property services will need to be enabled if a BACnet client requires to do anything other than simple monitoring e.g maintaining schedules, maintenance of Trend logs.

6.1 Compliance with BACnet Testing Laboratory Requirements

For compliance with the requirements of the BACnet standard, the IQ4 must be configured as described in **"Creating a BTL Compliant Strategy" on page 117**.

6.2 BACnet IP Communications

IQ4s that support BACnet are able to communicate with other BACnet devices over an Ethernet network. This allows them to access/adjust information in BACnet devices, send alarms to BACnet devices and for BACnet clients to access/change data in the IQ4. They can be viewed both as a Trend device, and as a BACnet device because they perform both Trend and BACnet communications.

IQ4s that support BACnet have the following additional modules and module types:

Network module 3 - Type 5 (BACnet IP)
Network module 6 - Type 13 (BACnet Application)
NTD (Non-Trend Device) module(s)
IC comms module(s) BACnet protocol type
Alarm Destination Module - Type 4 (BACnet)
Option module (BACnet type)

For details of these modules see **"IQ4 Modules" on page 165**.

The Type 5 (BACnet IP) Network module is enabled by default. It must be enabled for any of the Ethernet BACnet communications to operate. The other additional modules and types listed above can be configured in IQSET, but will not be operational until the Type 5 (BACnet IP) Network module is enabled and both the Type 5 (BACnet IP) and Type 13 (BACnet Application) Network modules correctly configured. Their default configuration will enable satisfactory operation in most cases.

IQ4s that support BACnet provide the following BACnet functionality:

BACnet COV Reporting and Receiving
BACnet Alarms
Trend Log Object
BACnet Schedule Object
Analogue and Binary BACnet Objects
Backup and Restore
BACnet User Alarm Acknowledgments

Note: The IQ4 does not support multistate objects, but it is able to write to multistate objects in BACnet devices that support them via ic comms.

6.2.1 BACnet COV Reporting and Receiving

BACnet Change Of Value (COV) services allow a COV client to receive reports from a COV server when the value of a referenced property changes.

6.2.1.1 COV Reporting

IQ4's COV reporting corresponds to BIBB - Data Sharing-COV-B. The IQ4 will send notifications to other devices when the selected values change by more than a specific amount. The IQ4 acts as the COV server, the client (BACnet device) must subscribe to the service to receive the notifications. From v3.43 the maximum number of COV subscriptions has increased from 400 to 1000 subject to limitations - see **"IC Comms/COV Subscription Limitations" on page 281**.

The IQ4 has COV reporting capability on its Analogue and Binary Inputs, Outputs, and Values (i.e. AI, BI, AO, BO, AV, BV). These map to Type 0 (Analogue) Sensor, Type 2 (Digital) Sensor, Digital Input, Knob, Switch and Driver Trend IQ4 strategy modules - see **"BACnet Properties Supported" on page 69**. The objects with analogue values each have a 'COV Increment' property (Trend 'COV Increment' parameter) so that the report is given when the change in value since the last report exceeds the COV increment. For binary properties the report is sent on a change of state.

Note: The 'COV Increment' parameter must not be set to '0'.

Note: The COV report is not sent as a Trend alarm, and does not appear in the Alarm Log.

6.2.1.2 COV Receiving

IQ4's COV receiving allows the IQ4 to act as the COV client to receive either unsolicited COV reports (BIBB - Data Sharing-COV-Unsolicited-A), or solicited reports (BIBB - Data Sharing-COV-A) from the server (via the NTD module). Both solicited and unsolicited reports use Data From IC communications modules (Type 7 (From Analogue), Type 8 (From Byte), Type 9 (From Bit)) set to communicate over BACnet to implement the service.

The IQ4 can receive reports from analogue, binary, and multi-state objects (i.e. AI, BI, MI, AO, BO, MO, AV, BV, MV). These map to Type 0 (Analogue) Sensor, Type 2 (Digital) Sensor, Digital Input, Knob, Switch and Driver IQ4 modules - see **"BACnet Properties Supported" on page 69**. The property sent to the IQ4 is always the object's 'Present Value'. The server sets the 'COV increment' (used for analogue values) so Trend have no control over this, and the server will send the report when the change in value since the last report exceeds the 'COV increment'. For binary properties the report is sent on a change of state.

Receiving COV Reports Unsolicited: In this case the BACnet Data From IC communications module is set up with COV Mode enabled and the 'Retransmission Interval' parameter set to '0'. This causes the IQ4 to listen for the required values being broadcast by the server.

Receiving COV Reports Solicited: In this case the BACnet Data From IC communications module is set up with COV Mode enabled and the 'Retransmission Interval' parameter set to non-zero. The IQ4 will subscribe to the COV server which will cause the COV server to send the value to the IQ4, as an unconfirmed message, on change of value (or change of state) until the retransmission interval (BACnet Lifetime) expires. The IQ4 will re-subscribe before the retransmission interval expires to keep the subscription going. Disabling the IC comms module or disabling its COV Mode will cause the IQ4 to unsubscribe to the service.

6.2.2 BACnet Alarms

An IQ4 that supports BACnet can send its Item alarms to BACnet devices over the BACnet network as standard BACnet alarms. The IQ4's item alarm and alarm generation corresponds to BACnet intrinsic reporting - see **"Intrinsic Reporting" on page 65**, and sending the alarm via the alarm handling strategy corresponds to the BACnet Notification Class object - see **"BACnet Notification Class Object" on page 568**.

6.2.2.1 Intrinsic Reporting

Intrinsic reporting allows a BACnet device to provide one or more event sources (modules in Trend terms) for that device, that generate event notifications (alarms in Trend terms). These may be directed to one or more destinations using a Notification Class object. In the IQ4, intrinsic reporting of events corresponds to the generation of Item alarms, The mapping between Trend Item alarms and intrinsic events is described in the following table:

Trend			*BACnet*		*Comments*
Module	***Alarm***	***Code***	***Object***	***To State***	
Sensor	High	HIGH	Analogue Input	OffNormal	
	Low	LOW	Analogue Input	OffNormal	
	Out of Limits	OUTL	Analogue Input	Fault	
	Read	READ	Analogue Input	Fault	
Digital Input	Status (Required state is 1)	DI = 0	Binary Input	OffNormal	
	Status (Required state is 0)	DI = 1	Binary Input	OffNormal	
Plot	Buffer Ready	BBUF	Trend Log	Normal	BBUF has no Clear alarm reporting

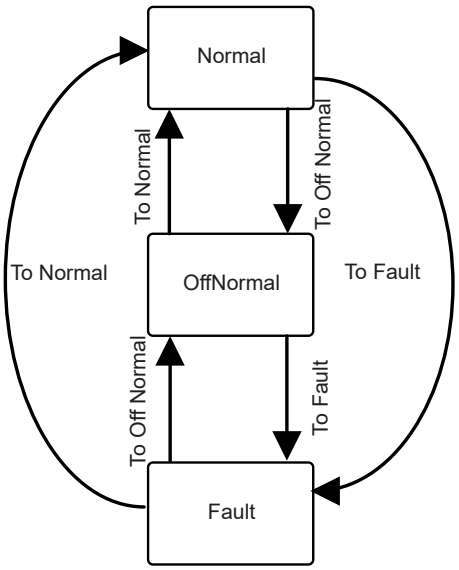
Trend Item alarms have corresponding Clear alarms but the actual corresponding BACnet events are dependent on the return path through the Trend alarm states - see **"Alarm State Transitions" on page 65**.

6.2.2.2 Alarm State Transitions

In Trend protocol each alarm has a clear alarm (excluding the Plot module BBUF event) and each is treated independently, but in the BACnet protocol the alarms follow paths between Normal, OffNormal, and Fault states. There are three different alarm state transitions:

To OffNormal	The value has changed to one that is not in the normal range.
To Fault	The value has changed to one that is the in fault range.
To Normal	The value has returned to the normal range.

The BACnet alarm also distinguishes the starting condition so there are two 'To OffNormal' alarms, 'Fault To Off Normal' and 'Normal To OffNormal'. There are also possible alarms within the transition states so a Trend change of state from OUTL to OUTL + READ will be represented by a BACnet alarm 'Fault to Fault'.



The possible BACnet transition paths, as shown in the diagram above, are directly enabled/disabled by the Alarm Destination module's 'Transition' parameter.

The way in which these transitions correspond to the various Trend alarms is shown in the table below.

Trend Alarm	*BACnet Alarm*
Sensor High	Analogue Input object to OffNormal
Sensor Low	Analogue Input object to OffNormal
Sensor Out of Limits	Analogue Input object to Fault
Sensor Read	Analogue Input object to Fault
Digital Input Status	Binary Input to OffNormal
Plot Buffer Ready	Trend Log to Normal

For details of the alarm state transitions for each module see **"Sensor BACnet Alarm State Transitions" on page 457**, **"Digital Input BACnet Alarm State Transitions" on page 196**, and **"Plot Module BACnet Alarm State Transitions" on page 435.**

6.2.3 Trend Log Object

An IQ4 that supports BACnet can make Type 3 (Periodic) and Type 4 (COV) Plot modules available to a BACnet device - see **"BACnet Trend Log Object" on page 571**.

6.2.4 BACnet Schedule Object

An IQ4 that supports BACnet can make Time Schedule modules available to a BACnet device - see **"BACnet Schedule Object" on page 569**.

6.2.5 Analogue and Binary BACnet Objects

An IQ4 that supports BACnet can make its analogue and digital points available to a BACnet client as analogue input, analogue output, analogue value, binary input, binary output, and binary value BACnet objects - see **"BACnet Properties Supported" on page 557**.

6.2.6 Backup and Restore

IQ4's BACnet backup and restore functionality complies with BIBB DM-BR-B,DM-RD-B. It enables a client to backup or restore an IQ4 over BACnet.

Note: If BACnet Backup/Restore is to be used the 'BACnet Writable Properties' and BACnet Backup/Restore' parameters must be set to enabled. The BACnet Services Password must be set up, this can only be done if the controller contains Type 0 User Modules with a user level of 99).

BACnet is an insecure protocol, therefore in the interests of security Trend has disabled certain BACnet services by default - see **"BACnet Communications" on page 63**. These can be enabled if required.

6.2.6.1 Backup

When the backup procedure is started, the following files are included in the backup and transferred to the client:

Strategy File.
Backdrop files that the strategy file references
XNC directory (/XNC variants only)
The content of the Languages directory

The files included in the backup will be made available in the controller as a single compressed and encrypted file.

Performing a BACnet Backup:

1. Ensure BACnet Services Password is set in the controller. Remember the password and store it securely for your own records, as the controller has no way to remind you of it later.
2. Start the backup using your preferred BACnet client.
3. When prompted enter the BACnet Services Password.

Important: Remember the BACnet Services Password it is required to restore the backup.

6.2.6.2 Restore

Restoring a Backup

The IQ4's BACnet restore functionality will restore a backup.

Restore a BACnet Backup:

1. Ensure you have the backup file and the BACnet Services Password used when it was created.
2. If restoring the backup to a different controller ensure that the controller's device ID (BACnet Application Network module's 'Device Instance' parameter) and controller type match the controller that the backup file was taken from. Also ensure that you configure the controller's BACnet Services Password to be the same as the one used when the backup was created and the controller's security is enabled.
3. Start the restore using your preferred BACnet client.
4. When prompted enter the BACnet Services Password.

6.2.7 BACnet User Alarm Acknowledgments

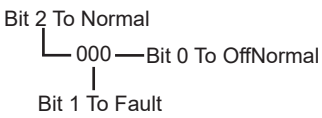
IQ4's BACnet alarm user acknowledgments enables the IQ4 to send BACnet alarms to a BACnet client and expect notification when it is acknowledged by a user on the BACnet client (BIBB - AE_ACK_B).

The following IQ4 modules support BACnet user alarm acknowledgments:

Type 1 (Internal) Digital Input
Type 2 (External) Digital Input
Type 0 (Analogue) Sensor
Type 1 (Internal Analog)

IQ4's BACnet alarm user acknowledgment functionality is enabled/disabled using the 'User ACK Enable' parameter in the Alarm Group module the alarm is connected to (all alarms that are connected to the group are treated the same).

The 'User ACK Enable' parameter contains 3 bits, one for To OffNormal, one for To Fault and one for To Normal alarms.

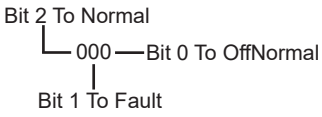


This enables user alarm acknowledgement to be enabled/disabled for each type of alarm. The appropriate bit is set to '1' to enable user alarm acknowledgment.

When an alarm condition is generated by the controller it is sent to each BACnet destination as configured in the BACnet Alarm Destination module (recipient list) and the appropriate bit in the 'ACK Transitions' parameter of the module generating the alarm set to '0' to indicate that a user acknowledgement is required.

If the user acknowledges the alarm from the BACnet client then Acknowledgement alarm will be sent by the BACnet client to the controller. Once the controller receives the Acknowledgement alarm, and the appropriate bit in the 'ACK Transitions' parameter set to '1' to indicate that the alarm has been acknowledged by a user and the controller will send an Acknowledgement notification to all BACnet clients as configured in the recipient list.

The 'ACK Transitions' parameter contains 3 bits: one for To OffNormal, one for To Fault and one for To Normal alarms.



Each bit will be '1' if either no Alarm of the associated type has occurred or if it has been ACKed by a user and set to '0' when an alarm of this type is generated.

6.2.8 Writes over BACnet

IQ4 allows its parameters to be written to over BACnet. If required this feature can be disabled using the 'BACnet Writable Properties' parameter.

To provide the most secure system if BACnet communications are required it is recommend that this feature is disabled using the Type 13 (BACnet Application) Network module's 'BACnet Writable Properties' parameter. However, this will mean that the strategy created is not BTL compliant.

BACnet is an insecure protocol, therefore in the interests of security Trend has disabled certain BACnet services by default, these can be enabled if required - see **"BACnet Communications" on page 63**.

6.2.9 Create/Delete Service

IQ4 allows the creation/deletion of Calendar modules by a BACnet client via the BACnet Create/Delete service. This feature can be enabled/disabled using the 'Create/Delete Service' parameter.

To provide the most secure system if BACnet communications are required it is recommend that this feature is disabled using the Type 13 (BACnet Application) Network module's 'Create/Delete Service' parameter.

6.2.10 BACnet with Automatic IP Addressing

When using IQ4's on BACnet with automatic IP addressing (e.g. DHCP) there are several things to remember.

BACnet addressing is performed in one of two ways:

Device Instance
BACnet Address

Device Instance: When the Device Instance is used the IQ4 is addressed on the BACnet network using its 'Device instance' parameter. This is similar to using a host name on Ethernet.

BACnet Address: When a BACnet Address is used the IQ4 is addressed using a combination of the BACnet 'Network Number', 'IP Address' and 'UDP port' parameters.

If the IQ4 is set up for DHCP the Device Instance will not change, but from time to time its BACnet Address will (due to change in IP address).

The IQ4's BACnet Alarm Destination modules and the NTD modules both allow the recipient's address to be selected using, either Device Instance, or by BACnet Address (Network Number and MAC Address). It is recommended that the Device Instance be used, and if the remote devices use automatic IP addressing, the Device Instance addressing must be used.

Note: Certain BACnet devices are unable to support Device Instance addressing and so must have static IP addresses.

The Type 5 (BACnet IP) Network module has a 'BBMD' parameter which enables or disables BBMD functionality. This functionality will only work correctly if all the BBMDs (including the IQ4) use static IP addressing.

Remote devices in the BACnet system communicating with the IQ4 which have to be explicitly configured with the MAC Address of the device they are communicating with (Static Device Binding) will no longer communicate with the IQ4 if its IP address changes due to DHCP operation until their configuration is changed. In this case the IQ4 would have to have fixed IP addressing.

Remote devices in the BACnet system communicating with the IQ4 which resolve the BACnet MAC address at run-time (Dynamic Device Binding) will be able to resolve the change in MAC address, but may themselves take a while to detect that the address has changed (this time depends on the local algorithm used in the device for responding to failed communication).

In summary, if there are devices which only support static device binding in the BACnet system, or devices with unacceptable delays in detecting MAC address changes then the IQ4s must be set up with static IP addresses in order to ensure reliable communications are maintained.

6.2.11 BACnet Properties Supported

Each BACnet device has a number of objects (roughly corresponding to Trend modules), and each object has a number of properties (roughly corresponding to Trend module parameters). The table below is a summary of how BACnet objects map to Trend modules in the IQ4.

BACnet Object	*Trend Module(s)*
Analogue Input Type A	Type 0 (Analogue) Sensor
Analogue Input Type B	Type 2 (Digital) Sensor
Analogue Output	Type 2 (Analogue), Type 3 (Time Proportional), Type 4 (Raise Lower with End Stop), Type 5 (Binary Hysteresis), Type 6 (Time Proportional + Override), Type 7 (Raise Lower Continuous), Type 8 (Multistage Digital) Driver.
Analogue Value	Knob
Binary Input	Digital Input
Binary Output	Type 1 (Digital) Driver
Binary Value	Switch
Calendar Object	Calendar
Device	Address, Program, Network, and Time
File Object	No mapping to Trend module, BACnet only.
Notification Class (Recipient)	Alarm Group, Alarm Route, and Alarm Destination
Schedule	Time Schedule
Trend Log Object	Type 3 (Periodic) and Type 4 (COV) Plot

E.g. BACnet analogue input 1 (AI1) is equivalent to Trend Sensor module 1 (S1).

The Notification Class object relates to alarm delivery, and has a Recipient List (list of alarm destinations) in which each recipient maps to Alarm Route, and Alarm Destination modules.

The number of BACnet schedules is the same as the number of Time Schedule modules. Eight notification classes are supported.

There is no limit on the number of trendlog object instances, however the total number of values that can be logged depends upon number of log points available in the controller - any value logged by a Trend Plot module will be logged by a BACnet trendlog object.

The supported BACnet properties are explained in **"BACnet Properties Supported" on page 557**.

6.3 BACnet over MS/TP

IQ4NCs in Ethernet Internetwork to MS/TP LAN or Current Loop Internetwork to MS/TP LAN modes are capable of both BACnet over IP, and BACnet over MS/TP communications.

IQ4NCs have the BACnet capability of the /BAC variants over the Ethernet network, but have additional capability to enable them to act as an interface to an MS/TP network.

IQ4NCs are able to communicate with other BACnet devices over the MS/TP trunk. This allows them to access/adjust information in BACnet devices on MS/TP, and for BACnet devices to access/change data in the IQ4NC. They can be viewed both as a Trend device, and as a BACnet device because they perform both Trend and BACnet communications.

When operating in Ethernet Internetwork to MS/TP LAN or Current Loop Internetwork to MS/TP LAN mode IQ4NCs have the following additional modules:

Network module 5 - Type 12 (BACnet MSTP)
Option (MSTP LAN Card)
Option (INC)

For full details of the objects, properties, and BIBBS supported by the IQ4NC - see the IQ4NC/.../ BACnet Protocol Implementation Conformance Statement (TP201286).

7 ALARMS

An alarm condition is generated by an IQ4 when a particular plant or self checking parameter is not in the required state or is outside the defined range. The alarm condition can be used by the strategy, monitored on a local display, and transmitted as an alarm message to another device (e.g. Supervisor) to alert a system operator. Alarms are also stored in the IQ4's Alarm Log.

7.1 Alarm Types

There are three types of alarm:

Network Alarms
General Alarms
Item Alarms

7.1.1 Network Alarms

The IQ4 generates network alarms when it detects a change in the communication status of the system. There are several different types of network alarms:

Local LAN Alarms
Internetwork Alarms
Local Device Alarms
Duplicate Address Alarms
Deaf Ethernet Node Alarm
DHCP Alarms

Network Alarm Reporting

The IQ4 sends any network alarms it generates to it's own local supervisor port (RS232), engineering port (USB), and vCNC where it can be processed by any device e.g. supervisor connected to the port.

> Important: This means that if nothing is connected to the either the local supervisor port (RS232), engineering port (USB), or vCNC the alarm will be lost.

For IQ4NCs the alarms are also put in the Alarm Log.

Note: The IQ4NC has additional functionality which enables it to send network alarms to the IQ4's local supervisor port (RS232), engineering port (USB), and vCNC; it can also use the alarm handling strategy (Alarm Group, Alarm Route and Alarm Destination modules) if the 'Alarm Group' parameter in the appropriate Network module is set to non-zero. There are separate 'Alarm Group' parameters for Ethernet IP, current loop, and BACnet MSTP networks. The use of Alarm Group modules is explained in section **"Alarm Handling Strategy" on page 79**.

7.1.1.1 Local LAN Alarms

There are three different Local LAN alarms:

LAN Changed (NKCH)
LAN Broken (NKBK)
LAN OK (NKOK)

Note: These alarms do not give details of which devices are added to or removed from the network.

7.1.1.1.1 Local LAN Alarm Generation

LAN Changed alarm: A LAN Changed alarm is generated if a device is added to or removed from the LAN. When an IQ4 detects the next or previous device has changed, it causes the LAN to be rebuilt. Once the LAN has been successfully rebuilt the IQ4 will generate a LAN Changed alarm and report it as described in **"Network Alarm Reporting" on page 71**.

LAN Broken alarm: A LAN Broken alarm is generated if an IQ4 detects that it is the only device on the LAN i.e there is no next or previous device. The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

LAN OK alarm: A LAN OK alarm is generated if an IQ4 detects another Trend device on its LAN, i.e there is a next or previous device. The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

For a description of how the IQ4 knows what the next and previous device is see **"What is the Next/Previous Device" on page 75**.

Note: When operating on MS/TP the IQ4 does not know what the previous device is so Local LAN alarms are only generated when the next device changes.

7.1.1.1.2 Format of Local LAN Alarms

LAN Changed Alarm Format

Ethernet

Local LAN Reported by IQ4 IP LAN Changed <CR>
NKCH

Current Loop

Local LAN Reported by IQ4 IQLAN Changed <CR>
NKCH

MS/TP

Local LAN Reported by IQ4 MSTP LAN Changed <CR>
NKBK

LAN Broken Alarm Format

Ethernet

Local LAN Reported by IQ4 IP LAN Broken <CR>
NKBK

Current Loop

Local LAN Reported by IQ4 IQLAN Broken<CR>
NKBK

MS/TP

Local LAN Reported by IQ4 MSTP LAN Broken <CR>
NKBK

LAN OK Alarm Format

Ethernet

Local LAN Reported by IQ4 IP LAN OK <CR>
NKOK

Current Loop

Local LAN Reported by IQ4 IQLAN OK <CR>
NKOK

MS/TP

Local LAN Reported by IQ4 MSTP LAN OK <CR>
NKOK

7.1.1.2 Internetwork Alarms

There are three different internetwork alarms:

Internetwork Broken (NKBK)
Internetwork Changed (NKCH)
Internetwork OK (NKOK)

Note: These alarms do not give details of which IQ4s are added to or removed from the network.

7.1.1.2.1 Internetwork Alarm Generation

Internetwork Changed alarm: An Internetwork Changed alarm is generated if a device is added to or removed from the internetwork. When an IQ4 detects the next or previous device has changed, it causes the internetwork to be rebuilt. Once the internetwork has been successfully rebuilt the IQ4 will generate an Internetwork Changed alarm and report it as described in **"Network Alarm Reporting" on page 71**.

Internetwork Broken alarm: An Internetwork Broken alarm is generated if an IQ4 detects that it is the only device on the internetwork i.e there is no next or previous device. The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

Internetwork OK alarm: An Internetwork OK alarm is generated if an IQ4 detects another Trend device on its internetwork (there is a next or previous device). The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

For a description of how the IQ4 knows what the next and previous device is see **"What is the Next/Previous Device" on page 75**.

Note: When operating on MS/TP the IQ4 does not know what the previous device is so Internetwork alarms are only generated when the next device changes.

7.1.1.2.2 Format of Internetwork Alarms

Internetwork Changed Alarm Format

Ethernet

Internetwork Reported by IQ4 IP Iwrk Changed <CR>
NKCH

Current Loop

Internetwork Reported by IQ4 IQ Iwrk Changed <CR>
NKCH

Internetwork Broken Alarm Format

Ethernet

Internetwork Reported by IQ4 IP Iwrk Broken <CR>
NKBK

Current Loop

Internetwork Reported by IQ4 IQ Iwrk Broken <CR>
NKBK

Internetwork OK Alarm Format

Ethernet

Internetwork Reported by IQ4IP Iwrk OK <CR>
NKOK

Current Loop

Internetwork Reported by IQ4 IQ Iwrk OK <CR>
NKOK

7.1.1.3 Local Device Alarms

There are two different local device alarms:

Device Dead (DVDD)
Device OK (DVOK)

7.1.1.3.1 Local Device Alarm Generation

Device Dead alarms: A Device Dead alarm is generated if a device (e.g. supervisor) connected to the IQ4's local supervisor port (RS232), engineering port (USB), or vCNC ceases communicating with the IQ4. The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

Device OK alarms: A Device OK alarm is generated if a device which had previously caused a Device Dead alarm starts to communicate again. The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

Note: For IQ4NCs Local Device Alarms cannot be sent using alarm handling strategy.

7.1.1.3.2 Format of Local Device Alarms

Device Dead Alarm Format

Local Lan Reported by IQ4 Device Dead <CR>
DVDD

Device Online Alarm Format

Local LAN Reported by IQ4 Device Online <CR>
DVOK

7.1.1.4 Duplicate Address Alarms

There are two different duplicate address alarms:

Duplicate Device (NKDA)
Duplicate LAN (NKDA)

7.1.1.4.1 Duplicate Address Alarm Generation

When the network is mapped, the mapping message is passed around the network from one device to the next. When the IQ4 receives a mapping message it adds its own network address (LAN and device address) and passes it on to the next device. If an IQ4 discovers its own address in the mapping message it generates a 'Duplicate Address' alarm.

Duplicate Device alarms: A Duplicate Device alarm is generated when the IQ4 detects another device on its LAN with the same device address. The alarm is reported it as described in **"Network Alarm Reporting" on page 71**.

Duplicate LAN alarms: A Duplicate LAN alarm is generated when the IQ4 that is providing the INC functionality detects another device on the internetwork with the same LAN number. The alarm is reported it as described in **"Network Alarm Reporting" on page 71**.

7.1.1.4.2 Format of Duplicate Address Alarms

Duplicate Address Alarm Format

Local LAN Reported by IQ4 DUPLICATE ADDRESS <CR>
NKDA

Duplicate LAN Alarm Format

Internetwork Reported by IQ4 DUPLICATE-LAN <CR>
NKDA

7.1.1.5 Deaf Ethernet Node Alarm

The Deaf Ethernet Node alarm (NKHD) only applies when the IQ4 is connected to an Ethernet network.

7.1.1.5.1 Deaf Ethernet Alarm Generation

Deaf Ethernet Node alarms: A Deaf Ethernet Node alarm is generated if the IQ4 detects that an adjacent Trend Ethernet device is sending but not receiving messages over Ethernet. This would probably be due to an Ethernet wiring or connection problem. The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

7.1.1.5.2 Format of Deaf Ethernet Node Alarms

Internetwork xxx.yyy Deaf Ethernet Node <CR>
NKDD

Where xxx and yyy are the last two sets of numbers in the deaf node's IP address.

7.1.1.6 DHCP Alarms

The DHCP alarms only apply when the IQ4 is connected to an Ethernet network and it is configured to obtain its IP addressing information from a DHCP server. There are two different DHCP alarms:

Chgd to DHCP Lklocal (NKAL)
Changed to DHCP auto (NKAA)

7.1.1.6.1 DHCP Alarm Generation

Chgd to DHCP Lklocal alarms: A Chgd to DHCP Lklocal alarm is generated if the DHCP server fails to provide the IP addressing parameters to the IQ4 causing the it to enter link/local mode. The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

Changed to DHCP auto alarms: A Changed to DHCP auto alarm is generated if after failing to provide the IP addressing parameters the DHCP server then provides them to the IQ4 causing it to return to DHCP server operation. The alarm is reported as described in **"Network Alarm Reporting" on page 71**.

7.1.1.6.2 Format of DHCP Alarms

Chgd to DHCP Lklocal Alarm Format

Internetwork xxx.yyy Chgd to DHCP Lklocal <CR>
NKAL

Where xxx and yyy are the last two sets of numbers in the IP address of the device that triggered the change.

Changed to DHCP auto Alarm Format

Internetwork xxx.yyy Changed to DHCP auto <CR>
NKAA

Where xxx and yyy are the last two sets of numbers in the IP address of the device that triggered the change.

7.1.1.7 What is the Next/Previous Device

The order of devices on a LAN or internetwork is important as an individual device on the LAN or internetwork needs to know what the next and previous device is in order to generate Local LAN and Internetwork alarms. The way the IQ4 calculates its next and previous device depends on the media used by the LAN or internetwork.

Ethernet

When the LAN or internetwork is using Ethernet the order of the devices is determined by their IP addresses. The device with the lowest IP Address is the first device in the order, and the device with the highest IP address is the last. The LAN or internetwork loops round at maximum and minimum IP address.

Current Loop

When the LAN or internetwork is using the current loop the order of the devices is determined by the way they are physically wired together.

MS/TP

When the LAN is using MS/TP the order of the devices is determined by their address on the MS/TP network. The device with the lowest address is the first device in the order, and the device with the highest address is the last. The LAN or internetwork loops round at maximum and minimum.

7.1.2 General Alarms

General Alarms are generated by the IQ4 when it detects an alarm condition within its own hardware or program. The following alarm conditions are monitored:

Controller on Line (CONL)
Corrupt Strategy (HELP)
Failed RTC (FRTC)
Sequence Table Overrun (STOR)
Archive Failure (FARC)
File Error (FLER)
SSL Failure (FSSL) (v3.30 or greater)

7.1.2.1 Controller on Line (CONL)

Controller on Line alarms are generated when the IQ4 restarts successfully due to power up or an internal soft reset and loads and runs a valid strategy. They are not cause for concern, unless the frequency is considered to be unreasonable.

7.1.2.2 Corrupt Strategy (HELP)

Corrupt Strategy alarms are sent when there is a checksum failure. It indicates a corrupt strategy has been detected when the IQ4 started due to power up or an internal soft reset. Under these conditions it is unlikely that an alarm will be sent as the strategy will not run. However, the alarm condition can be seen on the Address module's web page. A valid strategy file should be downloaded to clear this status.

7.1.2.3 Failed RTC (FRTC)

Failed RTC alarms are generated if any of the parameters of the Real Time Clock (RTC) are outside the range i.e. the RTC has failed. The RTC is a device that maintains the real time of day and date in terms of day, month and year. It has to be interrogated at regular intervals by the microprocessor in order that this information can be used by the IQ4.

Note: Failed RTC alarms may occur at power up with the RTC in an uninitialised state. If the fail condition remains, the FRTC alarm will be resent once every hour. It does not cause a restart or CONL.

7.1.2.4 Sequence Table Overrun (STOR)

Sequence Table Overrun alarms are generated when the IQ4 is unable to process all the modules in the sequence in the cycle time.

7.1.2.5 Archive Failure (FARC)

Archive Failure alarms are generated if the archive of the running strategy file to flash memory has failed.

7.1.2.6 File Error (FLER)

File Error alarms are generated if the IQ4 is unable to load its current strategy file, or the backup, and has loaded the original downloaded file. It will have lost any changes made since the download (e.g. changes made using web pages).

7.1.2.7 SSL Failure (FSSL) (v3.30 or greater)

SSL Failure alarms are generated if the IQ4 is unable to send an email alarm due to an SSL failure. If the issue is resolved a clear alarm is generated (CSSL).

7.1.2.8 General Alarm Reporting

General alarms are reported via alarm handling strategy (Alarm Group, Alarm Route and Alarm Destination modules). They are sent to the Alarm Group module specified by the Address module's 'General Alarm Group' parameter which is set by default to Alarm Group module 0 which means they are not sent but will be placed in the Alarm Log. For the alarms to be reported the 'General Alarm Group' parameter must be to an Alarm Group module other than '0', and the alarm handling strategy must be set up - see **"Alarm Handling Strategy" on page 79**. The status of the general alarms are available as outputs from the Address module.

7.1.2.9 Format of General Alarms

General alarms are sent using alarm handling strategy in either Coded, Text, Attribute or Email format - see **"Format of Alarms Sent Using the Alarm Handling Strategy" on page 82**.

7.1.3 Item Alarms

Item Alarms are generated by the following IQ4 modules, normally due to a faulty plant condition:

Sensor
Digital Input
Plot
Driver
Loop

The following alarm conditions are checked for by an IQ4.

Module	*Alarm*	*Alarm Code*	*Clear Code*
Digital Input	Digin (Req'd State = 1)	DI = 0	CDI0
	Digin (Req'd State = 0)	DI = 1	CDI1
Driver	Maintenance	MINT	CMNT
	Readback	SDGT	CDGT
	Hand/OFF/AUTO	HOA	CHOA
Loop	PV Fail	PVFL	CPVF
	SP Deviation	SDEV	CSDV
Sensor	High	HIGH	CHIH
	Low	LOW	CLOW
	Out of Limits	OUTL	COUT
	Read	READ	O/K
Plot	Buffer Ready	BBUF	-

Note: The Plot module's BBUF alarm is considered an 'Event' rather than a true alarm, and does not have a corresponding clear alarm.

Item alarms have digital outputs which are available to be used in the strategy. They can be individually disabled using the appropriate alarm 'Enabled' parameter.

7.1.3.1 Item Alarm Reporting

Item alarms are reported via alarm handling strategy (Alarm Group, Alarm Route and Alarm Destination modules). They are sent to the Alarm Group module specified by the source module's 'Alarm Group' parameter for that particular alarm. For the alarms to be reported the 'General Alarm Group' parameter must be to an Alarm Group module other than 0 (Alarm Group module 1 onwards), and the alarm handling strategy must be set up - see **"Alarm Handling Strategy" on page 79**. If an alarm group is parameter is sent to '0' the alarm will be logged, but not sent.

7.1.3.2 Item Alarm Generation

Each item alarm (other than BBUF) is generated by an alarm sub-module embedded into the main module. For example a Sensor module has four alarm sub-modules:

Read
Out of limits
Low
High

One for each of the sensor module's item alarms.

The alarm sub-module sets its 'Alarm State' parameter to '1' when the alarm condition exists for longer than the alarm delay (specified by the 'Alarm Delay(secs)' parameter). It does this regardless of its 'Enable' parameter. The alarm sub-module's 'Alarm State' parameter will be set back to '0' if the alarm clears for longer than the clear alarm delay specified by the 'Clear Alarm Delay(secs)' parameter). The state of this parameter is reflected in a parameter in the main module with the same name as the alarm.

If the alarm sub-module's 'Enabled' parameter is set to '1', its 'Alarm State' parameter is set to '1' the main module's 'In Alarm' parameter is also set to '1' and the alarm generated. The alarm is sent to the Alarm Group module specified by the alarm sub-module's 'Alarm Group' parameter and the alarm sub-module's 'Alarm Sent' parameter is set to '1'.

When the alarm sub-module's 'Alarm State' parameter is set back to '0' the main module's 'In Alarm' parameter is also set to '0' and if the alarm sub-module's 'Clear Alarm' parameter is set to '1' and the clear alarm is enabled a clear alarm is generated. The clear alarm is sent to the Alarm Group module specified by the alarm sub-module's 'Alarm Group' parameter and the alarm sub-module's 'Alarm Sent' parameter is set to '0'.

For details of BBUF alarm generation see **"Threshold Notification" on page 436**.

7.1.3.3 Analogue Alarm Status Bits

Analogue Sensor modules can generate four alarm conditions:

Read
Outside of Limits
Low
High

The status of these alarms is represented by the analogue alarm status bits.

Bit 0 = Status of the High Alarm
Bit 1 = Status of the Low Alarm
Bit 2 = Status of the Out of Limits Alarm
Bit 3 = Status of the Read Alarm

'1' indicates an alarm condition.

Sensor and Analogue Node Modules

If the Sensor module is Type 0 (Analogue) the alarm status bits reflect the validity of the value of the externally connected hardware sensor. They accompany the Sensor module's value as it is transferred to the next module. They will travel through Knob, Function, Loop, and Schedule Offset modules without being exposed although the module can take appropriate action if the status bits indicate that the input is invalid.

The bits are exposed by analogue Sensor modules, and by Analogue Node modules connected to the Sensor module's 'Value' parameter; these modules make the status bits available as outputs to be used in the strategy.

IC Communications Modules

If the alarm bits are attached to a value, and that value is transferred by Trend analogue Data To, Global To, and Data From IC comms, and if the remote point is explicitly set to an analogue node (e.g. set by absolute item), the alarm bits are also transferred. The alarm bits are exposed in the receiving analogue node.

Loop Modules

The Loop module can take a predefined sensor fail action if the input has either a Read or Out of Limits alarm.

Function Modules

The way in which a Function module behaves with respect to alarm bits on its input(s) varies with the Function module type. It may pass on the appropriate alarm bits to accompany its output. Some Function modules will take action if an input has a Read or Out of Limits alarm - see **"Function Module Alarms" on page 241**.

Schedule Offset Modules

A Schedule Offset module will transfer the alarm bits of the selected input to accompany its output.

7.1.3.4 Format of Item Alarms

Item alarms are sent using alarm handling strategy in either Coded, Text Attribute or Email format - see **"Format of Alarms Sent Using the Alarm Handling Strategy" on page 82**.

7.2 Alarm Handling Strategy

Alarm handling strategy enables different alarms to be sent to different destinations, or to determine when the alarms are sent. The alarm handling strategy can send alarms to the following types of destination:

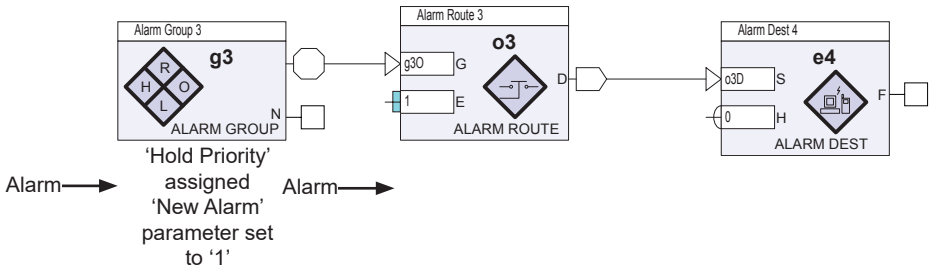
IQ Lan Alarm Destinations
IP Address Alarm Destinations
Email Alarm Destinations
BACnet Alarm Destinations (/BACnet variants and IQ4NC only)

Alarm handling strategy uses the IQ4's Alarm Group, Alarm Route, and Alarm Destination modules to send alarms to the required destination. This alarm handling strategy can also enable an alarm to be sent to another destination if the preferred destination is not available.

Alarms that are to be processed in the same way should be configured to be sent to an Alarm Group module which will process the alarms and send them to the required destination using an Alarm Destination module via an Alarm Route module.

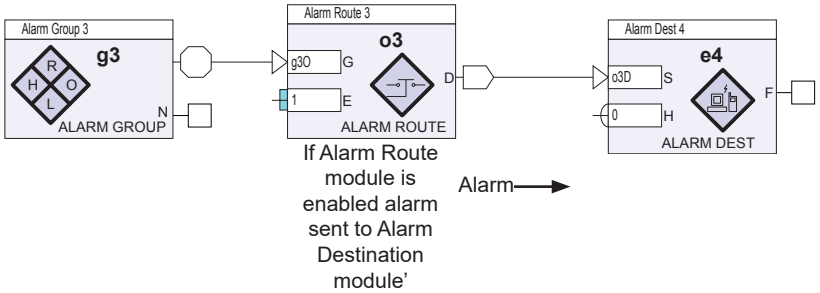
When an alarm condition occurs, the module that generated the alarm places it in the Alarm Log. It is then sent to the Alarm Group module (specified by its appropriate 'Alarm Group' parameter) and sets the Alarm Group module's 'New Alarm' parameter to '1' for one cycle of the sequence table. This could be used by the strategy, for example, to turn on an annunciator whenever an alarm occurs. If alarms are occurring faster than once per sequence table cycle, the 'New Alarm' parameter will have a value of '1' until one sequence cycle after the last alarm occurred.

The Alarm Group module assigns the alarm a hold priority (specified by the Alarm Group module's 'Hold Priority' parameter) and passes it to the connected Alarm Route module.



Note: The 'Hold Priority' parameter can be a specific value, or can be connected to the strategy. Connecting to strategy enables the hold priority level to be lowered when communications demand is lower (e.g. during non-occupation) to ensure that all held alarms are sent.

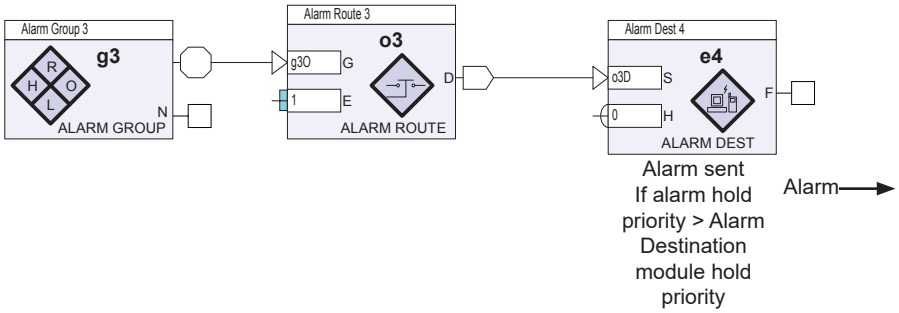
If the Alarm Route module's 'Enable' parameter is set to '1' the alarm will be passed to the specified Alarm Destination module.



Note: The 'Enable' parameter can be a specific value, or can be connected to the strategy. Connecting to strategy enables the destination to which alarms are sent to be changed by the strategy. For example, an alternative destination can be used if the first one fails.

If none of the Alarm Route modules to which the Alarm Group module is connected are enabled, the alarm status in the Alarm Log will be set to ‘Completed’, and the alarm will not be sent to a destination i.e. it is only logged.

When an alarm reaches the Alarm Destination module its hold priority is compared against the Alarm Destination module’s ‘Hold Priority’ parameter. If the priority is below the hold level it will be held. If alarm’s hold priority is above the Alarm Destination module’s hold priority it, and all held alarms, will be sent to the destination specified by the Alarm Destination module. The destination will stay open to low priority alarms for 5 seconds after the last alarm is sent. The Alarm Destination module can only hold the alarm it is trying to deliver (although other alarms may be waiting in a ‘stack’).



If the target device receives the alarm it will send an acknowledgment and when this is received the alarm’s status in the Alarm Log is set to ‘Completed’. If the alarm has been sent to multiple destinations its status in the Alarm Log will only be set to ‘Completed’ once an acknowledgement has been received from all destinations and once ‘Completed’ no further attempts to send the alarm will be made.

7.2.1 Delivery Failure

The Alarm Destination module defines the destination of the alarm and takes the responsibility of delivering the alarm. If an Alarm Destination module is unable to deliver the alarm, after the allotted retries it will set its ‘Failed’ parameter to ‘1’ and continue to try and send the alarm (even if the route to the Alarm Destination module is subsequently disabled); this enables the resumption of communication with the target device to be detected. The status of the ‘Failed’ parameter can be used by the strategy to switch Alarm Route modules to send alarms to alternative destinations.

Once the target device starts acknowledging alarms, the ‘Failed’ parameter is set to ‘0’, and the Alarm Destination module is free to send another alarm.

Note: The Alarm Destination module can only attempt to deliver one alarm at a time, and if its receipt by the target device is not being acknowledged, it is effectively blocked until the acknowledge is received.

7.2.2 IQ Lan Alarm Destinations

The IQ4 can send an alarm over the Trend network to another device on the Trend network using a Trend network address (LAN number and device address) a using Type 1 (IQ Lan) Alarm Destination module - see **“Type 1 (IQ Lan)” on page 174**.

*Note: If a device is permanently connected the alarm can be sent to the network address of the CNC the device is using to connect to the Trend network. If the device is not permanently connected (e.g. using TCP/IP dial up) the alarm can either be addressed to a vCNC in alarm mode (e.g. in an XTEND) that is configured to send the alarm over the IT network to its final destination, or using the IQ4NC’s alarm forwarding facility - see **“INC Alarm Forwarding (IQ4NC only)” on page 91**. Sending alarms to a device not permanently connected is best achieved by sending the alarm to an IP Address.*

7.2.2.1 Format of IQ Lan Destination Type Alarms

IQ Lan Destination Type alarms can be in either Coded, Text Attribute or Email format - see **“Format of Alarms Sent Using the Alarm Handling Strategy” on page 82**.

7.2.3 IP Address Alarm Destinations

The IQ4 can send an alarm over an Ethernet network to a device connected to Ethernet (i.e. a 963 supervisor) using its IP Address or host name, using a Type 2 (IP Address) Alarm destination module - see **"Type 2 (IP Address)" on page 174**. The device that is to receive the alarm must be set up to listen for alarms on the specified port number.

7.2.3.1 Format of IP Address Destination Type Alarms

IP Address Destination Type alarms can be in either Coded, Text Attribute or Email format - see **"Format of Alarms Sent Using the Alarm Handling Strategy" on page 82**.

7.2.4 Email Alarm Destinations

The IQ4 can send an alarm by email to a device configured to receive email using the appropriate email address using a Type 3 (Email) Alarm destination module - see **"Type 3 (Email)" on page 175**.

v3.30 firmware and greater provides secure transmission the email alarms (SSL or STARTTLS) providing the Type 4 (Ethernet IP) Network module's 'Email Port', 'Email Security', and 'Ignore Email Certificate' parameters are set up correctly.

The following parameters have to be set up:

Module	*Parameter*
Type 4 (Ethernet IP) Network module	Email Server Address
	Email Username
	Email Password
	Email Port
	Email Security
	Ignore Email Certificate
	Email Minimum TLS Version
Type 3 (Email) Alarm Destination Module	From Email Address
	To Email Address
	Extra Text

7.2.4.1 Format of Email Destination Type Alarms

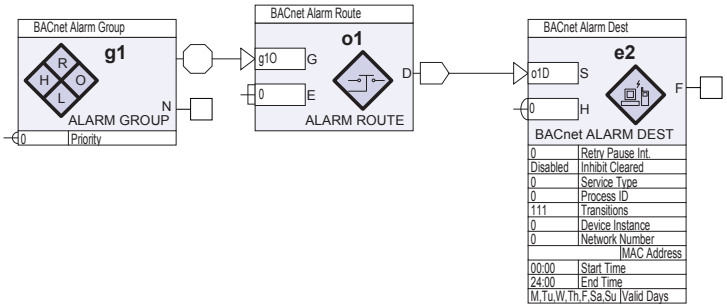
Email Destination Type alarms can only be sent in Email format - see **"Email Format" on page 83**.

7.2.5 BACnet Alarm Destinations (/BACnet variants and IQ4NC only)

The IQ4 can send an alarm to a BACnet device over the BACnet network as a BACnet alarm, using a Type 4 (BACnet) Alarm destination module - see **"Type 4 (BACnet)" on page 176**. The Type 5 (BACnet IP) Network module must be enabled and set up in order to be able to send BACnet type alarms.

Only alarms from Sensor, Digital Input, and Plot modules can be sent as BACnet alarms; if other alarms are sent to a BACnet Device Destination module they will be ignored. This type of alarm reporting is known as intrinsic reporting in BACnet terminology - see **"IP Address Alarm Destinations" on page 81**.

Type 4 (BACnet) Alarm Destination modules can only be added to the strategy as part of IQSET's BACnet Alarm Block, which consists of an Alarm Group, Alarm Route and BACnet Alarm Destination.



This strategy block conforms to BACnet standard BIBB AE-N-I-B and provides compatibility with the 'BACnet Notification Class Object'.

7.2.5.1 Format of BACnet Destination Type Alarms

BACnet Destination Type alarms are sent as BACnet alarms - see **"BACnet Format" on page 84**.

7.2.6 Format of Alarms Sent Using the Alarm Handling Strategy

Alarms sent to IQ Lan and IP Address type destinations using the alarm handling strategy can be sent in one of three formats:

Coded Format
Text Format
Attribute Format

Alarms sent to Email type destinations are sent in email format - see **"Email Format" on page 83**. Alarms sent to BACnet type destinations are sent in BACnet format - see **"BACnet Format" on page 84**.

7.2.6.1 Coded Format

Coded alarms are formated as below. They require interpretation by a Trend supervisor/tool or display to be understood by a user.

<Module Type><Module Number><Alarm Code><Time>

Where <Module Type> is the 'module identifier' (e.g. S is used for Sensor modules). <Module Number> is the hexadecimal representation of the module number. <Alarm Code> is the alarm's alarm code. The alarm codes are listed in **"General Alarms" on page 76** and **"Item Alarms" on page 77**. <Time> is the time the alarm occurred.

E.g.

SCHIGH17.48

Specifies a sensor HIGH alarm sent from sensor module 12 at 17:48.

7.2.6.2 Text Format

Text format alarms are formatted as below. They are in a format that can be read by a user.

<Controller Label><Item Label><Alarm Label><Transition><Time><Date>O<Device Address><Module Type><Module Number><Alarm Code>

Where <Controller Label> is the identifier of the IQ4 sending the alarm. <Item Label> is the label of the module that generated the alarm. <Alarm Label> is the alarm's label. <Transition> is specifies whether the alarm is an occurred or cleared alarm. <Time> the time the alarm occurred. <Date> the date the alarm occurred. <Module Type> is the 'module identifier' (e.g. S is used for Sensor modules). <Module Number> is the module number. <Alarm Code> is the alarm's alarm code. The alarm codes are listed in **"General Alarms" on page 76** and **"Item Alarms" on page 77**. <Time> is the time the alarm occurred.

E.g.

IQ14 TRAINING ROOM SENSOR FAIL OCCURRED 17.24 29/11/14 O02S001OUTL

Specifies a sensor fail occurred alarm (out of limits) from a Sensor module 1 with a label of 'TRAINING ROOM' from an IQ4 with a device address of '2' and an identifier of 'IQ14' that was generated at 17:24 on 29/11/2014.

7.2.6.3 Attribute Format

Attribute format alarms contain text communications from the alarm log, e.g.

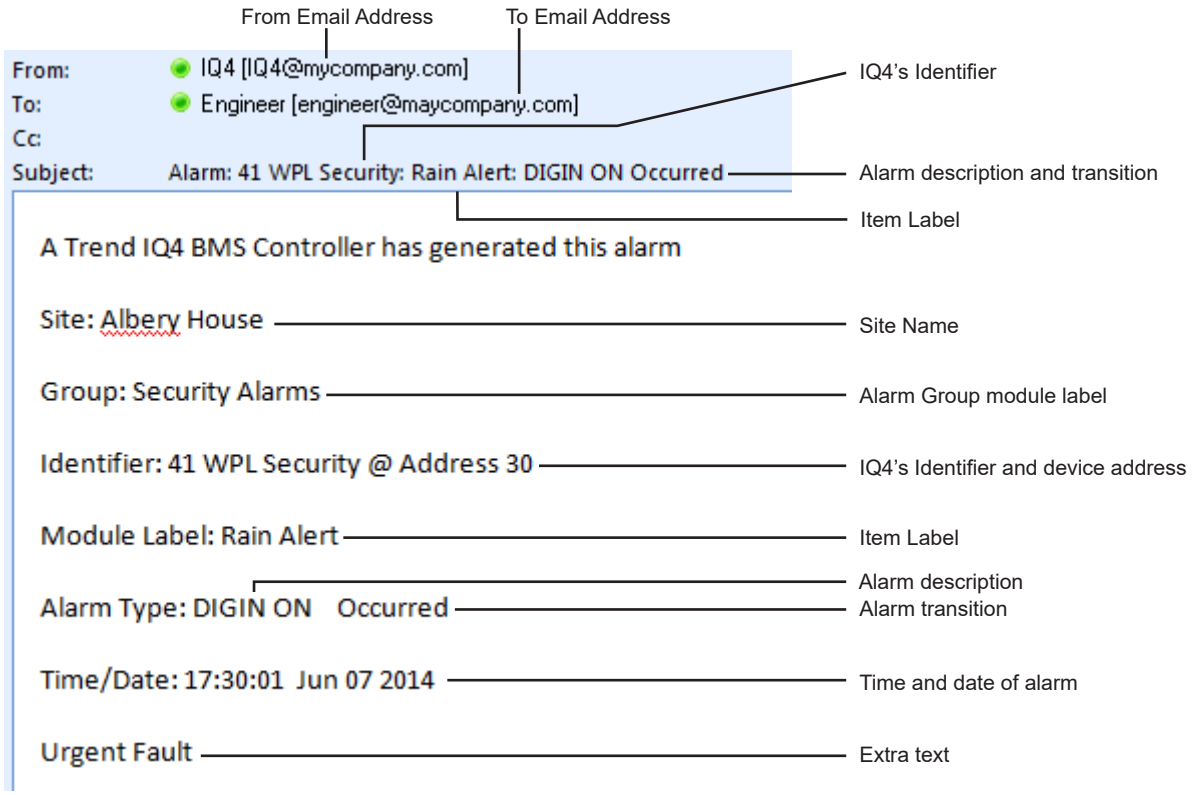
v(O = S, U = 1, P = 0, V = 24.04, R = 1, H = 16, N = 22, S = 4, D = 30, M = 5, Y = 0, G = 0, I = 0)

These parameters O, U, P etc. are Alarm Log module parameters- see **"Alarm Log Modules" on page 181**.

This form of alarm is for use by software applications that can parse the data and enter it into a database. The source address of the IQ4 is embedded in the alarm message and can be decoded by the application.

7.2.6.4 Email Format

Email format alarms are sent in standard email format and contain the following information as shown.



7.2.6.5 BACnet Format

BACnet format alarms are sent in standard BACnet alarm format and contain the following information:

<table>
<tr><th>BACnet Property</th><th>Description</th></tr>
<tr><td>Process Identifier</td><td>The Alarm Destination module's 'Process ID' parameter.</td></tr>
<tr><td>Initiating Device Identifier</td><td>The value of the Type 13 (BACnet Application) Network module's 'Device Instance' parameter.</td></tr>
<tr><td>Event Object Identifier</td><td>This comes from the source module's equivalent BACnet object's Object Identifier. E.g. AI8. The BACnet object identifiers for the Trend modules are AIn for Sensor module 1 and DIn for Digital Input module n.</td></tr>
<tr><td>Time Stamp</td><td>Time and date of alarm as in Alarm Log.</td></tr>
<tr><td>Notification Class</td><td>The Alarm Group module's number.</td></tr>
<tr><td>Priority</td><td>255 minus the value of the Alarm Group module's 'Hold Priority' parameter.</td></tr>
<tr><td>Event Type</td><td>Set according to the Trend alarm. Each alarm is associated with an event type.
<table>
<tr><th rowspan="2">Event Type</th><th colspan="4">Sensor</th><th colspan="2">Digital Input</th><th>Plot</th></tr>
<tr><th>HIGH</th><th>LOW</th><th>OUTL</th><th>READ</th><th>DI = 0</th><th>DI = 1</th><th>BBUF</th></tr>
<tr><td>CHANGE_OF_STATE</td><td></td><td></td><td></td><td></td><td>✓</td><td>✓</td><td></td></tr>
<tr><td>COMMAND_FAILURE</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>OUT_OF_RANGE</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td></td><td></td><td></td></tr>
<tr><td>BUFFER_READY</td><td></td><td></td><td></td><td></td><td></td><td></td><td>✓</td></tr>
</table></td></tr>
<tr><td>Message Text</td><td>The text format of the alarm - see "Text Format" on page 82.</td></tr>
<tr><td>Notify Type</td><td>Always set to 'ALARM' except for the Plot module BBUF alarm when it is set to 'EVENT'.</td></tr>
<tr><td>AckRequired</td><td>Always set to false.</td></tr>
<tr><td>From State</td><td>Previous state, either Normal, Off Normal, or Fault.</td></tr>
<tr><td>To State</td><td>The current state, either Normal, Off Normal, or Fault.</td></tr>
<tr><td>Event Values</td><td>Trend alarm. Each alarm is associated with an Event value.
<table>
<tr><th>Alarm</th><th>Event Value</th></tr>
<tr><td>Sensor HIGH</td><td>Current Value, Status Flags, Deadband, High Alarm Level</td></tr>
<tr><td>Sensor LOW</td><td>Current Value, Status Flags, Deadband, Low Alarm Level</td></tr>
<tr><td>Sensor OUTL</td><td>Current Value, Status Flags, Deadband, Upper or Lower Limit</td></tr>
<tr><td>Sensor READ</td><td>Current Value, Status Flags, Deadband, Input channel Limit</td></tr>
<tr><td>Digital Input DI = 0</td><td>Current Value, Status Flags</td></tr>
<tr><td>Digital Input DI = 1</td><td>Current Value, Status Flags</td></tr>
<tr><td>Plot Module BBF</td><td>Source, Last Notify Record, Previous Last Notify record</td></tr>
</table>
The Status Flags sent in the Event Values depend on the Event Type as shown in the table below:
<table>
<tr><th rowspan="2">Status Flags</th><th colspan="2">Trend module</th></tr>
<tr><th>Sensor</th><th>Digital Input</th></tr>
<tr><td>Bit 3</td><td>Out of Service, always '0'</td><td>Out of Service, always '0'.</td></tr>
<tr><td>Bit 2</td><td>Overridden, Sensor Overridden</td><td>Overridden, Digital Input Overridden</td></tr>
<tr><td>Bit 1</td><td>Fault, READ or OUTL</td><td>Fault, always '0'</td></tr>
<tr><td>Bit 0</td><td>In alarm, HIGH or LOW</td><td>In alarm, DI = 0 or DI = 1</td></tr>
</table></td></tr>
</table>

The IQ4 is also able to respond to a BACnet request for alarm information (BACnet GetEventInformation Service). This is a request for summary details all the relevant modules (i.e. sensor, digital input) in alarm which in BACnet is any module which is not in the 'Normal' state. The response message (List of Event Summaries) contains the following information in the previous table for each alarm.

BACnet Property	*Description*
Object Identifier:	The source module's equivalent BACnet object's Object Identifier. E.g. AI8. The BACnet object identifiers for the Trend modules are AIn for Sensor module 1 and DIn for Digital Input module n.
Event State	The current alarm state. It roughly corresponds to the Trend 'In Alarm' parameter. It is declared as Normal, OffNormal, Fault, High Limit, or Low Limit - see **"What is the Next/Previous Device" on page 75**.
Acknowledged Transitions	The relevant bits are always set to '1' (i.e. acknowledged) as the IQ4 does not support the BACnet acknowledge alarm service.
Event Time Stamps	The year, date, and time of the last transitions into OffNormal, Fault, and Normal states respectively.
Notify Type	Always set to 'ALARM' except for Plot module BBUF alarms when it is set to 'EVENT'.
Event Enable	Three bits are the enable bits for To Normal, To Fault, and To OffNormal; they are the equivalent of the Alarm Route module's 'Alarm Enable' parameters.
Event Priorities	These are the priorities for the three types of alarm transitions from the Module Notification Class object that transmitted the alarm. They are derived from the Alarm Group module's 'Hold Priority' parameter and are all set equal to 255 minus the value of the Alarm Group module's 'Hold Priority' parameter.

7.2.7 Alarm Format Validation (IQ4NC only)

If the alarm handling strategy is being used by an IQ4NC to send network alarms, forward alarms received from other controllers (see **"INC Alarm Forwarding (IQ4NC only)" on page 91**), or send general/item type alarms (strategy alarms) not all possible combinations of the alarm source, alarm format, and alarm destination are supported because the IQ4NC may not have enough information to send the alarm as specified.

Sources of alarms to the destination module are:

Alarms received from other controllers: These are alarms sent from other controllers to the IQ4NC - see **"INC Alarm Forwarding (IQ4NC only)" on page 91**.

Internal strategy alarms: These are general or item alarms from the modules within the IQ4NC.

Network alarms: These are network alarms generated by the IQ4NC that have been configured to be sent using the alarm handling strategy by setting the 'Alarm Group' parameter in the appropriate network module to non-zero.

The two Alarm Destination module parameters affecting the way the alarms are transmitted are the 'Type' and 'Message Format' parameters.

The alarm destination types are:

IQ Lan Alarm Destinations
IP Address Alarm Destinations
Email Alarm Destinations
BACnet Alarm Destinations

The 'Message Format' parameter only applies to IQ Lan Alarm and IP Address Alarm destinations, and can be set to:

Coded
Text
Attribute

The Alarm Destination module will validate its alarm transmission. Internal strategy alarms are sent in the format specified in the Alarm Destination module. The other alarms are validated as indicated in the table below:

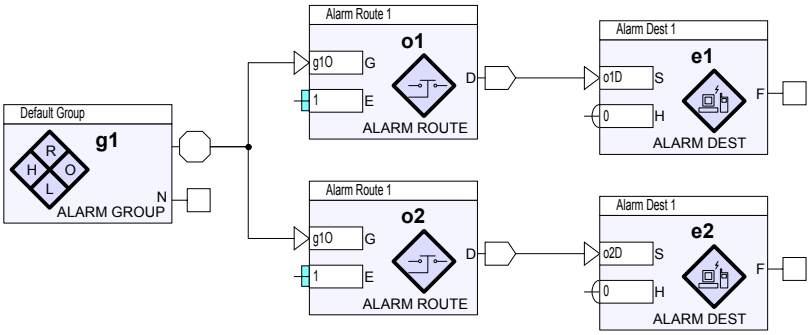
Alarm Source	*Original Alarm Format*	*Alarm Destination*			
		IQ Lan and IP Address IQ4NC Alarm Format			*Email*	*BACnet*
		Coded	*Text*	*Attribute*		
Alarms received from other controllers	Coded	Sent in Coded format			Not sent	N/A
	Text	Sent in Text format			Sent in email format	N/A
	Attribute	Sent in Attribute format			Not sent	N/A
Network alarms		Alarm sent in Text format			Alarm sent in email format	N/A

7.2.8 Alarm Handling Strategy Examples

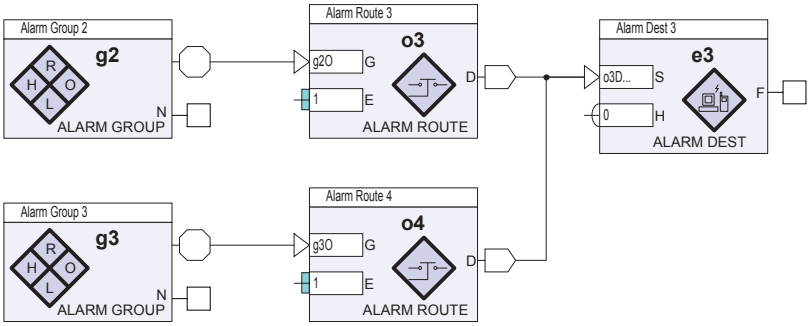
7.2.8.1 Sending Alarms to Multiple Destinations

Although an Alarm Route module can only be connected to one Alarm Destination module, an Alarm Group module can connect to several Alarm Route modules, and several Alarm Route modules can connect to the same Alarm Destination module. This flexibility enables groups of alarms to be sent to several destinations, and to share destinations with other groups.

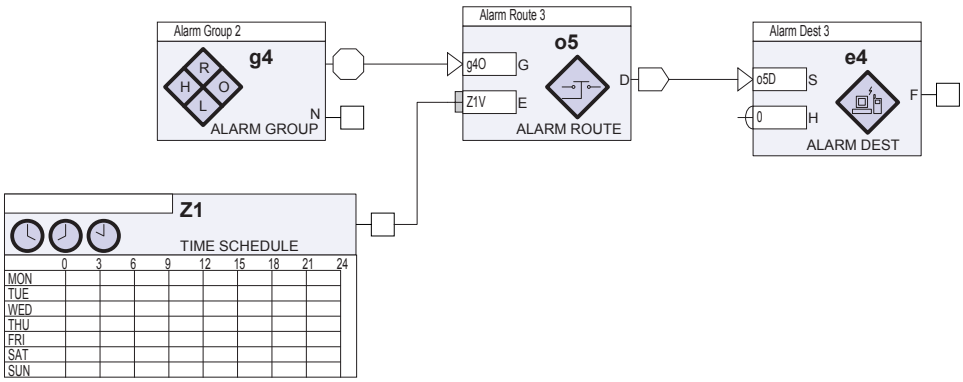
Alarms sent to two destinations



Two alarm groups using the same destination

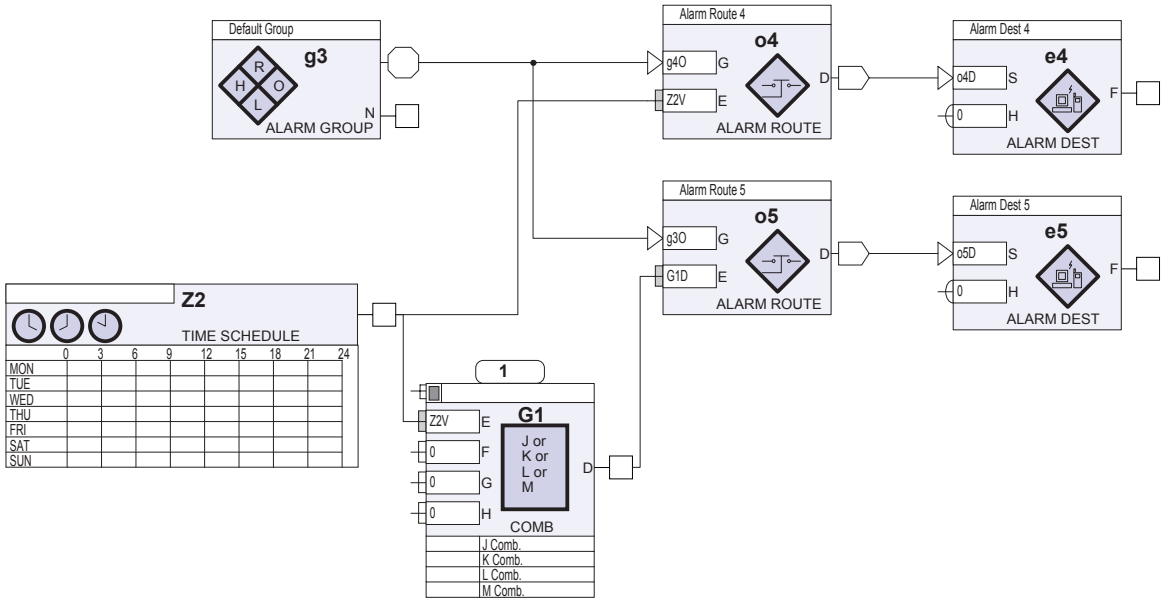


7.2.8.2 Route Time Enabled



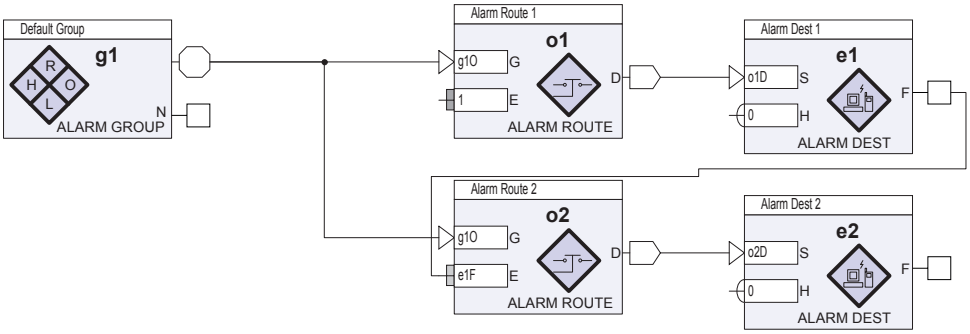
In this example, there is a single Alarm Route module, enabled by a Time Schedule module, so the Alarm Route module is enabled during Time Schedule module's occupation period. During non-occupation alarms are logged, but not sent, as with no route open, they are considered completed as soon as they are put in the Alarm Log. If the destination is failing, it will hold the alarm it is retrying, and if during this time the Time Schedule module switches to unoccupied, then any alarms stacked up in the log, and any further alarms will just be logged, not sent, and considered completed. However, the original alarm will still be active until it is acknowledged. Once the destination receives its acknowledge and occupation returns, the destination can start sending alarms again.

7.2.8.3 Two Routes, Occupied/Unoccupied



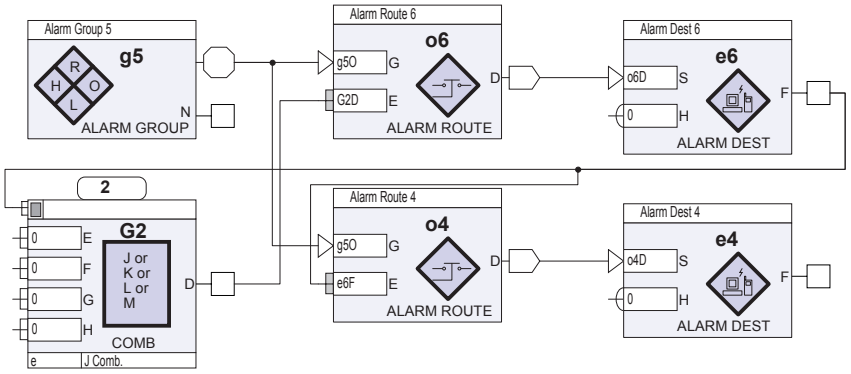
This strategy is a development of the previous in that now there is an alternative destination during non-occupation via a second Alarm Route module (o5). Alarms are sent to Alarm Destination module 4 during occupation, and to Alarm Destination module 5 during non-occupation. If Alarm Destination module 4 is failing awaiting an acknowledge, and the Time Schedule module changes to unoccupied, then the failing alarm will be sent to Alarm Destination module 5. If this is successful, any alarms stacked up will also be sent regardless of their original alarm time.

7.2.8.4 Two Routes; Use Second on Fail



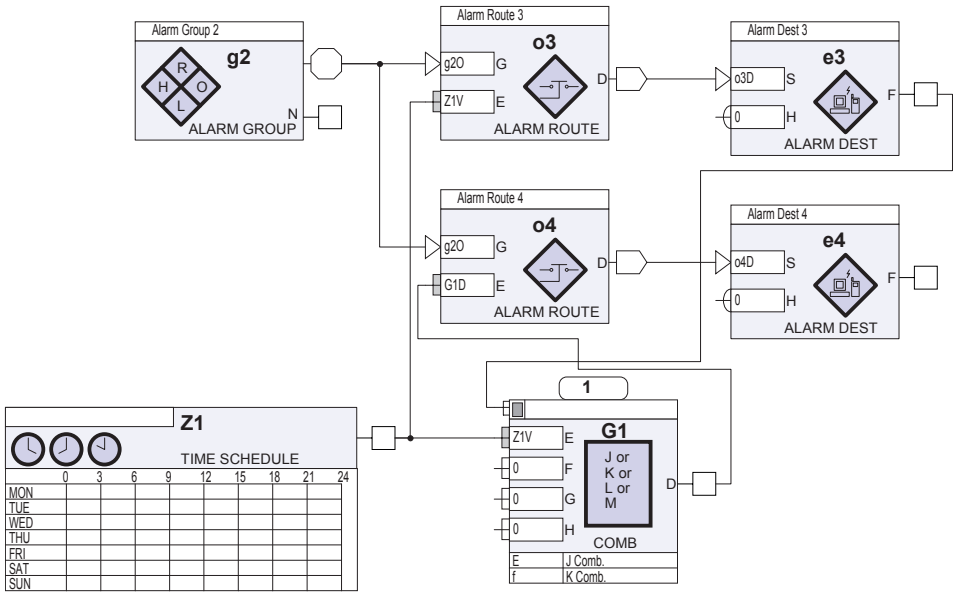
In this strategy, all alarms are sent to the first destination until it fails, at which point the Fail bit enables the second destination, and the failing alarm and any subsequent alarms are sent to the second destination. Once the first Alarm Destination module receives its acknowledge, the 'Failed' parameter is set '0' and the second Alarm Route module is disabled. At this point any alarms that have been sent to the second destination in the meantime, will also now get sent to the first destination because Alarm Route module 1 has never been disabled and thus during the time Alarm Route module 2 was enabled, those alarms had two destinations.

7.2.8.5 Two Routes, Use only Second on Fail



This strategy is a development of the previous, in that the first Alarm Route module (o6) is disabled while the e6 is in a fail state. This swops the route enables so that if an alarm occurs while the Alarm Route module (o7) is enabled, there is only one route open so it doesn't get sent to o6 when e6 recovers. Normally o6 is enabled and all alarms are sent to e6, but if e6 fails, the 'Failed parameter enables o7 and disables o6, and the failing alarm and any subsequent alarms are sent to the e7. Once e6 receives its acknowledge, the 'Failed parameter is set to '0', o7 is disabled, and alarms are sent to e6 again.

7.2.8.6 Two Routes, Use Second on Fail, or Unoccupied

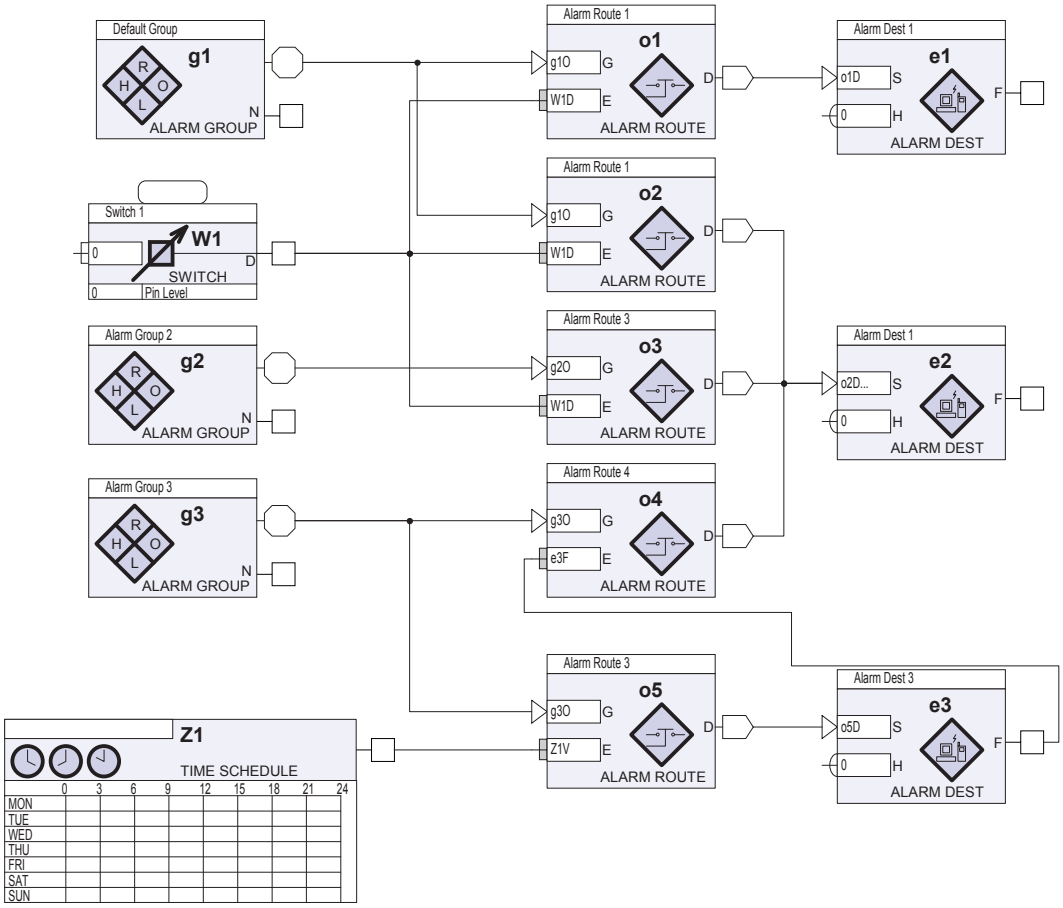


This strategy is a combination of 'Use second on fail', and 'Occupied/unoccupied'. The second route (route 4) is used both during non-occupation, and if the first route (route 3) fails during occupation. This is done by enabling route 3 during occupation, and using the logic module to enable route 4 either during non-occupation, or when the destination 3 fail bit is set.

As a result if route 3 fails during occupation, the failing alarm and any subsequent alarms will be sent to route 4. However, like the example shown in 11.6.1.3 above, if destination 3 recovers during occupation, then any alarms sent to destination 4 will also get sent to destination 3, as the route is valid.

However, if destination 3 fails, and then occupation status changes, any alarms sent to destination 4 during the unoccupied phase would not get sent to destination 3 after it recovers, and occupation returns.

7.2.8.7 Combining Groups and Destinations



This is an example of groups going to two destinations via two routes, (g1 via o1 and o2, and g3 via o4 and o5), and of several routes connecting to one destination (o2, o3, and o4 connecting to e2).

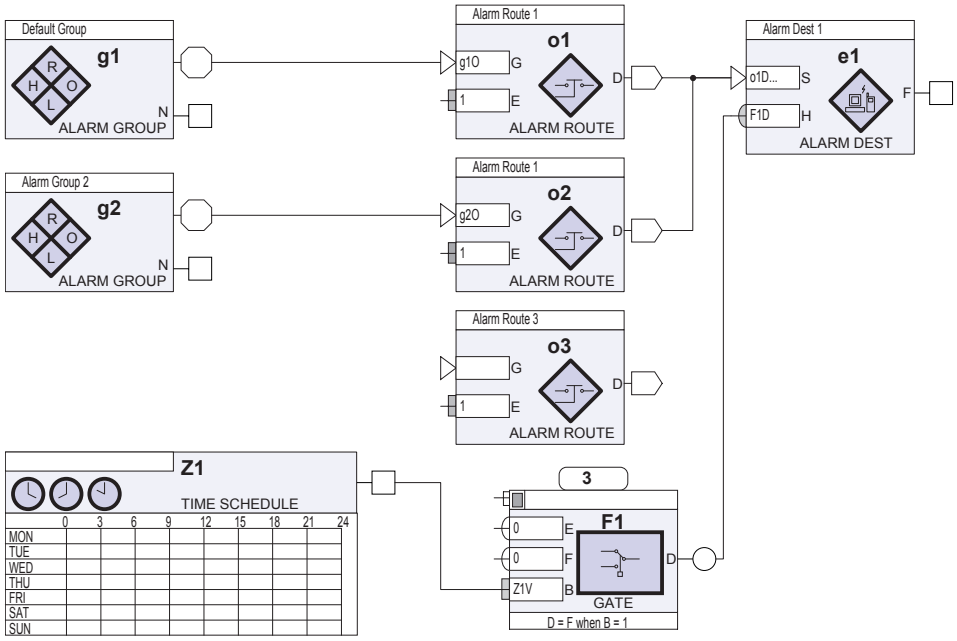
The Maintenance alarms from g1 are sent to the Maintenance company via o1 and e1, and to the Remote Bureau via o2 and e2. The Plant Failure alarms from g2 are sent to Remote Bureau via o3 and e2.

When the engineer is on site, he switches switch 1 off, disabling the reporting of maintenance and plant failure alarms. In this state the alarms will just get logged, but not reported.

The comfort alarms from g3 are reported to the Facilities Manager supervisor via o5 and e3 during occupation as defined by Time Schedule 1's output. If the alarm being sent to the Facilities Manager is not acknowledged, then the Fail bit of e3 will enable o4 to send it to the Remote Bureau via e2.

Note: In this case, if the Facilities Manager supervisor continues to fail to acknowledge alarms, Alarm Route module o4 will remain open after the original Alarm Route module o5 closes for non-occupation, and alarms will continue to be sent to the Remote Bureau until the e3 alarms are acknowledged. Additional logic would be required to enable Alarm Route module o4 only if both occupied and failed bit conditions were present.

7.2.9 Autodialled Destination with Routine Transmission of Low Priority Alarms



This strategy has the alarms split into two Alarm Group modules, Maintenance at priority 10, and Plant Failure at Priority 50. The Alarm Destination module e1 has its priority switched between 0 and 20 by the action of Time Schedule 1 on Function module 3.

Time Schedule 1 has its occupation set to the period of cheaper phone tariff. Thus during ‘phone times’, the hold priority level of Alarm Route module e1 will be set to ‘0’, so that maintenance alarms will be allowed to dial out, and all held alarms will be cleared. During the non ‘phone times’ period, only Plant Failure alarms will be allowed to send alarms to the auto-dialled destination, the Maintenance alarms will be held (unless a plant fail alarm causes a dial out). By carefully setting the Time Schedule this low priority dial out can be restricted say to one hour in the night, reducing the telephone costs.

Note: If a plant failure occurs during the day, any held maintenance alarms will also be sent.

7.3 INC Alarm Forwarding (IQ4NC only)

INC alarm forwarding enables the IQ4NC to forward alarms to either:

an IP address of a remote Trend supervisor making temporary connection to the site, or
an email address.

It satisfies a similar requirement to that provided by the XTEND when the vCNC is set up in alarm mode. It can be used by any IQ controller but is particularly useful for controllers which do not have direct Ethernet access or do not have IP Address or Email alarm destination types, such as IQ1, IQ2, IQL, IQeco, IQ3/LAN, and IQ4/LAN.

Note: Alarm forwarding can only be used to forward alarms to IP Address and Email alarm destinations (i.e the alarm handling strategy in the IQ4NC should only have Type 2 (IP Address) and Type 3 (Email) Alarm Destination modules. It cannot be used for forwarding alarms to another vCNC. If this is required the alarms can be targeted directly to that vCNC rather than using alarm forwarding.

The controller sends the alarms to be forwarded to the IQ4NC. Two address options are available for sending alarms to an IQ4NC:

- For all IQ1, IQ2, IQ3, and IQ4 pre v3.2 controllers the alarms should be sent to the IQ4NC's device address on its own LAN number. This option provides benefits to existing sites whereby existing controller configurations do not have to be modified and are left at their existing alarm addresses. For example, replacing an EINC with an IQ4NC, where the controllers previously sent their alarms to a vCNC which was configured to send their alarms to an IP address.

- For new projects using IQeco, IQ2, IQ3, IQ4 v3.2 or greater controllers, the alarms can be sent to either IQ4NC's device address on its own LAN number or at address 126 on its own LAN number. This provides the benefit for all future systems to be able to use template strategies which are configured to send the alarms to address 126 without knowing the specific destination address of the particular site.

The table below gives a summary of how to set up the alarm target addresses and LAN numbers to send alarms to an IQ4NC. This is dependent on the controller initiating the alarm and whether the IQ4NC is on the Local LAN or on a Remote LAN.

<table>
<tr><th rowspan="2">Initiating Controller</th><th colspan="2">IQ4NC on Local LAN</th><th colspan="2">IQ4NC on Remote LAN</th></tr>
<tr><th>Device</th><th>LAN</th><th>Device</th><th>LAN</th></tr>
<tr><td>IQeco pre v2.2</td><td>126</td><td>IQ4NC's LAN Number</td><td rowspan="3">126</td><td rowspan="4">IQ4NC's LAN Number</td></tr>
<tr><td>IQeco v2.2 or greater</td><td>126</td><td>0 or IQ4NC's LAN Number</td></tr>
<tr><td>IQ4 v3.2 or greater</td><td>126</td><td>0</td></tr>
<tr><td>IQ1, IQ2, IQ3, IQ4 pre v3.2</td><td>IQ4NC's device address</td><td>0</td><td>IQ4NC's device address</td></tr>
</table>

When the IQ4NC receives the alarm it is sent to the Alarm Group module specified by the 'Alarm Group' parameter of the Network module for the media from which the alarm is received e.g. alarms received from the current loop network will be sent to the Alarm Group module specified by the Type 1 (IQ LAN) Network module.

Note: This means that the forwarded alarms and network alarms from that particular network are processed using the same alarm handling strategy.

Note: The IQ4NC must have the 'Alarm Group' parameter set up in this Network module.

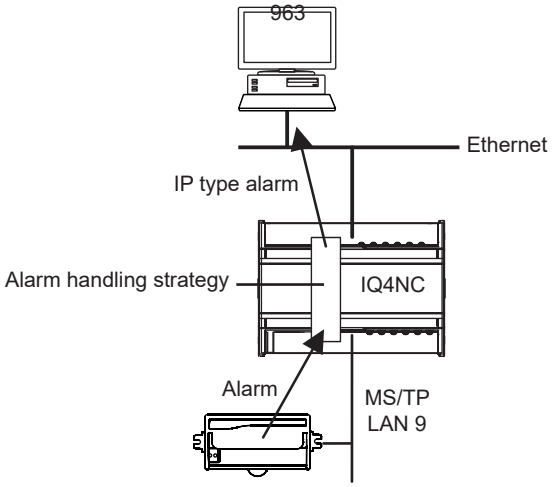
Once the alarm is received by the Alarm Group module it will be processed according to the alarm handling strategy. Care should be taken with the configuration of the alarm format in the sending controller as only certain formats are supported for different destinations - see **"Alarm Format Validation (IQ4NC only)" on page 85**.

7.3.1 Alarm Forwarding Examples

7.3.1.1 Alarm Forwarding for IQeco Using Local IQ4NC

One of the most common uses of alarm forwarding is for an IQeco to send its alarms to a supervisor that only has a temporary connection the system via the IQ4NC connecting the MS/TP LAN to the internetwork.

In this example an IQeco on LAN 9 which is an MS/TP LAN uses the IQ4NC (operating in Ethernet to MS/TP mode) also on LAN 9 to forward an alarm to the 963.



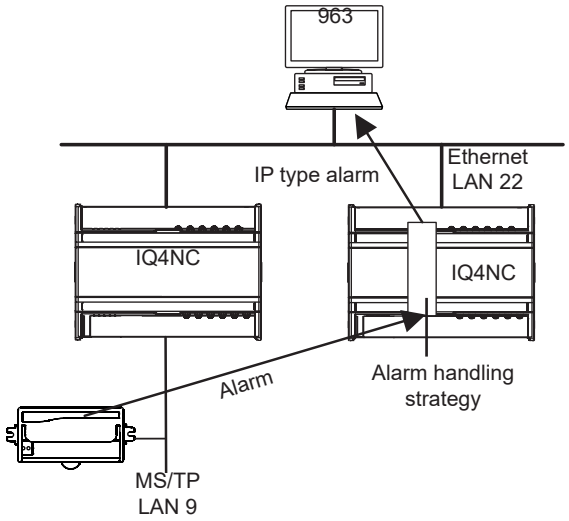
The IQeco generating an alarm is configured to send it to address 126 on LAN 0. The IQ4NC will receive the alarm via its Type 12 (BACnet MSTP) Network module. The Type 12 (BACnet MSTP) Network module's 'Alarm Group' parameter is set up to send the alarm to the required Alarm Group module for processing by the alarm handling strategy. The alarm handling strategy has a contains Type 2 (IP Address) Alarm Destination module which forwards the alarm to the 963.

Note: Pre v2.2 IQecos should send the alarm to the Local LAN number, instead of LAN 0.

7.3.1.2 Alarm Forwarding Using Remote IQ4NC

The IQ4NC forwarding the alarms does not have to be on the same LAN as the controller generating the alarm, i.e. it can be on a remote LAN.

In this example an IQeco on LAN 9 which is an MS/TP LAN uses the IQ4NC on LAN 22 to forward an alarm to the 963 that only has a temporary connection the system.

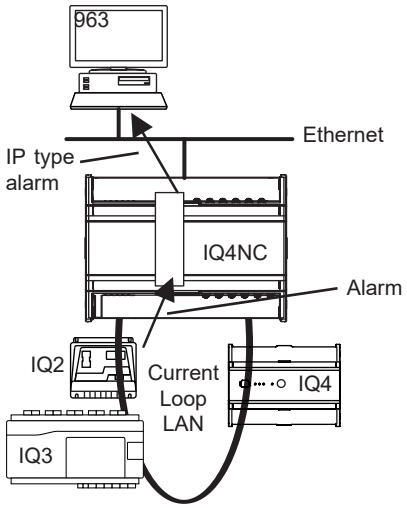


The IQeco generating an alarm is configured to send it to address 126 on LAN 22. The IQ4NC will receive the alarm via its Type 4 (Ethernet IP) Network module. The Type 4 (Ethernet IP) Network module's 'Alarm Group' parameter is set up to send the alarm to the required Alarm Group module for processing by the alarm handling strategy. The alarm handling strategy contains a Type 2 (IP Address) Alarm Destination module which forwards the alarm to the 963.

7.3.1.3 Alarm forwarding from a Current Loop LAN (e.g. EINC Replacement)

A common requirement for alarm forwarding is for a controller on a current loop LAN to be able to send alarms to a supervisor that only has a temporary connection to the system.

In this example the IQ2, IQ3, and IQ4 controllers on the current loop LAN (LAN 9) all use the IQ4NC also on LAN 9 to forward alarms to the 963. The diagram below shows the path taken by alarms generated by the IQ2.



The controllers generating an alarm are configured to send the alarms either to address 126 on LAN 0 or to the IQ4NC device address on LAN 0 (depending on controller type and version - see **"INC Alarm Forwarding (IQ4NC only)" on page 91)**. The IQ4NC will receive the alarm via its Type 1 (IQ LAN) Network module. The Type 1 (IQ LAN) Network module's 'Alarm Group' parameter is set up to send the alarm to the required Alarm Group module for processing by the alarm handling strategy. The alarm handling strategy contains a Type 2 (IP Address) Alarm Destination module which forwards the alarm to the 963.

A specific case of alarm forwarding from a current loop LAN is where an INC type node (e.g. XTEND) is being replaced by an IQ4NC and the INC type node is using a vCNC operating in alarm mode to send alarms to a supervisor with a temporary connection to the system . In this case the IQ4NC's device address should be set to the address of the vCNC (in the INC type node) it is replacing which was being used to send the alarms. Since the alarms to be forwarded were previously directed to this address by the IQs on the LAN, they will now be directed to the IQ4NC instead and will be forwarded by the IQ4NC.

8 WEB SERVER

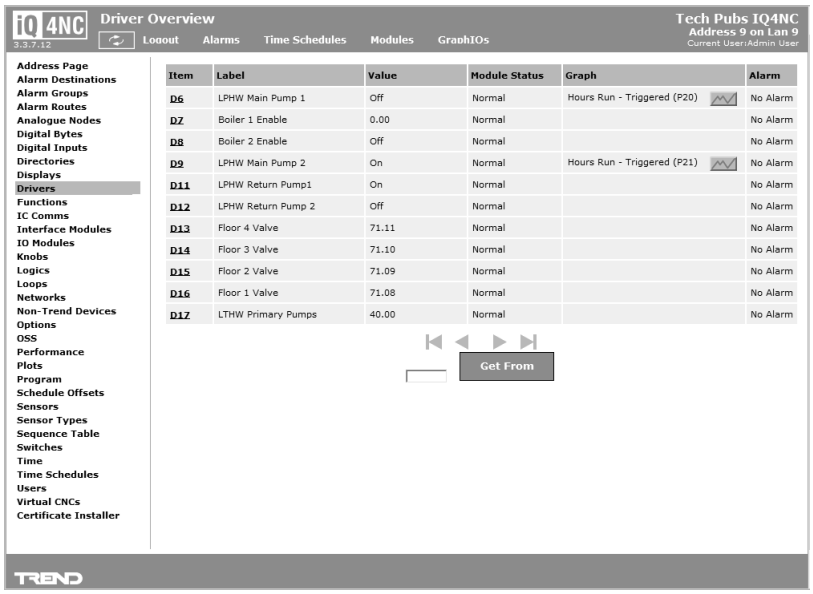
Information from an IQ4 can be viewed and adjusted from a web client over any TCP/IP network (e.g. company intranets) using secure HTTPS protocol. All that is required is the IP address or host name of the IQ4, and a valid user name and password.

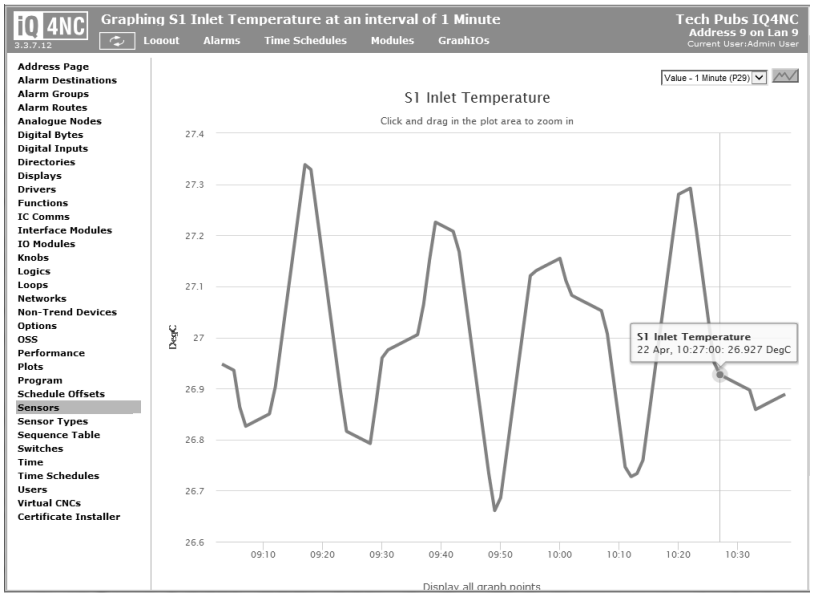
8.1 Functionality

The IQ4's web server allows:

- Viewing and adjustment of the daily profile
- Viewing the alarm history
- Viewing individual module parameters
- Adjustment of individual module parameters
- Graphing individual module parameters

The IQ4 is provided with a standard set of web pages which cover all accessible modules.



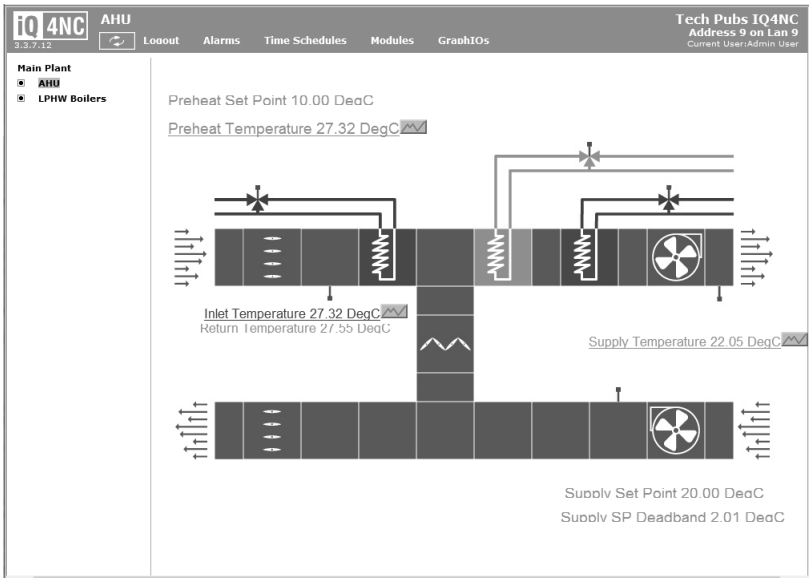


Note: Module parameters may be monitored and changed using web pages but module creation, deletion and linking can only be done using IQSET.

8.1.1 Graphical Display Pages

Graphical display pages (GraphIQs) can be created to suite user requirements: The pages can contain:

a backdrop graphic
live values from the IQ4
static text
adjustment of parameter
graphing of parameters
links to other GraphIQs, web pages and URLs.



GraphIQs are created by setting up Display and Directory modules using the Graphical Display Page Editor which is supplied with IQSET.

For further details of web pages and their configuration see the IQ4 Web User Guide (TC201256), and the Graphical Display Pages Editor Manual (TE200629).

8.2 Clients

The IQ4 web server can be accessed using a web client on:

- PCs
- Smartphones
- Tablet

IQ4 has been tested with most modern browsers, smartphones, and tablets available at the date of release of the product. For details of compatibility see the appropriate IQ4 data sheet.

Note: IQ4 has not been tested with all devices and Trend cannot guarantee a particular device's compatibility. Potential changes beyond Trend's control mean that IQ4's compatibility with future browser releases cannot be guaranteed.

8.3 Web Server Security

When using the HTTP web server the connection to the IQ4's web server uses Hypertext Transfer Protocol (HTTP). The HTTP protocol is used to send data between the IQ4 and the client.

When using the HTTPS web server the connection to the IQ4's web server uses Hypertext Transfer Protocol Secure (HTTPS). The HTTPS protocol is used to encrypt the data being sent between the IQ4 and the client with TLS encryption. This ensures that all data exchanged between the IQ4 and client is encrypted and secure, if someone were to capture the data being transferred it would be unrecognizable.

In order to encrypt and decrypt the data being transferred the IQ4 uses X.509 certificates which are small data files that digitally bind a cryptographic key to an organisation's details. The IQ4 web server must be configured with an X.509 certificate which ensures that any outgoing communications are encrypted and any client accessing it must have a matching client certificate in order to decrypt the data.

9 SECURITY

This section is divided into two sections:

Securing IQ4 - Describes what needs to be done to secure the IQ4 on the network(s) it is connected to, and steps to enable recovery in the event of failure of the IQ4.
IQ4 Security System - Describes the IQ4s security system.

9.1 Securing IQ4

IQ4 is a networked product and as such must have its security correctly configured to reduce the risk of unauthorised access. For general information about securing Trend products see the 'General Security Best Practice for Trend Products Information Sheet' (TP201331).

In addition to the actions described in the 'General Security Best Practice for Trend Products Information Sheet' the advice described in the following sections must be followed.

Adopting normal installation and security best practice guidelines can mitigate the risk of a malicious IT attack from a skilled and equipped IT individual.

9.1.1 Security Checklist

- ☐ Strategy file (.IQ4) included in the disaster recovery plan.
- ☐ IQSET files included in the disaster recovery plan.
- ☐ Physical access to IQ4 is restricted.
- ☐ IQ4 is running latest release of firmware.
- ☐ Ethernet network secured - see **"Network Planning and Security" on page 100**.
- ☐ Type 0 User Module created with a security level of 99.
- ☐ 'Guest' user added and configured or minimum access.
- ☐ Type 0 User Modules created as appropriate.

9.1.2 Developing a Security Program

Refer to the 'General Security Best Practice for Trend Products Information Sheet' (TP201331).

9.1.3 Disaster Recovery Planning

When developing the disaster recovery plan you must ensure that the strategy file (.IQ4) downloaded to the IQ4 and the IQSET files used to create it are included.

9.1.4 Physical and Environmental Consideration

The IQ4 must be installed within a locked environment e.g. located in a secured plant room, or a locked cabinet.

Note: Ensure adequate ventilation.

9.1.5 Security Updates and Service Packs

Ensure the IQ4 is running the latest release of firmware.

9.1.6 Virus Protection

Not applicable to IQ4.

9.1.7 Network Planning and Security

Ethernet Network

It is recommended that the Ethernet network used by the BMS system is separated from the normal office network using an air gap, or virtual private network. Physical access to the Ethernet network infrastructure must be restricted. You must also ensure that the installation complies with your company's IT policy.

IQ4s must not be connected directly to the internet.

Web Server

The IQ4 provides both HTTP and HTTPS web servers. If a web server is not required it is recommend that both web servers are disabled - see **"Configure Web Servers" on page 124**.

Trend Current Loop Network (/LAN and IQ4NC only)

Physical access to the Trend current loop network infrastructure must be restricted. If the Trend current loop network is not being used the IQ4's Type 1 (IQ LAN) Network module must be disabled by setting the 'Disable Module' parameter to '1'.

BACnet IP Network (/BAC and IQ4NC only)

Due to the insecure nature of the BACnet protocol IQ4s that support BACnet IP MUST not be connected to the internet under any circumstance. The IQ4's security system does not protect against BACnet writes. Physical access to the BACnet IP network infrastructure must be restricted. If BACnet IP communications are not required the IQ4's Type 5 (BACnet IP) Network module must be disabled by setting the 'Disable Module' parameter to '1'.

If BACnet communications are required it is strongly recommend that the BACnet Backup/Restore, Reinitalise Device and BACnet Writable services are not enabled. However, this will mean that the strategy created is not BTL compliant - see **"BACnet Communications" on page 63**.

MS/TP (IQ4NC only)

Physical access to the MS/TP network infrastructure must be restricted. If the MS/TP network is not required the IQ4's Type 12 (BACnet MSTP) Network module must be disabled by setting the 'Disable Module' parameter to '1'.

I/O Bus (IQ4E only)

Physical access to the IQ4E's I/O bus must be restricted.

Wallbus (Not IQ422/00/XNC/.... or IQ4NC/00/... variants)

Physical access to the IQ4's Wallbus must be restricted. If the Wallbus is not required the IQ4's Type 14 (Wallbus) Network module should not be included in the strategy.

USB

Physical access to the IQ4's USB Local Engineering Port must be restricted.

RS232 (including MODBUS / INT only)

Physical access to the IQ4's RS232 Local Supervisor Port must be restricted. If not required any Network modules connected to the port should not be included in the strategy.

RS485 (including MODBUS / INT only)

Physical access to the IQ4's RS485 port must be restricted. If not required any Network modules connected to the port should not be included in the strategy.

MODBUS IP Network (/INT only)

Due to the insecure nature of the MODBUS protocol IQ4s that support MODBUS IP MUST not be connected to the internet under any circumstance. Physical access to the MODBUS IP network infrastructure must be restricted. If MODBUS IP communications are not required the IQ4's Type 19 (Modbus IP) Network module should not be included in the strategy.

9.1.8 Virtual Environments

Not applicable to IQ4.

9.1.9 Securing Wireless Devices

If a wireless network is being used it must be secured according to your company's IT policy.

9.1.10 System Monitoring

Not applicable to IQ4.

9.1.11 Windows Domains

Not applicable to IQ4.

9.1.12 Securing Access to the Operating System

Ensure that the IQ4's Type 0 User Modules are set up to ensure that access or changes require password or PIN authorisation. For details of the IQ4s security system - see **"IQ4 Security System" on page 101**.

9.1.13 Security Features

For details of the IQ4s security features - see **"IQ4 Security System" on page 101**.

9.2 IQ4 Security System

IQ4's security is provided by its User modules. Security is enabled as soon as a User module (Type 0) is set up.

If no Type 0 User Modules are set up, the IQ4's security is disabled and any change can be made without the need for a password or PIN. This should be seen as a serious security risk and installing IQ4s without Users is not recommended, and seen as bad practice.

Note: IQ4s are shipped from the factory without any Type 0 User Modules set up i.e no security. On new strategies IQSET will enforce the addition of default users and it is strongly recommended that they are NOT removed.

The IQ4's security system protects against:

- **Changes Using Text Communications**
- **Strategy Upload and Download**
- **Access to Web Pages**
- **Access to XML Web Services**
- **Network Use (v3.30 or greater)**
- **Connection to vCNCs (v3.60 or greater)**

Note: For /BAC variants and IQ4NCs even if the IQ4's security is correctly configured, the IQ4 is not protected from BACnet writes. It is only protected from BACnet writes if the IQ4's Type 5 (BACnet IP) Network module is disabled.

Each Type 0 User Module has a password, user name, PIN and user level.

Password, User name and PIN

The user name and password are used to protect access to web pages, and strategy upload and download. The PIN is used to protect text communications with the IQ4 and strategy upload and download, and is only required for Type 0 User Modules with network access rights.

Note: If a user's user name, password and PIN are forgotten they can be reset - see **"Password Recovery" on page 514**.

User Level

The user level determines what parameters a user can adjust, and what information they can see on the IQ4's web pages once a valid PIN or user name and password has been supplied.

Some changes require the user level to be at or above a certain fixed level before they can be made - see ""Fixed User Levels" on page 511. Other changes have levels that are configurable (e.g. Knob Value, Switch Status, Time Schedule parameters) - see "Configurable User Levels" on page 512. Because of this there may be many Type 0 User Modules set up in an IQ4 enabling different users to have different sets of parameters that they may change.

9.2.1 Changes Using Text Communications

The following changes use text communications:

- Changes using a supervisor, tool or display
- Changes using IC communications
- Timemaster time synchronization
- IQSET Live Values/Live Edit

Text communications are protected by Type 0 User Modules that have network access rights ('Disable Network Use' parameter set to '0' (OFF). In order to protect changes made using text communications there must be a least one Type 0 User Module with network access rights. If there are Type 0 User Modules set up but none have network access rights then changes made using text communications will not be protected.

Note: If protection is required for the web server but this would cause problems with communications on the Trend network Type 0 User Modules with network use disabled should be used to protect the web server - see **"Network Use (v3.30 or greater)" on page 104.**

If changes using text communications are protected, the device making the change must include a PIN in the communications message that makes the change. This PIN must match a PIN in one of the IQ4's Type 0 User Modules with network access rights, and the PIN level associated with that Type 0 User Module must be high enough to authorise the change - see **"Fixed User Levels" on page 511**.

If the device initiating the change is a supervisor, tool, or display, the PIN sent is specified during the configuration process of the device making the change. For IC communications and time synchronisation messages the controller initiating the message will include the PIN from its own Type 0 User Module 1 if it is set up; if this Type 0 User Module is not set up or its does not have network access rights the controller will not be able to make the change.

9.2.2 Strategy Upload and Download

When security is enabled the upload and download of strategy files is protected by user name, password and PIN. IQSET includes a user name, password and PIN in the communications message when uploading or downloading the strategy.

The user name, password and PIN must match those of a Type 0 User Module in the IQ4, and the PIN level associated with that Type 0 User Module must be high enough to authorise the upload or download - see **"Fixed User Levels" on page 511**.

Note: If none of the Type 0 User Modules have network access rights strategy upload and download is only protected by user name and password, a PIN is not required.

Note: The Type 0 User Module that matches the user name and password and the Type 0 User Module that matches the PIN can be different modules as long as the User Level allows file Upload/Download.

9.2.3 Access to Web Pages

9.2.3.1 Access with Security Enabled

When security is enabled access to web pages is protected by user name and password - see **"Logged in Users" on page 103**. Users that access the IQ4 without a user name and password will be treated as a casual user - see **"Casual Users" on page 103**.

9.2.3.1.1 Logged in Users

To access the IQ4s web pages as a logged in user a user must enter a user name and password that is a match to one of the IQ4's Type 0 User Modules. Once logged in the specified home page is displayed - see **"Home Page" on page 513**. The user's access is determined by the Type 0 User Module's user level- see **"User Level" on page 511** and view defaults - see **"View Defaults" on page 512**. The user level determines what parameters the user is able to view/adjust and the view defaults determine the pages they are able to access via the links on the banner across the top of the page and the Navigator on GraphIQs pages.

Once the user has finished they must logout to prevent unauthorised changes (although the login will time out automatically after period of key inactivity defined by the 'Timeout (Minutes)' parameter).

All users who are going to access the IQ4 via the web pages must be given their own user name and password (i.e a Type 0 User Module for each user).

9.2.3.1.2 Casual Users

A casual user is someone who accesses the IQ4s web pages without logging in. When security is enabled the access given to such a user is controlled by a special Type 0 User Module known as the 'Guest' user - see **"The 'Guest' User" on page 103**.

The table below summarises the web page security features for casual users when security is enabled:

Configuration	***Description***
'Guest' user set up with no password.	The IQ4 will log the user in as the 'Guest' user - see **"The 'Guest' User" on page 103**
'Guest' user does not exist or is set up with password.	The IQ4 will log the user in as the 'System Guest' user with a user level of '-1' - see **"The 'System Guest' User" on page 103**.

9.2.3.2 Access with Security Disabled

When security is disabled, access to the IQ4's web pages are not protected and anyone accessing the IQ4 will have full access to all information and the ability to make changes. This is achieved by automatically logging the user in as the 'System Guest' user with a user level of '100' - see **"The 'System Guest' User" on page 103**.

9.2.3.3 The 'Guest' User

If the 'Guest' user exists, when the IQ4's web pages are accessed without logging in then the user is given the access rights (user level, home page, and view defaults) specified by the 'Guest' user.

By default the 'Guest' user only provides access to the IQ4's 'Welcome' page and has a user level of '0'. This means that a user accessing the IQ4 without logging in will have no access to the IQ4's web pages. In order to give more access to the casual user the 'Guest' user can be configured in the same way as any other Type 0 User Module.

IQSET prevents the 'Guest' user being given a password, PIN, or user level higher than '0'. It does allow a home page and view defaults to be configured.

It is strongly recommended that the Guest user is left with the default configuration (user level of '0' and no view rights).

Note: If the 'Guest' user module has a password set up it loses its special purpose, and a casual user will be logged in as the 'System Guest' user with a user level of -1 - see **"The 'System Guest' User" on page 103**.

9.2.3.4 The 'System Guest' User

The 'System Guest' user controls access to the IQ4's web pages when security is disabled, the 'Guest' user has been configured with a password, or the 'Guest' user does not exist.

The table below summarises the web page security features for casual users when the 'System Guest' user is used:

Configuration	*Description*
No Type 0 User Modules	The IQ4 will log the user in as the 'System Guest' with a user level of '100'..
Type 0 User Modules set up but the 'Guest' user does not exist.	The IQ4 will log the user in as the 'System Guest' with a user level of '-1'.
Type 0 User Modules set up and the Guest' user is set up with password.	

'System Guest' User with User Level of '100'
When accessing the IQ4's web pages as the 'System Guest' user with User Level of '100' the user will have full access to the IQ4s web pages and will be able to make adjustments.

'System Guest' User with User Level of '-1'
When accessing the IQ4's web pages as the 'System Guest' user with User Level of '-1' the user will have access to Sensor, Digital Input, Knob, Switch, Driver, Schedule, Time Schedule, Time, Plot modules, the Alarm Log, and GraphIQs but not the ability to make changes.

9.2.4 Access to XML Web Services

When security is enabled reads and writes using the XML Web Services is protected by an application key. This key must match the key in one of the IQ4s Type 0 User Modules - see **"XML Syntax" on page 527**.

9.2.5 Network Use (v3.30 or greater)

Type 0 User Modules can be configured to only provide security to the IQ4's web pages using the 'Disable Network Use' parameter. When the 'Disable Network Use' parameter is set to '1' the Type 0 User Module will protect web page access, file upload/download and XML web services but not text communications (.i.e. the Type 0 User Module does not enforce network access rights).

As a best practice Trend recommends that all controllers should have Type 0 User Modules added, however if this would affect Trend network communications (e.g. on legacy sites), then setting this parameter will still allow the Trend network communications to operate while providing protection to the IQ4's web server.

Note: If security is enabled but none have network access rights then changes made using text communications will not be protected.

9.2.6 Connection to vCNCs (v3.60 or greater)

Type 1 CNC User Modules are used to control access to the IQ4's vCNCs when they are operating in 'Secure' mode - see **Type 1 CNC User Modules (v3.60 or greater).** It is recommended that if vCNCs are enabled that they are configured to operate in 'Secure' mode.

10 XML WEB SERVICES

The XML Web Services feature provides communications access for read and write operations (like SCADA, Supervisory Control and Data Acquisition).

XML (Extensible Markup Language) is a general purpose specification for creating custom markup languages. It helps developers create web pages and also provides a basic syntax that enables information to be shared by different computers and different applications. The IQ4 XML syntax is described in **"XML Syntax" on page 527**.

A web service call is achieved by sending an HTTP request to the IQ4, the response from the IQ4 being an XML document. The data is specified using text communications.

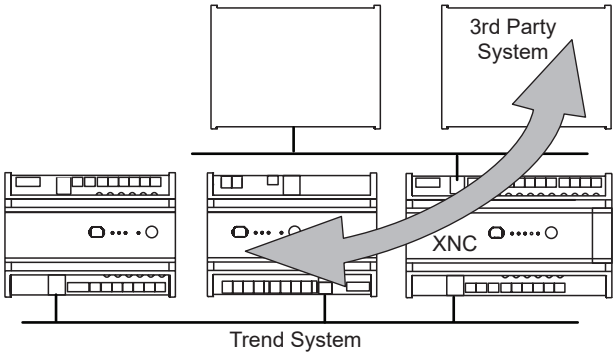
XML Webs Services Security

Writes to the IQ4 require an 'Application Key' to be included with the write if the IQ4 is protected by an engineered Type 0 User Module. The key will limit access to the IQ4 to only those users that are eligible. This key must match details generated from an engineered Type 0 User Module (not a System Guest). Enhanced security using encryption is described in **"Authentication Security" on page 540**.

The XML Web Services validate the HTTP request, and return an error if it fails the validation.

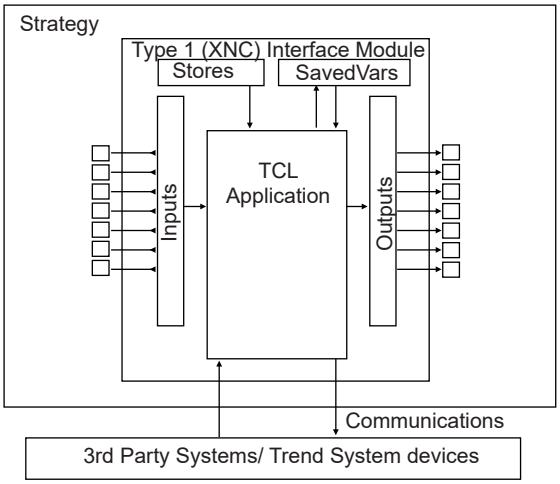
11 XNC FACILITY

The IQ4's XNC facility is only available on /XNC variants. It enables interfacing between the Trend system and 3rd party systems allowing information from other systems to appear as though from an IQ controller and for parameters within the 3rd party system to be adjusted using Trend Supervisor and Tool software.



It is able to communicate with 3rd party systems over Ethernet, RS232, or RS485. It can communicate with other Trend devices over Ethernet, or Trend current loop networks.

The XNC facility uses the Type 1 (XNC) Interface module and appropriate Network modules to provide the interface between the 3rd party system and the strategy as shown below:



11.1 Strategy

The IQ4's strategy passes information into the Type 1 (XNC) Interface module by linking module values (e.g. Knob and Switch modules) in the strategy to its inputs. Information is passed from the Type 1 (XNC) Interface module to the strategy by linking its outputs to module values (e.g. Sensor, and Digital Input modules) in the strategy.

11.2 Type 1 (XNC) Interface Module

The Type 1 (XNC) Interface module is the interface between the TCL application and the strategy, it contains the TCL application, Inputs, Outputs, Stores, and SavedVars. It is where the TCL application is stored, compiled, and executed. Once downloaded with a TCL application, the device will continue to execute that application until it is manually halted. If the device is turned off the application will restart once the device is turned on. It communicates with the third party system, and can write values to, and read values from the strategy, 3rd party system, and Trend network allowing data to be sent directly to other devices on the Trend system.

For details of Type 1 (XNC) Interface modules see **"Type 1 (XNC)" on page 309**.

11.3 Trend Custom Language

TCL is used to program the interface. It is a programming language, based on BASIC. Amongst other things it enables interfacing between the Trend System, and 3rd party systems. A full description of the language can be found in the IQ4/XNC TCL Manual (TE201360).

TCL applications are created by Trend Interface Specialists (TISs) who can provide both bespoke applications and generic applications for commonly used equipment. The TIS may provide the application in soft format, or pre-loaded into the /XNC. The application can be tied to one IQ4 or available for general download.

A security mechanism prevents applications written by a TIS from being used without the TIS's authority. The application may be designated to run on any IQ4, or to run on a particular IQ4. The on site commissioning engineer is able to request an unlock code from the TIS.

11.3.1 TCL Application Configuration

The TCL application is configured using TCLTool2. This creates the XNC program file which can be downloaded using TCLTool2 or IQSET. Use of TCLTool2 is described in the TCLTool2 Manual (TE200923).

Note: The XNC program file should be downloaded before the strategy file is downloaded.

11.4 Compatibility with IQ3 TCL

TCL applications written for use with IQ3/XNC variants can be used by IQ4/XNC variants with some minor modifications - see the IQ4/XNC TCL Manual (TE201360).

12 IQ INTERFACING

IQ interfacing is an optional feature only available on /INT variants that enables easy integration of Serial (RS485) MODBUS, IP MODBUS and M-bus devices (e.g. meters, sensors/chillers/VRF Systems) to the IQ4. This enables the IQ4 to read and write values directly. If required an IQ4 can be connected to both MODBUS and M-bus.

Standard strategy blocks, available in IQSET, allow easy strategy configuration using drag & drop for common MODBUS and M-bus devices - see https://partners.trendcontrols.com/partners/index.aspx#Downloads:SET for the latest plant blocks.

Engineering with standard IQ4 Network and Interface modules allows connection to other MODBUS or M-Bus devices. Strategy blocks can be created for favourite devices to save engineering time. /INT variants have extra brIQs - see **"Modules" on page 7**.

Note: XNC variants cannot be licenced for IQ interfacing.

12.1 MODBUS

Maximum number of inputs/outputs per device: Each MODBUS Interface module can have up to 500 inputs and 500 outputs, this allows 500 inputs and 500 outputs per device. If more than this is required additional MODBUS Interface modules can be connected to the same MODBUS address (device).

Maximum number of devices: Although the maximum number of MODBUS devices connected to the IQ4 is determined by the number of available interface modules in the IQ4 (max 500), in reality, the limit is set by the MODBUS rules.

Maximum number of connected MODBUS networks: Each connected network requires one MODBUS Network module. There is a maximum of 10 MODBUS IP network modules. For Serial MODBUS network modules, the number is set by the IQ4's hardware.

Devices from different manufacturers can be connected to the same network providing the network parameters configuration is the same.

Note: IQ4 cannot operate as a slave so does not have registers.

12.2 M-bus

Maximum number of outputs per device: Each M-bus Interface module can have up to 1000 outputs, this allows 1000 outputs per device. If more than this is required additional M-bus Interface modules can be connected to the same M-bus address (device).

Note: An individual M-bus device is limited to 255 values. Therefore, although there can be 1000 outputs on the interface module, each one can only be set to one of 255 values form the M-bus device.

Maximum number of devices: Although the maximum number of interface modules is 500, the maximum number of M-bus devices is limited to 250 but is dependant on the converter used.

When values are requested from an M-bus device ALL values are sent to the IQ4. Therefore it is necessary to consider the data flow rate.

Note: The more values and devices that are required the larger the strategy requirements therefore the brIQ count must be considered.

Maximum number of connected M-bus networks: Each connected network requires one M-bus Network module. The number of M-bus net work modules is set by the IQ4's hardware.

Devices from different manufacturers can be connected to the same network providing the network parameters configuration is the same.

12.3 Configuring IQ Interfacing

Standard strategy blocks, available in IQSET, allow easy strategy configuration using drag & drop for common devices.

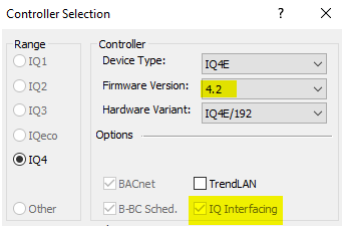
For each device, there are two strategy blocks, one labelled 'Energy Only' the other just the name of the meter. The 'Energy Only' strategy block only obtain the meter's energy reading. The other strategy block reads the most commonly used parameters.

If the required device is not available as a strategy block, or the block does not completely suit your requirements it is easy to create strategy to interface with it by:

- Adding and configuring an interface network module.
- Adding and configuring a Modbus Interface module.

To use the IQ interfacing strategy blocks:

1. Ensure the IQ4 that is to be used has the IQ interfacing feature licenced.
2. Create the controller in IQSET. Ensure that the firmware variant is set to 4.2 or greater, and the IQ Interfacing option selected.



3. Add a Modbus Serial, Modbus IP or M-bus Interface network module for the communications port that is to be used by the interface and configure it as necessary to enable communications with the device (e.g. Baud Rate). Refer the device documentation for details.

   Note: Only one interface network module can be connected to a physical port. Therefore all devices connected to that port must be capable of operating with the same network settings, e.g. baud rate.

   Note: The RS485, RS232 ports cannot be used for MS/TP, M-bus or MODBUS at the same time.

4. Drag the strategy block for the required device from the **Strategy Library** onto a strategy page. The **Strategy Block Questions and Answers** dialogue box will be displayed.
   - Specify the network module, the one added above
   - Enter the answer to the other questions in the appropriate boxes.

     The result of the text replacement for the selected question will be shown at the bottom of the dialogue box. Items that the answer will cause to be truncated will be indicated in red text, this enables you to adjust the answers to the question appropriately.

   - Click **OK**.

   *Note: Clicking **Preview** will display all items in the that contain placeholder text.*

5. Locate the Modbus or M-bus Interface module for the device and ensure the Address is set to the address of the device.
6. If there are parameters and functionality in the strategy block that are not required, you can delete the unwanted strategy, and, if required, save it your own strategy block.

PART 2

This part of the manual provides an overview of installing and configuring IQ4s. It is divided into the following sections:

Installing an IQ4
Installing I/O Modules
Creating a Strategy
Configuring an IQ4
Configure I/O Modules

1 INSTALLING AN IQ4

The installation of an IQ4 is described in the installation instructions for the individual variant:

- IQ41x Series Controllers Installation Instructions - Mounting (TG201250)
- IQ4NC, IQ422 Controllers Installation Instructions - Mounting (TG201264)
- IQ4E/.., IQ4NC/16/.., IQ4NC/32/.. Installation Instructions - Mounting (TG201338)

1.1 Installation Procedure

Mount
Connect Power
Connect Ethernet
Connect Inputs
Connect Outputs
Connect Auxiliary Supply (Not IQ41x)
Connect Trend Current Loop Network (IQ4NC and /LAN only)
Connect MS/TP Trunk (IQ4NC only)
Connect Wallbus (Not IQ422/00/XNC/.... or IQ4NC/00/... variants)
Connect to XNC 3rd Party System (/XNC only)
Install I/O Modules (IQ4E only)
Connect Serial MODBUS (/INT only)
Connect M-Bus (/INT only)

1.1.1 Mount

The IQ4 should be mounted on TS35 DIN rail complying to DIN EN500022 in an enclosure in a location that provides an environment that is within the IQ4's operating limits as specified in the IQ4's installation instructions.

1.1.2 Connect Power

The IQ4 should be connected to an appropriate power supply with an appropriate method of isolating it from the power supply as described in the IQ4's installation instructions.

1.1.3 Connect Ethernet

If the IQ4 is to use Ethernet to build LANs and internetworks, serve web pages, communicate over BACnet, or interface with a 3rd party system it must be connected to the Ethernet network using a standard Ethernet cable.

1.1.4 Connect Inputs

Connection of the external devices providing inputs to the IQ4 should be made as described in the IQ4's installation instructions. For details of the device being connected you should refer to the manufacturer's documentation.

Note: If the IQ4's inputs are to provide power to the connected devices, ensure that the power used by the inputs, outputs, auxiliary supply, RS232, Wallbus and I/O bus (IQ4E only) does not exceed the maximum available from the IQ4's combined supply. For details of calculating this see **"Calculate the Power Used by the Combined Supply" on page 116**.

1.1.5 Connect Outputs

Connection of the external devices which the IQ4 is controlling should be made as described in the IQ4's installation instructions. For details of the device being connected you should refer to the manufacturer's documentation.

Note: If the IQ4's outputs are to provide power to the connected devices, ensure that the power used by the inputs, outputs, auxiliary supply, RS232, Wallbus and I/O bus (IQ4E only) does not exceed the maximum available from the IQ4's combined supply. For details of calculating this see **"Calculate the Power Used by the Combined Supply" on page 116**.

1.1.6 Connect Auxiliary Supply

If the IQ4's auxiliary supply is to be used to power external devices it should be connected as described in the IQ4's installation instructions.

Note: Ensure that the power used by the inputs, outputs, auxiliary supply, RS232, Wallbus and I/O bus (IQ4E only) does not exceed the maximum available from the IQ4's combined supply. For details of calculating this see **"Calculate the Power Used by the Combined Supply" on page 116**.

Note: Not all IQ4 variants have an auxiliary supply.

1.1.7 Connect Trend Current Loop Network (IQ4NC and /LAN only)

If the IQ4 is to be connected to a Trend current loop network it should be connected to the Trend current loop network described in the IQ4's installation instructions.

1.1.8 Connect MS/TP Trunk (IQ4NC only)

If the IQ4NC is to be connected to an MS/TP trunk it should be connected to the MS/TP trunk as described in the IQ4's installation instructions.

1.1.9 Connect Wallbus (Not IQ422/00/XNC/.... or IQ4NC/00/... variants)

If devices are to be connected to the IQ4 via the Wallbus network they should be connected to the IQ4's Wallbus connector as described in the IQ4's installation instructions and the documentation supplied with the device. More than one device can be connected providing the Wallbus is able to provide enough power for each device.

Although the addressing allows 15 devices (the IQ4 plus 14 devices) it may not be possible to connect 14 devices to the IQ4 as the IQ4 may not be able to provide enough power. Therefore it is necessary to calculate whether there is enough power for all the required devices.

The power requirement of each Trend Wallbus device is specified as a percentage of the total Wallbus Power Budget (WPB). You should ensure that the WPB of all the connected Wallbus devices does not exceed 100%.

Note: If the Wallbus is to be used it requires 50mA. The power available for the Wallbus is affected by the demands on the combined supply and it is necessary to ensure that 50mA is available for use by the Wallbus - see **"Calculate the Power Used by the Combined Supply" on page 116**.

1.1.10 Connect to XNC 3rd Party System (/XNC only)

If the /XNC variant is to interface with a 3rd party system the connection must be made using the RS232 Port, RS485 Port, or Ethernet Port as required.

1.1.11 Install I/O Modules (IQ4E only)

If the IQ4E is to have its I/O count expanded by connecting external I/O modules. The required I/O modules should be connected. IQ4E supports Trend IQ4/IO modules, and Trend XCITE I/O modules.

The **'Installing I/O Modules'** section of this manual provides a brief overview of the process, but for detailed instructions refer to the installation instructions for the module(s) being used:

- IQ4/IO Modules Installation Instructions - Mounting (TG201342)
- XCITE/IO Standard I/O Modules Installation Instructions - Mounting (TG200627)

Note: Ensure that the power used by the inputs, outputs, auxiliary supply, RS232, and I/O bus (IQ4E only) does not exceed the maximum available from the IQ4's combined supply. For details of calculating - see **"Calculate the Power Used by the Combined Supply" on page 116**.

1.1.12 Connect Serial MODBUS (/INT only)

If the IQ4 is to connect to a serial MODBUS device it is necessary to connect to the MODBUS as described in the IQ4's installation instructions.

If connecting using the IQ4's RS232 connector it is necessary to use the RJ11 to 9 way female D type (CABLE/EJ101442) and an independently powered RS232 to RS485 convertor.

The convertor must be configured to match the communications settings on the MODBUS e.g. baud rate and data size (typically 19k2 baud and 11-bit data size) although it is dependant on the devices connected to the MODBUS and their configuration - refer to the documentation supplied with the convertor and the documentation for the other MODBUS devices for details. The IQ4 Type 8 (Modbus Serial) must also be configured with the same settings.

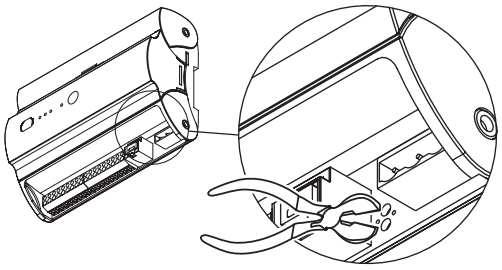
1.1.13 Connect M-Bus (/INT only)

If the IQ4 is to connect to a M-bus device it is necessary to connect to the M-bus as described in the IQ4's installation instructions using a suitable M-bus convertor.

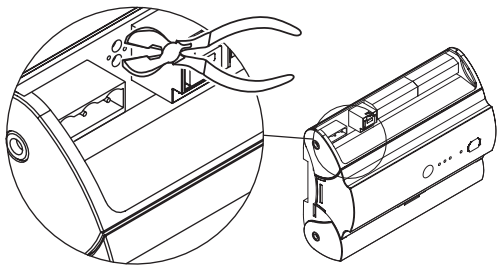
1.2 Cutting the Earth Link Track (/230 variants only)

On /230 variants it may be necessary to isolate the IQ4's electronics earth from the earth terminal on the IQ4's power connector in order to reduce electrical noise (e.g. if an input or output device (e.g. externally powered sensor) is earthed separately. This is done by cutting the earth link track (located on the board adjacent to the power connector) with suitable cutters.

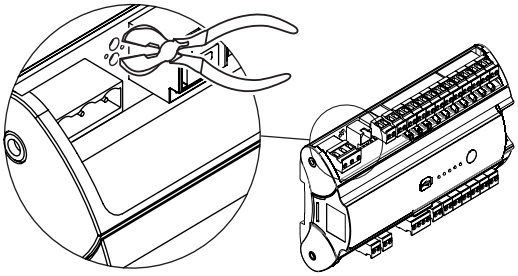
IQ41x



IQ422 or IQ4NC



IQ4E, IQ4NC/16/XNC or IQ4NC/32/XNC



1.3 Calculate the Power Used by the Combined Supply

The IQ4's combined supply provides power for:

- Inputs
- Outputs
- RS232 supply
- Auxiliary supply
- Wallbus
- I/O Bus (IQ4E only)

If the power requirement of the combined supply is more than the IQ4 can supply it will be necessary to power some to the items externally. The amount of power available depends on the IQ4 variant - see the appropriate data sheet.

Note: On IQ422/230, IQ41x/230 the combined supply can fulfil all power requirements up to the maximum specified on the appropriate data sheet.

The amount of power used by the combined supply can be calculated as follows:

- Power required for on board inputs and outputs, plus
- Power required for RS232 supply, plus
- Power required for Auxiliary supply, plus
- Power required for Wallbus, plus
- Power required for I/O Bus (IQ4E only)

Power required for on board inputs and outputs

The table below list the power requirements for each type of input/output:

Item	***Maximum Current Consumption***
I/O Universal Input	
Thermistor Input	0 mA
Voltage Input	0 mA
Digital Input	0 mA
Loop powered Current Input	20 mA
Externally powered Current Input	0 mA
Analogue Output	Dependant on what is connected, maximum 20 mA

The power requirement of the analogue output assumes that it is supplying its maximum load current. If this is not the case then the actual supply current should be used. The actual supply current can be used in the calculation if the calculation of an IQ4's power requirement returns a value that is too high, and it is certain that the outputs can never be fully ON.

Power Required for RS232 Supply

The power required by the RS232 port depends on the device that is connected to the port see the appropriate data sheet for details.

Power Required for Auxiliary Supply

The power required by the auxiliary supply depends on what is connected to the supply see the appropriate data sheet for details.

Power Required for Wallbus

If devices are to be connected to the Wallbus 50mA is required. Therefore if Wallbus devices are to be connected to ensure that the other demands on the combined supply leave 50mA available for the Wallbus.

Power Required for I/O Bus (IQ4E only)

The power required for the I/O bus is dependant on the power consumption of the I/O modules connected to it - see **"Calculating I/O Module Power Requirements" on page 129**.

2 CREATING A STRATEGY

The IQ4 strategy is created using Trend's System Engineering Tool (IQSET). Strategy modules can only be interconnected, created, or deleted using IQSET.

IQSET facilitates strategy creation by providing a library of pre-configured 'Strategy Blocks'. Similarly complete strategies are available as 'Solutions'.

This section describes the process of creating a strategy.

Plan the System
Create a IQSET Project and Add Controllers
Configure the Strategy
Test the Strategy
Create the Strategy File

For details of using IQSET see the IQSET Manual (TE200147), for details of using IQ4 web pages see the IQ4 Web User Guide (TC201256).

2.1 Plan the System

Good planning is necessary prior to starting work because it reduces the actual amount of time spent engineering the system, reduces the number of errors, and produces a better result. It is important to understand the HVAC equipment that is being controlled, and the hardware that interfaces with it, to ensure that the strategy is written so that the IQ4 correctly interfaces with the HVAC equipment. The HVAC equipment will need to be maintained; therefore it is necessary to know the maintenance intervals of the pieces of HVAC equipment so that this can be built into the strategy. It is also necessary to understand how the system is to be used so that the appropriate adjustments can be included. This is helpful in deciding on the setpoints, OSS settings etc.

When engineering a system that uses the Trend network, it is not only necessary to understand the requirements of the control strategy within individual controllers; it is also necessary to understand how various controllers interact with each other. This enables IC communications and the devices' addresses to be set up correctly.

If the strategy is to be BTL compliant you should follow the advice in the **"Creating a BTL Compliant Strategy" on page 117**.

2.1.1 Creating a BTL Compliant Strategy

The IQ4 firmware is certified as a BACnet Building Controller (B-BC). Compatibility is defined in IQ4 Product Implementation Conformance Statement (TP201248) and the IQ4NC/.../ BACnet Protocol Implementation Conformance Statement (TP201286).

By default /BAC variants or IQ4NCs can communicate over a BACnet network when correctly configured. For compliance with the requirements of the BACnet standard, the following steps need to be taken:

- Enable the Type 5 (BACnet IP) Network module.
- Ensure that the following parameters in the Type 13 (BACnet Application) Network module are enabled.
  - BACnet Backup/Restore
  - Cold & Warm Restart Service
  - BACnet Writable Properties
  - and the 'BO State Count And Active Time Properties' parameter is disabled
- If alarms are to be sent over BACnet:
  - Only connect one Alarm Route module to each BACnet Alarm Destination module. The Alarm Route module should not be connected to any other modules.
  - Only connect a single BACnet Alarm Destination module to each BACnet Alarm Route module's 'Enable' parameter.

2.2 Create a IQSET Project and Add Controllers

In order to create the strategy for an IQ4 using IQSET, a project file has to be created and the necessary IQ4 added. See the IQSET Manual (TE200147) for more details.

2.3 Configure the Strategy

To configure the strategy the following process should be followed:

Set up the Address Module
Set up the Type 1 (IQ LAN) Network Module (/LAN variants and IQ4NC only)
Set up the Type 4 (Ethernet IP) Network Module
Set up the Type 12 (BACnet MSTP) Network Module (IQ4NC only)
Set up the Type 5 (BACnet IP) Network Module (/BAC variants and IQ4NC only)
Set up the Type 13 (BACnet Application) Network Module (/BAC variants and IQ4NC only)
Set up the Sensor Type Modules
Add States Category Modules
Write TCL Application (/XNC variants only)
Set up the Type 17 (XNC Serial) Network Module (/XNC variants only)
Set up the Type 1 (XNC) Interface Module (/XNC variants only)
Add and Link Modules
Set up the Alarm Handling Strategy
Add I/O Modules (IQ4E only)
Assign I/O Channels
Set up Strategy for Use with Wallbus Devices (v3.30 or greater)
Edit Module Parameters
Set up the Sequence Table
Set up Event Driven Strategy
Set up Daily Profile
Set up Calendar Modules (v3.50 or greater)
Set up Exceptions
Set up IC Communications
Set up Security
Set up CNC User Modules (v3.60 or greater)
Set up Virtual CNC Modules (v3.60 or greater)
Configure Web Servers
Set up Display and Directory Modules

2.3.1 Set up the Address Module

The Address module determines information about how the IQ4 communicates over the Trend network. The following parameters should be set up:

Identifier
Attributes 2, 3, 4, 5, 6 and 7
Local LAN
Local Address
Supervisor Port (if required)
USB Supervisor Port (if required)
General Alarm Group
Language Selection

Note: The recommended method of configuring the 'Local Lan', 'Local Address', 'Supervisor Port', and 'USB Supervisor Port' parameters is as part of configuring the IQ4 (see **"Configuring an IQ4" on page 131)** *using IQTool Addresser Applet and IP Tool before the strategy file is downloaded, i.e not using IQSET. These parameters can be specified in IQSET and then excluded from the download file if required - see* **"Parameter Download Exclusions" on page 136***. If these parameters are excluded from the strategy file they will still be uploaded by IQSET.*

For details of the module and its parameters see **"Address Module" on page 167**.

Note: When IQ4s are on a Trend current loop LAN, their LAN number for communication purposes is dictated by the INC type node for the LAN. However, for their alarm communications to operate correctly the Local LAN parameter should be set to the same value.

2.3.2 Set up the Type 1 (IQ LAN) Network Module (/LAN variants and IQ4NC only)

Type 1 (IQ LAN) Network modules determine the parameters specific to the Trend current loop network. It should only be set up for IQ4s that are to build LANs and internetworks on the Trend current loop. It must be disabled for IQ4s (except IQ4NC) that are to build LANs and internetworks over an Ethernet network. For Ethernet only IQ4s this module must always be disabled. The following parameters should be set up using IQSET:

Label
Disable Module
LAN Label
LAN Number (IQ4NC only)
Trend Mode (IQ4NC only)
Baud Rate
Alarm Group

Note: The recommended method of configuring the 'Autobaud' and 'Baud Rate' parameters is as part of configuring the IQ4 (see **"Set up Current Loop Baud Rate (/LAN or IQ4NC only)" on page 135)** *using IP Tool before the strategy file is downloaded. These parameters can be specified in IQSET and then excluded from the download file if required - see* **"Parameter Download Exclusions" on page 136***. If these parameters are excluded from the strategy file they will still be uploaded by IQSET.*

Note: For IQ4NCs the recommended method of configuring the 'Disable Module', 'LAN Number' and 'Trend Mode' parameters is as part of configuring the IQ4 (see **"Set up IQ4NC Communication Mode (IQ4NC only)" on page 134)** *using the IQ4s web pages before the strategy file is downloaded, i.e. not using IQSET. These parameters can be specified in IQSET and then excluded from the download file if required - see* **"Parameter Download Exclusions" on page 136***. If these parameters are excluded from the strategy file they will still be uploaded by IQSET.*

For details of the module and its parameters see **"Type 1 (IQ LAN)" on page 390**.

2.3.3 Set up the Type 4 (Ethernet IP) Network Module

The Type 4 (Ethernet IP) Network module determines the Ethernet settings for IQ4. It needs to be set up if the IQ4 is to connect to the Trend network using Ethernet, if its vCNC is to be used, if IP alarms are to be used, or the web server is to be accessed. The following parameters should be set up using IQSET:

LAN Label
IP Address
Subnet Mask
Router 1
UDP Port
WINS Servers
DNS Servers
Standalone Mode
Web Server Port Number or Secure Web Server Port Number - see **"Configure Web Servers" on page 124**.
XML Web Services (if required)
Alarm Group
Email Server Address (if required)
Email Username (if required)
Email Password (if required)
Hostname

The recommended method of configuring the 'IP address', 'Subnet Mask', 'Router 1', 'WINS Servers', 'DNS Servers', 'Remote Devices', and 'Remote Devices Subnet' parameters is as part of configuring the IQ4 (see **"Configuring an IQ4" on page 131**) using IP Tool before the strategy file is downloaded, i.e. not using IQSET. These parameters can be specified in IQSET and then excluded from the download file if required. If these parameters are excluded from the strategy file they will still be uploaded by IQSET.

For details of the module and its parameters see **"Type 4 (Ethernet IP)" on page 392**.

2.3.4 Set up the Type 12 (BACnet MSTP) Network Module (IQ4NC only)

Type 12 (BACnet MSTP) Network modules determine the MS/TP settings. The following parameters should be set up using IQSET:

Label
Disable Module
LAN Number
LAN Label
Trend Mode
Manual MAC Address
MAC Address (only if MAC address is to be set manually)
Baud Rate
Manual Network Number
Network Number (only if network number is to be set manually)
Alarm Group
Max Info Frames
Max Master
Service Class

Note: The recommended method of configuring the 'Disable Module', 'LAN Number' and 'Trend Mode' parameters is as part of configuring the IQ4 (see **"Set up IQ4NC Communication Mode (IQ4NC only)" on page 134)** *using the IQ4's web pages before the strategy file is downloaded not using IQSET.*

For details of the module and its parameters see **"Type 12 (BACnet MSTP)" on page 400**.

2.3.5 Set up the Type 5 (BACnet IP) Network Module (/BAC variants and IQ4NC only)

Type 5 (BACnet IP) Network module determines the IQ4's BACnet IP settings. It needs to be to be set up if the IQ4 is to connect to a BACnet network on Ethernet. The following parameters should be set up using IQSET:

Label
Disable Module
UDP Port
BBMD
Network Number

Once the module is enabled the default settings will normally allow the IQ4 to communicate over BACnet. For details of the module and its parameters see **"Type 5 (BACnet IP)" on page 397**.

Note: There are certain issues with communicating over BACnet where the IQ4 is set up for automatic IP addressing - see **"BACnet with Automatic IP Addressing" on page 68**.

2.3.6 Set up the Type 13 (BACnet Application) Network Module (/BAC variants and IQ4NC only)

Type 13 (BACnet Application) Network modules determine the IQ4's BACnet settings. It needs to be to be set up if the IQ4 is to connect to a BACnet network. The following parameters should be set up using IQSET:

Label
Manual device instance
Device instance (if 'Manual device instance' parameter set to 'Manual')
Send I-Am at Startup
BACnet Backup/Restore
Cold & Warm Restart Service
BACnet Writable Properties
Create/Delete Service
State Count And Active Time Properties' parameter
Object Name Prefixes

For details of the module and its parameters see "Type 13 (BACnet Application)" on page 401.

2.3.7 Set up the Sensor Type Modules

The IQ4's Sensor Type modules need to be set up to define how the sensors scale their inputs. IQSET allows the Sensor Type modules to be defined either using definitions stored in a library of sensor definitions or by entering manually.

2.3.8 Add States Category Modules

If multistate values are to be used in the strategy States Category modules must be added.

2.3.9 Write TCL Application (/XNC variants only)

If configuring an /XNC variant it is necessary to write the TCL application that is to define the interface to the 3rd party system using TCLTool 2 - see the TCLTool 2 Manual (TE200923) and IQ4/XNC TCL Manual (TE201360).

2.3.10 Set up the Type 17 (XNC Serial) Network Module (/XNC variants only)

If configuring an /XNC variant that is to communicate with a 3rd Party system via a serial connection it is necessary to set up a Type 17 (XNC Serial) Network module for each different serial connection that is to be used.

For details of the module and its parameters see "Type 17 (XNC Serial)" on page 408.

2.3.11 Set up the Type 1 (XNC) Interface Module (/XNC variants only)

If configuring an /XNC variant it is necessary to set up the Type 1 (XNC) Interface module to define the Alarm Group module to which runtime alarms are sent, the user alarm labels, the initial value of the Store module's strings, the linking of the TCL application's inputs and outputs to modules in the strategy, and if interfacing using a serial connection the number of the Type 17 (XNC Serial) Network module that specifies the serial connection.

For details of the module and its parameters see "Type 1 (XNC)" on page 309.

2.3.12 Add and Link Modules

Adding and linking modules is the main part of configuring the control strategy. The modules should be added to the strategy pages in IQSET, and linked as appropriate to achieve the desired functionality.

For details of the modules and their parameters see "IQ4 Modules" on page 165.

2.3.13 Set up the Alarm Handling Strategy

The alarm handling part of the strategy provides the facility to send different alarms to different destinations, or to determine when the alarms are sent. It also enables alarms to be sent to another destination if the preferred destination is not available.

For details of how the alarm handling strategy works see "Alarm Handling Strategy" on page 79.

2.3.14 Add I/O Modules (IQ4E only)

If an IQ4E is to use external I/O modules a strategy I/O module must be added to the strategy and appropriately configured for each external I/O module so that the Driver, Type 0 (Analogue) Sensor, Type 2 (Digital) Sensor, and Type 2 (External) Digital Input modules can be connected to them. The following parameters should be set up using IQSET:

Label
Configured Type
Address

For details of I/O module parameters see "IQ4 Modules" on page 165.

2.3.15 Assign I/O Channels

The actual physical piece of hardware from which, Type 0 (Analogue) Sensor, Type 2 (Digital) Sensor, and Type 2 (External) Digital Input modules obtain their inputs, and where Driver modules send their outputs to must be specified. This is done by setting the 'Hardware Module' parameter, and 'Hardware Channel' parameter or the 'Hardware Module' parameter and 'In-Phase Hardware Channel' and 'Anti-phase Hardware Channel' parameters for Driver modules - see **"Connection to I/O Channels" on page 26**.

2.3.16 Set up Strategy for Use with Wallbus Devices (v3.30 or greater)

If Wallbus devices are to be connected to the IQ4 the IQ4's strategy must be configured appropriately:

- The Type 14 (Wallbus) Network module must be added to the strategy.

  Note: Ensure that the module is enabled, 'Disable Module' parameter set to '0'.

- A Type 2 (Wallbus) Interface module must be added for each device that is to be connected to the IQ4's Wallbus network and configured as required. The module's 'Address' parameter must be set to the address of the Wallbus device on the Wallbus. The module's inputs and outputs must be connected to the modules that they are to read values from, or write values to, and each input and output must be connected to the appropriate Process Variable (PV) in the Wallbus device. Refer to the Wallbus device's data sheet for details of the values that are available and the PVs used.

  Strategy blocks for each of the supported Trend Wallbus devices are provided in IQSET to facilitate connection of those displays. If using a non-standard configuration these standard strategy blocks can be edited after they have been added to the strategy.

For details of the Type 14 (Wallbus) Network module's parameters see **"Type 14 (Wallbus)" on page 405**, and for details of Type 2 (Wallbus) Interface module's parameters see **"Type 2 (Wallbus)" on page 312**.

2.3.17 Edit Module Parameters

Once the modules have been added to the strategy and linked, and the alarm handling strategy set up the module's parameters can be configured to change them from the default values to ensure that they perform the required tasks.

For details of the modules and their parameters see **"IQ4 Modules" on page 165**.

2.3.18 Set up the Sequence Table

The sequence table must be set up using IQSET to ensure that the strategy is serviced in the correct order.

For details of module servicing see **"Module Servicing" on page 26**.

2.3.19 Set up Event Driven Strategy

Event driven strategy can be used to cause a particular piece of strategy to be processed when an event occurs. It is designed for applications which must have an immediate response to an input e.g. for lighting strategies where the light must be turned on as soon as a switch operated. If this type of operation is required it should be set up using IQSET to ensure that the required modules are not in the sequence table.

For details of event driven strategy see **"Event Driven Servicing" on page 28**.

2.3.20 Set up Daily Profile

The IQ4's Time Schedule module must be configured with the required daily profile to allow the strategy to control as expected.

For details of Time Schedule modules and their parameters see **"Time Schedule Modules (Time Zone Modules)" on page 495**.

Note: On most systems the daily profile will be controlled by a supervisor.

2.3.21 Set up Calendar Modules (v3.50 or greater)

If Calendar modules are to be used to specify the date(s) for an exception he require module must be added to the strategy.

For details of the Calendar modules and their parameters see **"Calendar Modules" on page 189**.

2.3.22 Set up Exceptions

For IQ4 controllers any times that the controller is to be in occupancy that are different to the normal occupation time profile can be specified by setting up exceptions.

For details of the Exceptions and their parameters see **"Time Schedule Modules (Time Zone Modules)" on page 495**.

2.3.23 Set up IC Communications

If the IQ4 is to share information with other IQ controller's or BACnet devices (/BAC or IQ4NC only) the IQ4's IC communications modules must be set up using IQSET.

For details of the IC comms modules and their parameters see **"IC Comms Modules" on page 279**.

Note: If the IQ4 is receiving data to, or global to IC comms messages from a pre IQ3 controller, Analogue Node or Digital Byte modules must be added to receive the data. E.g. if an IC comms message being sent to analogue node 104 is to be received by an IQ4, Analogue Node module 104 must be added to the IQ4's strategy. In a similar way if an IQ4 is to receive a bit or byte value from a pre IQ3 controller that is to be written to byte 47, Digital Byte module 47 must be added to the IQ4's strategy.

2.3.24 Set up Security

The IQ4 is shipped from the factory with its security disabled. The IQ4's Type 0 User Modules MUST be set up to prevent any unauthorised access to the IQ4. It is recommended that:

- The 'Guest' user is set up to control access to the IQ4's web pages for the casual user.
- Type 0 User Modules are created for each user who is going to access the IQ4's web pages.
- A Type 0 User Module configured with network access to provide protection for text communications e.g. for use by a supervisor or display and file upload/download.

Note: If installing the IQ4 into a legacy system with no network comms protection then it is not necessary to configure a Type 0 User Module with network access. Any Type 0 User Modules should be configured with network use disabled - see **"Network Use (v3.30 or greater)" on page 104**.

Note: By default security is disabled (i.e. no Type 0 User Modules). When the first Type 0 User Module is created it must have a user level of 99 or greater. Further Type 0 User Modules can then be created giving restricted access.

Note: If the 'Guest' user is set up without a password, or if any Type 0 User Module is created with a password less than 99, it will not be possible to download strategy files unless there is a Type 0 User Module set up with a user level of '>=99'.

For details of the Type 0 User Modules and their parameters see **"Type 0 User Modules" on page 511**, and for details of how IQ4 security works see **"Security" on page 99**.

2.3.25 Set up CNC User Modules (v3.60 or greater)

If clients (e.g. IQSET) are to connect to a vCNC in the IQ4 that is operating in secure mode Type 1 CNC User modules must be set up to specify the username and password required to connect to the vCNC - see **"Set up Virtual CNCs (if required)" on page 133**.

For details of the Type 1 CNC User Modules and their parameters see **"Type 1 CNC User Modules (v3.60 or greater)" on page 515**.

2.3.26 Set up Virtual CNC Modules (v3.60 or greater)

If clients (e.g. IQSET) are to connect to a vCNC in the IQ4 that the controller's vCNC modules must be set up. This can be done by:

Adding the vCNC modules and CNC User modules to the strategy as part of the strategy creation and downloading them to the IQ4 via another connection.
From IQSET's System View once the controller is connected to the Trend network, requires another connection. By using IP Tool to enable them and then connecting to the vCNC using IQSET, requires physical access to the controller.

For more details see **"Set up Virtual CNCs (if required)" on page 133**.

For details of the Virtual CNC Modules and their parameters see **"Virtual CNC Modules" on page 517**.

2.3.27 Configure Web Servers

The IQ4 is shipped from the factory with its HTTP web server enabled. If the web servers are not required they should be disabled. If a web server is required the HTTPS web server should be used.

Configure the HTTP Web Server

- Set the Type 4 (Ethernet IP) Network module's 'Web Server Port Number' parameter to the port number that is to be used (default 80). To disable the HTTP web server set the 'Web Server Port Number' parameter to '0'.

Configure the Using the HTTPS Web Server

- Set the Type 4 (Ethernet IP) Network module's 'Secure Web Server Port Number' parameter to the port number that is to be used (e.g 443). To disable the HTTPS web server set the 'Secure Web Server Port Number' parameter to '0'.

  Note: When HTTPS is first enabled the IQ4 will automatically generate a self signed certificate, however this is not suitable for long term use, and a full trusted certificate should be created as soon as possible - see **"Configure SSL Certificates (if required)" on page 136**.

- If required use the Type 4 (Ethernet IP) Network module's 'Redirect to https' parameter to specify whether web server requests to the HTTP server are redirected to the HTTPS web server.
- Configure the SSL certificates - see **"Configure SSL Certificates (if required)" on page 136**.

For details of Type 4 (Ethernet IP) Network modules and their parameters see **"Type 4 (Ethernet IP)" on page 392**.

Note: If disabling the web servers via the IQ4's web pages it will not be possible to enable them again using the IQ4's web pages.

2.3.28 Set up Display and Directory Modules

Display modules and Directory modules enable a set of displays to be defined within the IQ4. Each display can show a mixture of different item types, e.g. Sensor module values, Knob module values etc, and, where appropriate, enable adjustments to be made, and values to be graphed. They can then be viewed on supervisory products such as the IQView4. They are also used to define the IQ4's GraphIQs pages. If either of these facilities are required it is necessary to set up the Display and Directory modules using IQSET.

For details of how Display and Directory modules are used together see **"Display and Directory Modules" on page 31**. For details of Directory modules and their parameters see **"Directory Modules" on page 201**. For details of Display modules and their parameters see **"Display Modules" on page 203**.

2.4 Test the Strategy

Once the strategy is complete it may be tested prior to download by using the IQSET's simulation mode.

2.5 Create the Strategy File

Once the strategy is complete it is necessary to create the strategy file (.IQ4) ready to be downloaded to the IQ4 using IQSET

3 INSTALLING I/O MODULES

The IQ4E, IQ4NC/16/XNC or IQ4NC/32/XNC can connect to IQ4/IO modules and XCITE I/O modules.

3.1 IQ4/IO Module Installation Procedure

The installation of IQ4/IO modules is described in the following document:

- IQ4/IO Modules Installation Instructions - Mounting (TG201342)

This section provides a summary of this information.

Mount
Connect I/O Bus
Connect Power
Terminate I/O Bus
Connect Inputs
Connect Outputs

3.1.1 Mount

The IQ4/IO modules should be mounted in an enclosure in a location that provides an environment that is within the module's operating limits as specified in the module's installation instructions on TS35 DIN rail complying to DIN EN500022.

3.1.2 Connect I/O Bus

IQ4/IO modules interconnect with each other and the IQ4 via the IQ4's I/O bus. A short rigid interconnector is available to allow easy connection between adjacent IQ4/IO modules. One interconnector is supplied with each I/O module; spare interconnectors are also available.

Where modules are mounted separately from the IQ4 (or from neighbouring I/O modules) the interconnection may be made using a breakout connector and the specified screened cable. The cable type required is determined by the total bus length - see the IQ4/IO Modules Data Sheet (TA201341).

3.1.3 Connect Power

IQ4/IO modules are typically powered from the IQ4 via the I/O bus. This is derived from the IQ4's combined 24 Vdc supply and for most installations this will be sufficient.

However, in some circumstances the IQ4 will not be able to supply enough power:

- Module power requirements higher than power available for I/O bus
- Power due to length of cable run

To determine whether the IQ4 will be able to power the connected I/O modules it is necessary to calculate the maximum power required by all the I/O modules - see **"Calculating I/O Module Power Requirements" on page 129**, and compare it with the amount of power the IQ4 can supply to the I/O bus - see **"Calculate the Amount of Power the IQ4 can Supply to the I/O Bus" on page 129**.

In both of these circumstances an external 24 Vdc supply can be used to provide additional power to the bus - see **"Using a Additional 24 Vdc Supply" on page 129**.

Additionally the maximum current that can flow through the power line of a single IQ4/IO module must not exceed 1 A. To meet this requirement the I/O bus power can be split into separately powered groups of modules, each group consuming less than 1 A.

3.1.4 Terminate I/O Bus

The I/O bus must be terminated at the farthest end from the IQ4. If the module is at the end of the bus a suitable plug-in terminator must be used to terminate the bus at the I/O module.

3.1.5 Connect Inputs

Connection of the external devices providing inputs to the IQ4 should be made as described in the I/O module's installation instructions. For details of the device being connected you should refer to the manufacturer's documentation.

3.1.6 Connect Outputs

Connection of the external devices which the IQ4 is controlling should be made as described in the I/O module's installation instructions. For details of the device being connected you should refer to the manufacturer's documentation.

3.2 XCITE I/O Module Installation Procedure

The installation of XCITE I/O modules is described in the following documents:

- XCITE/IO Standard I/O Modules Installation Instructions - Mounting (TG200627)

This section provides a summary of this information.

Mount
Connect I/O Bus
Connect Power
Terminate I/O Bus
Connect Inputs
Connect Outputs

3.2.1 Mount

The XCITE I/O modules should be mounted in an enclosure in a location that provides an environment that is within the module's operating limits as specified in the module's installation instructions on TS35 DIN rail complying to DIN EN500022.

3.2.2 Connect I/O Bus

XCITE I/O modules interconnect with each other and the IQ4 via the IQ4's I/O bus. A cable is required to make the connection between the IQ4 and an XCITE I/O module. A short rigid interconnector (different to that required for connecting IQ4/IO modules) is available to allow easy connection between adjacent XCITE I/O modules. One interconnector is supplied with each I/O module; spare interconnectors are also available.

Where modules are mounted separately from the IQ4 (or from neighbouring I/O modules) the interconnection may be made using a breakout connector and the specified screened cable. The cable type required is determined by the total bus length - see the XCITE/IO/.. Standard I/O Module Data Sheet (TA201352).

3.2.3 Connect Power

XCITE I/O modules are typically powered from the IQ4 via the I/O bus. This is derived from the IQ4's combined 24 Vdc supply and for most installations this will be sufficient.

However, in some circumstances the IQ4 will not be able to supply enough power:

- Module power requirements higher than power available for I/O bus
- Power due to length of cable run

To determine whether the IQ4 will be able to power the connected I/O modules it is necessary to calculate the maximum power required by all the I/O modules - see **"Calculating I/O Module Power Requirements" on page 129**, and compare it with the amount of power the IQ4 can supply to the I/O bus - see **"Calculate the Amount of Power the IQ4 can Supply to the I/O Bus" on page 129**.

In both of these circumstances an external 24 Vdc supply can be used to provide additional power to the bus - see **"Using a Additional 24 Vdc Supply" on page 129**.

Additionally the maximum current that can flow through the power line of a single XCITE I/O module must not exceed 1.6 A. To meet this requirement the I/O bus power can be split into separately powered groups of modules, each group consuming less than 1.6 A.

3.2.4 Terminate I/O Bus

The I/O bus must be terminated at the farthest end from the IQ4. If the module is at the end of the bus a suitable plug-in terminator (different to that required for IQ4/IO modules) must be used to terminate the bus at the I/O module.

3.2.5 Connect Inputs

Connection of the external devices providing inputs to the IQ4 should be made as described in the I/O module's installation instructions. For details of the device being connected you should refer to the manufacturer's documentation.

3.2.6 Connect Outputs

Connection of the external devices which the IQ4 is controlling should be made as described in the I/O module's installation instructions. For details of the device being connected you should refer to the manufacturer's documentation.

3.3 Calculate the Amount of Power the IQ4 can Supply to the I/O Bus

The power available to the I/O bus is provided by the IQ4's combined supply. **"Calculate the Power Used by the Combined Supply" on page 116** describes how to calculate the power used by the combined supply.

In order to calculate the amount of power available for the I/O bus exclude the I/O bus consumption in the combined supply power calculation and subtract the I/O bus current from the calculated power available to the combined supply. The amount of power available depends on the IQ4 variant - see the appropriate IQ4 data sheet.

3.4 Calculating I/O Module Power Requirements

The power requirements for the I/O modules connected to an IQ4 can be calculated by adding the power requirement for each connected I/O module (IQ4/IO and XCITE I/O).

The current required by each type of IQ4/IO module is described in the IQ4/IO Modules Data Sheet (TA201341). The IQ3 Configuration Manual (TE200768) provides details of calculating the power consumptions of XCITE I/O modules. These values can be used to estimate the total current required by the I/O modules connected to an IQ4.

3.5 Using a Additional 24 Vdc Supply

If the power requirement for the I/O bus exceeds the maximum available from the IQ4, an external 24 Vdc supply can be used.

Note: This type of installation is NOT UL compliant.

3.6 Mixing IQ4/IO and XCITE I/O Modules

IQ4 is compatible with IQ4/IO and XCITE I/O modules however if the installation uses both types of I/O module there are some limitations:

- **I/O bus length reduced:** The maximum length of the I/O bus is reduced to 30m.
- **Only cable interconnections can be used:** When connecting an XCITE/IO module to an IQ4/IO module or IQ4E a rigid interconnector cannot be used, the connection must be made with a cable interconnector.

  Note: Check the wiring carefully as the cable connections are reversed on either side of an XCITE/IO module - see XCITE/IO/.. Standard I/O Module Installation Instructions - Mounting (TG200627).
- **Number of I/O module reduced:** It is recommended that a maximum of 15 I/O modules.

3.7 Replacing an XCITE I/O Module with an IQ4/IO Module

When replacing an XCITE I/O module with an IQ4/IO module there are a number of things to be considered:

- Physical space - IQ4/IO modules are smaller and will therefore fit in the space of the XCITE I/O module but cabled connections are required to connect XCITE I/O module to IQ4/IO modules which take up more space so it may be necessary to move the modules - see the IQ4/IO Modules Installation Instructions - Mounting (TG201342).
- If cabled connections are to be used the existing cable may need to be replaced as IQ4/IO modules have different cable adaptors - see the IQ4/IO Modules Installation Instructions - Mounting (TG201342).
- I/O connections may be different therefore the I/O may need to be rewired - see the IQ4/IO Modules Installation Instructions - Mounting (TG201342).
- IQ4/IO modules only allow 1A to pass through compared with 1.6 A allowed by XCITE I/O modules, therefore the power distribution may need to be changed - see the IQ4/IO Modules Installation Instructions - Mounting (TG201342).
- Controller strategy will need to be modified to specify the new I/O module.

4 CONFIGURING AN IQ4

The configuration of an IQ4 is described in the installation instructions for the individual variant:

- IQ41x Series Controllers Installation Instructions - Configuring (TG201251)
- IQ4NC, IQ422 Controllers Installation Instructions - Configuring (TG201265)
- IQ4E/.. IQ4NC/16/..., IQ4NC/32/... Installation Instructions - Configuring (TG201339)

This section provides a summary of the information in these documents.

Prepare Ethernet Network
Configure I/O Modules (IQ4E only)
Connect PC/Laptop
Check Watchdog/Power Indicators
Check Ethernet Indicators
Connect to IQ4 Using IP Tool
Set up IP Addressing Details
Set up Virtual CNCs (if required)
Set up IQ4NC Communication Mode (IQ4NC only)
Prepare Ethernet Network
Check MS/TP Trunk Indicators (IQ4NC only)
Set up Ethernet Access Across Routers
Set up Trend Addressing
Set up Current Loop Baud Rate (/LAN or IQ4NC only)
Set up RS232/USB Port Addressing
Check LAN OK Indicator (/LAN only)
Download Strategy
Check Input & Output Operation
Connect Supervisors/Tools/Display Panels (if required)
Configure SSL Certificates (if required)
Backup the Strategy

4.1 Prepare Ethernet Network

If the IQ4 is to connect to an Ethernet network it is necessary to ensure that the network is correctly configured.

- If the IP address settings (IP address, subnet mask, router 1 (default router), WINS and DNS servers' addresses) are to be supplied by a DHCP server, ensure one is installed on the Ethernet segment to which IQ4 is connected.
- If host names are to be used for IP addressing across a router, a WINS server must be installed.
- If email alarms are to be sent, and the email server address is identified by internet domain name, a DNS server must be installed.

4.1.1 Configure Firewalls

Firewalls on the Ethernet network must be configured to allow communication between controllers that need to communicate with each other, and any clients that need to connect to the controller.

4.1.1.1 Connect to Controller's vCNC

If connecting to the a vCNC in the controller all firewalls between the client and the controller must be configured to allow communication. The following must be allowed:

SSH protocol
Port number used for SSH communications, specified by the 'Secure Access Port' parameter in the Type 4 (Ethernet IP) Network module, default 22.

4.1.1.2 Connect to Controller's Web Server

If connecting to the controller's web server all firewalls between the client and the controller must be configured to allow communication.

Connecting to HTTPS Web Server (recommended)
The following must be allowed:

HTTPS protocol
Port number used by the web server, specified by the 'Secure Web Server Port Number' parameter in the Type 4 (Ethernet IP) Network module, default 443.

Connecting to HTTP Web Server (not recommended)
The following must be allowed:

HTTP protocol
Port number used by the web server, specified by the 'Web Server Port Number' parameter in the Type 4 (Ethernet IP) Network module, default 80.

4.2 Configure I/O Modules (IQ4E only)

If an IQ4E is using I/O modules they should be configured so that each has a unique address on the I/O bus, and that address matches the address of the I/O module in the strategy that connects physical input and outputs to the Driver, Type 0 (Analogue) Sensor, Type 2 (Digital) Sensor, and Type 2 (External) Digital Input modules - see **"Configure I/O Modules" on page 159**.

4.3 Connect PC/Laptop

To configure the IQ4 a PC/laptop should be connected to the IQ4 either directly to its Ethernet port or by connecting it to the same Ethernet network segment as the IQ4.

4.4 Check Watchdog/Power Indicators

Check that the Power indicator is ON, and the watchdog indicator is OFF. If the power indicator is not ON check the input power supply. If the Watchdog indicator is ON the device is faulty and should be reset to factory defaults.

4.5 Check Ethernet Indicators

If the IQ4 is connected to an Ethernet network check that the Ethernet OK indicator is ON and the Ethernet RX indicator is flashing. If the Ethernet OK indicator is OFF check the Ethernet connection.

4.6 Connect to IQ4 Using IP Tool

Connect to the Ethernet network to which the IQ4 is connected using IP Tool, and select the IQ4 from the list of Trend devices. The IQ4 can be identified by its MAC address which is located on the label, and is displayed in IP Tool.

4.7 Set up IP Addressing Details

If there is an Ethernet connection the following parameters need to be set up:

IP Address
Subnet Mask
Router 1
WINS Server

Note: If sending email alarms, and the email server is identified by a domain name, a DNS server must be specified.

Note: If necessary the UDP port can also be changed.

This can be done automatically (default) or manually by specifying the parameters using IP Tool - see **"IP Addressing" on page 50**.

4.8 Set up Virtual CNCs (if required)

A Virtual CNC (vCNC) must be set up if connection to a supervisor tool or display is required. This can be done by:

Adding the vCNC modules and CNC User modules to the strategy
From IQSET's System View
Using IP Tool to enable it and then connecting to the vCNC using IQSET

Note: For IQ4 v3.60 or greater when a vCNC is enabled it will be configured to operate in 'Secure' mode. To connect to a secure vCNC it is necessary to configure a Type 1 CNC User Module. If the vCNC is to operate in 'Unsecured' mode (see **"Security Mode 'None'" on page 517***) it is necessary to change to mode to during the set up process.*

Note: Secure vCNC connections use an SSH port. If the default is not suitable e.g. the port is being used on the network for something else, the SSH port can be changed using the 'Secure Access Port' parameter in the Ethernet IP Network module.

4.8.1 Adding the vCNC modules and CNC User Modules to the Strategy (v3.60 or greater)

vCNC modules and CNC User modules can be added to the strategy and then downloaded to the controller using IQSET. This requires IQSET to have access to the IQ4 over the Trend network IQ4 via another connection.

See the IQSET Manual (TE200147) for details.

4.8.2 From IQSET's System View (v3.60 or greater)

vCNC modules can be set up by learning the network the IQ4 is connected to in IQSET's System View, via another connection, and then using IQSET's 'Configure vCNCs' option to configure the vCNCs. This will also enable a CNC User module to be added if there are none in the strategy.

See the IQSET Manual (TE200147) for details.

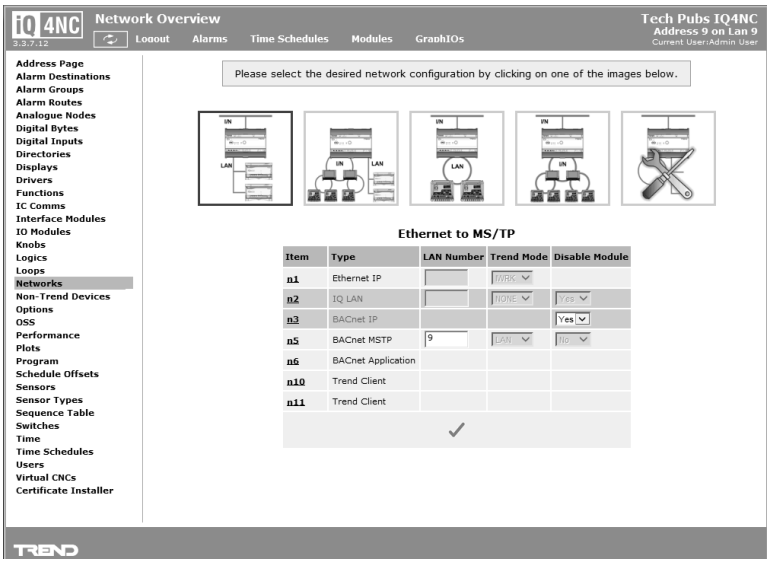
4.8.3 Using IP Tool to Enable vCNCs and Connecting to the vCNC Using IQSET

vCNC modules can be set up by using IP Tool to enable the required vCNCs and set the port number. For IQ4 v3.60 or greater, it is then necessary to connect to one of the controller's vCNCs using IQSET and follow the on-screen instructions. See **"Connecting to a vCNC operating in Secure Mode" on page 517**.

Note: For v3.60 or greater physical access to the controller is required.

4.9 Set up IQ4NC Communication Mode (IQ4NC only)

If configuring an IQ4NC its communication mode must be configured before the strategy download as it affects which Network modules are enabled and hence the ability of IQSET to download the strategy files. This is done by accessing the IQ4's Network modules summary web page and clicking the required icon.



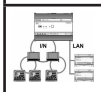
By default the IQ4NC is set to provide a link between an Ethernet Internetwork and MS/TP LAN.

Selecting any of the pre-defined configurations will set the Network modules appropriately as indicated on the web page. The 'Trend Mode' parameter indicates which networks are internetwork, LAN, and not used for Trend communications. Those cells not greyed out enable settings to be made from the summary page.



Ethernet Internetwork to MS/TP Lan (default)

Parameter	*Notes*
Lan number	1, 4 to 9, 11 to 119
BACnet IP	Can be disabled



Current Loop Internetwork to MS/TP Lan

Parameter	*Notes*
Lan number	1, 4 to 9, 11 to 119
BACnet IP	Can be disabled



Ethernet Internetwork to Current Loop Lan

Parameter	*Notes*
Lan number	1, 4 to 9, 11 to 119
BACnet IP	Can be disabled
BACnet MSTP	Can be enabled e.g. for connection of PEHA devices



Ethernet Internetwork to Current loop Internetwork *Note address no longer needs to be > 100.*

Parameter	*Notes*
Lan number	1, 4 to 9, 11 to 119
BACnet IP	Can be disabled
BACnet MSTP	Can be enabled e.g. for connection of PEHA devices.



Manual configuration

E.g. Ethernet Internetwork to Current loop Internetwork plus MS/TP Lan

If complete control is required the manual option should be selected. Care must be taken when using this. Certain configurations are invalid and either cannot be selected, or will be ignored when the changes are committed.

For more details of the IQ4NC communication modes see **"IQ4NC Communication Modes" on page 59**.

4.10 Check Trend Current Loop Network Indicators (/LAN or IQ4NC only)

If the current loop network is being used check that the current loop network TX and RX indicators are ON. If either is OFF check the next/previous devices on the current loop network, and the connection to them.

4.11 Check MS/TP Trunk Indicators (IQ4NC only)

If the MS/TP Trunk is being used check that the MS/TP TX and RX indicators are ON. If either is OFF check the MS/TP connection and baud rate.

4.12 Set up Ethernet Access Across Routers

If the internetwork is to be built across a router(s) details of devices (on each subnet) must be specified to enable cross-router communications using IP Tool to specify IP settings for up to 20 devices in the Remote Trend Devices list.

It is recommended that at least two devices from each subnet are specified. For increased reliability details of additional devices should be specified. The same list must be copied to other EINC type devices on the network.

4.13 Set up Trend Addressing

The IQ4s device address is set in the factory on a rolling basis in the range 11 to 119, and the LAN number is set to 20. This reduces the likelihood of an address clash but the addresses should be set up as part of the installation. This is done using either IP Tool or IQTool Addresser Applet depending on the system configuration.

Single IQ4 or LAN of IQ4s on Ethernet

Use IP Tool to set the device address (ensuring it is unique on the LAN) and LAN number.

LAN of IQ4s on current loop

Use the IQTool Addresser Applet to automatically address the IQ4s.

Note: The IQ4's LAN number is dictated by its INC type node. However, for its alarm communications to operate correctly it should have the same LAN number set in its Address module; this can be done from IQSET's Device View.

Mixture of IQ4 and pre IQ4 controllers on current loop

Use the IQTool Addresser Applet to add the IQ4 to the grid manually and specify the serial number, and device address and then use IQSET's System View to set the LAN number.

See the IP Tool Manual (TE200638) for details of using the IP Tool. See the IQTool Addresser Applet Manual (TE201299) for details of using the IQTool Addresser Applet.

4.14 Set up Current Loop Baud Rate (/LAN or IQ4NC only)

If the IQ4 is communicating on a current loop network it will automatically detect the baud rate being used on the network and set itself accordingly. On systems where there is only IQ4s on the current loop network, the IQ4s will decide amongst themselves what baud rate to use; this may not be the one that is required. In this situation it is necessary for one IQ4 to be set to a specific baud rate using IQSET's IQTool Monitor Applet to ensure the other IQ4s select the required baud rate.

See the IQTool Monitor Applet Manual (TE201298) for details of using the IQTool monitor Applet.

4.15 Set up RS232/USB Port Addressing

If the IQ4 is to provide connection to the network for supervisors and tools it is necessary to setup the Supervisor Port or USB Engineering Port as required using IQSET's IQTool Monitor Applet.

See the IQTool Monitor Applet Manual (TE201298) for details of using the IQTool Monitor Applet.

4.16 Check LAN OK Indicator (/LAN only)

Check the LAN OK indicator is ON.

Note: Indication that the IQ4 has built a LAN with other devices does not mean that the LAN has built as expected only that the IQ4 has built a LAN with another device.

4.17 Download Strategy

The strategy (IQ4 file) must be downloaded to the IQ4 using IQSET. This can be done over most connections to the IQ4, but Ethernet is recommended.

Note: The strategy file and other configuration files (backdrop, and XNC) can be downloaded using an Ethernet connection, but only the strategy file can be downloaded or uploaded through the current loop Lan.

Note: IQSET will set the IQ4's time to that of the PC running IQSET if required during the download process. If the IQ4 is to obtain its time from an NTP server ensure that this option is selected because if the IQ4's time is different from the NTP server's time by more than 49 days the NTP server will be unable to set the time.

Note: Before downloading a strategy to an IQ4 it is recommended that the strategy operation is checked using IQSET's simulation mode.

4.17.1 Parameter Download Exclusions

The following parameters make a strategy bespoke to a specific IQ4:

Module	***Group***	***Parameter(s)***
Address module	1	Identifier
	2	Local Lan and Local Node address
	3	RS232 Supervisor port address and USB Engineering port address
Type 4 (Ethernet IP) network module	1	IP Address
	2	Ethernet Addressing Mode, Subnet Mask, UDP port, Router 1
	3	Hostname, WINS servers, DNS servers
	4	Addresses and Subnet Masks of Remote Devices 1 to 20
Type 1 (IQ LAN) network module	1	Module Disable, Autobaud, and Baud Rate

If the same strategy is to be used in a number of IQ4s these parameters should configured prior to the download using IPTool and then excluded from the download. IQSET's default is to exclude these parameters. For each group of parameters there is a 'Not Sent' check box which can be unchecked if the parameters are to be downloaded.

Note: If the settings have been included in the download and are different to those previously made (e.g. using IPTool) then when the file is downloaded the address parameters will change to those included in the file.

4.18 Check Input & Output Operation

Once the strategy is downloaded and running it can be fault found/modified using the web pages (parameter changes only, and if Ethernet is connected), by IQSET Live Values, and IQSET Live Edit.

4.19 Connect Supervisors/Tools/Display Panels (if required)

Connect PCs (e.g. for supervisors/tools) and displays. Connection of supervisors, tools, display panels etc can be made directly to controller via the RS232 Local supervisor Port, USB Local Engineering Port, or Ethernet port. Connection can also be made over the Trend network over Ethernet or the current loop.

4.20 Configure SSL Certificates (if required)

If the IQ4's HTTPS web server is being used it is necessary to create an SSL certificate and upload it to the IQ4.

A separate certificate must be created and uploaded to each IQ4 whose web server is to be accessed and the appropriate certificate must be trusted on the client device.

The following procedure should be followed:

- Create the SSL Certificates
- Upload the Controller Certificate to the IQ4
- Test Access

Note: When HTTPS is first enabled the IQ4 will automatically generate a self signed certificate, however this is not suitable for long term use, and a full trusted certificate should be created as soon as possible.

4.20.1 Create the SSL Certificates

The SSL certificate should be created as specified by your companies IT policy. A self signed certificate or a certificate generated by a certificate issuing authority can be used. If in doubt about how to create a certificate you should consult an IT professional with appropriate skills.

Recommended settings for compatibility with controller

Certificate Requirements

- **Max key size:** Elliptic Curve (EC) Certificates recommended with a 256bit Key (eg. P-256) RSA keys are also supported, but are no longer recommended.

  Note: Maximum 2048 bit RSA as larger key sizes will slow down the IQ4 web serving significantly.

- **Signature Algorithm:** SHA256 (MD5, SHA1 are supported but not recommended) SHA512 is also supported.

IQ4 will only accept certificates in PKCS#12 format (.pfx .p12) For the IQ4 server certificate these should be saved with the full certificate chain and private key and a password to allow the IQ4 to open the file.

The IQ4 does not currently produce a Certificate Signing Request (CSR) therefore you will need to use a suitable certificate generation tool to create the key pair and CSR.

Once you have your certificate from your Certificate Authority (CA) you will need to package these into a PKCS #12 file along with the relevant root certificates supplied by them.

Note: Do not include other (unrelated) root certificates in the controller's PFX/P12 file, this is for the IQ4 server and chain certificates only.

The following sections describe the common scenarios using XCA which is free to download from '**http://sourceforge.net/projects/xca/**'.

Creating a Fully Trusted Signed Certificate
Create a Self Signed Controller Certificate

Note: The use of XCA Certificate and Key management software is described in this manual for illustrative proposes only. Trend does not endorse or recommend XCA Certificate and Key management software other products are available.

4.20.1.1 Creating a Fully Trusted Signed Certificate

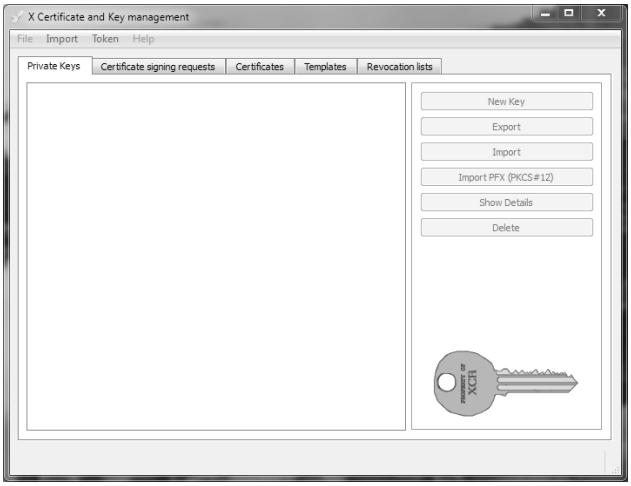
To create a fully trusted signed certificate:

1. Create a trusted signed certificate request - see **"Create a Trusted Signed Certificate Request" on page 138**.
2. Export the certificate signing request - see **"Export the Certificate Signing Request" on page 144**.
3. Get the certificate signed - see **"Get the Certificate Signed" on page 144**.
4. Upload the controller certificate to the required controller - **"Upload the Controller Certificate to the IQ4" on page 155**.

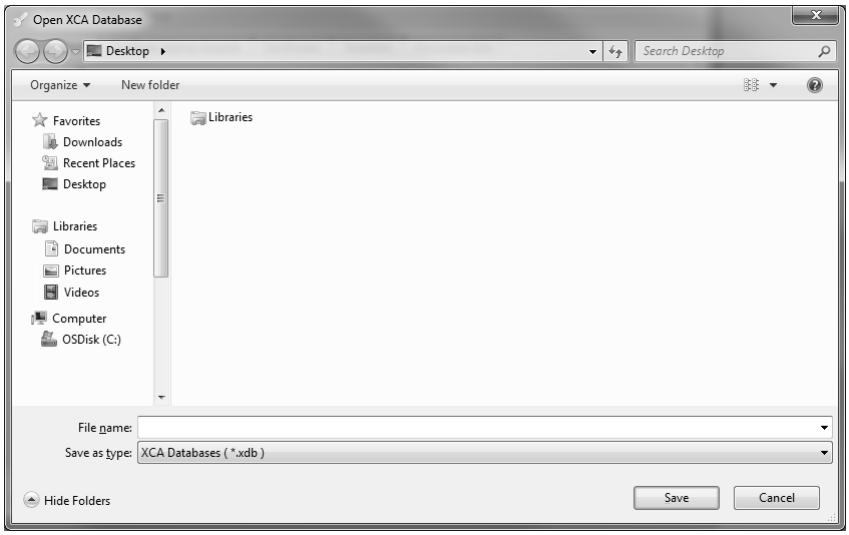
4.20.1.1.1 Create a Trusted Signed Certificate Request

Create a trusted signed certificate request:

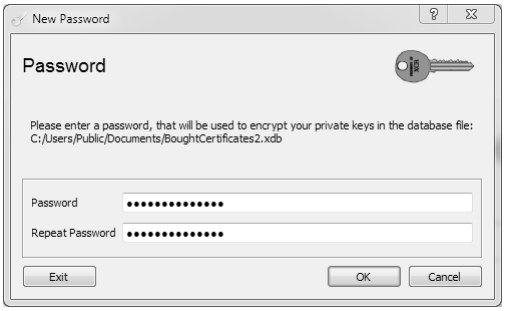
1. Run XCA - refer to the documentation suppled with the product.



2. Select **File**>**New Database**. The **Open XCA Database** dialogue box is displayed.

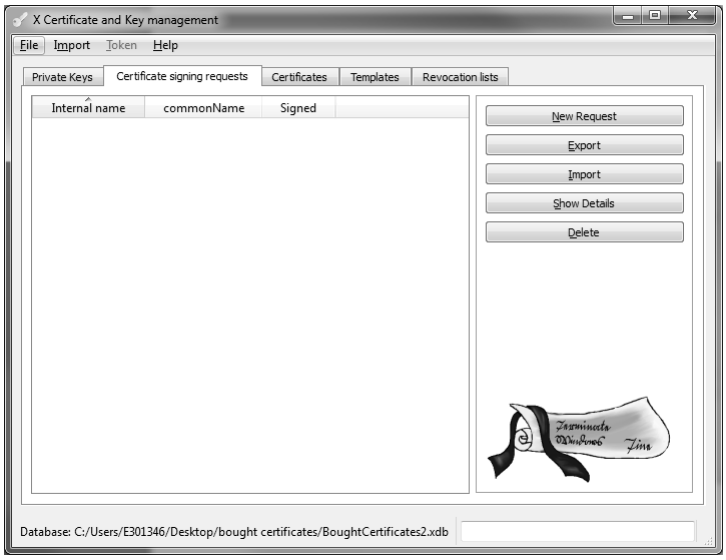


3. Navigate to the location where the database is to be created and in the **File name** box enter the name of the database.
4. Click **Save**. The **New Password** dialogue box is displayed.

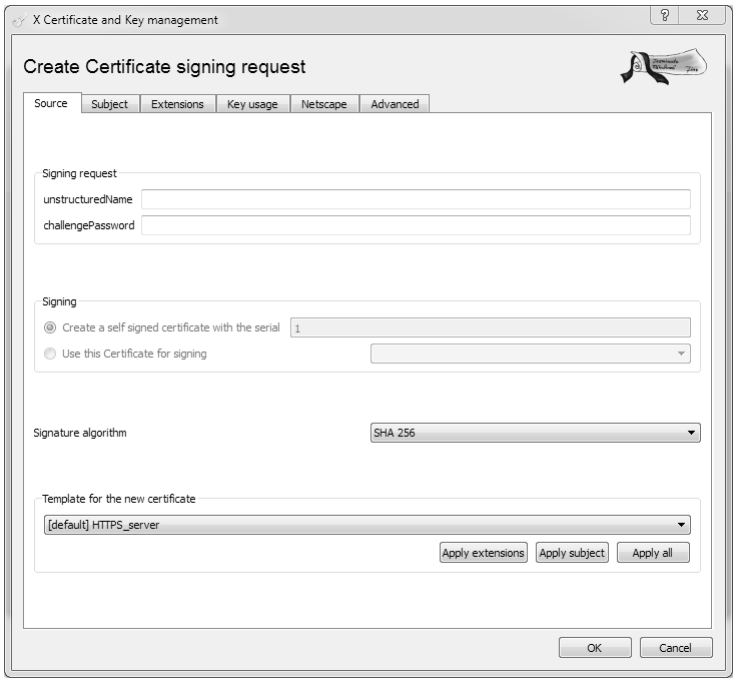


5. In the **Password** box enter the required password. It is important that the password is secure and that you remember it as it will be protecting the private key.
6. In the **Repeat Password** box re-enter the password.
7. Click **OK**.

8. Select the **Certificate Signing Request** Tab.

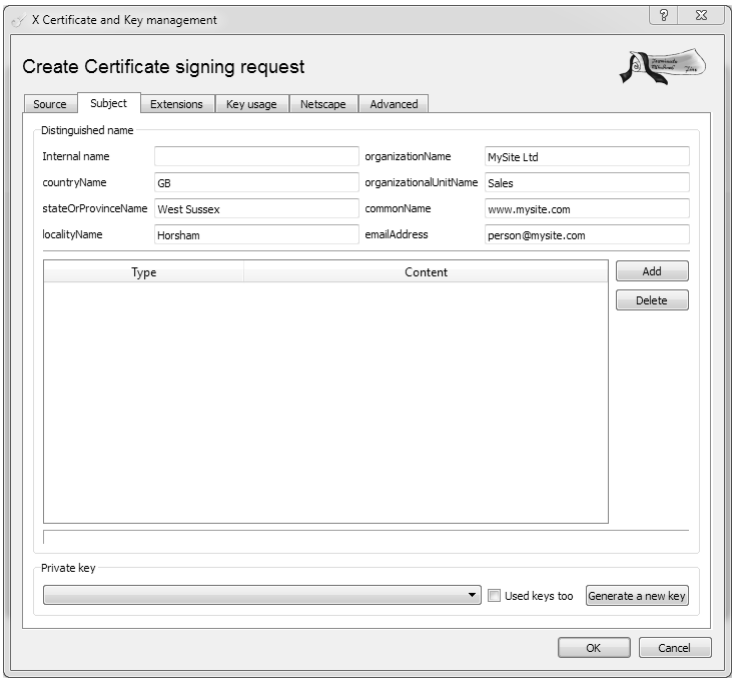


9. Click **New Request**.



10. Ensure that 'HTTPS_server' is selected in the **Template for new certificate** box.
11. In the **Signature Algorithm** box select 'SHA 256'.
12. Click **Apply all**.

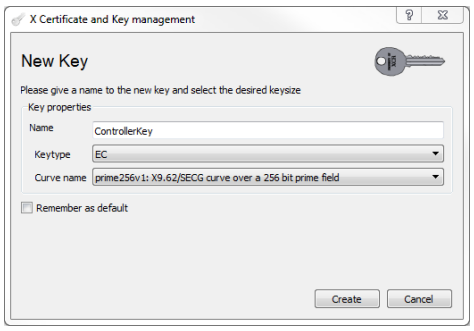
13. Select the **Subject** tab.



14. In the **commonName** box enter to the domain name of the IQ4 do not use spaces.

    - Remember that this MUST be the same name as you use to access the IQ4.

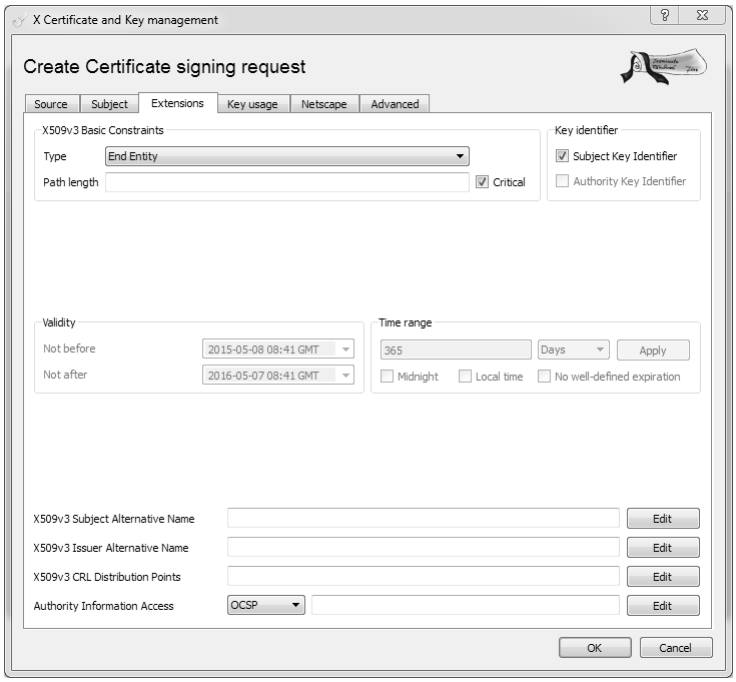
15. Fill in the other fields as appropriate, ‘commonName’ is a mandatory field, but the others my be required by your certificate authority, please check for their requirements.
16. Click **Generate a new key**.



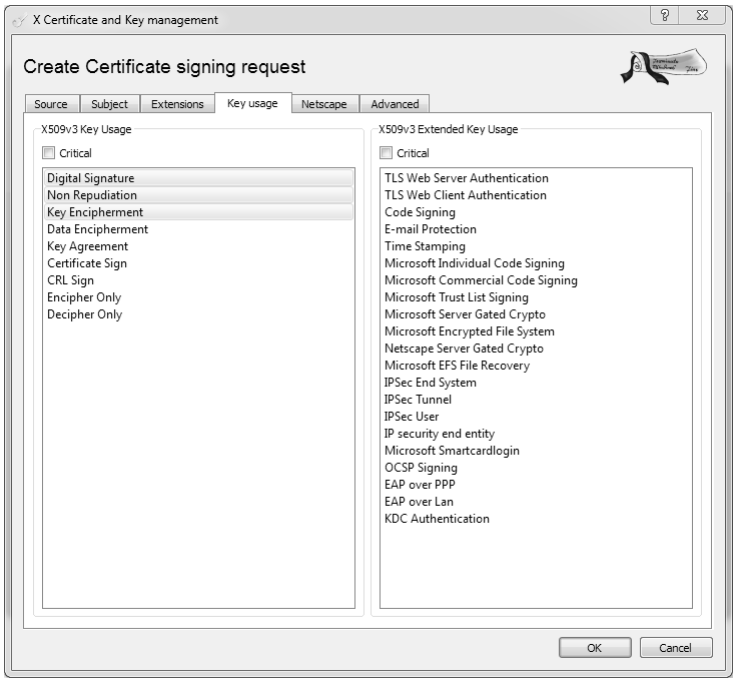
The name should ideally match the name of the certificate.

17. In the **Keytype** box select ‘EC’.
18. In the **Curve name** box select ‘prime256v1’.
19. Click **Create**. A confirmation message is displayed, click **OK**.

20. Select the **Extensions** tab.

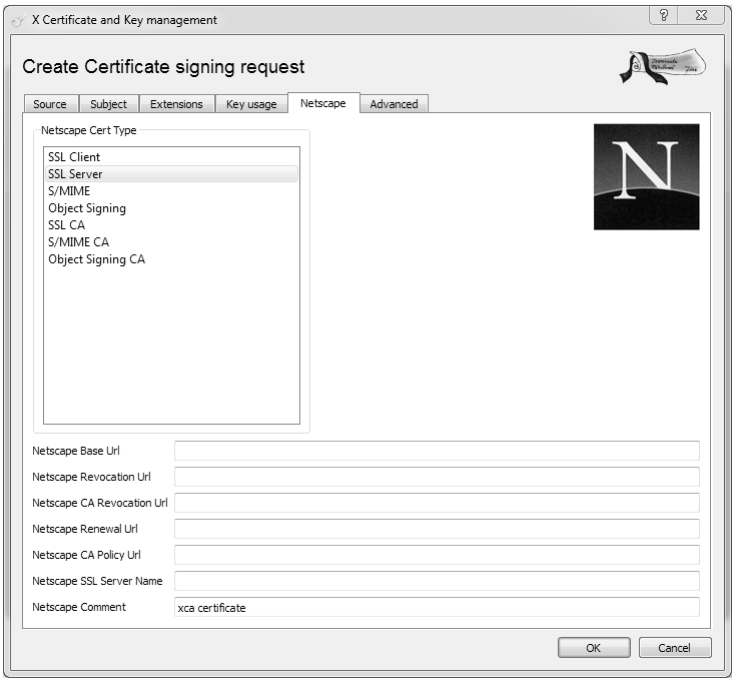


21. In the **Type** box select **End Entity**.
22. Select the **Key usage** tab.

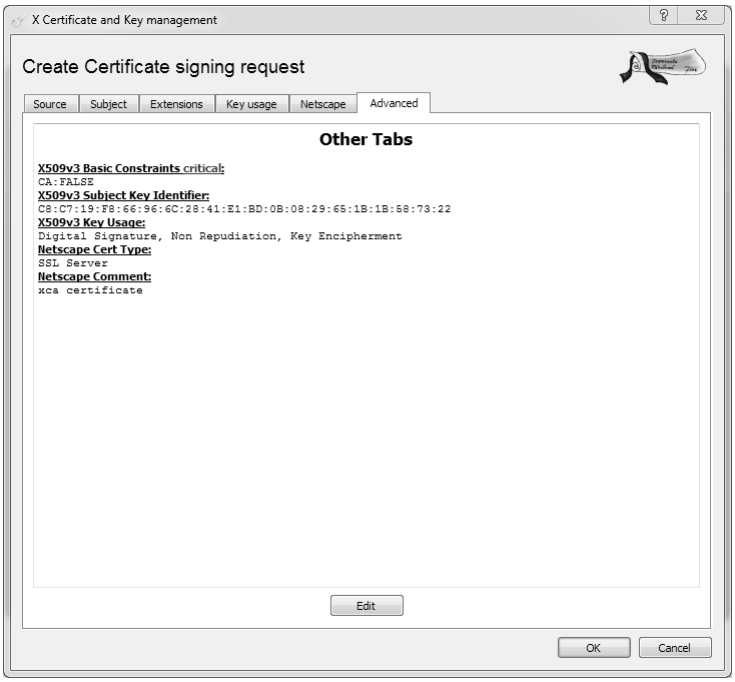


23. Ensure ‘Digital Signature’, ‘Non Repudiation’ and ‘Encipherment’ are selected in the **X509v3 Key Usage** box.

24. Select the **Netscape** tab.

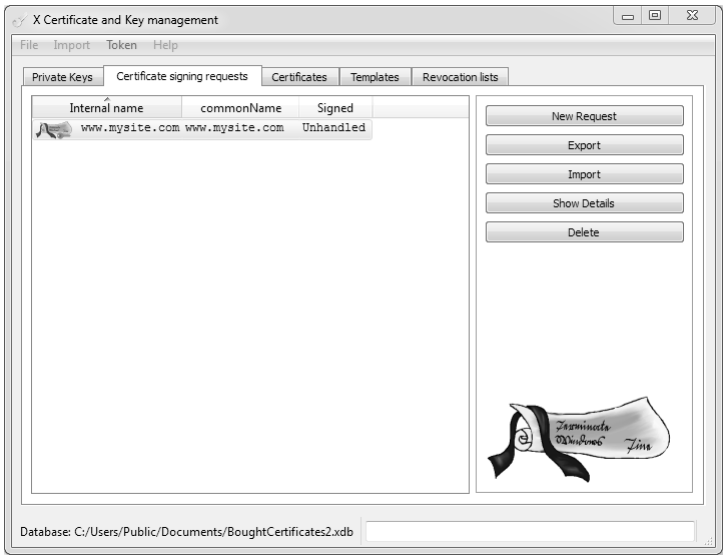


25. Ensure that 'SSL Server' is selected in the **Netscape Cert Type** box.
26. Select the **Advanced** tab.



27. Check the settings.

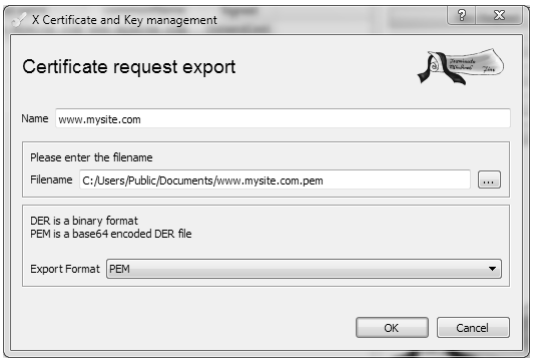
28. Click **OK**. A confirmation box is displayed, click **OK**. XCA will create the certificate.



4.20.1.1.2 Export the Certificate Signing Request

To export the certificate signing request:

1. Create a trusted signed certificate request - see **"Create a Trusted Signed Certificate Request" on page 138**.
2. Run XCA and load the required database - refer to the documentation suppled with the product.
3. Select the trusted signed certificate request and click **Export**.



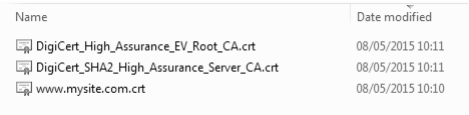
4. Select an appropriate name for the exported certificate (it will default to the name of the certificate).
5. In the **Export Format** box select 'PEM'.
6. In the **Please enter the filename for the certificate** box specify a filename for the certificate request and the location where it will be saved and click **Save**.
7. Click **OK**. XCA will now produce a PEM file in the location specified. Alternatively you can right click on the CSR in XCA and select **Export->Clipboard** (this will allow you to paste the data into an email to send to the Certificate Signing Authority (CA).

4.20.1.1.3 Get the Certificate Signed

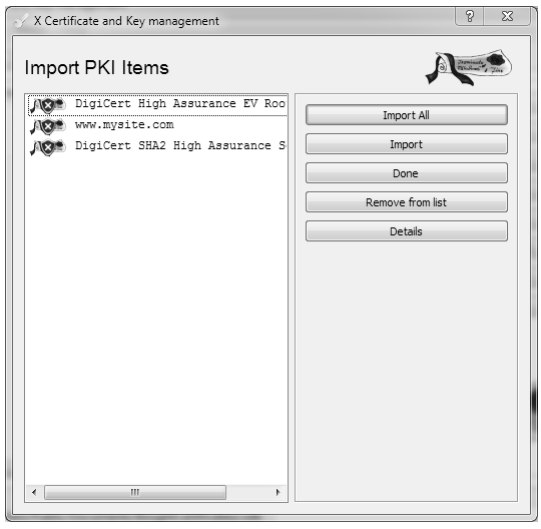
To get the certificate signed:

1. Email the PEM file created in **"Export the Certificate Signing Request" on page 144** to your certificate signing authority (CA) e.g. Verisign/Digicert etc.

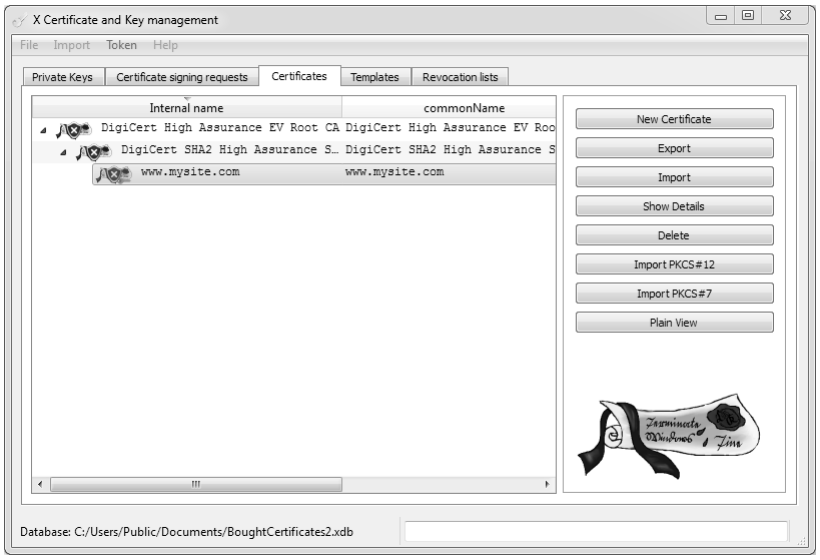
   Once the certificate is signed you should receive back a certificate and any root certificates required to complete the certificate chain.



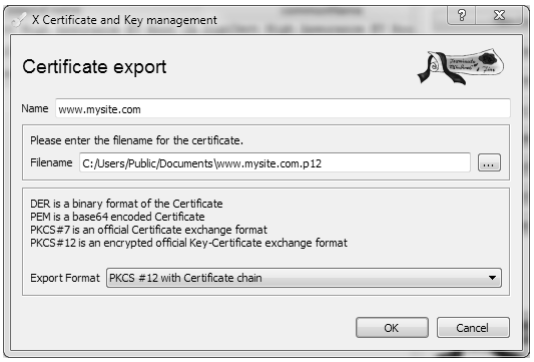
2. Run XCA and load the required database - refer to the documentation suppled with the product.
3. Select the **Certificates** tab.
4. Drag the received files into the **Certificates** tab.



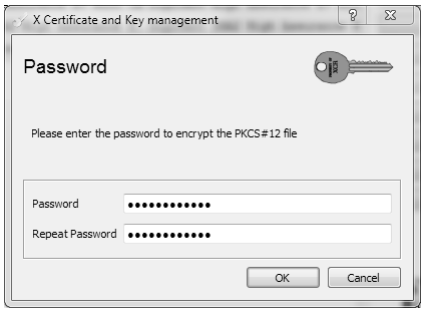
5. Click **Import All**. You should now see a chain of certificates from the CA.



6. Select the IQ4's certificate (here called www.mysite.com).
7. Click **Export**.



8. Specify the location where the certificate will be saved.
9. Ensure that the export format is specified as 'PKCS #12 with Certificate chain'.
10. Click **OK**.



11. In the **Password** box enter the required password. It is important that the password is secure and that you remember it.
12. In the **Repeat Password** box re-enter the password.
13. Click **OK**. This will now create a p12 certificate file, that contains the new certificate, the private key for the IQ4 and the root certificates to complete the chain. This file can now be loaded into the IQ4 - see **"Upload the Controller Certificate to the IQ4" on page 155**.

4.20.1.2 Create a Self Signed Controller Certificate

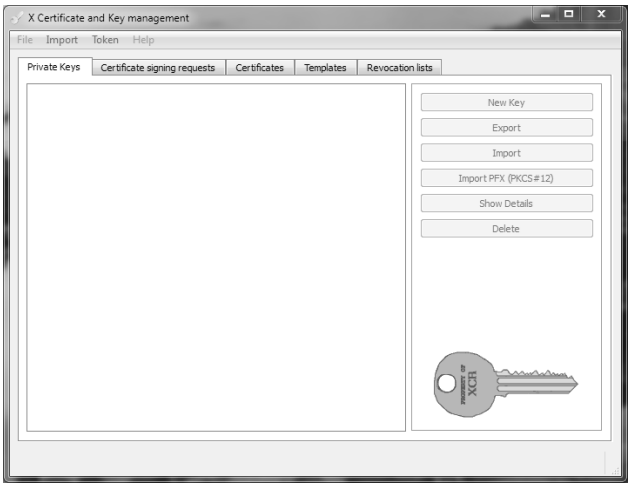
To create a self signed controller certificate:

1. Create a controller certificate for the IQ4 - see **"Create the Controller Certificate" on page 146**.
2. Export the controller certificate - see **"Export the Controller Certificate" on page 151**.
3. Upload the controller certificate to the required IQ4 - **"Upload the Controller Certificate to the IQ4" on page 155**.
4. Trust the IQ4 - see **"Import the Controller Certificate to the Client Device" on page 152**.

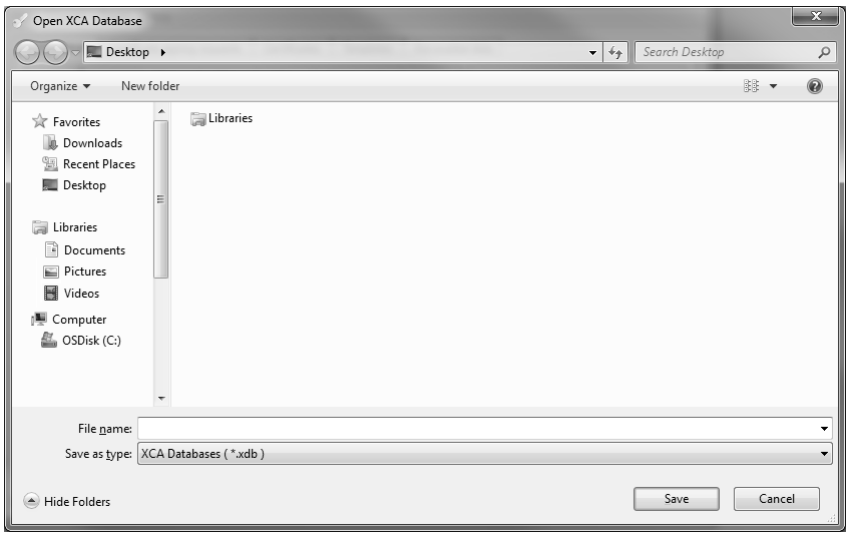
4.20.1.2.1 Create the Controller Certificate

To create the controller certificate:

1. Run XCA - refer to the documentation supplied with the product.



2. Select **File**>**New Database**. The **Open XCA Database** dialogue box is displayed.

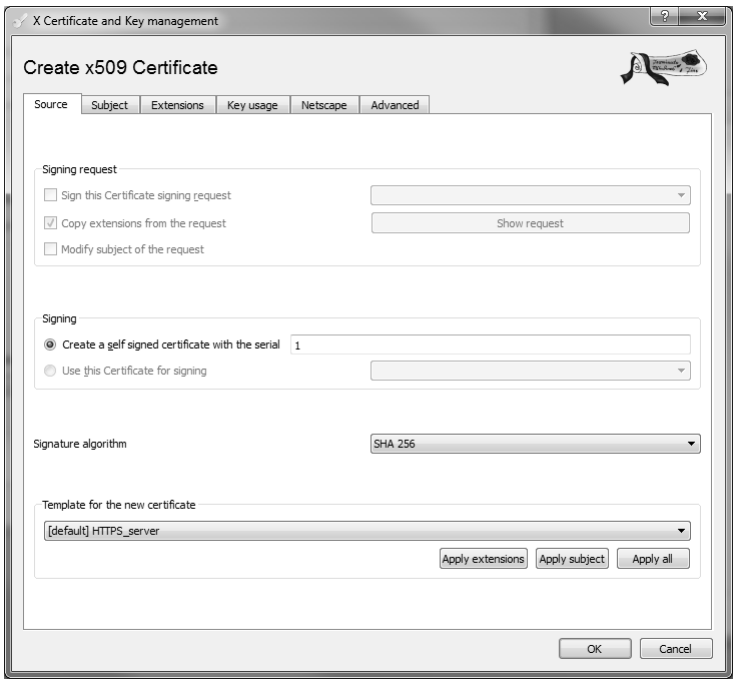


3. Navigate to the location where the database is to be created and in the **File name** box enter the name of the database.

4. Click **Save**. The **New Password** dialogue box is displayed.

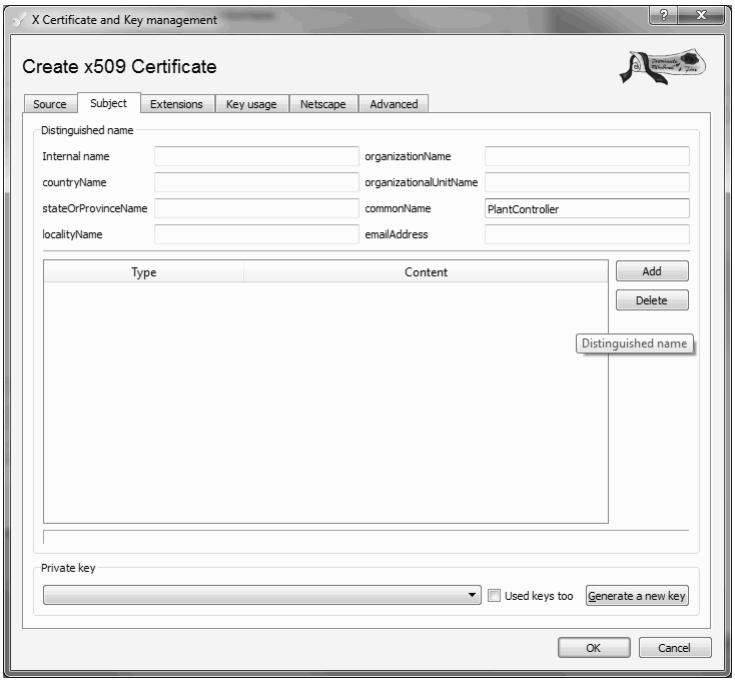


5. In the **Password** box enter the required password. It is important that the password is secure and that you remember it as it will be protecting the private key.
6. In the **Repeat Password** box re-enter the password.
7. Click **OK**.
8. Click **New Certificate**.
9. Select the **Source** tab.



10. Ensure that the signature algorithm is set to ‘SHA256’ and that the template is set to ‘HTTPS_server’.
11. Click **Apply all**.

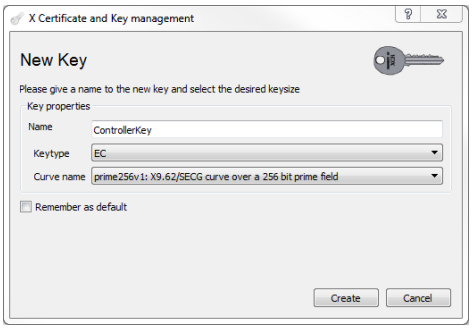
12. Select the **Subject** tab.



13. In the **commonName** box enter to the domain name or WINS name (hostname) of the IQ4 do not use spaces.

- Remember that this MUST be the same name as you use to access the IQ4 i.e. the IQ4's hostname
- If you use Static IP addresses then alternatively you can put in the IP address here.

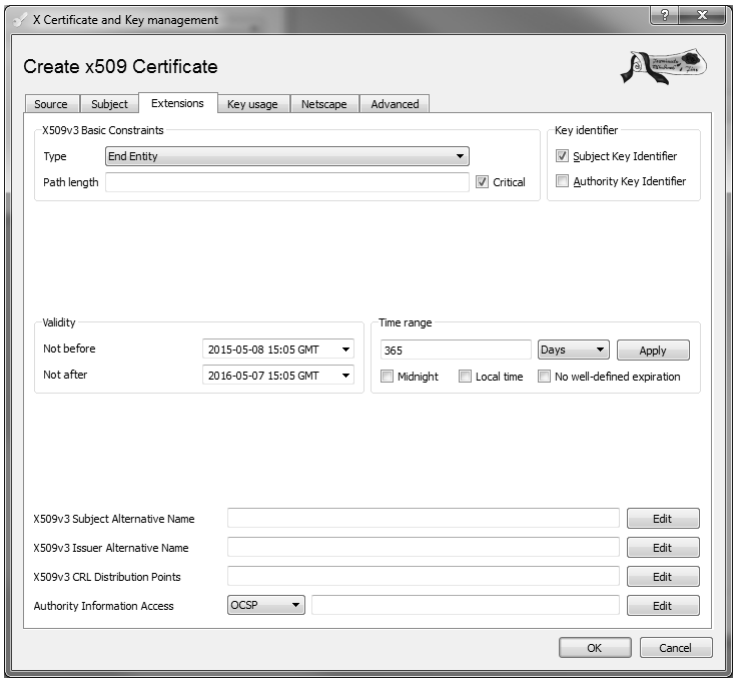
14. Click **Generate a new key**.



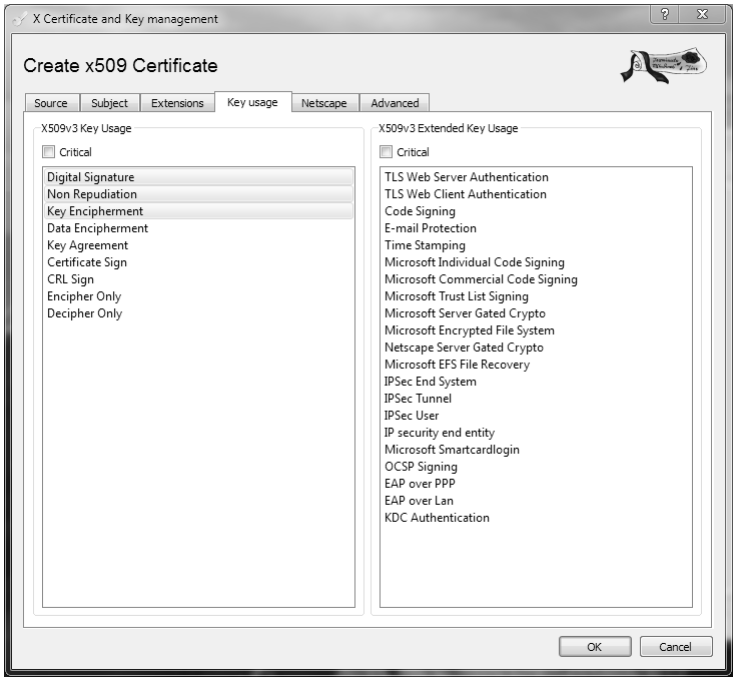
The name should ideally match the name of the certificate.

15. In the **Keytype** box select 'EC'.
16. In the **Curve name** box select 'prime256v1'.
17. Click **Create**. A confirmation message is displayed, click **OK**.

18. Select the **Extensions** tab.



19. In the **Type** box select **End Entity**.
20. Set the Time range for as long as you want the controller certificate to last (do not make this too long, 1 or 2 years is normal).
21. Click **Apply**.
22. Select the **Key usage** tab.

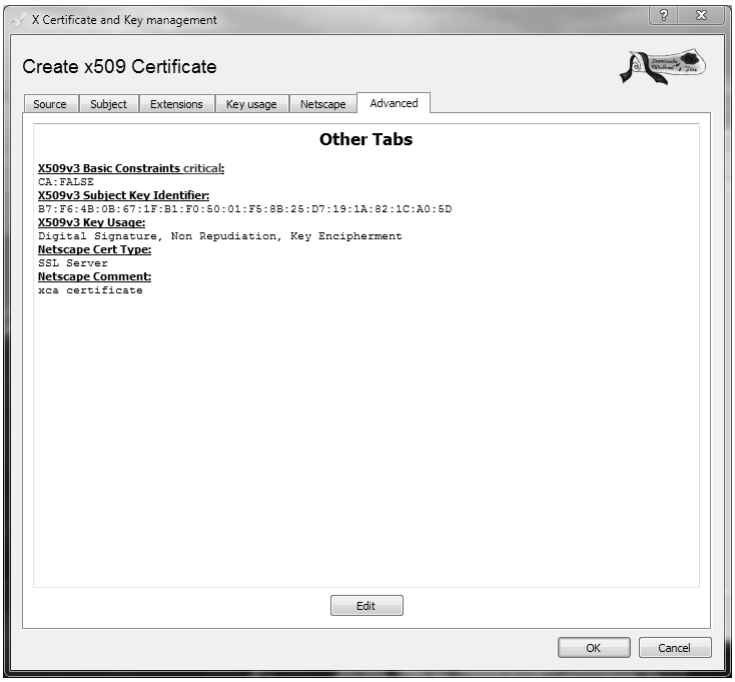


23. Ensure ‘Digital Signature’, ‘Non Repudiation’ and ‘Encipherment’ are selected in the **X509v3 Key Usage** box.

24. Select the **Netscape** tab.

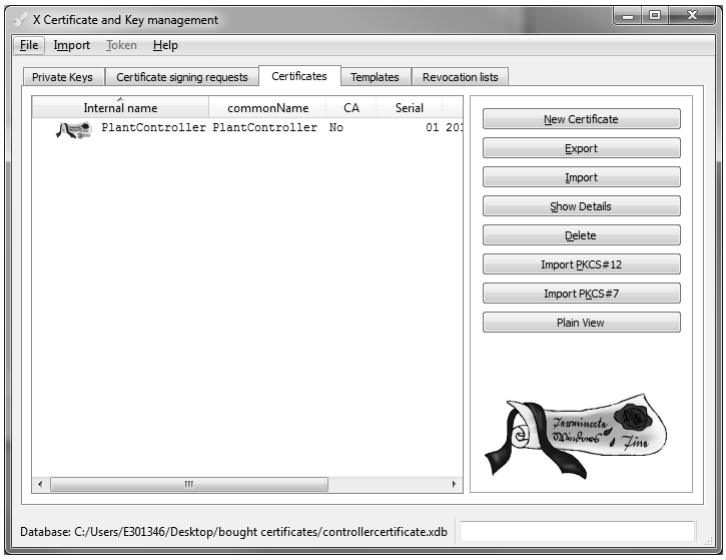


25. Ensure that 'SSL Server' is selected in the **Netscape Cert Type** box.
26. Select the **Advanced** tab.



27. Check the settings.

28. Click **OK**. A confirmation box is displayed, click **OK**. XCA will create your certificate.



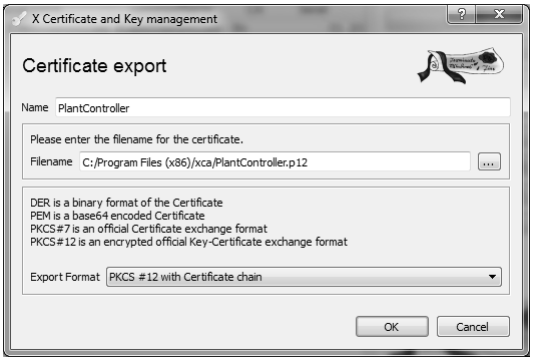
4.20.1.2.2 Export the Controller Certificate

It is necessary to export the controller certificate in two formats, 'PKCS #12 with Certificate Chain' (p12 file) which needs to be imported into the IQ4 and 'PEM' (crt file) which needs to be imported onto any client device that is to access the IQ4.

4.20.1.2.2.1 Export the Controller Certificate as a p12 File

To export the controller certificate as a p12 file:

1. Create a controller certificate - see **"Create the Controller Certificate" on page 146**.
2. Run XCA and load the required database - refer to the documentation suppled with the product.
3. Select the controller certificate and click **Export**.



4. Select an appropriate name for the exported certificate (it will default to the name of the certificate).
5. In the **Export Format** box select 'PKCS #12 with Certificate chain'.
6. In the **Please enter the filename for the certificate** box specify a filename for the certificate and the location where it will be saved and click **Save**.
7. Click **OK**.

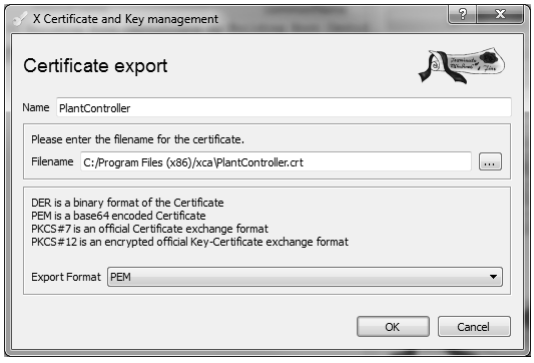


8. In the **Password** box enter a strong password for the certificate. This password protects the IQ4's private key, and MUST be kept secret.
9. In the **Repeat Password** box re-enter the password.
10. Click **OK**. XCA will now produce a certificate file (p12 file) in the location specified.

4.20.1.2.2.2 Export the Controller Certificate as a crt File

To export the controller certificate as a crt file:

1. Create a controller certificate - see **"Create the Controller Certificate" on page 146**.
2. Run XCA and load the required database - refer to the documentation suppled with the product.
3. Select the controller certificate and click **Export**.



4. Select an appropriate name for the exported certificate (it will default to the name of the certificate).
5. In the **Export Format** box select 'PEM'.
6. In the **Please enter the filename for the certificate** box specify a filename for the certificate and the location where it will be saved and click **Save**.
7. Click **OK**. XCA will now produce a certificate file (crt file) in the location specified.

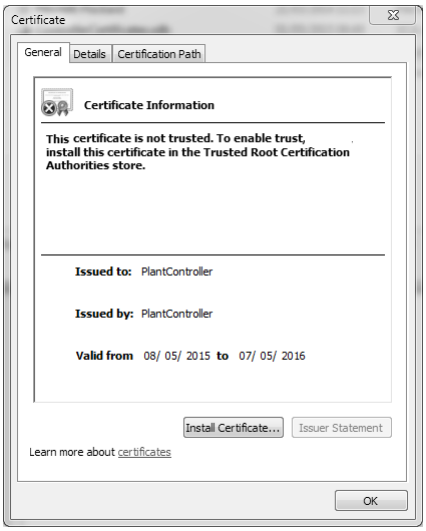
4.20.1.2.3 Import the Controller Certificate to the Client Device

To allow a client device to trust the IQ4's certificate it is necessary to import the controller certificate (crt file) to the device that is to access the IQ4.

The instructions below apply to Windows 7 and Internet Explorer. The process depends on the operating system and/or the browser being used. Refer to the appropriate 3rd party documentation.

To import the controller certificate:

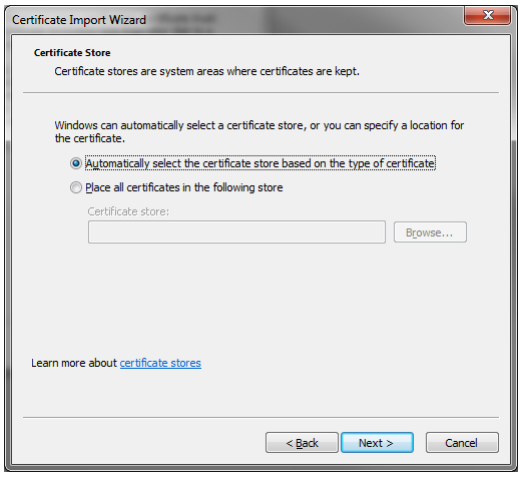
1. On the computer to which the certificate is to be imported (i.e. on that is required to access the IQ4's web pages) navigate to where you saved the certificate as a crt file using Windows Explorer and double click on the (.crt file) certificate. The certificate details are be displayed.



2. Click **Install Certificate...**. The **Certificate Import Wizard** is displayed.



3. Click **Next**.

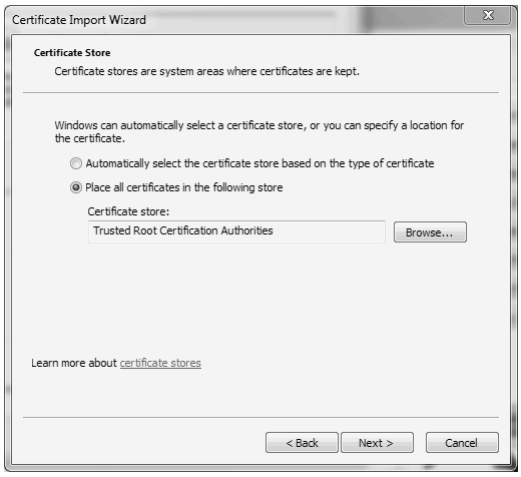


4. Select **Place all certificates in the following store**.
5. Click **Browse...** The **Select Certificate Store** dialogue box is displayed

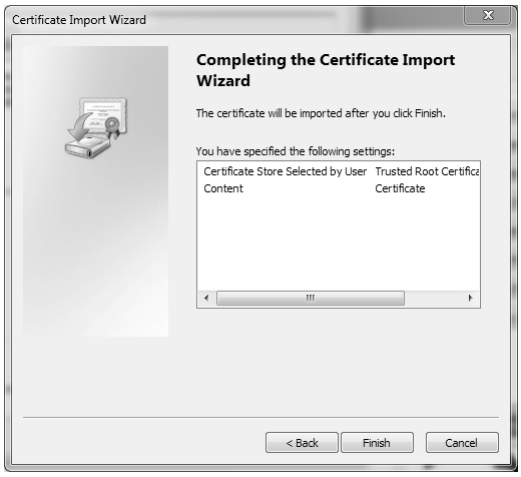


6. Select 'Trusted Root Certification Authorities'.

7. Click **OK**.



8. Click **Next**.



9. Click **Finish** then wait for a few seconds until the following dialogue box is displayed.



10. Click **Yes** to accept the installation of the certificate. A notification message is displayed, click **OK**.
11. Click **OK** on the **Certificate** dialogue box.

Note: Before to accessing an IQ4 close all browser windows.

4.20.2 Upload the Controller Certificate to the IQ4

The controller certificate (p12 file) must be uploaded to the IQ4 in order for it to protect the controller.

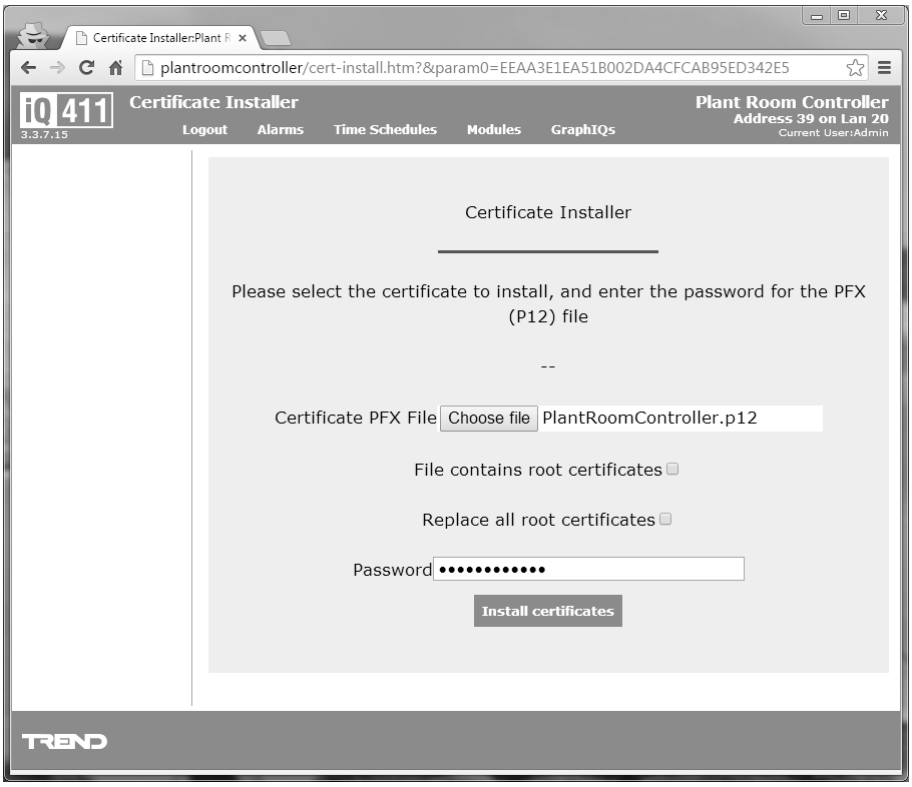
To upload the controller certificate to the IQ4:

1. Before to accessing the IQ4 close all browser windows.
2. Login to the IQ4 using a web client to which the certificate is to be uploaded as a user with a PIN Level of '100'.

   Note: Ignore any security warnings given by the web client in order to access the IQ4.

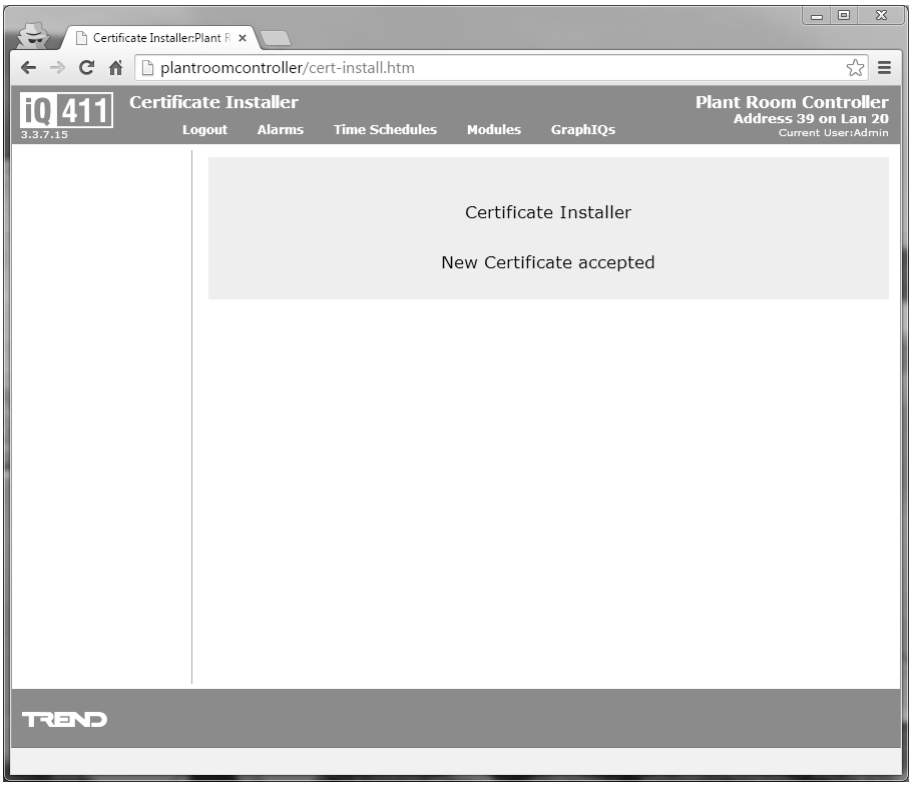
   Note: If the IQ4 has no users set up a user must be configured before certificates an be set up, and HTTPS must be enabled so that the transmission of the certificate is protected.

3. Navigate to the Networks page and select n1 (Ethernet IP) and ensure that the Secure Web server port number is set to '443' and click tick at the bottom of the page.
4. Select **Modules**>**Certificate Installer**.



5. Click **Browse**. The **Choose File to Upload** dialogue box is displayed.
6. Navigate to the folder the controller certificate file (p12) is stored in and select it.
7. Click **Open**.
8. Ensure that the **File contains root certificates** and the **Replace all root certificates** check boxes are NOT selected.
9. Enter the certificate's password in the **Password** box.

10. Click **Install certificates**. After a few seconds the IQ4 will display a page indicating that the new certificate was accepted.



If the IQ4 displays with a page with 'Unable to read certificate file' this could be because you entered the password incorrectly, or the certificate was in a format that the IQ4 did not understand

11. Upon successful completion of loading the certificate, navigate to the Networks page and select n1 (Ethernet IP). Scroll down to the HTTPS section, the name of the certificate, and its expiry time are displayed.

HTTPS Certificate Name	Tech PubsIQ41x
HTTPS Certificate Expiry	05/05/2016 10:11:00

At this stage DO NOT specify 'Redirect to HTTPS', as you may lock yourself out of the web access to the IQ4.

4.20.3 Test Access

Once the SSL certificate have been configured you should check that you are able to access the IQ4 and the certificate is valid.

To test access:

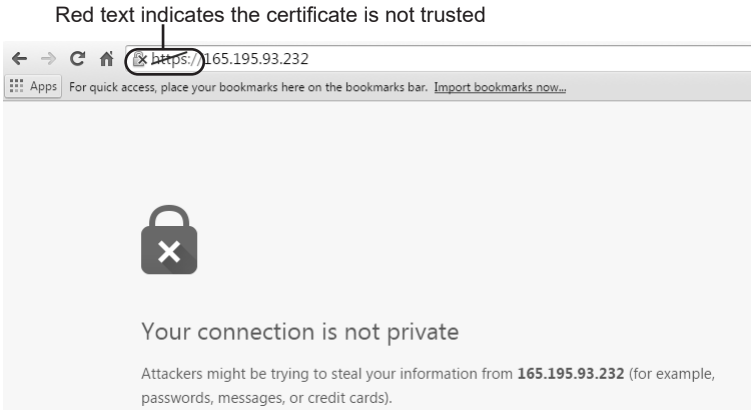
1. Ensure that controller certificate's 'commonName' matches IQ4's hostname or IP address.
2. Close all web browser windows.
3. Run the web browser and access the controller using the name specified by the controller certificate's 'commonName' using HTTPS e.g.

   https://techpubsnc

   If the controller trusts the certificate on the PC it will be indicated by the browser. The indication is dependant on the browser being used. An example is shown below:



   If the controller does not trust the certificate on the PC it will be indicated by the browser. The indication is dependant on the browser being used. An example is shown below:



4.21 Backup the Strategy

Once the strategy is operating as expected use IQSET to upload the strategy and compare it with the strategy held by IQSET. This ensures that the strategy running is the IQ4 is the same as that held by IQSET.

5 CONFIGURE I/O MODULES

The IQ4E, IQ4NC/16/XNC or IQ4NC/32/XNC can connect to IQ4/IO modules and XCITE I/O modules. The configuring of IO modules is described in the following documents:

- IQ4 I/O Modules Installation Instructions - Configuring (TG201343)
- XCITE/IO Standard I/O Modules Installation Instructions - Configuration (TG201161)

This section provides a summary of this information.

Select Universal Input Channel Type (XCITE/UI type modules only)
Cut Screen Earth (Ground) Link(s) (XCITE/UI type modules only)
Set I/O Module Address (XCITE I/O modules only)
Check the Power and Watchdog Indicators
Check Supply Voltage (IQ4/IO modules only)
Check the I/O Bus Indicator
Set Physical I/O Module Addresses (if required)
Resolve Physical I/O Module Duplicate Addresses (if required)
Check Physical I/O Module Addressing
Check Strategy I/O Modules are Set Correctly

5.1 Select Universal Input Channel Type (XCITE/UI type modules only)

The input channel type of the XCITE I/O module input channels must be specified. This is done by positioning the input channel links.

5.2 Cut Screen Earth (Ground) Link(s) (XCITE/UI type modules only)

If required for universal inputs the earth screen should be segregated from I/O module's earth by cutting the wire link.

5.3 Set I/O Module Address (XCITE I/O modules only)

Set the I/O module's address (range 1 to 15 labelled 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F) on the I/O bus by adjusting the address switch to the required position.

5.4 Check the Power and Watchdog Indicators

Check the Power indicator is ON, and the Watchdog indicator is OFF.

5.5 Check Supply Voltage (IQ4/IO modules only)

For IQ4/IO modules located a long way from the IQ4 and that do not use a local power supply, it is necessary to check that the module supply voltage is still within the specified limits as a voltage drop may occur due to the cable length.

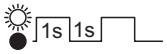
5.6 Check the I/O Bus Indicator

5.6.1 IQ4/IO Modules

Check the I/O Bus indicator on each IQ4/IO module is green. If I/O Bus indicator is green it is matched with a strategy I/O module. Where there is more than one physical I/O module it is recommended that you check for the correct address assigned to each module - see **"Check Physical I/O Module Addressing" on page 161**.

If the I/O Bus indicator indicates a fault (not green) the fault must be corrected:

Fault indications (red)	
☼	No comms with IQ4. Check I/O bus wiring - see IQ4E/.., IQ4NC/16/.., IQ4NC/32/.. Installation Instructions - Mounting (TG201338). If wiring correct the cable adapter fuse may have blown; replace the cable adapter after locating and correcting the reason for failure. *Note: The cable adapter is a sacrificial part and will must be replaced if the fuse blows.*

Fault indications (red)	
	'SOS' flash - module address not set or out of range. Set module address as required - see **"Set Physical I/O Module Addresses (if required)" on page 160**.
	Duplicate address (all affected modules will flash). Resolve duplicate addresses as required - see **"Resolve Physical I/O Module Duplicate Addresses (if required)" on page 161**.
Green  Red	Firmware upgrade is in progress. Please wait and then check this indicator again.

5.6.2 XCITE/IO Modules

Check the I/O Bus indicator on each XCITE/IO module is OFF. If I/O Bus indicators are OFF the physical I/O modules are matched with a strategy I/O module. Where there is more than one physical I/O module it is recommended that you check that the correct address is assigned to each module - see **"Check Physical I/O Module Addressing" on page 161**.

If the I/O Bus indicator indicates a fault (ON) the fault must be corrected:

Fault indications	
	Bus wiring fault. Check for short circuit or data hi/lo connected to 24 V or 0V.
	No comms with controller.
 (Fast flash)	Module address not set (i.e. set to 0) Set module address as required- see **"Set Physical I/O Module Addresses (if required)" on page 160** or Duplicate address with another module (indicator will flash on both modules). Set module address as required - see **"Resolve Physical I/O Module Duplicate Addresses (if required)" on page 161**.

5.7 Set Physical I/O Module Addresses (if required)

5.7.1 IQ4/IO Modules

The IQ4 will automatically attempt to match each physical IQ4/IO module with its corresponding strategy I/O module and set its address accordingly.

The IQ4 will be unable to set the address of a module if:

- *there is more than one unallocated strategy module of the same type, or
- *there is more than on unallocated physical I/O module of the same type, or
- there is no corresponding I/O module in the strategy.

**Note: This is because there must be a unique match between physical and strategy I/O modules.*

In either case, the I/O Bus indicator on the affected module(s) will be flashing red ("SOS") and it will be necessary to set the module address manually as follows:

Either:

- Press the module's service button for 3s to assign it to the next available strategy I/O module of its type or the next unused I/O bus address if there is no equivalent strategy module. If successful, the I/O bus change to solid green.

Or

- Use the IQ Tool Soft Addresser Applet in IQSET to assign the physical I/O module to the appropriate strategy I/O module - see the IQ Tool Soft Addresser Applet Manual (TE201356).

Alternatively, you can view and change module addresses using the I/O Module pages in the IQ4's web pages.

5.7.2 XCITE/IO Modules

The address of XCITE/IO modules is set with a physical switch. If the address is '0' or a duplicate the address switch should be set to the required address - see the XCITE/IO/.. Installation Instructions - Configuration (TG201161).

5.8 Resolve Physical I/O Module Duplicate Addresses (if required)

5.8.1 IQ4/IO Modules

Duplicate addresses will normally only occur if:

- two or more modules have been accidentally given the same address in the web interface
- an XCITE/IO module has its address switch set to the same address
- a module has been installed that already has an address.

In each case, the I/O Bus indicator on the affected modules will be flashing red and it will be necessary to change the module address as follows:

Either:

- Remove power to the module by disconnecting the I/O bus interconnector(s) and/or cable adapter(s).
- Press and hold the service button while reapplying power. After a few seconds the module will reboot with an address of 63.
- Cycle the module power again. After a few seconds the module will reboot into a default state with no address. The IQ4 will then attempt to assign an address automatically.

Or

- Use the IQ Tool Soft Addresser Applet in IQSET to assign the physical I/O module to the appropriate strategy I/O module - see the IQ Tool Soft Addresser Applet Manual (TE201356).

Alternatively, you can view and change module addresses using the I/O Module pages in the IQ4's web pages.

5.8.2 XCITE/IO Modules

The address of XCITE/IO modules is set with a physical switch. If the address is a duplicate, the address switch should be set to the required address - see XCITE/IO/.. Installation Instructions - Configuration (TG201161).

5.9 Check Physical I/O Module Addressing

5.9.1 IQ4/IO Modules

Once all IQ4/IO modules have been addressed, i.e. all I/O bus indicators are showing solid green, it is recommended that you check that each module has been addressed as expected.

Either:

- Press each module's service button in turn. The I/O bus indicator will flash to show the module's address, e.g. an address of 13 will comprise one long flash, followed by three short flashes:

  Note: The IQ4E's I/O bus indicator will also flash the I/O module's address.

Or

- Use the IQ Tool Soft Addresser Applet in IQSET to check the address - see the IQ Tool Soft Addresser Applet Manual (TE201356).

Alternatively, you can view and change module addresses using the I/O Module pages in the IQ4's web pages.

5.9.2 XCITE/IO Modules

Once all XCITE/IO modules have been addressed, i.e. all I/O bus indicators are OFF, it is recommended that you check that each module has been addressed as expected.

Either:

- Check each module's address switch is set as expected.

Or

- Use the IQ Tool Soft Addresser Applet in IQSET to check the address - see the IQ Tool Soft Addresser Applet Manual (TE201356).

Alternatively, you can view module addresses using the I/O Module pages in the IQ4's web pages.

5.10 Check Strategy I/O Modules are Set Correctly

- Check that the strategy I/O modules have been set up appropriately.
- Check each physical I/O module is linked to its own strategy I/O module. If extra strategy I/O modules are required they must be added to the strategy using IQSET and downloaded to the IQ4

PART 3

This part of the manual describes the functionality of each IQ4 module and their parameters. It is divided into the following sections:

IQ4 Modules
Address Module
Alarm Destination Modules
Alarm Group Modules
Alarm Log Modules
Alarm Route Modules
Analogue Node Modules
Calendar Modules
Digital Byte Modules
Digital Input Modules
Directory Modules
Display Modules
Driver Modules
Function Modules
IC Comms Modules
Interface Modules
I/O Modules
Knob Modules
Logic Modules
Loop Modules
Network Modules
Non-Trend Device Modules
Optimum Start Stop Modules
Options Module
Page Modules
Performance Module
Plot Modules
Program Module
Schedule Offset Modules (Schedule Modules)
Sensor Modules
Sensor Type Modules
Sequence Table Module
States Category Modules
Switch Modules
Time Module
Time Schedule Modules (Time Zone Modules)
User Modules
Virtual CNC Modules

1 IQ4 MODULES

The following sections describe the functionality of each IQ4 strategy module and the associated parameters. At the beginning of each section the module's overview parameters are defined; these list the parameters shown when the module is used as a dynamic item on a GraphIQ page. The functionality of the module is then defined, and a table of parameters which provides a description of each parameter, its range, default value, type and text communications identifier (which should be used when accessing the module with text communications) is provided.

A parameter's type can be one of the following - see **"Module Parameters" on page 23**:

Parameter Type	*Abbreviation*
Input	I
Output	O
Internal	N

1.1 Common IQ4 Module Parameters

This section provides a description of parameters common to most IQ4 modules.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Disable Module	Enables/disables the module. When disabled the module is not serviced. Its internal states and output will be maintained at their values prior to disabling - see **"Disabling Modules" on page 29**.	0 or 1 0 = Disabled 1 = Enabled	0	I	~
Label	User-friendly label for the module.	0 to 30 character string except \ /	-	N	$
Page	Page number containing the module.	0 to 255	0	N	p
Sequence Step	The sequence step at which the module is serviced - see **"Sequenced Servicing" on page 27**. It may be entered into the sequence table more than once. If set to '0' the module will be event driven.	IQ41x: 0 to 300 IQ422: 0 to 600 IQ4NC/00/..: 0 to 600 IQ4NC/12/24VAC: 0 to 600 IQ4NC/16/XNC/...: 0 to 600 IQ4NC/32/XNC/...: 0 to 640 IQ4E: 0 to 600 IQ4E/32: 0 to 600 IQ4E/64: 0 to 640 IQ4E/96: 0 to 960 IQ4E/96/XNC: 1280 IQ4E/128: 0 to 1280 IQ4E/160: 0 to 1600 IQ4E/192: 0 to 1920 IQ422/00/XNC: 0 to 1280 IQ422/12/XNC: 0 to 600	0	N	@
Sequence Table	Specifies whether the module is sequenced (in sequence table) or event driven.	0 to 1 0 = Event driven 1 = Sequenced	0	N	#
Xco-ordinate	The horizontal position of the module on a strategy page.	0 to 255	0	N	x
Yco-ordinate	The vertical position of the module on a strategy page.	0 to 255	0	N	y

Note: Although these parameters are common they may not all exist in all modules.

2 ADDRESS MODULE

Module Identifier	R1
Default Overview Parameters	None
Size in brIQ	24

2.1 Functionality

The Address module is used to store the addressing data used for Trend communications. It also holds the general alarm status, the IQ4 type details, and details of memory available and memory used. It does not appear on the strategy page and does not need to be sequenced.

2.2 Address Module Parameters

<table>
<tr><th>Parameter</th><th>Description</th><th>Range</th><th>Default</th><th>Type</th><th>Text Comms</th></tr>
<tr><td>Archive Failure</td><td>FARC alarm - Indicates if the archive of the running strategy file to flash memory has failed - see “Archive Failure (FARC)” on page 76.</td><td>0 or 1
0 = OK
1 = Failure</td><td>-</td><td>O</td><td>E4</td></tr>
<tr><td>Attribute 2</td><td rowspan="6">Labels used with the identifier to help identify the IQ4. The seven attributes (Attributes 2-7 plus the Identifier) are used in text communications for two functions:
<table><tr><td>Message Reception</td><td>Allow the IQ4 to be identified either uniquely or as part of a group, so that it may receive relevant communications.</td></tr><tr><td>Message Transmission</td><td>Can be used by the IC comms module to identify a controller or group of controllers with which this IQ4 wishes to communicate.</td></tr></table>& | * ? . are reserved for special functions. Single letters should also be avoided (S, K, W etc). For further details see “Attribute Selection - Wildcards” on page 285.</td><td rowspan="6">0 to 30 character string except \ /</td><td rowspan="6">Null</td><td rowspan="6">N</td><td>F</td></tr>
<tr><td>Attribute 3</td><td>G</td></tr>
<tr><td>Attribute 4</td><td>H</td></tr>
<tr><td>Attribute 5</td><td>I</td></tr>
<tr><td>Attribute 6</td><td>J</td></tr>
<tr><td>Attribute 7</td><td>K</td></tr>
<tr><td>Baseboard version</td><td>Baseboard version used.</td><td>Fixed</td><td>-</td><td>N</td><td>b</td></tr>
<tr><td>Bootloader version</td><td>Bootloader version used.</td><td>Fixed</td><td>-</td><td>N</td><td>c</td></tr>
<tr><td>brIQ capacity</td><td>The total amount of strategy memory in the current IQ4. Measured in brIQs.</td><td>-</td><td>-</td><td>N</td><td>a</td></tr>
<tr><td>brIQs used</td><td>The amount of memory used for the current strategy. Measured in brIQs.</td><td>-</td><td>-</td><td>N</td><td>t</td></tr>
<tr><td>Build</td><td>Build number of the firmware version.</td><td>Fixed</td><td>-</td><td>N</td><td>B</td></tr>
</table>

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Checksum OK	The status of the strategy data file in the IQ4. The checksum is stored at power down (or power failure) and compared against the stored value at power restoration. If the checksum, value is the same, it is set to '1' and a CONL alarm generated. If different it is set to '0', and the 'Corrupt Strategy ' parameter is set to '1'. The IQ4 will cease controlling (i.e. outputs will freeze - in the case of power up the outputs will remain frozen in the state they were at power down).	0 or 1 1 = Strategy running 0 = Strategy not running	-	O	e
Controller Online	CONL alarm - Indicates when the IQ4 restarts successfully due to power up or an internal soft reset and loads up and runs a valid strategy - see **"Controller on Line (CONL)" on page 76**.	0 or 1 1 = Successful restart 0 = Unsuccessful restart	-	O	S0
Corrupt Strategy	HELP alarm - Indicates when there is a checksum failure (corrupt strategy). Set when the IQ4 starts due to power up or an internal soft reset and is unable to load up and run a valid strategy - see **"Corrupt Strategy (HELP)" on page 76**.	0 or 1 0 = OK 1 = Failure	-	O	S1
Corrupt Time	Not used	-	-	-	-
Device Discovered	Indicates whether the IQ4 has been discovered by a tool.	0 or 1 0 = Not discovered 1 = Discovered	0	O	-
External IO Count	Total number of external I/O channels connected - only applies to IQ4E, set to '0' for all other types of IQ4.	0 to 176	-	N	o
External IO Enabled	Indicates whether the external I/O modules may be connected - only applies to IQ4E, set to '0' for all other types of IQ4.	0 or 1 0 = No external I/O modules 1 = External I/O modules may be connected	-	N	k
External IO Limit	The number of external I/O channels that may be connected.	0 to 176	-	N	l
Fail SLL	FSSL alarm - indicates that an email alarm has not been sent due to an SLL failure. When the fault is rectified an CSSL alarm is generated.	0 or 1 0 = OK 1 = Fail	-	N	-
Failed Peripheral Interface Adapter	Not used	-	-	-	-
Failed PROM	Not used	-	-	-	-
Failed RTC	FRTC alarm - Indicates if any of the parameters of the real time clock are outside the range i.e. the RTC has failed - see **"Failed RTC (FRTC)" on page 76**.	0 or 1 0 = OK 1 = Fail	-	O	S3
Failed Software	Not used	-	-	-	-
File Error	FLER alarm - Indicates if the IQ4 is unable to load its current strategy file, or the backup to this file, and has had to load the original downloaded file - see **"File Error (FLER)" on page 76**.	0 or 1 0 = OK 1 = Fail	-	O	E6
File List	A number which refers to a list of files included in this version of IQ4. For Trend use only.	-	-	N	f

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
General Alarm Group	The Alarm Group module to which general alarms from the IQ4 are sent. By default set to group 0; alarms in group 0 are logged, but not sent. The alarms must be attached to another group (and the alarm handling strategy constructed) in order to be sent.	0 to 500 0 = not Sent	0	N	g
Identifier	A label which describes the IQ4. It is sent as part of a text alarm message to identify its source and should be unique on the site. It is also used as attribute 1 for text communications.	0 to 30 character string except \ /	Null	N	D
IO Bus Failure	Not used	-	-	-	-
Language selection	The language used for the display of web pages and display and transmission of alarms.	Dansk Deutsch English Espanol Francais Hebrew Italiano Nederlands Norsk Polski Portugues Romanian Russian Simplified Chinese Suomi Svenska Traditional Chinese	English	N	P
Licence Error	Not used	-	-	-	-
Local Address	The network address of the IQ4. It must be unique on the Local LAN. *Note: IQ4s do not have the timekeeper functionality so address 11 is not reserved for this purpose, but it is recommended for timemaster - see* **"Timemaster Operation" on page 488**.	1, 4 to 119 excluding 10	20	N	L
Local Lan	The LAN number of the IQ4. It must be unique on the site. All devices on the Local LAN must have the same LAN number. It is important that the LAN number matches that of the INC type device on the Local LAN. Must be set up for /LAN variants or IQ4NC. *Note: The data is actually held in the Address module but it can also be viewed and changed from the appropriate network module (e.g. Type 4 (Ethernet IP), Type 1 (IQ LAN), and Type 12 (BACnet MSTP) where it is known as 'LAN Number'.*	1, 4 to 119 excluding 10	20	N	N
Low Flash	Not used	-	-	-	-
Low Memory	Not used	-	-	-	-
Onboard IO Count	Number of I/O channels available in the IQ4.	0 to 16	-	N	h

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Override Timer (mins)	The time (minutes) for which a module set to override will stay in override before reverting to normal control. It should be set to the required time, and it will start counting down to '0' when a module is overridden at which point this override (and any other overrides set up) will be removed. This does not apply to overrides which are sourced from a strategy module.	0 to 1440 0 = No override	0	N	T
Plot Memory Size	Memory capacity for plots (log points).	IQ41x: 1000000 IQ422: 1000000 IQ4NC: 1000000 IQ4E: 1000000 IQ4E/32: 1000000 IQ4E/64: 1000000 IQ4E/96: 1000000 IQ4E/128: 1000000 IQ4E/160: 1250000 IQ4E/192: 1500000	-	N	V
Plot Memory Used	The amount of memory already used for plots.	-	-	N	w
Power Reset	Set to '1' on power up for one sequence table cycle.	0 or 1	-	O	P
Send Service PIN	Causes the IQ4 to send its service PIN message (same as the message generated when the Service Button is pressed).	0 or 1 1 = Send message	0	N	!
Sequence Table Overrun	STOR alarm - Indicates when the IQ4 is unable to complete the sequence of modules in the required time - see **"Sequence Table Overrun (STOR)" on page 76**.	0 or 1 0 = No alarm 1 = Alarm	-	O	E1
Serial Comms Failure	Not used	-	-	-	-
Serial Number	The serial number for the IQ4. Set up during manufacture.	String	-	N	M
Site GUID	A unique machine readable alphanumeric code that identifies the site to a computer. It is a world wide unique code (GUID = Global Unique Identifier) and all controllers on the site should have the same site GUID. It is specifically used in IP alarms to identify the source on a multi-site system. It is created in IQSET along with the site name, and can be managed using the IQSET Site Code Manager Applet; this enables the Site Name and Site GUID (together known as Site Codes) to be set the same for the entire site.	-	Null	N	O
Site Name	A human readable site identifier. All controllers on the site should have the same site name. It is specifically used in remote alarms to identify the alarm source on a multi-site system. It is created in IQSET and along with the site GUID and can be managed using the IQSET Site Code Manager Applet; this enables the Site Name and Site GUID (together known as Site Codes) to be set the same for the entire site.	0 to 30 character string	IQ4's MAC address	N	$
Strategy Revision	The revision of strategy. Incremented by IQSET each time strategy download file is created.	1 to max	1	N	d

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Supervisor Port	The network address of the local supervisor port on Local LAN. Unique on LAN. If set to '0' the device connected to the port can only communicate with the IQ4. If set to non-zero it allows the connected device to talk over the network.	0, 1, 4 to 119 excluding 10 0 = Local device only	0	N	U
Type Name	The IQ4 type.	IQ41x, IQ422, IQ4E, or IQ4NC	IQ41x, IQ422, IQ4E, IQ4NC	N	y
USB Supervisor Port	The network address of the sCNC associated with the local engineering (USB) port. This port is for use by IQSET and enables it to access the whole network. If the set to '0' a sCNC at address 125 is created for the period of the IQSET session. When the IQSET PC is removed the sCNC times out and no longer exists on the network. If set to non-zero, the sCNC remains on when the IQSET PC is removed.	0, 1, 4 to 119 excluding 10	0	N	Y
Version	String indicating IQ4 type, firmware issue, I/O count and date.	25 character string	-	N	C
XNC Enabled	Indicates if XNC functionality is enabled - only applies to /XNC variants, set to '0' for all other variants.	0 or 1 0 = Disabled 1 = Enabled	-	N	x

2.3 Address Module Connectives

The following parameters are available for connection to other modules in the strategy:

Archive Failure
Controller Online
Corrupt Strategy
Failed RTC
Local Address
Local Lan
Power Reset
Sequence Table Overrun

3 ALARM DESTINATION MODULES

Module Identifier	e
Default Overview Parameters	$, F
Size in brIQs	14

3.1 Functionality

Alarm Destination modules define the destination to which general, item and network alarms (IQ4NC only), are sent and when the alarms are sent. They contain the parameters necessary to deliver the alarms. The parameters are dependent on the type of Alarm Destination module - see **"Alarm Destination Module Types" on page 173**.

When an alarm is submitted for delivery the module will compare the priority of the alarm with its own hold priority and if the alarm's priority level is greater, the alarm will be sent. If the alarm is not sent, it will be stored until the priority of another incoming alarm is higher than the Alarm Destination module's 'Hold Priority' parameter. If an alarm is sent, then any other alarms being held will also be sent. This allows calls to autodialled destinations to be limited to times when a high priority alarm is to be sent. The hold priority may be changed according to a time schedule in order to ensure that all low priority alarms are sent every day (e.g. during time of low call charges) in the event that a high priority alarm is not generated.

Alarm Destination modules appear on the strategy page but do not need to be sequence table or event driven.

For a detailed description of alarm handling see **"Alarm Handling Strategy" on page 79.** On IQ4NCs the Alarm Destination modules can also be used for alarm forwarding - see **"INC Alarm Forwarding (IQ4NC only)" on page 91**, and for sending network alarms to a specific destination - see **"Network Alarms" on page 71**.

3.1.1 Delivery failure

If the alarm is unable to reach the destination it will retry as follows:

1. Retry 5 times at 30 second intervals (i.e. a total of 6 times before the wait).
2. Wait length of time defined by the 'Retry Interval' parameter.
3. Retry 6 times at 30 second intervals.
4. Set 'Failed' parameter to '1'.
5. Repeat steps 2 and 3 until an is acknowledge is received.
6. Once an acknowledge is received the 'Failed' parameter is set to '0'.

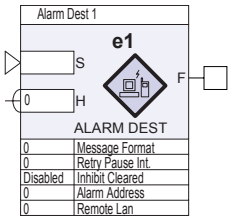
The Alarm Destination module can only send one alarm at a time. If the destination has failed the destination is blocked and the 'Failed' parameter set to '1'. The 'Failed' parameter may be used to switch Route modules to send alarms to alternative destinations - see **"Two Routes, Use only Second on Fail" on page 88**.

3.2 Alarm Destination Module Types

There are four different type of Alarm Destination modules specified using the 'Type' parameter:

Module Type	*Description*
Type 1 (IQ Lan)	Alarms are sent to another Trend device (e.g. a supervisor) using Trend communications protocols addressed using the 'LAN Number' and 'Local Address' parameter.
Type 2 (IP Address)	Alarms are sent to a device using IP protocol (e.g. to devices connected to Ethernet).
Type 3 (Email)	Alarms are sent to an email address.
Type 4 (BACnet)	Alarms are sent to a BACnet device using BACnet protocol (/BAC variants and IQ4NC only).

3.2.1 Type 1 (IQ Lan)



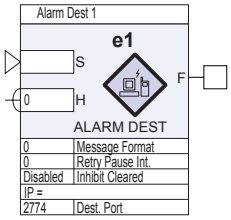
Type 1 (IQ Lan) Alarm Destination modules function in the way described in **"Functionality" on page 173** enabling alarms to be sent to a Trend address (LAN number and device address).

3.2.1.1 Type 1 (IQ Lan) Alarm Destination Modules Parameters

Type 1 (IQ Lan) Alarm Destination modules include the parameters common to all Alarm Destination modules - see **"Alarm Destination Module Common Parameters" on page 177**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Address	Device address of the device to which alarms are to be sent.	0, 1, 4 to 119 excluding 10 0 = not used	0	N	A
LAN	LAN number of the device to which alarms are to be sent.	0, 1, 4 to 119 excluding 10 0 = Local LAN	0	N	N
Message Format	The format of the alarm message. Coded and attribute alarms are for decoding by supervisors. Text alarms are man-readable - see **"Format of Alarms Sent Using the Alarm Handling Strategy" on page 82**. Always use Attribute alarms unless sending to a legacy supervisor or display that does not support them use in which case 'Text'. *Note: The IQ4NC is able to forward alarms, but in doing so it will validate the destination type with respect to the incoming alarm format and may overrule the type used or ignore the alarm altogether - see* **"Alarm Format Validation (IQ4NC only)" on page 85**.	0, 1, 2, 3 0 = Coded 1 = Text 2 = Attribute 3 = Enhanced (not used)	0	N	M

3.2.2 Type 2 (IP Address)



Type 2 (IP Address) Alarm Destination modules function in the way described in **"Functionality" on page 173** enabling alarms to be sent to an IP address or host name on a specific port.

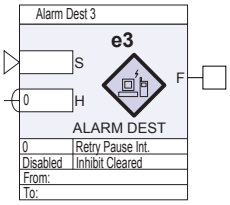
3.2.2.1 Type 2 (IP Address) Alarm Destination Module Parameters

Type 2 (IP Address) Alarm Destination modules include the parameters common to all Alarm Destination modules - see **"Alarm Destination Module Common Parameters" on page 177**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The IP address or hostname of the device to which alarms are to be sent.	IP addresses or host name	128.1.1.4	N	i

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination Port	The TCP port on the device to which alarms are to be sent , where the receiving device will 'listen' for alarms.	0 to 65535	2774	N	P
Message Format	It can be set to either Coded, Text or Attribute. Coded and attribute alarms are for decoding by supervisors. Text alarms are man-readable - see **"Format of Alarms Sent Using the Alarm Handling Strategy" on page 82**. *Note: The IQ4NC is able to forward alarms, but in doing so it will validate the destination type with respect to the incoming alarm format and may overrule the type used or ignore the alarm altogether see* **"Alarm Format Validation (IQ4NC only)" on page 85**.	0, 1, 2, 3 0 = Coded 1 = Text 2 = Attribute 3 = Enhanced (not used)	0	N	M

3.2.3 Type 3 (Email)



Type 3 (Email) Alarm Destination modules function in the way described in **"Functionality" on page 173** enabling alarms to be sent to an email address. v3.30 firmware and greater provide secure transmission of the email (SSL or STARTTLS) providing the Type 4 (Ethernet IP) Network module's 'Email Port', 'Email Security', and 'Ignore Email Certificate' parameters are set up correctly - see **"Type 4 (Ethernet IP) Network Module Parameters" on page 393**.

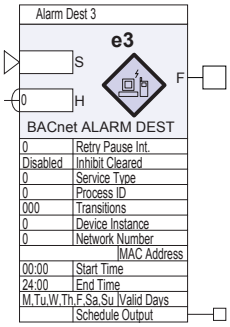
See **"Email Alarm Destinations" on page 81** for a description of the email alarm format.

3.2.3.1 Type 3 (Email) Alarm Destination Module Parameters

Type 3 (Email) Alarm Destination modules include the parameters common to all Alarm Destination modules - see **"Alarm Destination Module Common Parameters" on page 177**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Extra Text	Text which is added to the end of the alarm message.	0 to 256 character string.	Blank	N	e
From Email Address	The email address which can receive email replies from the device to which alarms are to be sent.	0 to 64 character string.	Blank	N	f
To Email Address	The email address of the device to which alarms are to be sent.	0 to 64 character string.	Blank	N	a

3.2.4 Type 4 (BACnet)



Type 4 (BACnet) Alarm Destination modules only available in /BAC variants and IQ4NCs. They function in the way described in **"Functionality" on page 173** enabling alarms to be sent to a BACnet device using BACnet protocol as a BACnet notification.

Note: The IQ4's Type 5 (BACnet IP) Network module must be enabled and set up in order to be able to send BACnet type alarms.

3.2.4.1 Type 4 (BACnet) Alarm Destination Module Parameters

Type 4 (BACnet) Alarm Destination modules include the parameters common to all Alarm Destination modules - see **"Alarm Destination Module Common Parameters" on page 177**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Device instance	The BACnet device instance of the device to which the alarms are to be sent. If set to '0' the 'Network Number' and 'MAC Address' parameters are used to address the BACnet device to which the alarms are to be sent, otherwise it is addressed using the 'Device instance' parameter. It must be set up if remote device is using DHCP.	0 to 419439302	0	N	i
End Time	IQ4 v3.40 or greater. The end time of the period during which the alarms are to be sent to the destination in the format HH:MM:SS.	00:00:00 to 23:59:59	00:00:00	I	e
MAC Address	The BACnet MAC Address of the device to which the alarms are to be sent. When the 'Device instance' parameter is set to '0' it is used in conjunction with the 'Network Number' parameter to address the BACnet device to which alarms are to be sent. *Note: BACnet MAC addresses do not have colon separators (e.g A5C3A1BABAC0).*	Valid MAC address	Blank	N	a
Network Number	The network number of the BACnet device to which the alarms are to be sent. When the 'Device instance' parameter is set to '0' it is used in conjunction with the 'MAC Address' parameter to address the BACnet device to which alarms are to be sent.	0 to 65534	0	N	n
Process ID	An integer that can be used in alarm messages to identify the alarm and enable the device receiving the alarm to process it in a particular way, e.g. a particular ID could be used to identify all HVAC alarms.	0 to 4294967295	0	N	P
Schedule Output	IQ4 v3.40 or greater. Indicates if the module is in the period alarms are sent to the destination i.e. time is between the start and end time and valid day bit is set.	0 or 1 0 = Outside period 1 = Inside period	0	O	O

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Service Type	Defines whether the IQ4 expects an acknowledgement for the alarm (Confirmed) or will just send once and not expect an acknowledgement (Unconfirmed). If set to Unconfirmed the IQ4 acknowledges the alarm itself so the 'Failed' parameter will never get set to '1' as the IQ4 has no way of knowing if the alarm transmission was successful. When set to confirmed, the IQ4 will expect an acknowledgement of the alarm, and will continue to send the alarm until one is received, or it has determined that a failure has occurred.	0 or 1 0 = Confirmed 1 = Unconfirmed	0	N	T
Start Time	IQ4 v3.40 or greater. The start time of the period during which the alarms are to be sent to the destination in the format HH:MM:SS.	00:00:00 to 23:59:59	00:00:00	I	s
Transitions	Specifies the changes in status that cause alarms to be sent. There are three different changes in alarm state that can generate alarms: *State Change* / *Description* To Normal – The value has returned to the normal range. To Fault – The value has changed to one that is the fault range. To OffNormal – The value has changed to one that is not in the normal range. It is bit field containing 3 bits: the left most bit enables/disables the To-Normal alarm, the middle bit enables/disables the To-Fault alarm, and the right most bit enables/disables the To-OffNormal alarm - see "Alarm State Transitions" on page 65.	000 to 111	111	N	R
Valid Days	IQ4 v3.40 or greater. The days during which the alarms are to be sent to the destination formatted as a bit string where the first bit represents Monday and last bit Sunday. E.g. 1000001 would be Monday and Sunday'.	0000000 to 1111111	0000000	I	w

3.3 Alarm Destination Module Common Parameters

Alarm destination modules include the following parameters that are common to most modules: 'Label', 'Page,' 'X co-ordinate' and 'Y co-ordinate' - see "Common IQ4 Module Parameters" on page 165, plus the following:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Failed	Indicates when the module is unable to reach the specified destination. Set to '1' when the destination device fails to acknowledge the receipt of an alarm after the second retry - see "Delivery failure" on page 173.	0 or 1 0 = Received 1 = Failed	0	O	F
Hold Priority	The value defines the priority level below which alarms are held back. If an alarm is received above this level it, and all held alarms, will be sent. It may be overridden if a high number of alarms are waiting to be sent - see "High Water Mark" on page 181. It can be set by strategy to enable the hold priority level to be lowered when communications demand is lower (e.g. during non-occupation) to ensure that all held alarms are sent, or it can be set to a specific value.	0 to 255	0	I	H

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Inhibit Clear	Specifies whether the Alarm Destination module will send clear alarms.	0 or 1 0 = Clear alarms sent 1 = Clear alarms not sent	0	N	C
Retry Interval	The interval (in minutes) between batches of attempts to send alarm to destination. The interval should be set to '0' for autodialling to allow continual retries at 30 seconds.	0 to 32767	0	N	I
Source	The Alarm Route module(s) from which the Alarm Destination module accepts alarms.	Route modules	No connection	I	Sc
Type	The type of destination to which alarms are to be sent. It is set up in IQSET, and cannot be changed from the web page. The available parameters change with the destination type. *Note: The IQ4NC is able to forward alarms, but in doing so it will validate the destination type with respect to the incoming alarm format and may overrule the type used or ignore the alarm altogether see* **"Alarm Format Validation (IQ4NC only)" on page 85**.	1, 2, 3, or 4 1 = IQ LAN 2 = IP address 3 = Email 4 = BACnet (IQ4/BAC, IQ4NC only)	1	N	Y

3.4 Alarm Destination Module Connectives

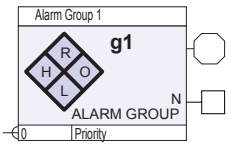
The following parameter is available for connection to other modules in the strategy:

Failed

4 ALARM GROUP MODULES

Module Identifier	g
Default Overview Parameter	$
Size in brIQs	9

4.1 Functionality



Alarm Group modules are used to group alarms that are to be handled in the same way. When the module receives an alarm it assigns the alarm the priority defined by the 'Hold Priority' parameter and passes it to the Alarm Route module specified by its 'Output' parameter. Alarms should be linked to the same Alarm Group module if they are to have the same priority and are to go to the same destination(s).

4.2 Alarm Group Module Parameters

Alarm Group modules include the following parameters that are common to most modules: 'Label', 'Page', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Hold Priority	The priority of the alarm.	0 to 255 255 = highest priority	0	I	P
New Alarm	Indicates when a new alarm occurs. It is set to '1' for one sequence cycle whenever a new alarm, attached to the Alarm Group module, is generated. If alarms are occurring faster that once per sequence cycle, it will remain at '1' until one sequence cycle after the last alarm occurred.	0 or 1 1 = Alarm 0 = No alarm	0	O	N
Output	The Alarm Route module(s) used to route the alarms to their destination(s).	Alarm Route modules	No connection	O	Oc
User ACK Enable	v3.50 or greater. Specifies which BACnet alarm state transitions require user acknowledgement - see **"BACnet User Alarm Acknowledgments" on page 67**.	000 to 111	000	O	U

4.3 Alarm Group Module Connectives

The following parameter is available for connection to other modules in the strategy:

New Alarm
Output

5 ALARM LOG MODULES

Module Identifier	vn (where n = alarm log record number)

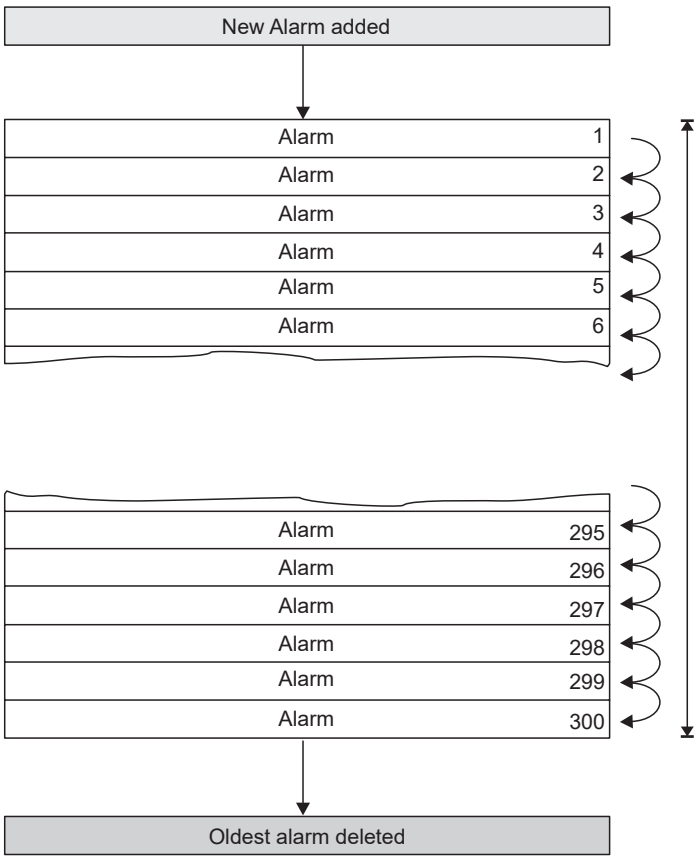
5.1 Functionality

The alarm log contains a list of the last 300 alarms that have occurred in the IQ4. All network (IQ4NC only), general, and item alarms are stored in the alarm log. Each logged alarm is stored in an Alarm Log module. Alarm Log module 1 is the most recent entry in the log.

Alarm Log modules do not appear on the strategy page and do not need to be sequenced.

5.1.1 Add/Removing Alarms

The alarm log has a maximum length of 300 alarms (Alarm Log modules). A new alarm is always added as alarm 1; the other alarms 'roll on' one position. Once the log is full, when a new alarm is added, the oldest alarm is deleted. It will be lost from the log, and no further attempts will be made to send it.



Note: Because Alarm Destination modules only hold the alarm they are trying to send (i.e. destination is blocked) the Alarm Destination module may have other alarms waiting in a stack. If an alarm is deleted as the oldest alarm in the alarm log it will also be deleted from any Alarm Destination modules' stacks. However, if the Alarm Destination module is currently trying to send the alarm, it will not be deleted from the Alarm Destination module.

5.1.2 High Water Mark

In order to maximise the chances of an alarm being sent before it is deleted there is a level, known as the 'high water mark', above which any alarms that have not been sent (because their hold priority is below the hold priority of the Alarm Destination module being used to send them) will be sent regardless of their priority. This in effect ignores the hold priority in the Alarm Destination module.

The high water mark is set at 80% (i.e. the 241st alarm will trigger the high water mark).

5.1.3 Alarm Log Format

The alarm log is designed so that its database can be queried using text communications.

5.2 Alarm Log Module Parameters

<table>
<tr><th>Parameter</th><th>Description</th><th>Range</th><th>Default</th><th>Type</th><th>Text Comms</th></tr>
<tr><td>Alarm status</td><td>Indicates whether the alarm has been sent (i.e. acknowledged by the destination(s) required at the time). If the alarm is being sent to multiple destinations it will remain ‘waiting’ until all destinations have acknowledged at which point it will change to ‘sent’.</td><td>0, 1, or 2
0 = unused
1 = sent
2 = waiting</td><td>0</td><td>O</td><td>L</td></tr>
<tr><td>Alarm type</td><td>The type of alarm. Meaning varies with module type:
<table>
<tr><th>Alarm Type</th><th>Description</th></tr>
<tr><td>Sensor</td><td>0 = high
1 = low
2 = out of limits
3 = read</td></tr>
<tr><td>Digital Input</td><td>0 = digin off
1 = digin on</td></tr>
<tr><td>Loop</td><td>0 = SP dev
1 = PV fail</td></tr>
<tr><td>Driver</td><td>0 = readback
1 = maintenance</td></tr>
<tr><td>Plot:</td><td>0 = buffer ready (Event)</td></tr>
<tr><td>General</td><td>0 = CONL
1 = data error
2 = failed PIA
3 = failed RTC
4 = failed RAM*
5 = failed SWR*
6 = failed DART*
7 = failed PROM
8 = overrun sequence table
9 = low memory*
10 = fail I/O bus*
11 = low flash*
12 = fail archive
13 = time corrupt*
14 = not used*
15 = file error*
16 = licence error*
*These alarms do not operate</td></tr>
<tr><td>Network</td><td>Not used</td></tr>
</table></td><td>0 to 16</td><td>0</td><td>N</td><td>P</td></tr>
<tr><td>Alarm Type</td><td>Text of alarm type e.g. ‘High Value’.</td><td>0 to 11 characters</td><td>Null</td><td>N</td><td>T</td></tr>
<tr><td>day</td><td>Day part of date alarm occurred.</td><td>1 to 31</td><td>0</td><td>N</td><td>D</td></tr>
<tr><td>group</td><td>Alarm group module to which the alarm is linked.</td><td>0 to 500</td><td>0</td><td>N</td><td>G</td></tr>
<tr><td>Hours</td><td>Hours part of time alarm occurred.</td><td>0 to 23</td><td>0</td><td>N</td><td>H</td></tr>
<tr><td>Label</td><td>Label of the module in alarm.</td><td>0 to 47 characters</td><td>Null</td><td>N</td><td>$</td></tr>
<tr><td>minutes</td><td>Minutes part of time alarm occurred.</td><td>0 to 59</td><td>0</td><td>N</td><td>N</td></tr>
<tr><td>module number</td><td>The number of the module generating alarm (not used for general alarms).</td><td>0 to 255</td><td>0</td><td>N</td><td>U</td></tr>
</table>

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
module type	The type of module sending the alarm.	A to Z, a to z S = sensor I = digital input L = loop D = driver G = general n = network	Null	N	O
month	Month part of date alarm occurred.	1 to 12	0	N	M
priority	The priority attached to the alarm.	0 to 255	0	N	I
reported state	Indicates the alarm state 'Clear' or 'Occurred'.	0 or 1 0 = Clear alarm 1 = Occurred	0	N	R
seconds	Seconds part of time alarm occurred.	0 to 59	0	N	S
value	Value of module in alarm at the time alarm was generated.	Real number or bit status	-	N	V
year	Year part of date alarm occurred.	00 to 99	0	N	Y

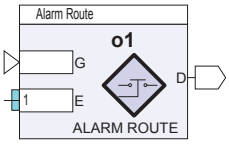
5.3 Alarm Log Module Connectives

None

6 ALARM ROUTE MODULES

Module Identifier	o
Default Overview Parameters	$,E
Size in brIQs	9

6.1 Functionality



Alarm Route modules connect an Alarm Group module to an Alarm Destination module. When the module receives an alarm from the Alarm Group module it passes the alarm to the Alarm Destination module unless its 'Enable' parameter is set to '0'. Several Alarm Route modules can be connected to the same Alarm Destination module. Alarm Route modules appear on the strategy page but do not need to be sequenced.

6.2 Alarm Route Module Parameters

Alarm Route modules include the following parameters that are common to most modules: 'Label', 'Page', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm Group	The Alarm Group module(s) for which it is to route alarms.	Alarm Group modules	Blank	N	Gc
Destination	The Alarm Destination module to which the module will send alarms.	Alarm Destination module only	No connection	O	Dc
Enable	Enables/Disables the route. When enabled it will pass alarms to the Alarm Destination module, when disabled no alarms are passed. It can be sourced from a digital output or set to a specific value. Connection to a digital output allows the strategy to select the destination to which alarms are routed. For example an alternative destination may be used if the first one fails - see **"Two Routes, Use only Second on Fail" on page 88**.	0 or 1 0 = Disabled 1 = Enabled	1	N	E

6.3 Alarm Route Module Connectives

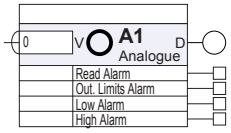
The following parameter is available for connection to other modules in the strategy:

Destination

7 ANALOGUE NODE MODULES

Module Identifier	A
Default Overview Parameters	$,D
Size in brIQs	14

7.1 Functionality



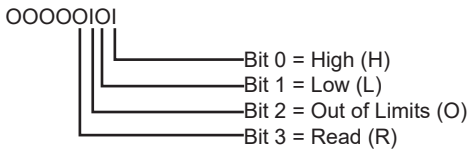
Analogue Node modules store analogue values. They are used for several functions:

- For reception of IC communications, or for use with devices using text communications to nodes (e.g. wireless sensor receiver).
- Provision of a constant value shared between several module inputs.
- Compatibility with supervisors set up for IQ1 or IQ2 strategies.
- Detection of alarm status bits from Function module outputs.

Analogue Node modules appear on the strategy page and are event driven and cannot be allocated a sequence step. Since Analogue Node modules are always event driven, the output should never be connected back to the input.

7.2 Analogue Node Module Parameters

Analogue Node modules include the following parameters that are common to most modules: ‘Label’, ‘Page’, ‘X co-ordinate’ and ‘Y co-ordinate’ - see **“Common IQ4 Module Parameters” on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm Status	The alarm status of the analogue value connected to the ‘Value’ parameter. Remains at the default if the ‘Value’ parameter is set to a specific value. The least significant four alarm bits correspond sensor module’s four alarm bits R (Read Alarm), O (Out of Limits), L (Low), H (High).  The web page shows eight status bits; these can be accessed by text communications. Only the four lowest bits (ROLH) are available as connectives and can be used in the strategy. Status bits 4 and 5 can be set up by an IC communications message, but can only be accessed by text communications.	00000000 to 11111111	0	N	S
High Alarm	The module’s high alarm output, bit 0 from the ‘Alarm Status’ parameter.	0 or 1 0 = No alarm 1 = Alarm	0	O	H
Low Alarm	The module’s low alarm output, bit 1 from the ‘Alarm Status’ parameter.	0 or 1 0 = No alarm 1 = Alarm	0	O	L
Out of Limits Alarm	The module’s out of limits alarm output, bit 2 from the ‘Alarm Status’ parameter.	0 or 1 0 = No alarm 1 = Alarm	0	O	O
Output	The module’s output, it is equal to the ‘Value’ parameter.	Full analogue	0	O	D
Read Alarm	The module’s read alarm output, bit 3 from the ‘Alarm Status’ parameter.	0 or 1 0 = No alarm 1 = Alarm	0	O	R

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Value	The value of the Analogue Node module. It can be connected to a module's analogue output, can be a specific value, or an input from another controller using IC communications. If connected to an analogue output it will expose the status bits accompanying the analogue value (e.g. from function module). If configured as a specific value then it can provide a constant value for use by other modules.	Full analogue	0	I	V

7.3 Analogue Node Module Connectives

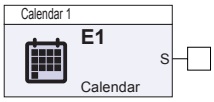
The following parameters are available for connection to other modules in the strategy:

High Alarm
Low Alarm
Out of Limits Alarm
Read Alarm
Output

8 CALENDAR MODULES

Module Identifier	E
Default overview parameters	-
Size in brIQs	-

8.1 Functionality



v3.50 or greater firmware only. Calendar modules are designed to provide a Time Schedule module or exception with the date(s) it is to apply to. The output ('Module Status' parameter) is set to '0' or '1' depending on the schedule set in the module.

8.1.1 Calendar Entries

The module's schedule is made up of one or more Calendar Entries (Calendar Entry sub-modules) each of which define a date or range of dates where the schedule is ON and the Calendar module's output set to '1'. There are three different ways the dates a Calender Entry applies to can be specified, determined by the 'Date Option' parameter:

Date or Date Pattern
Range of Dates
Week and Day

8.1.1.1 Date or Date Pattern

Date or Date Pattern determines the day(s) a Calendar Entry applies to based on the combination of the Calendar Entry sub-module's 'Start Day', 'Start Month' and 'Start Year' parameters e.g:

Start Day = 25
Start Month = 12
Start Year = 2017

would apply the Calendar Entry on 25th of December 2017.

Wildcards

Wildcards can be used for one or more of the 'Start Day', 'Start Month' and 'Start Year' parameters to enable a date pattern to be specified, e.g. to repeat a date every year. To specify the 25th of December every year the 'Start Day', 'Start Month' and 'Start Year' parameters would be set as shown:

Start Day = 25
Start Month = 12
Start Year = 0 (Unspecified)

Start Day wildcards: It is possible to specify any day, odd days, even days or the last day of the specified month by setting the 'Start Day' parameter appropriately:

0 = Unspecified (any day of specified month)
32 = Last day of specified month
33 = Every odd day of specified month
34 = Every even day of specified month

Start Month wildcards: It is possible to specify any month, odd months (January, March, May, July, September, November), or even months (February, April, June, August, October, December) of the specified year by setting the 'Start Month' parameter appropriately:

0 = Unspecified (any month of specified year)
13 = Every odd month
14 = Every even month

Start Year wildcards: It is possible to specify any year by setting the 'Start Year' parameter appropriately:

0 = Unspecified (any year)

8.1.1.2 Range of Dates

Range of Dates determines the day(s) a Calendar Entry applies to as the period(s) between the start date (determined by the combination of the Calendar Entry sub-module's 'Start Day', 'Start Month' and 'Start Year' parameters) and the end date (determined by the combination of the Calendar Entry sub-module's 'End Day', 'End Month' and 'End Year' parameters) e.g:

Start Day = 1
Start Month = 12
Start Year = 2017
End Day = 31
End Month = 12
End Year = 2017

would apply the Calendar Entry for all of December 2017.

Wildcards

Start date: The 'Start Day', 'Start Month', and 'Start Year' parameters can be left as '0' (Unspecified) to start the date range from now.

Note: If wildcards are being used all three parameters must be set to unspecified. If one ore more is specified (no zero) the start date will be invalid.

End date: The 'End Day', 'End Month', and 'End Year' parameters can be left as '0' (Unspecified) to continue the date range forever.

Note: If wildcards are being used all three parameters must be set to unspecified. If one ore more is specified (no zero) the end date will be invalid.

Example: To specify the range to start on any day and end on 31st of December 2017 the 'Start Day', 'Start Month', 'Start Year', 'End Day', 'End Month' and 'End Year' parameters would be set as shown:

Start Day = 0 (Unspecified)
Start Month = 0 (Unspecified)
Start Year = 0 (Unspecified)
End Day = 31
End Month = 12
End Year = 2017

8.1.1.3 Week and Day

Week and Day enables the day(s) a Calendar Entry applies to be a specific day of the specified week, specific week of the specified month, or a specific month using a combination of the 'Day of Week', 'Week of Month' and 'Start Month' parameter, e.g. every Tuesday in March could be specified by setting the parameters as shown:

Day of Week = 2 (Tuesday)
Week of Month = 0 (Unspecified)
Start Month = 3 (March)

Wildcards

Wildcards can be used for one or more of the 'Day of Week', 'Week of Month' and 'Start Month' parameters.

Day of Week wildcards: It is possible to specify any day of the week:

0 = Unspecified (any day of the week)

Week of Month wildcards: It is possible to specify any week or the last week of the setting the 'Week of Month' parameter appropriately:

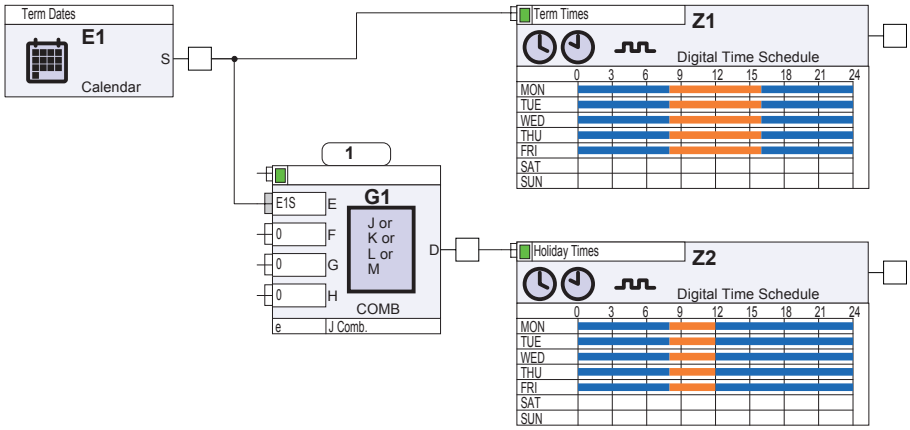
0 = Unspecified (any week of the month)
6 = Last week of the month

Start Month wildcards: It is possible to specify any month or odd /even months by setting the 'Month of Year' parameter appropriately:

0 = Unspecified (any month)
13 = Every odd month
14 = Every even month

8.1.2 Link to Time Schedule Modules

Calendar modules can be linked to the 'Disable Module' parameter of a Time Schedule module. This enables a Time Schedule module to be enabled during a period specified by the Calendar module. E.g a Time Schedule module could be configured so that it is only used during the school holidays. The Calendar module would contain the complete list of holidays or term time periods. This would enable and disable the term time and holiday Time Schedule module via a Logic module.



Note: This configuration requires the connected Time Schedule module's effective period to be set to be ON during any period the Calendar module might be ON - see **"When the Time Schedule Applies (Effective Period)" on page 495**.

8.1.3 Link to Time Schedule Module Exceptions

Calendar modules can be linked to the 'Calendar Reference' parameter of a Time Schedule module's Exception. When an Exception has a Calendar module specified the Calendar module determines the date(s) the Exception applies to. The times used are those specified by the exception and the dates are provided by the linked Calendar module. It is not possible to specify any dates. This enables easy control of the days different exceptions apply. E.g A Calendar module, connected to exceptions in different Time Schedule modules, would contain the complete list of banks holidays. This would apply the connected exceptions, which could specify a different occupation profile, on the days specified by the Calendar module.

8.2 Calendar Module Parameters

8.2.1 Module Properties

Calendar modules include the following parameter common to most modules: 'Label' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Date List	A list of Calendar Entries that have been added to the module.	List of Calendar Entries	-	N	D
Module Status	The module's status.	0 or 1 1 = Schedule indicates ON.	0	O	S
PIN Level	The PIN level required before the module can be adjusted.	0 to 99	0	N	P

8.2.2 Calendar Entry Sub-Module Parameters

Each Calendar Entry sub-module includes the following parameter common to most modules: 'Label' see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Each of the Calendar Entry sub-modules contains a number parameters which are identified in text comms as:

En(Dn(<text comms>))

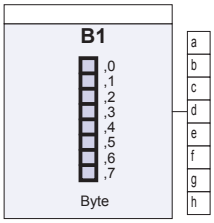
Where En specifies the Calendar module, Dn specifies the Calendar Entry sub-module and <text comms> is the Calendar Entry sub-module's text comms identifier.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Day of the Week	The day of the week when the schedule will come ON. When set to 'Unspecified' any day is used. Only available when 'Date Option' is set to 'Week and Day'.	0 to 7 0 = Unspecified 1 = Monday	-	N	A
End Day	The day of the month when the schedule will go OFF. When set to 'Unspecified' any day is used. Only available when 'Date Option' is set to 'Range of Dates'.	0 to 31 0 = Unspecified	-	N	E
End Month	The month of the year when the schedule go OFF. When set to 'Unspecified' any month is used. Only available when 'Date Option' is set to 'Range of Dates'.	0 to 12 0 = Unspecified 1 = January	-	N	T
End Year	The year when the schedule will go off. When set to 'Unspecified' any year is used. Only available when 'Date Option' is set to 'Range of Dates'.	0 to 9999 0 = Unspecified	-	N	R
Start Day	The day of the month when the schedule will come ON. When set to 'Unspecified' any day is used. Not used when 'Date Option' is set to 'Week and Day'.	0 to 34 0 = Unspecified 1 to 31 = Day of month 32 = Last day of month 33 = Every odd day 34 = Every even day	-	N	S
Start Month	The month of the year when the schedule will come ON. When set to 'Unspecified' any month is used.	0 to 14 0 = Unspecified 1 to 12 = Month of year (1 = January) 13 = Every odd month 14 = Every even month	-	N	O
Start Year	The year when the schedule will come ON. When set to 'Unspecified' any year is used. Not used when 'Date Option' is set to 'Week and Day'.	0 to 9999 0 = Unspecified	-	N	Y
Week of Month	The week of the month when the schedule will come ON. When set to 'Unspecified' any week is used. Only available when 'Date Option' is set to 'Week and Day'.	0 to 6 0 = Unspecified 1 = First (days 1 to 7) 2 = Second (days 8 to 14) 3 = Third (days 15 to 21) 4 = Forth (days 22 to 28) 5 = Fifth (days 29 to 31) 6 = Last	-	N	W
Date Option	Specifies how the Calendar entry is to define the date(s).	0 to 3 1 = Date or Date Pattern 2 = Range of dates 3 = Week and Day	1	N	u

9 DIGITAL BYTE MODULES

Module Identifier	B
Default Overview Parameters	$,S
Size in brIQs	16

9.1 Functionality



Digital Byte modules store a byte of information (8 bits 7, 6, 5, 4, 3, 2, 1, 0 e.g. 01010100) identified as ‘h’ to ‘a’ respectively. They are intended to be used to receive digital IC communications.

Digital Byte modules appear on the strategy page and are event driven and cannot be allocated a sequence step. Since Digital Byte modules are always event driven, the output should never be connected back to the input.

9.2 Digital Byte Module Parameters

Digital Byte modules include the following parameters that are common to most modules: ‘Label’, ‘Page’, ‘X co-ordinate’ and ‘Y co-ordinate’ - see **“Common IQ4 Module Parameters” on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Bit 0	The status of bits 7 to 0 respectively. A single bit (e.g. B1(S0 = I)) may be written to using IC comms.	0 or 1	0	O	a or S0
Bit 1					b or S1
Bit 2					c or S2
Bit 3					d or S3
Bit 4					e or S4
Bit 5					f or S5
Bit 6					g or S6
Bit 7					h or S7
Value	The digital byte status in binary form (e.g. 00001111, Bit 7 through to Bit 0) showing all 8 bits. The whole byte (e.g. B1(S = OOIIOIIO)) may be written to using IC communications.	00000000 to 11111111	0	O	S

9.3 Digital Byte Module Connectives

The following parameters are available for connection to other modules in the strategy (connectives):

Bit 0
Bit 1
Bit 2
Bit 3
Bit 4
Bit 5
Bit 6
Bit 7

10 DIGITAL INPUT MODULES

Module Identifier	I
Default Overview Parameters	$,S
Size in brIQs	28

10.1 Functionality

Digital Input modules take a digital input from a digital value in the IQ4 or from a physical external input and make it available for use by other modules. For this reason there are two types of Digital Input module:

Type 1 (Internal)
Type 2 (External)

Digital inputs are simple ON or OFF inputs and may be derived from devices such as pressure switches, feedback contacts on a motor starter, duty/standby changeover relays or from local override switches for ventilation systems etc. They will also accept pulse train type signals such as those derived from meters (electricity, gas, water flow etc.), which, in conjunction with a Type 4 (Counter) Logic module, can be used to supply such information as consumption, flow rate etc. This will allow, for example, an electricity meter input to show total energy consumption rate in kilowatts (kW) or total energy usage (kWh).

Both module types provide hours run, and number of starts information in addition to the current state of the input, and can generate an alarm if the input is not at the required state. The hours run, and number of starts can be reset by text communications or from the web page. The Digital Input module's 'State' parameter can be overridden to a particular value.

10.1.1 Number of Starts, Hours Run and State

Digital Input modules maintain a count of the number of starts, and the total hours run. The module counts the number of times its input ('Source' parameter) changes from '0' to '1' and makes it available in the 'Number of Starts' parameter. A cumulative sum of the time for which the input is in the '1' state is available in the 'Hours Run' parameter.

Type 1 (Internal) Digital Input modules check the status of the input each time it is serviced, adjusting the 'Number of Starts', 'Hours Run' and 'State' parameters appropriately.

Type 2 (External) Digital Input modules check the status of the input at 80 Hz to facilitate monitoring of pulse inputs of up to 30 Hz (minimum pulse width 16.6 milliseconds), adjusting the 'Number of Starts', 'Hours Run' and 'State' parameters appropriately. In order to be monitored, the input must be ON or OFF for a minimum of 16.6 mS. The number of starts, hours run, and current state information are passed from the hardware to the Digital Input module on the first four changes. In order to avoid errors due to a rapidly changing input value the 4th change initiates a hold OFF for one second, so that if a 5th transition occurs within the one second hold OFF, the values will not be passed to the Digital Input module until that one second hold OFF has elapsed.

10.1.2 Alarms

Digital Input modules generate a single alarm (DIGIN) when the module's 'State' parameter is different from the required state for longer than the alarm delay. The alarm is expressed by one of two alarm codes dependent on the required state:

DI = 0 - Indicates that the required state is 1 and the digital input state is 0.
DI = 1 - Indicates that the required state is 0 and the digital input state is 1.

When the module's 'State' parameter returns to the required state for longer than the clear alarm delay a clear alarm (CLEAR DIGIN) is generated:

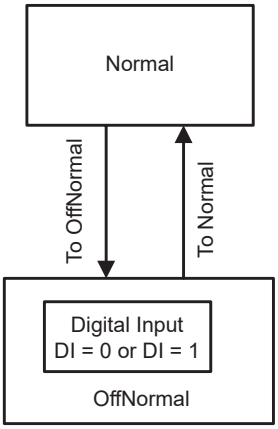
CDI0 - The alarm code for a clear DI = 0 alarm
CDI1 - The alarm code for a clear DI = 1 alarm

The module's alarm generation is performed by an alarm sub-module which sets its 'Alarm State' parameter to '1' when the module's 'State' parameter has been different to the required state for longer than the alarm delay. For details of alarm generation - see **"Item Alarm Generation" on page 77.**

10.1.2.1 Digital Input BACnet Alarm State Transitions

The digital input alarm (DI = 0 or DI = 1) is considered to be an OffNormal state and there are no fault states, so the only transitions between states are To Normal and To OffNormal.

The transition from no alarms present into the digital input alarm state will generate a 'From Normal to OffNormal' transition. The clear of the digital input alarm would then generate a 'From OffNormal to Normal' transition. The diagram shows possible Digital Input module alarm state transitions with the downward paths being alarms, and the upward paths being clear alarms.



The BACnet alarm state transitions are explained in **"Alarm State Transitions" on page 65**.

Each digital input module has a BACnet Event Time Stamps array that contains three values which are the times (date and time) that the last transitions into the BACnet alarm states occurred:

n1: Time into OffNormal
n2: Time into Fault (always 0)
n3: Time into Normal

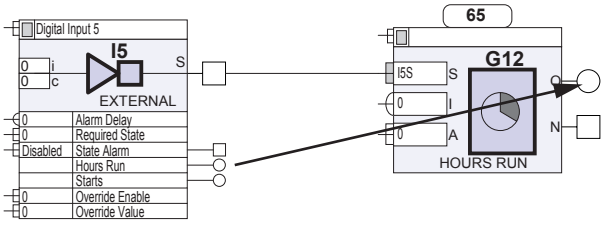
10.1.3 Module Override

The module's input can be overridden to a value specified by the 'Override Status' parameter by setting the 'Override Output' parameter to '1'. For details on how module overrides work see **"Module Overrides" on page 31**.

Note: The Starts Count and Hours Count functions use the module's direct input as source so number of starts and hours run values are not affected by the Override function.

10.1.4 Use with Type 6 (Hours Run) Logic Module

If a Digital Input module is connected to a Type 6 (Hours Run) Logic module as shown below, the Digital Input module's 'Hours Run' parameter's value is transferred directly to the Type 6 (Hours Run) Logic module's 'Hours' parameter without scaling when the Type 6 (Hours Run) Logic module is serviced.



If the Type 6 (Hours Run) Logic module is reset this will reset both its 'Hours' parameter and the Digital Input module's 'Hours Run' parameter. Similarly if the Digital Input module's 'Hours Run' parameter is changed, this will be transferred to the Type 6 (Hours Run) Logic module's 'Hours' parameter.

This special relationship between the two modules exists mainly for compatibility with IQ2 controllers, although it also enables an hours run interval exceeded output to be generated.

10.1.5 Module Disable

If required the module can be disabled - see **"Disabling Modules" on page 29**.

10.1.6 COV Operation

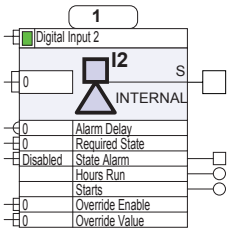
COV operation enables Digital Input modules to send its value to a list of subscribers using BACnet protocol when the value changes.

10.2 Digital Input Module Types

There are two type of Digital Input modules specified by the 'Type' parameter:

Module Type	*Description*
Type 1 (Internal)	Type 1 (Internal) Digital Input modules take the status of a digital value in the IQ4, and pass it to the strategy.
Type 2 (External)	Type 2 (External) Digital Input modules take the input from a physical external input on the IQ4 or attached I/O module and pass it to the strategy via its 'State' parameter.

10.2.1 Type 1 (Internal)

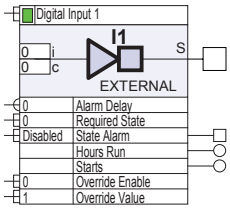


Type 1 (Internal) Digital Input modules function in the way described in **"Functionality" on page 195** enabling the status of a digital value in the IQ4 to be passed to the strategy via the 'State' parameter which can be used as an input to other modules. They appear on the strategy page and can be either sequence table driven, or event driven.

10.2.1.1 Type 1 (Internal) Digital Input Module Parameters

Type 1 (Internal) Digital Input modules include the parameters common to all digital input modules - see **"Digital Input Module Common Parameters" on page 199**, plus the 'Sequence Step' parameter that is common to most modules - see **"IQ4 Modules" on page 165**.

10.2.2 Type 2 (External)



Type 2 (External) Digital Input modules function in the way described in **"Functionality" on page 195** enabling external digital inputs to be passed from a physical external input to the strategy via its 'State' parameter which can be used as an input to other modules. They have additional parameters to define the external channel (I/O module, channel number) used. They appear on the strategy page and are automatically event driven and cannot be allocated a sequence step.

10.2.2.1 Type 2 (External) Digital Input Module Parameters

Type 2 (External) Digital Input modules include the parameters common to all digital input modules - see **"Digital Input Module Common Parameters" on page 199**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Hardware Channel	The hardware channel which the digital input is monitoring. The range of channel numbers available depends on the type of IQ4 used.	0 to 32 0 = Disabled	1	N	c
Hardware Module	The address of the physical I/O module that contains the input channel providing the module's input - only applies to IQ4s that support external I/O modules, set to '0' for all other types of IQ4.	0 to 30 0 = IQ4 1 to 30 = I/O modules	0	N	i

10.3 Digital Input Module Common Parameters

Digital Input modules include the following parameters that are common to most modules: 'Disable Module', 'Label', 'Page', 'Sequence Table' (not type 2), Sequence Step (not type 2), 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
ACK Transactions	v3.50 or greater. The status of the modules user alarm acknowledgement - see **"BACnet User Alarm Acknowledgments" on page 67**.	000 to 111	111	O	U
Alarm Delay (secs)	The length of time in seconds that the value must be in the wrong state before a digital input alarm is generated.	0 to 172800	0	I	d1(D)
Alarm Group	The Alarm Group module to which the alarm is sent.	0 to 500	1	N	d1(G)
Alarm Sent	Indicates if the alarm has been sent to the alarm group. *Note: The alarm will be sent if the alarm state has been present for longer than the alarm delay, and the alarm is enabled.*	0 or 1 0 = Not sent 1 = Sent	0	O	d1(K)
Alarm State	The state of the module's alarm sub-module (the same as the 'Digin' parameter in the parent module). See **"Alarms" on page 195** for details of how this parameter is set.	0 or 1 1 = Alarm 0 = No alarm	0	O	d1(S)
BACnet Event Time Stamps	BACnet only. An array of three parameters which hold the times of the last transitions into the BACnet alarm states (n1 = To OffNormal, n2 = To Fault, and n3 = To Normal) in date and time format. This parameter is only available when accessing the IQ4 using BACnet protocols it is not available when using Trend communication protocols.	date&time; yyyy -mm-dd &Thh:mm:ss	0000-00- 00 T00:00:00	O	n1 to n3
Clear Alarm Delay (secs)	v3.50 or greater. The length of time in seconds that digital input alarm must be cleared before a clear alarm is generated.	0 to 172800	0	I	d1(d)
Clear Alarm	Enables/Disables the reporting of the clear alarm.	0 or 1 1 = Enabled 0 = Disabled	1	N	d1(C)
Digin	The state of the module's alarm sub-module. See **"Alarms" on page 195**. Clicking on **Digin** on the web page will display the digital input's alarm sub-module.	0 or 1 1 = Alarm 0 = No alarm	0	O	-
Enabled	Enables/Disables the reporting of the alarm.	0 or 1 0 = Disabled 1 = Enabled	0	I	d1(E)
Hours Run	The number of hours for which the 'Source' parameter has been set to '1'.	0 to 87840 hours	0.00	O	H
In Alarm	The module's current alarm state. See **"Alarms" on page 195** for details of how this parameter is set.	0 or 1 0 = No alarm 1 = Alarm	0	O	!
Last number of Starts	Used internally in the software when restoring the number of starts after a software restart.	0	0	N	l
Number of Starts	The number of times the input has transitioned from OFF to ON.	0 to 10^9	0	O	N
Override Output	Overrides the module's output to the status set in the 'Override Status' parameter. For details on how module overrides work - see **"Module Override" on page 196**.	0 or 1 1 = Override 0 = No override	0	I	^

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Override Status	The status to which the module is overridden when the 'Override Output' parameter is set to '1'.	0 or 1	0	I	v
Required State	The required (non-alarm) state of the input.	0 or 1	0	I	R
Source	The state of the input to the module. For an internal digital input it defines the source of the input. For an external digital input it is read only.	0 or 1	0	I	I
State	The state of the module's output. Normally connected to the input, but if the 'Override' parameter is set to '1' it is connected to the 'Override Status'.	0 or 1	0	O	S
Time in Alarm (secs)	The elapsed time since the alarm state commenced.	+ Full analogue	0	O	d1(T)
Time of Hours Run Reset	The time that the cumulative hours run was last reset to '0'.	date&time; yyyy - mm-dd & Thh:mm:ss	0000-00-00 T00:00:00	O	t
Type	Type of digital input (external or internal). It is set up when the module is created.	0 or 1 1 = Internal 0 = External	-	N	Y

10.4 Digital Input Module Connectives

The following parameters are available for connection to other modules in the strategy:

Hours Run
Number of Starts
Alarm State
State

11 DIRECTORY MODULES

Module Identifier	@n (where n is module number)
Default overview parameters	$
Size in brIQs	13

11.1 Functionality

Directory modules, together with Display modules, enable views to be configured in the IQ4 that allow data to be viewed by web clients, supervisors, tools, or displays without having to set them up separately in every device. For details of how Display and Directory modules are used together see **"Display and Directory Modules" on page 31.** Directory modules do not appear on the strategy page and do not need to be sequenced.

11.2 Directory Module Parameters

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Back Colour	The background colour for the page when displayed on a GraphIQ page. Specified using standard HTML colour format using 2 hex characters in the range 0 to 255 reflecting the colour intensity.	#RRGGBB	Blank	N	B
Backdrop	The filename of the file (.JPG or .GIF) that is to be used as the background for the page when displayed on a GraphIQ page. The filename cannot be longer than 34 characters. The use of backdrop files larger than 100k is not recommended. IQSET allows a maximum backdrop size of 200 kbytes, and a maximum capacity for all backdrops of 2 Mbytes. It will warn if a backdrop exceeds 100 kbytes. They should be sized to fit the screen being used to display the backdrop (see 'Width', 'Height' parameters).	Filename	Blank	N	D
Height	The height of the page in pixels.	240 to 1200	768	N	H
Label	Label for the module. This is the title of the page when displayed on a GraphIQ page. *Note: If engineering Directory modules that are to be navigated using an IQView then it is recommended that the length of the label does not exceed 12-characters. SDU will truncate this to the first 20 characters.*	0 to 30 character string.	Null	N	$
Parent	The number of the Directory module that is above the module in the hierarchy. *Note: The root is always Directory module 1, so a parent of '1' means that this module is in the next level below the root. Directory module 1 should itself have its parent set to '1' as '0' signifies that the directory module is not set up.*	0 to 500 1 = Directory 1 0 = Not configured	0	N	R
Refresh Rate	The interval (in seconds) at which all values on a GraphIQ page are updated.	0 to 60	5	N	E
User Level	The user level required for the user to be able to view the directory.	0 to 99 0 = No PIN required	0	N	P
Width	The width of the page in pixels.	320 to 1600	1024	N	W

11.3 Directory Module Connectives

None

12 DISPLAY MODULES

Module Identifier	~n (where n is module number)
Default overview parameters	None
Size in brIQs	19

12.1 Functionality

Display modules, together with Directory modules, enable views to be configured in the IQ4 that allow data to be viewed by web clients, supervisors, tools, or displays without having to set them up separately in every device. For details of how Display and Directory modules are used together see **"Display and Directory Modules" on page 31**. Display modules do not appear on the strategy page and do not need to be sequenced. There are two types of Display module - **"Display Module Types" on page 203**.

12.2 Display Module Types

There are two Display module types specified using the 'Display Subtype' parameter:

Module Type	*Description*
Static Text	Used by GraphIQs to display static text.
Dynamic Text	Used to display a value from the IQ4.

12.2.1 Static Text

Static Text Display modules are used specify static text displayed on GraphIQs.

12.2.1.1 Hyperlinks

The text will become a hyperlink if the 'URL' parameter is not blank. The hyperlink can be used to go to a page of information held in the IQ4, a URL that is not the IQ4, or an email address (loads mail client with a new email with email address set up).

12.2.2 Static Text Display Module Parameters

Static Text Display modules include the parameters common to all Display modules - see **"Display Module Common Parameters" on page 205**, plus the following parameters:

Parameter	*Comments*	*Range*	*Default*	*Type*	Text Comms
Label	Label for the module. Used as the text on a GraphIQ page.	0 to 30 character string.	Null	N	$
URL	The URL accessed if the user clicks on the module on a GraphIQ page - see **"URL" on page 204**.	Valid link	Blank	N	U

URL

<table>
<tr><th>Link to</th><th>URL format</th><th>Description</th></tr>
<tr><td>Page in the IQ4</td><td>trendModule:<Page>.htm

E.g. trendmodule:S3.htm</td><td><Page> is the page to be displayed. The format of <Page> depends on the destination.

<table>
<tr><th>Destination</th><th><Page></th></tr>
<tr><td>Welcome page</td><td>Index.htm</td></tr>
<tr><td>Alarm page</td><td>Alarms.htm</td></tr>
<tr><td>Module page</td><td>Modules.htm</td></tr>
<tr><td>Diagnostic page</td><td>^.htm</td></tr>
<tr><td>GraphIQ page</td><td>GraphIQs\<Directory Module Structure>.htm

<Directory Module Structure. The Directory module structure of the Display module that defines the page.

E.g. GraphIQs/Boilers/Boiler1/plant.htm</td></tr>
<tr><td>Modules List Page <Page></td><td><Module Type>.htm

<Module Type> Type of module using its module identifier. E.g. S.htm sensor module.</td></tr>
<tr><td>Modules Details Page</td><td><Module Type><Module Number>.htm

<Module Type> Module type using its module identifier. <Module Number> Number of the module. E.g. S3 for Sensor module 3.</td></tr>
</table>
</td></tr>
<tr><td>email address</td><td>mailto:<email address></td><td><email address> is the email address to which the email is to be sent.

E.g techpubs@trendcontrols.com</td></tr>
<tr><td>URL</td><td><URL></td><td><URL> is the required URL

The URL can link to a page in the IQ4, or controller by using the page's URL. If linking to another controller that has security set up add ? autoLogin = TRUE to the end of the URL. This will pass the current login information to the controller. It is compared with the destination controller's users and if it matches access is granted.

E.g. http://123.456.78.123/GraphIQs/index.htm ? autoLogin = TRUE</td></tr>
</table>

12.2.3 Dynamic Text

Dynamic Text Display modules are used to display a value from the IQ4.

12.2.4 Dynamic Text Display Module Parameters

Dynamic Text Display modules include the parameters common to all Display modules - see **"Display Module Common Parameters" on page 205**, plus the following parameters:

Parameter	*Comments*	*Range*	*Default*	*Type*	*Text Comms*
Clickable	Specifies whether the value is a link to the detail page for the parameter that is being displayed.	0 or 1 1 = Link 0 = No link	1	N	N
In Alarm Colour	The colour of the text if the parameter from the IQ4 being displayed is in an alarm condition. Specified using standard HTML colour format using 2 hex characters in the range 0 to 255 reflecting the colour intensity.	#RRGGBB*	Fore colour	N	A

Parameter	*Comments*	*Range*	*Default*	*Type*	*Text Comms*
Item	The module parameter that is to be displayed in the form: <Module><Module Parameter> <Module> specifies the type of module using its module identifier and module number (e.g., S1 for sensor 1). *Note: Only Sensor, Digital Input, Knob, Switch, Driver, and Time Schedule modules should be used. Other modules can be used. However, they are only for use on GraphIQs; use in supervisors and displays may cause problems.* <Module Parameter> specifies the parameter within the module that is to be displayed (e.g. sensor value). The module parameter is specified using the text comms identifier, and can be any parameter available within the specified module - see **"IQ4 Modules" on page 165**. If <Module Parameter> is not specified the default overview parameters for the module type are used. E.g. S1 would display the sensor label, value, and units. The default overview parameters for each module are described in the module section of this manual. Although multiple parameters are used as the default overview parameters it is only possible to specify single parameters.	Valid module parameter.	Blank	N	I

12.3 Display Module Common Parameters

Parameter	*Comments*	*Range*	*Default*	*Type*	*Text Comms*
Align	The alignment of the text on a GraphIQ page. Can only be set up if the 'Width' parameter is set to a value other than '0'.	0 to 2 0 = Left 1 = Centre 2 = Right	1	N	G
Back Colour	The background colour for the text when it is displayed on a GraphIQ page. Specified using standard HTML colour format using 2 hex characters in the range 0 to 255 reflecting the colour intensity.	#RRGGBB*	#000000	N	B
Display Subtype	Specifies the module type.	1 or 2 1 = Static Text 2 = Dynamic Text	-	N	T
Font Name	The font used to display the text on a GraphIQ page.	0, 1, 2, 3, 4 0 = Serif 1 = Sans-serif 2 = Cursive 3 = Fantasy 4 = Monospace	0	N	N
Font Size	The size of the font (in points) used to display the text on a GraphIQ page.	6 to 72	12	N	S
Fore Colour	The colour the text is displayed in on a GraphIQ page. Specified using standard HTML colour format using 2 hex characters in the range 0 to 255 reflecting the colour intensity.	#RRGGBB*	#000000	N	F

Parameter	*Comments*	*Range*	*Default*	*Type*	*Text Comms*
Left Position	The horizontal position of the top left corner of the value when it is displayed on a GraphIQ page.	0 to 1600	0	N	X
Parent	The Directory module to which this item applies.	0 to 500	0	N	R
Top Position	The vertical position of the top left corner of the value when it is displayed on a GraphIQ page.	0 to 1200	0	N	Y
User Level	The user level at which the user must be logged on before the display item becomes visible.	0 to 99 0 = No PIN required	0	N	P
Width	The width of the text box in which the text is displayed.	0 to 1600 0 = automatic sizing	0	N	W

12.4 Display Module Connectives

None

13 DRIVER MODULES

Module Identifier	Dn (where n is module number)
Default overview parameters	$, 1
Size in brIQs	52

13.1 Functionality

Driver modules are used to drive the HVAC equipment through the IQ4's output channels. They can be sourced from Loop modules to provide a closed loop system or have fixed input levels to provide load-cycling or timed switch-ON or switch-OFF. They can be used with or without feedback from HVAC equipment. If feedback from the equipment is used, alarms can be raised if the HVAC equipment does not respond within a defined time period (Grace-time). Sequence control of HVAC equipment connected to multiple outputs is supported by the Type 8 (Multistage Digital).

There are eight types of Driver module - see **"Driver Module Types" on page 212**. Driver types 1 to 7 have two output channels, type 8 drivers can have up to 8 output channels each of which may be assigned to the hardware outputs of the IQ4 or attached I/O modules.

Driver modules appear on the strategy page and can be either sequence table driven, or event driven.

The module's algorithm drives the in-phase and anti-phase outputs, its operation varies with module type. It is normally connected to the priority array, which has 16 slots, and receives the value from the highest priority slot.

When a source writes a value into a slot it becomes active, and it can make it inactive again by relinquishing its slot (setting it to null). The strategy (source input) writes its value to slot 10, which is normally used by the driver's algorithm. However, if a BACnet write to a higher level in the priority array occurs, its value will be used instead.

Note: A BACnet write to the Driver module's priority array (Present Value) includes a priority level in the write message which defines the array slot to be used; if the message does not contain a priority level, level 16 is assumed. If a Driver module has an unconnected source, then slot 10 will be null, and BACnet slots above 10 may take control. The Override Output works on the Driver module further down in the process, but if it is operating its value is put into the array at slot 4 so that a BACnet read of the array will indicate it is operational. The array slot currently affecting the Driver module is given by the 'Active Priority Level' parameter.

The 'Active' Priority Level ' parameter is set to '0' if the source is unconnected and there is no other active input to the Driver module. IC comms writes are put into slot 9.

The 'Value' parameter monitors the input to the Driver module's algorithm. If the input is outside the range 0 to 100 it will be clipped to be within that range by the algorithm. Exceptions to this are the Type 8 (Multistage Digital) Driver module whose range depends on the switching type, and the Type 1 (Digital) Driver module whose value is either 0 or 1, and the Type 4 (Raise Lower with End Stop) Driver module which has a special negative input function.

13.1.1 Alarms

Driver modules can report three alarms:

Readback Alarm
Maintenance Interval Alarm
Hand/OFF/AUTO Alarms (v3.70 or greater)

These alarms are unusual in that the alarm states are not detected by the Driver module they must be generated by strategy that passes the alarm condition to the Driver module.

13.1.1.1 Readback Alarm

The module's Readback alarm (SDGT) generation is performed by an alarm sub-module which sets its 'Alarm State' parameter to '1' when the module's 'Readback Alarm Input' parameter is '1' for longer than the alarm delay. When the Driver module's 'Readback Input' parameter returns to '0' for longer than the clear alarm delay a clear readback alarm (CDGT) is generated and the Driver module's 'Readback Alarm State' parameter set to '0'.

For details of alarm generation - see **"Item Alarm Reporting" on page 77.**

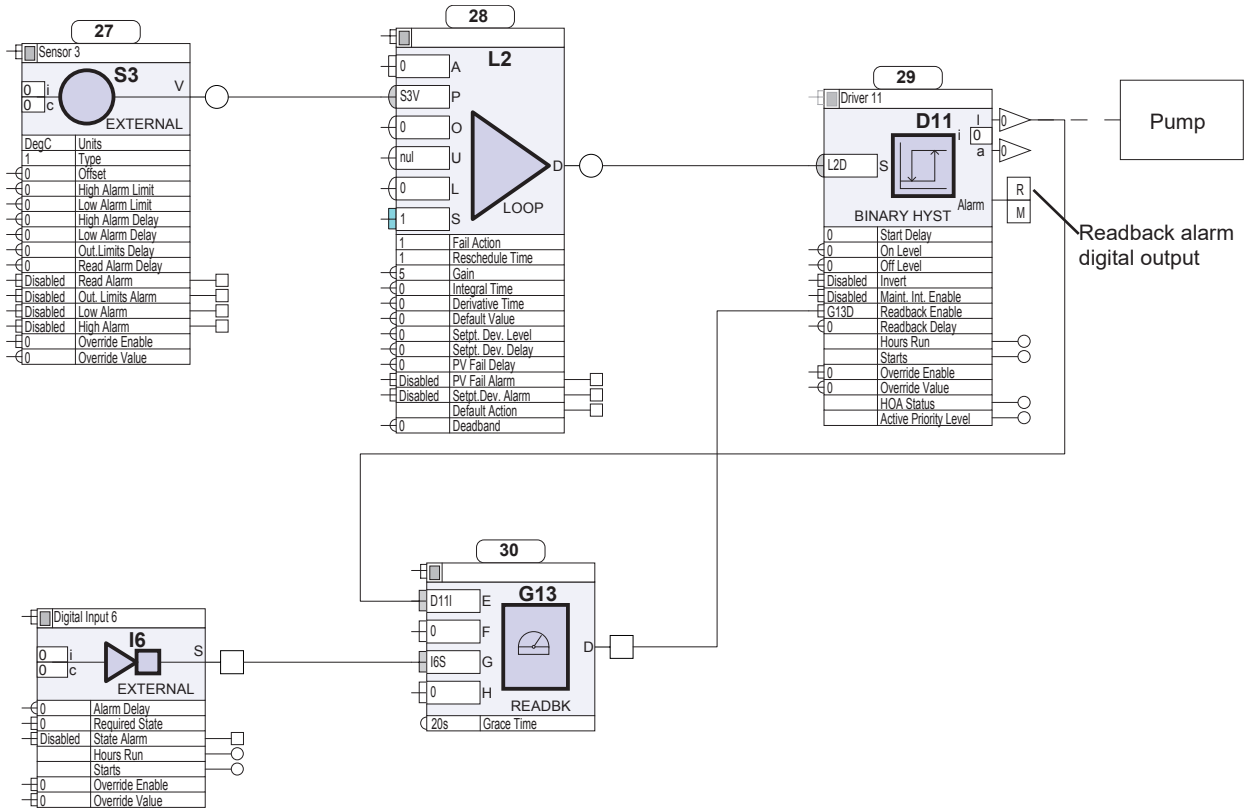
Driver modules do not detect the alarm state, in order for it to generate the alarm it is necessary to write strategy that produces a digital status of '1' when the alarm state occurs. This digital status must be connected to the Driver module's 'Readback Input' parameter.

Note: The Driver module's readback alarm delay is not necessary as its function can be achieved by using the strategy that generates the initial alarm state (e.g. adjusting the Type 3 (Readback) Logic module's grace time) so it should be left at '0'.

The driver's readback alarm output can be used by other modules to perform functions associated with the alarm condition, for example, to start a standby pump.

Example

Readback checking can be provided using the Type 3 (Readback) Logic module.



In the above, example, the output from L2 is driving D11, whose output is used to drive a pump ON or OFF through digital output 11. I6 accepts input from a flow switch. When the pump is in the OFF state, the flow switch should also be in the OFF state and with the pump in the ON state, the flow switch should be in the ON state. The Type 3 (Readback) Logic module (G13) compares the two signals at its inputs E and G; when they are same (i.e. both ON or both OFF) its output D is set to '0'. However, if, the pump is ON and the flow switch is OFF, or vice versa, there is a fault in the system and after the grace time (20s) has elapsed, G13's output D will go be set to '1'. G13's output D is connected to the D11's 'Readback Alarm Input' parameter.

13.1.1.2 Maintenance Interval Alarm

The module's Maintenance Interval alarm (MINT) alarm generation is performed by an alarm sub-module which sets its Maintenance Interval 'Alarm State' parameter to '1' when the module's 'Maintenance Alarm Input' parameter is '1' for longer than the alarm delay. When the Driver module's 'Maintenance Alarm Input' parameter returns to '0' for longer than the clear alarm delay a clear alarm (CMNT) is generated and the Driver module's 'Maintenance Alarm State' parameter set to '0'.

For details of alarm generation - see **"Item Alarm Reporting" on page 77..**

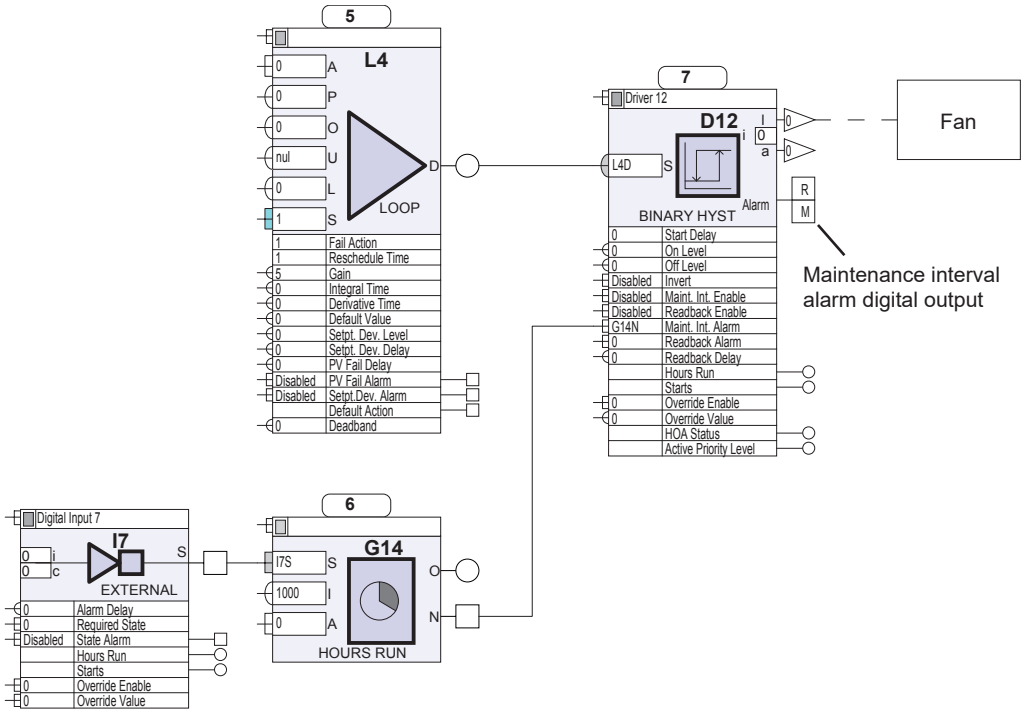
Driver modules do not detect the alarm state, in order for it to generate the alarm it is necessary to write strategy that produces a digital status of '1' when the alarm state occurs. This digital status must be connected to the Driver module's 'Maintenance Input' parameter.

Note: The Driver module's readback alarm delay is not necessary as its function can be achieved by using the strategy that generates the initial alarm state e.g. adjusting the Type 6 (Hours Run) Logic module's interval time, so it should be left at '0'.

The driver's maintenance alarm output can be used by other modules to perform functions associated with the alarm condition.

Example

Maintenance interval checking can be provided by using a Type 6 (Hours Run) Logic module.



In the above, example, the output from L4 is driving D12, whose output is used to drive a fan ON or OFF through digital output 12. I7 accepts input from a secondary contact on the fan contactor. When the number of hours the fan has been running exceeds the Type 6 (Hours Run) Logic module's 'Interval' parameter's value, the module's 'Interval Limit' parameter will be set to '1'. The Type 6 (Hours Run) Logic module output is connected to the Driver module's 'Maintenance Interval Alarm Input' parameter.

13.1.1.3 Hand/OFF/AUTO Alarms (v3.70 or greater)

Hand/OFF/AUTO (HOA) alarms are available on Type 1 (Digital), Type 2 (Analogue), Type 5 (Binary Hysteresis) and Type 8 (Multistage Digital) Driver modules.

The module's Hand/OFF/AUTO (HOA) alarm generation is performed by an alarm sub-module which sets its 'Alarm State' parameter to '1' when the module's 'HOA Input' parameter is '1' for longer than the alarm delay. When the Driver module's 'HOA Input' parameter returns to '0' for longer than the clear alarm delay a clear HOA alarm (CHOA) is generated and the Driver module's 'HOA Alarm State' parameter set to '0'.

For details of alarm generation - see **"Item Alarm Reporting" on page 77.**

13.1.2 Module Disable

If required Driver modules can be disabled using the 'Disable Module' parameter - see **"Disabling Modules" on page 29**. will stop the Driver module being serviced. The outputs will remain at their last calculated value or status before the driver was disabled. In the case of time proportional or raise/lower drivers they will continue to drive for their calculated time and then both channels will turn OFF.

13.1.3 Module Override

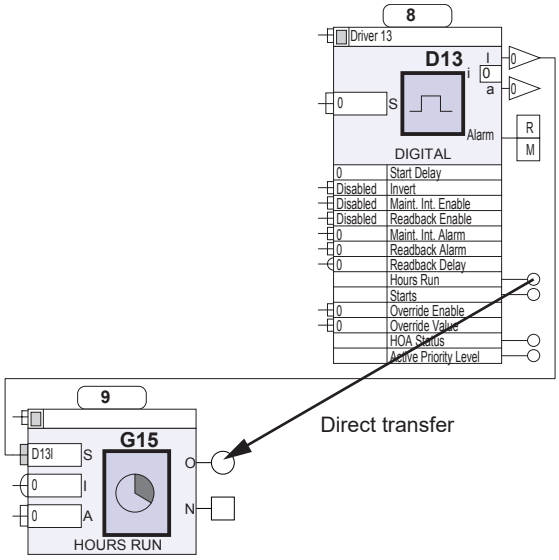
If required Driver modules can be overridden to the value specified by the 'Override Value'/'Override Status' parameter, and the value/status will be put into priority array slot 4 by setting the 'Override Output' parameter to '1' - see **"Module Overrides" on page 31** for more details.

13.1.4 Hours Run

The 'Hours Run' parameter measures the total time for which the in-phase output (uninverted) has been ON, this takes module override into account. It is stored and updated each time the Driver module's output is updated.

A special relationship exists between the Driver module and a Type 6 (Hours Run) Logic module connected to its output. This is provided mainly for compatibility with earlier controllers (IQ1, IQ2); it also provides a method of resetting the hours run.

If an Type 6 (Hours Run) Logic module is connected directly to a Driver module's output channel as shown (not type 2, 4 and 7 drivers), the Driver module's Hours Run is transferred directly to the 'Hours Run' parameter without scaling when the module is serviced. If the Type 6 (Hours Run) Logic module is reset, this will reset both the hours run module hours output, and the Driver modules Hours Run. Similarly if the Driver modules Hours Run is changed this will be copied to the Type 6 (Hours Run) Logic module's Hours output.



13.1.5 Driver Invert

The 'Invert' parameter causes a logical inversion between the driver algorithm outputs and the output channels; the actual effect on the outputs varies with driver type.

13.1.6 Hand Off Auto (HOA) Operation

When connected to an output channel that supports HOA operation the Driver modules's output can be set to the following statuses using a physical switch on the output channel (Trend I/O modules only).

HOA State	*Description*
AUTO	Driver module output controlled by strategy
Hand	Driver module output set to ON.
OFF	Driver module output set to OFF.

Note: Operation of the HOA module does not provide a safe environment for working on electrical equipment, and hence must not be used as a substitute for appropriate lockout-tagout procedures during maintenance operations.

13.1.6.1 Operation of 'HOA State' Parameter

The 'HOA State' parameter indicates the Hand/Off Auto status of the module's output.

Note: If the 'HOA State' parameter is connected to a Type 18 (Analogue to Digital Converter) Function module, the bit 0 output will indicate a manual state (non-Auto).

13.1.6.1.1 Driver Using In Phase Channel Only

For a Driver module only using the in phase channel the 'HOA State' parameter indicates the state of the HOA switch connected to the in phase channel.

13.1.6.1.2 Driver Using Anti-phase Channel Only

For a Driver module only using the anti-phase channel the 'HOA State' parameter indicates the state of the HOA switch connected to the anti-phase channel.

13.1.6.1.3 Driver Using both In Phase and Anti-phase Channels

For a Driver module using both in phase and anti-phase channels the 'HOA State' parameter's value is determined as below:

In phase	*Anti-phase*	*'HOA State' Parameter value*
Auto	Auto	Auto
Auto	Hand	Hand
Hand	Auto	Hand
Hand	Hand	Hand
Hand	OFF	Hand
OFF	Hand	Hand
Auto	OFF	OFF
OFF	Auto	OFF
OFF	OFF	OFF

13.1.6.1.4 Drivers with Multiple Outputs

For a Type 8 (Multistage Digital) Driver module the way the 'HOA State' parameter's value is determined depends on the module's Switching Type mode.

Cascade and Binary Mode

If any of the Driver's outputs are not in Auto, the 'HOA State' parameter is set to Hand.

Sequenced mode

If any of the Driver's outputs are not in Auto, the 'HOA State' parameter is set to Hand. All other outputs that are in Auto are set to OFF by the Driver module as soon as the Hand condition is detected.

13.1.6.1.5 I/O Error Condition

If there is an error condition on the I/O channel 'HOA State' parameter is set to '3', 'Error' and the channel assumed to be in Auto.

13.1.6.2 Power up

On power up each channel will be set to the value specified by the position of the HOA module's rotary switch.

13.1.6.3 Hand/OFF/AUTO Alarms (v3.70 or greater)

Hand/OFF/AUTO alarms are available on v3.70 or greater - see **"Hand/OFF/AUTO Alarms (v3.70 or greater)" on page 209**. HOA alarms enable notification when a driver's status is changed using an IQ4 HOA I/O module.

Note: Hand/OFF/AUTO alarms are **NOT** *available on all Driver module types - see* **"Hand/OFF/AUTO Alarms (v3.70 or greater)" on page 209**.

13.1.7 COV Operation

COV operation enables Driver modules to send a value to a list of subscribers using BACnet protocol when the value is changed by more than the value specified by the 'COV Increment' parameter.

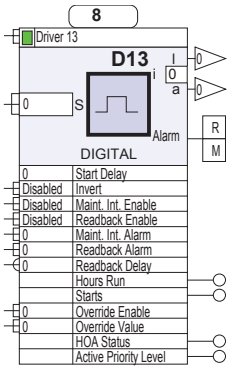
Note: Type 1 (Digital) Driver modules do not have the 'COV Increment' parameter and the value will be sent whenever the module's value changes.

13.2 Driver Module Types

There are several Driver module types specified using the 'Type' parameter:

Module Type	***Description***
Type 1 (Digital)	Provides a digital output.
Type 2 (Analogue)	Provides an analogue output.
Type 3 (Time Proportional)	Provides a digital output.
Type 4 (Raise Lower with End Stop)	Provides either analogue output requiring external relays or 2 digital outputs.
Type 5 (Binary Hysteresis)	Provides a digital output from an analogue input.
Type 6 (Time Proportional + OR)	Provides an analogue output requiring external relays.
Type 7 (Raise Lower Continuous)	Provides an analogue output requiring external relays or 2 digital outputs.
Type 8 (Multistage Digital)	Provides from 1 to 8 digital outputs from an analogue input.

13.2.1 Type 1 (Digital)



Type 1 (Digital) Driver modules provide an ON/OFF type output controlled by the state of a digital bit. They have a single digital source and provides a single digital output (with an optional anti-phase output).

They transfer the digital status of the 'Source' parameter directly to the output. The anti-phase output is the inverse of this (e.g. if the S is '1', the in-phase output is '1' and the anti-phase output is '0').

The actual physical digital output is achieved by using an analogue voltage output channel and an external relay module or, in the case of IQ4E relays on an I/O module.

13.2.1.1 Alarms

Supported - see **"Alarms" on page 207**.

13.2.1.2 Module Disable

Supported - see **"Module Disable" on page 209**.

13.2.1.3 Module Override

Supported - see **"Module Override" on page 210**.

13.2.1.4 Hours Run

Supported - see **"Hours Run" on page 210**.

13.2.1.5 Driver Invert

Supported. This can be used if the plant is driven via normally closed relay contacts - see **"Driver Invert" on page 210**.

13.2.1.6 HOA Operation

Supported - see **"Hand Off Auto (HOA) Operation" on page 211**.

13.2.1.7 COV Operation

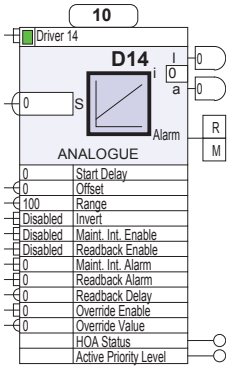
Supported - see **"COV Operation" on page 212**.

13.2.1.8 Type 1 (Digital) Driver Module Parameters

Type 1 (Digital) Driver modules include the parameters common to all driver modules - see **"Functionality" on page 207**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Time of Hours Run Reset	The time that the cumulative hours run was last reset to '0', in full date and time format.	date&time;yyyy-mm-dd&Thh:mm:ss	0 0 0 0 - 0 0 - 00T00:00:00	O	t
Time of Last Value Change	The time that the output last changed state, in full date time format	date&time; yyyy-mm-dd&Thh:mm:ss	0 0 0 0 - 0 0 - 00T00:00:00	O	c
Time of Number of Starts Reset	The time that the cumulative number of starts was last reset to '0', in full date and time format.	date&time;yyyy-mm-dd&Thh:mm:ss	0 0 0 0 - 0 0 - 00T00:00:00	O	e

13.2.2 Type 2 (Analogue)



Type 2 (Analogue) Driver modules provide an analogue output signal in the range 0 to 10 Vdc. They are normally used to drive valves or damper position actuators etc. They have a single source variable in the range 0 to 100 % which represents the full range of the analogue output. They can be used to drive one of a range of external relay modules via an analogue output channel (e.g. SRMV, 2RM, 3RM, 6RM). The strategy driving the module can specify which relays are energised within the module by setting the percentage input appropriately to match the relay's switching voltage.

13.2.2.1 Alarms

Supported - see - **"Alarms" on page 207**.

13.2.2.2 Module Disable

Supported - see **"Module Disable" on page 209**.

13.2.2.3 Module Override

Supported - see **"Module Override" on page 210**.

13.2.2.4 Hours Run

Not supported.

13.2.2.5 Driver Invert

Supported - see **"Driver Invert" on page 210**.

13.2.2.6 HOA Operation

Supported - see **"Hand Off Auto (HOA) Operation" on page 211**.

13.2.2.7 COV Operation

Supported - see **"COV Operation" on page 212**.

13.2.2.8 Offset and Range

In order to drive actuators where the linear range of the actuator corresponds to an output voltage range smaller than 0 to 10V the 'Offset' and 'Range' parameters can be adjusted to provide the required output range.

The standard output voltage range is 0 to 10 V, where 0 volts represents an input of 0 % input and 10 V represents an input of 100%. If an offset of 0 and a range of 100 is specified an input of 20%, results in an output of 2 V i.e. 10 % per volt.

For actuators where the linear range is smaller that the 0 to 10 V (e.g. 2 to 8 V) the offset and range parameters need to be calculated as follows:

Offset = Bottom of range voltage x 10 %
Range = (Top of range voltage - Bottom of range voltage) x 10 %

Note: Offset + Range < = 100

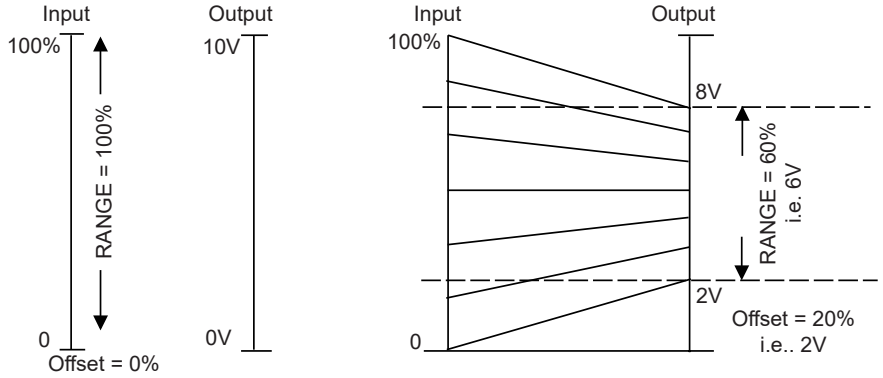
For example for an output voltage range of 2 to 8 V the offset is calculated as follows:

Offset = 2V x 10% = 20 %

The range is calculated as follows:

Range = (8V - 2V) x 10% = 60 %

The diagram below illustrates this. When the input to the module is 100%, the output voltage will be 8 V and when the input to the module is 0.01 %, the output voltage will be 2 V (0% will still give 0 V).



If the Driver module is inverted, its output level corresponds to its complement from 100 % e.g. 80 % inverted gives 20% output. If a range and offset are set up then the output is the complement of the range plus the offset. In the example above with an offset of 20 % and a range of 60 %, if the input were 80 % and the module inverted, the output is 20 % of the range (60 %) plus the offset (20 %).

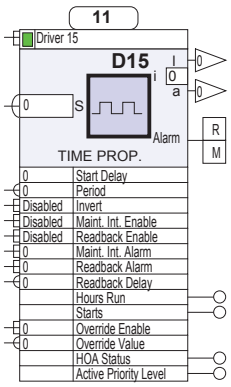
i.e. Output = (20 x 60)/100 + 20 = 32 % = 3.2 V

13.2.2.9 Type 2 (Analogue) Driver Module Parameters

Type 2 (Analogue) Driver modules include the parameters common to all Driver modules - see **"Functionality" on page 207**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Offset	The percentage of the output to which the 0% input level corresponds (i.e. 20 % offset means that 0% input corresponds to 2V output).	0 to 100	0	I	O
Range	The percentage of the output which corresponds to the 0 to 100% input range (i.e. a range of 60% means that an input range of 0 to 100 % corresponds to an output range of 6 V).	0 to 100	0	I	R

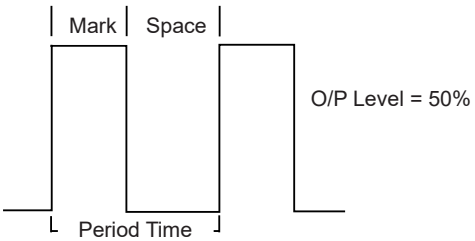
13.2.3 Type 3 (Time Proportional)



Type 3 (Time Proportional) Driver modules provide a variable mark/space ratio pulsed digital output (with an optional anti-phase output) from an analogue input.

The state of the output is determined by the value of the 'Source' Parameter, and the 'Period' parameter. If the 'Source' parameter is greater than '0', the in-phase output will be set to '1' for a length of time. This length of time is calculated as a percentage of the period time determined by the value of the 'Source' parameter (i.e. the output's mark-space ratio is dependent on the module input level).

The example below shows the module's 'In-Phase Hardware Channel' parameter where the Driver's 'Source' parameter is 50 %. The output waveform shows an ON pulse (mark) which is equal to half the period time. The period time is set in seconds.



The time proportional driver input defines the ratio ON time/PERIOD time in percent.

The period time is configured in the Driver module's 'Period' parameter, and then the ON time, or the time that the output is in the MARK state is determined by the input level taken as a percentage of the period time. The OFF time or time when the output is in the SPACE state is then the remainder of the period (or period time minus ON time).

Example: When the period time is 150 s.

- With the module input at 100 %, the output will be in the ON state for 150 s and the OFF state for 0 s.
- With the module input at 50 %, the output will be in the ON state for 75 s and the OFF state for 75 s.
- With the module input at 10 %, the output will be in the ON state for 15 s and in the OFF state for 135 s.

The actual physical digital outputs are achieved by using analogue voltage output channels and external relay modules or, in the case of IQ4E, relays on the I/O modules.

13.2.3.1 Alarms

Supported - see **"Alarms" on page 207**.

13.2.3.2 Module Disable

Supported - see **"Module Disable" on page 209**. Disabling the Driver module will allow the channels to continue to drive for their calculated time to the end of the period before turning both channels OFF.

13.2.3.3 Module Override

Supported - see **"Module Override" on page 210**.

13.2.3.4 Hours Run

Supported - see **"Hours Run" on page 210**.

13.2.3.5 Driver Invert

Supported - see **"Driver Invert" on page 210**.

13.2.3.6 HOA Operation

Supported - see **"Hand Off Auto (HOA) Operation" on page 211**.

13.2.3.7 COV Operation

Supported - see **"COV Operation" on page 212**.

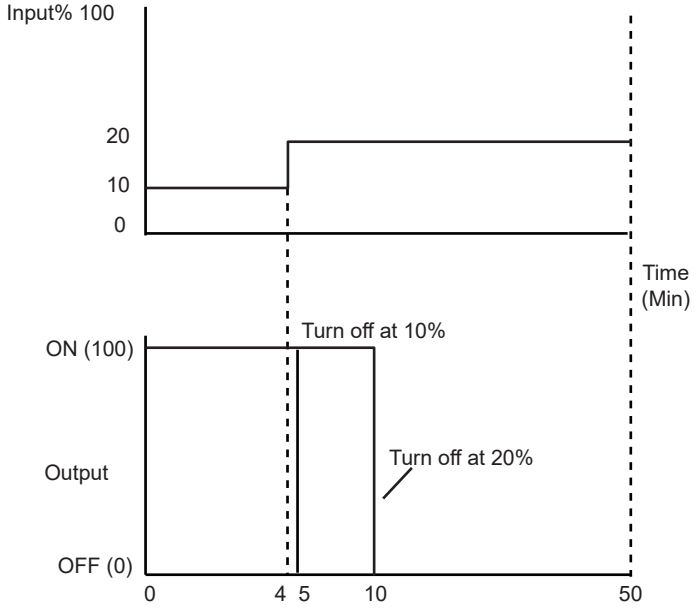
13.2.3.8 Response to Changing Inputs

When the module's input is greater than 0 % and at the start of the time period, the driver assumes the ON state. The output state is updated every 30 mS (so its turn off time is accurate to 30 mS), but it only monitors the input when the module is serviced (i.e. if sequenced, once per second, unless entered into the sequence table more than once).

As a general rule every time the module is serviced the input is examined and if the output is still ON, the time for which it will remain ON is recalculated, however once the output has turned OFF it will remain OFF until the start of the next period.

Note: There is a special exception to this action if the input is found to be 100 % - see **page 218**.

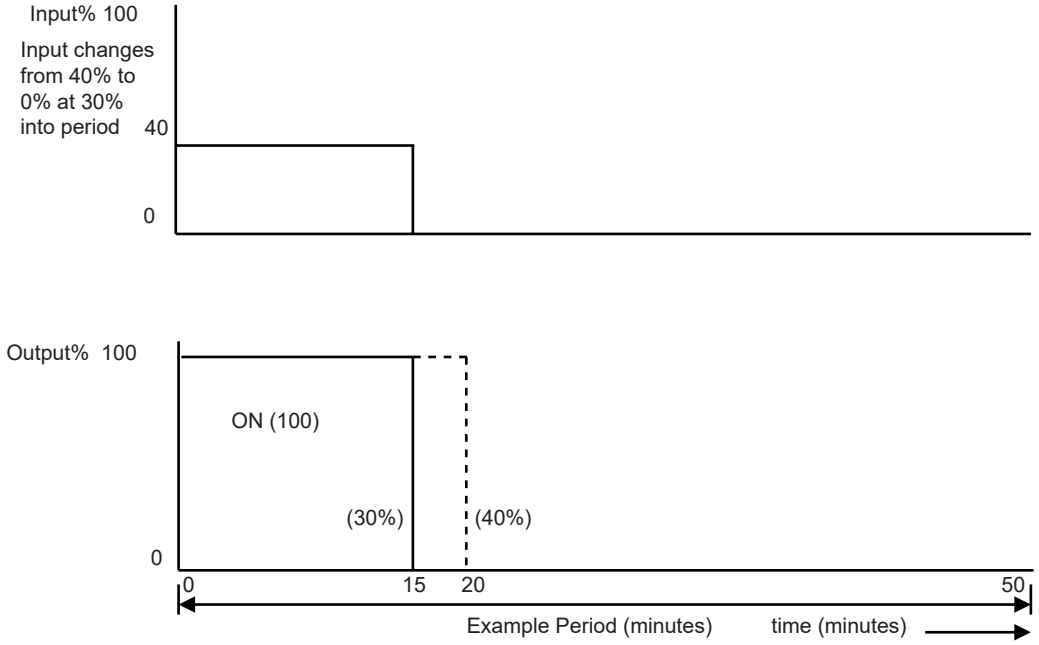
For example if the input at the start of a 50 minute period is at 10 %, then the output should remain ON for 5 minutes.



If at 4 minutes the input has changed to 20 % then the output will be set to stay ON until the 10 minutes. However if the input had changed to 5 % then the output will turn OFF immediately (as the ON period of 2½ minutes has been exceeded).

Similarly any change to 0 % will be actioned the next time the module is serviced.

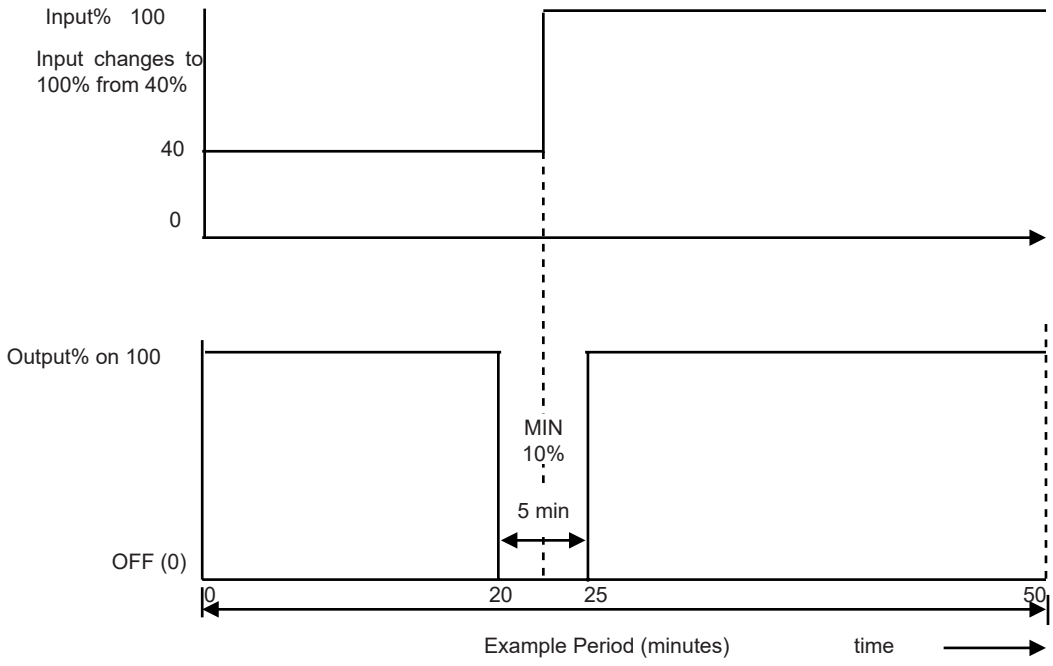
Example: change from 40 % to 0 %



In this example the output would stay ON for 20 minutes, but when the input changes to 0 % after 15 minutes, the output turns OFF the next time the module is serviced.

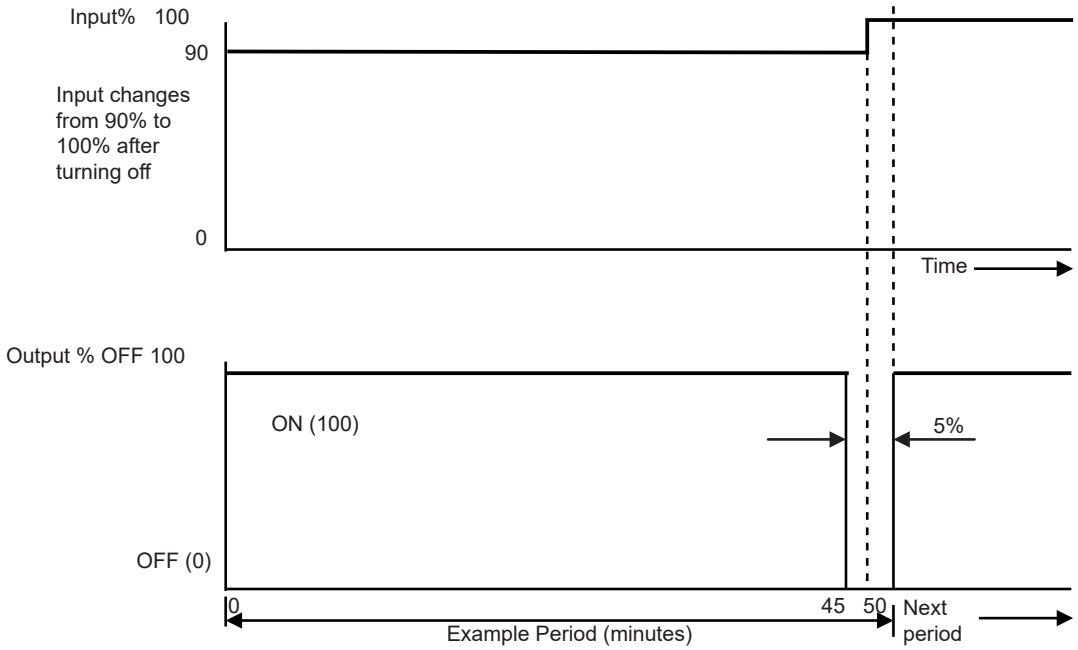
The exceptions to this action occur if the input has changed to 100 % when it is examined the next time the module is serviced. In this case if the output is ON it will stay ON, (as would be expected from the general action described above), but if the output is OFF it will again turn ON if it has been OFF for greater than the minimum off time of 10% of the period. In the following examples the period is set to 50 minutes.

E.g: Input changes from 40 % to 100 %.
In this case the input changes from 40 % to 100 % after 22½ minutes. This means it turned OFF at 20 minutes, so it will not turn ON again until it has exceeded the minimum OFF time of 5 minutes (10 % of 50 minutes). This point occurs at 25 minutes.



E.g: Input change for 90 % to 100 %

In this case the input changes from 90 % to 100 % after 47½ minutes. The output has turned OFF at 45 minutes so it cannot complete the minimum off time of 5 minutes until the start of the next period, so the output cannot turn ON again until the start of the next period.

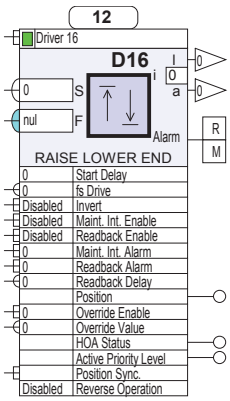


13.2.3.9 Type 3 (Time Proportional) Driver Module Parameters

Type 3 (Time Proportional) Driver modules include the parameters common to all driver modules - see **"Functionality" on page 207**, plus the following parameter:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Period	The amount of time (in seconds) being time proportioned.	0 to 32767	0	I	P

13.2.4 Type 4 (Raise Lower with End Stop)



Type 4 (Raise Lower with End Stop) Driver modules are designed to drive a split phase motor of, for example, a valve actuator. They should be used for actuators with reversible action motors without any built-in positioning or where there are no built-in limit switches, or clutches.

They provide either two digital outputs or a single analogue output. Setting the anti-phase output channel number to '0' selects a single analogue output; this enables it to drive an external raise lower or heat cool relay module using an analogue output channel. Setting the anti-phase output channel to a number other than '0' selects two digital outputs; this enables it to drive relay or triac outputs.

Feedback is optional. The module will do its best to ensure that the item it is driving has reached its end stop and then stop driving the item, as opposed to the Type 7 (Raise Lower Continuous) Driver module which will keep driving.

13.2.4.1 Alarms

Supported - see **"Alarms" on page 207**.

13.2.4.2 Module Disable

Supported - see **"Module Disable" on page 209**. Disabling the Driver module will allow the channels to continue to drive for their calculated time to the end of the period before turning both channels OFF.

13.2.4.3 Module Override

Supported - see **"Module Override" on page 210**.

13.2.4.4 Hours Run

Supported - see **"Hours Run" on page 210**.

13.2.4.5 Driver Invert

Supported - see **"Driver Invert" on page 210**. The invert is ignored on a driver with a single analogue output channel see **"Analogue Output Channel and Relay Module Operation" on page 221**. For a driver with 2 digital outputs, the driver will invert both channels, but it doesn't transpose the channels as the anti-phase channel is not necessarily the inverse of the in-phase channel. However if the driver is in the static phase, both channels are left at '0'.

13.2.4.6 HOA Operation

Supported - see **"Hand Off Auto (HOA) Operation" on page 211**.

13.2.4.7 COV Operation

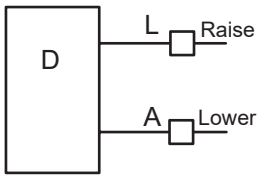
Supported - see **"COV Operation" on page 212**.

13.2.4.8 Reverse Operation

The reverse operation function will swap the raise and lower outputs. A driver with a single analogue output will perform an equivalent exchange of voltage levels. It can be used in conjunction with the invert function.

13.2.4.9 Raise Lower Driver with Two Digital Outputs

If connected to two I/O module relay (or triac) channels, these can be used to implement the raise lower function. Type 4 (Raise Lower with End Stop) Driver modules have one analogue source and provide two digital outputs. The first of the digital outputs is known as the in-phase or raise output and the second digital output is known as the anti-phase or lower output.

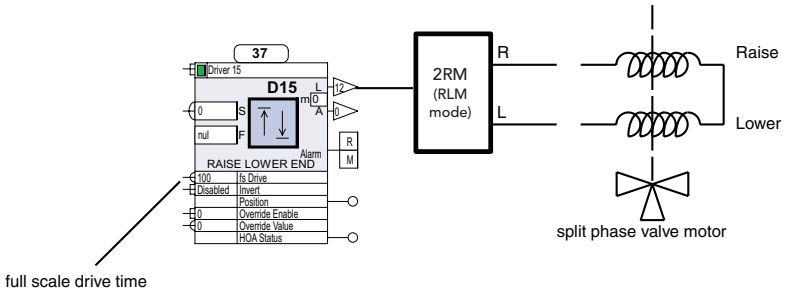


The outputs are driven as follows:

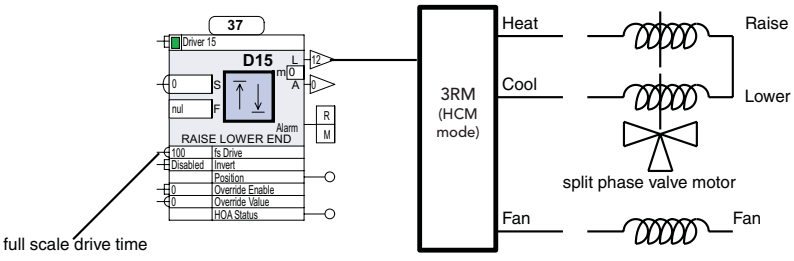
Raise: If the input = 100 % or greater, or for a time period calculated by the full scale drive time multiplied by any positive value determined by (input % minus last output position %).
Lower: If the input = 0 % or lower or for a time period calculated by the full scale drive time multiplied by any negative value determined by (input % minus last output position %).
Static: For any time period during which the requested output position is equal to the last output position.

13.2.4.10 Analogue Output Channel and Relay Module Operation

Raise Lower driver with 2RM (RLM mode)



Raise Lower Driver with 3RM (HCM mode)



When using the Type 4 (Raise Lower with End Stop) Driver module with an analogue output channel and a 2RM set to raise lower module mode (RLM) or 3RM set to heat cool module mode (HCM) to interface to HVAC equipment, the output of the Driver module as measured at the terminals of the assigned output channel (set to voltage) will be:

10 V: If the input = 100% or for a time period calculated by the full scale drive time multiplied by any positive value determined by (input % minus last output position %).
4 V: If the input = 0% or for a time period calculated by the full scale drive time multiplied by any negative value determined by (input % minus last output position %).
7 V: For any time period during which the requested output position is equal to the last output position.
0 V: As an overriding condition for any time period when the input is negative.

These voltages applied to relay modules will result in the following actions:

Voltage	*2RM (RLM mode)*			*3RM (HCM mode)*			
	Raise	*Lower*	*State*	*Fan*	*Cool*	*Heat*	*State*
0	OFF	OFF	OFF	OFF	OFF	OFF	OFF
4	OFF	ON	Lower	ON	ON	OFF	Fan + cool
7	OFF	OFF	Static	ON	OFF	OFF	Fan
10	ON	OFF	Raise	ON	OFF	ON	Fan + heat

13.2.4.11 Boundless Operation

If the 'Feedback' parameter is set to '0' then the Driver module will operate using an estimated value position.

The position of the valve is estimated using the full-scale drive time and the sum of the amounts of time that the valve has been driven in each direction. (E.g. If the full-scale drive time is 10 s and the valve has been driven open from the closed position for 8 s and then driven closed for 3 s making a total of 5 s in the open direction (8-3). The estimated value position would be 50 % open because the value has been driven open for half the full scale drive time). The valve position is verified when the value is required to be fully closed or fully open (0% or 100) by additionally driving for the full-scale drive time in the required direction. This ensures that the valve has reached the end stop.

The valve position will not be changed for input changes of less than 0.25 % unless the valve moves to or is at the 100 % or 0 % positions.

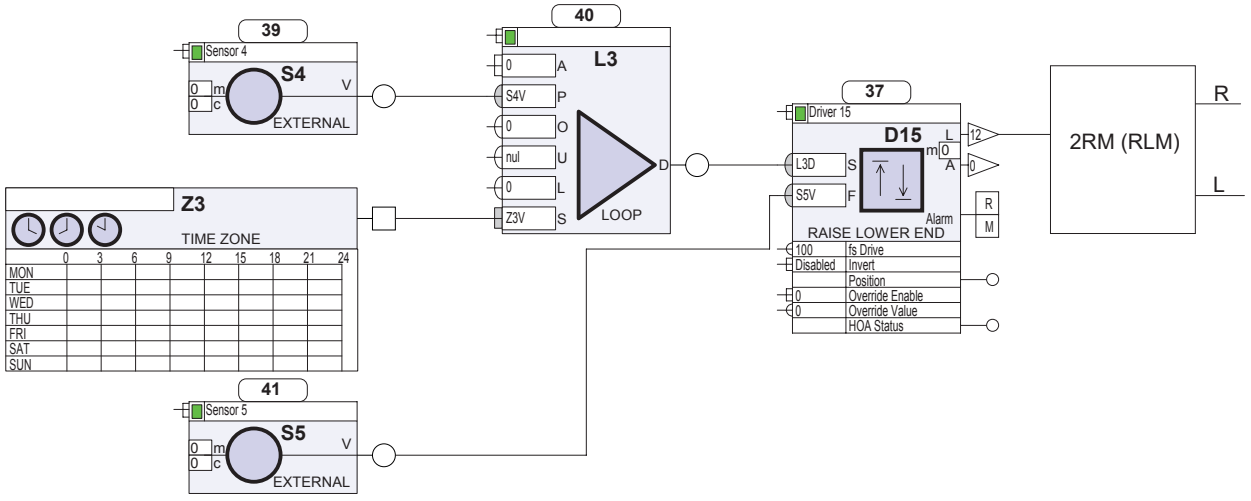
13.2.4.12 Operation with Proportional Feedback

If the valve has a position feedback potentiometer, this can be used to provide a feedback signal to the Driver module to determine the valve's actual position at any point in time. Each time the Driver module is serviced the position error is calculated. If the error is greater than 0.25 % then the appropriate output will be maintained for the time calculated to be necessary to reduce the error to 0. This is calculated as below:

Drive time = error % x full scale drive time

Note: The full scale drive time must be set up when configuring the Driver module.

If the Sensor module providing the feedback signal has either an out-of-limits or a read alarm then the Driver module will revert to boundless control. If positional information is incorrect, control may deteriorate. The feedback input is ignored if it is set to a specific value.



From the above implementation it can be seen that the strategy operates like a cascade control system with the primary loop L3 giving a position setpoint to the secondary loop D15 which compares this with its process variable, the potentiometer S5, to energise the relevant raise or lower line.

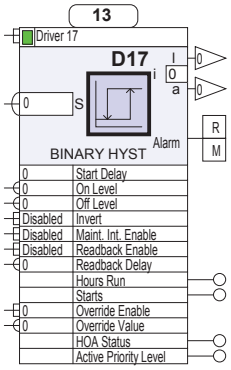
In both boundless and feedback operation the Driver module response to a change in input is dependent on the current valve position. If at either of its bounds, the Driver module has to go to a static position (i.e. for one cycle time), before recovering on the next cycle. If static in midrange, the Driver module will respond within one cycle time. If driving in one direction and then reversing, the driver will wait for 5s.

13.2.4.13 Type 4 (Raise Lower with End Stop) Driver Module Parameters

Type 4 (Raise Lower with End Stop) Driver modules include the parameters common to all driver modules - see **"Functionality" on page 207** plus the following parameters.

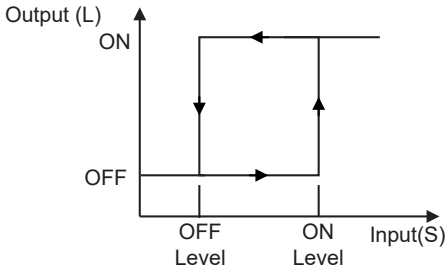
Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Drive Position	The current HVAC equipment item's position as a percentage of it full range as understood by the Driver module. If there is no feedback (boundless) it is estimated, otherwise it will reflect the 'Feedback' parameter's value.	0 to 100	0	O	P
Feedback	A positional feedback input to the Driver module. This enables the HVAC equipment item's actual position to be fed back to the driver. If set to a specific value, it will be ignored.	Full analogue	0	I	F
Full Scale Drive	The time in seconds taken to drive the HVAC equipment item (e.g actuator) over its full range. Can be connected to an analogue output.	0 to 32767 0 = Not enabled	0	I	D
Position Synchronisation	When enabled the Driver module will drive the actuator closed for 1.5 times the full scale drive time. This enables the actual actuator position to be re-calibrated regularly to ensure correct control. Once the synchronisation is complete the Driver module will return to the required control position.	0 or 1	0	I	Z
Reverse Operation	When enabled the behaviour of the raise and lower channels are swapped enabling the drive direction to be reversed i.e. the in-phase channel becomes the anti-phase channel and the anti-phase channel becomes the in-phase channel. The OFF situation is unchanged.	0 or 1 1 = Enabled 0 = Disabled	0	N	E
State	The current state of the module's outputs.	0, 1, 2, 3 0 = Off 1 = Stationary 2 = Lower 3 = Raise	0	O	B

13.2.5 Type 5 (Binary Hysteresis)



Type 5 (Binary Hysteresis) Driver modules provide an ON/OFF type output from an analogue input. They have an analogue input and provide a single digital output (with an optional antiphase output).

The state of output is calculated by comparing the value of the 'Source' parameter with the ON and OFF switching levels. If value of the 'Source' parameter goes above the ON switching level, the output will be set to '1', and if it is below the OFF switching level, it will be set to '0'. If an antiphase output is used its state will always be opposite to the state of the inphase output. The ON/OFF levels can be set apart to provide hysteresis.



The difference between the ON and OFF levels is the hysteresis, and the input can fluctuate within this band without the output changing.

Note: The ON and OFF switching levels can each be set between 0 and 255 although only 0 to 99 is operational.

When the module is first serviced (after a strategy download) the output is set to 0 and if the 'Source' parameter is between the OFF and ON levels then the output remains at '0'.

If it is to be used as a switch with no hysteresis, the ON and OFF levels can be set to the same value. However, the output may change rapidly as the input fluctuates about the ON/OFF level.

The digital outputs are achieved by using either analogue voltage output channels and external relay modules or in the case of IQ4E relays on the I/O modules.

13.2.5.1 Alarms

Supported - see **"Alarms" on page 207**.

13.2.5.2 Module Disable

Supported - see **"Module Disable" on page 209**. Disabling the module will leave the output channels in their last serviced state.

13.2.5.3 Module Override

Supported - see **"Module Override" on page 210**.

13.2.5.4 Hours Run

Supported - see **"Hours Run" on page 210**.

13.2.5.5 Driver Invert

Supported - see **"Driver Invert" on page 210**. If the module is inverted the channels are effectively cross-connected (transposed).

13.2.5.6 HOA Operation

Supported - see **"Hand Off Auto (HOA) Operation" on page 211**.

13.2.5.7 COV Operation

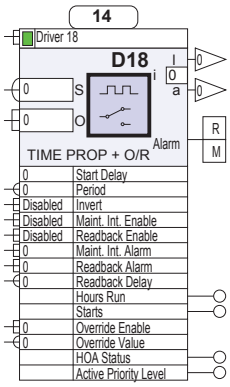
Supported - see **"COV Operation" on page 212**.

13.2.5.8 Type 5 (Binary Hysteresis) Driver Module Parameters

Type 5 (Binary Hysteresis) Driver modules include the parameters common to all driver modules - see **"Functionality" on page 207**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
OFF Level	The input level at and below which the Driver module will switch OFF. Can be connected to an analogue output.	0 to 100	0	I	F
ON Level	The input level at and above which the Driver module will switch ON. Can be connected to an analogue output.	0 to 100	0	I	O

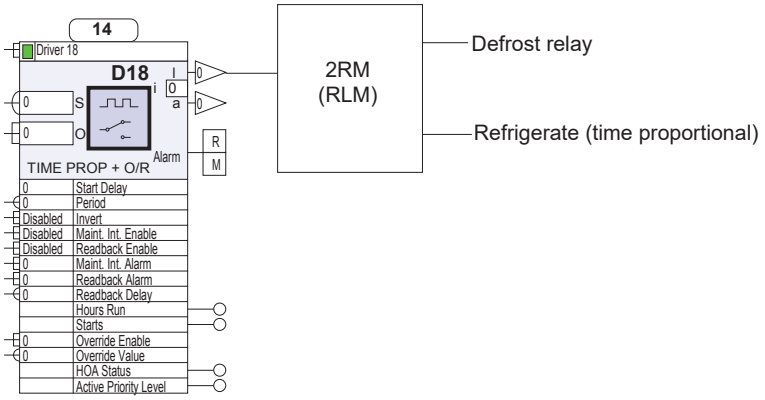
13.2.6 Type 6 (Time Proportional + OR)



Type 6 (Time Proportional + OR) Driver modules are designed to operate a raise lower relay module via an analogue output channel. They provide a time proportioning output on the lower relay or energise the raise relay in override. To do this they drive an analogue output channel which can have one of three values 10 V, 7 V and 4 V.

When the 'Override' parameter 'O' is set to '1' the analogue output channel is held at 10 V, otherwise they time proportion the output in a similar way to a time proportional driver but between 4 V (ON) and 7 V (OFF).

The principle application of this override facility is to drive a time proportional refrigeration stub of the type used in supermarkets. In this application, use of a raise lower module (2RM in RLM mode) halves the number of outputs and simplifies wiring.



When the 'Override' parameter 'O' is set to '1' the analogue output channel drives to 10V and holds the defrost relay (RAISE) ON and the refrigerate relay (LOWER) OFF. When the 'Override' parameter 'O' is set to '0' the defrost relay (RAISE) is OFF and the refrigeration relay (LOWER) comes ON for a time which is related to the period time by the input level (i.e. time proportional). Time proportional operation is described in **"Type 3 (Time Proportional)" on page 216**.

13.2.6.1 Alarms

Supported - see **"Alarms" on page 207**.

13.2.6.2 Module Disable

Supported - see **"Module Disable" on page 209**. Disabling the module allows the channels to continue to drive for their calculated time to the end of the period before turning both channels OFF. If the 'Override' parameter is ON the in-phase output will continue to be held ON.

13.2.6.3 Module Override

Supported - see **"Module Override" on page 210**.

13.2.6.4 Hours Run

Supported - see **"Hours Run" on page 210**.

13.2.6.5 Driver Invert

Supported - see **"Driver Invert" on page 210**.

13.2.6.6 HOA Operation

Supported - see **"Hand Off Auto (HOA) Operation" on page 211**.

13.2.6.7 COV Operation

Supported - see **"COV Operation" on page 212**.

13.2.6.8 Anti-phase Output

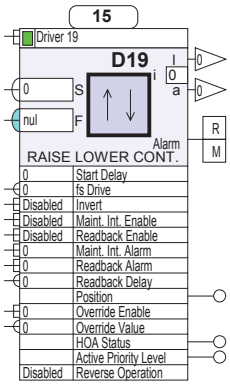
The anti-phase output is not normally used but it obeys the following rules. The anti-phase output channel (A) will be the inverse of the in-phase channel but only with respect to the time proportional 4 V (ON) and 7 V (OFF) levels, so if the override is ON the in-phase channel is at 10 V, i.e. the time proportional output is >7 V i.e. OFF so the anti-phase channel is ON i.e. 4 V.

13.2.6.9 Type 6 (Time Proportional + OR) Driver Module Parameters

Type 6 (Time Proportional + OR) Driver modules include the parameters common to all Driver modules - see **"Functionality" on page 207**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Override	Switches the module into override operation where it will drive the in-phase output to 10V.	0 or 1 1 = Override 0 = No override	0	I	O
Period	The amount of time (in seconds) being time proportioned.	0 to 32767	0	I	P

13.2.7 Type 7 (Raise Lower Continuous)



Type 7 (Raise Lower Continuous) Driver modules are very similar to Type 4 (Raise Lower with End Stop) Driver modules, see **page 220**, except for the way they deal with boundless operation - see **"Boundless Operation" on page 228**.

They are used for actuators with reversible action motors with built-in positioning or where there are built-in limit switches or clutches.

13.2.7.1 Alarms

Supported - see **"Alarms" on page 207**.

13.2.7.2 Module Disable

Supported - see **"Module Disable" on page 209**. Disabling the module allows the channels to continue to drive for their calculated time to the end of the period before turning both channels OFF. If the input is at 100 % the in-phase output will continue to be held on; if the input is at '0' the antiphase output will continue to be held ON.

13.2.7.3 Module Override

Supported - see **"Module Override" on page 210**.

13.2.7.4 Hours Run

Supported - see **"Hours Run" on page 210**.

13.2.7.5 Driver Invert

Supported - see **"Driver Invert" on page 210**. The invert is ignored on a Driver module with a single analogue output channel - see **"Analogue Output Channel and Relay Module Operation" on page 221**. For a driver with 2 digital outputs, the driver will invert both channels, but it doesn't transpose the channels as the anti-phase channel is not necessarily the inverse of the in-phase channel. However if the driver is in the static phase, both channels are left at '0'.

13.2.7.6 HOA Operation

Supported - see **"Hand Off Auto (HOA) Operation" on page 211**.

13.2.7.7 Reverse Operation

The reverse operation function will swap the raise and lower outputs. A Driver module with a single analogue output will perform an equivalent exchange of voltage levels. It can be used in conjunction with the invert function.

13.2.7.8 Boundless Operation

If the 'Feedback' parameter is set to '0' then the Driver module will operate using an estimated value position.

The position of the valve is estimated using the full-scale drive time and the sum of the amounts of time that the valve has been driven in each direction. (E.g. If the full-scale drive time is 10 s and the valve has been driven open from the closed position for 8 s and then driven closed for 3 s making a total of 5 s in the open direction (8-3). The estimated value position would be 50 % open because the value has been driven open for half the full scale drive time). The valve position is verified when the value is required to be fully closed or fully open (0% or 100) by driving continuously in as a the required direction. This ensures that the valve has reached the end stop.

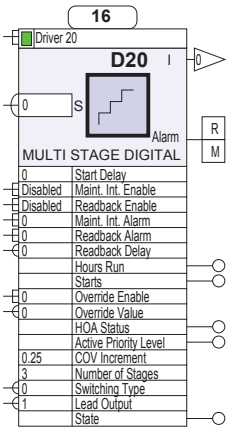
If the Feedback' parameters is set to '1' the module will operate in the same way as the Type 4 (Raise Lower with End Stop) Driver module - see **"Operation with Proportional Feedback" on page 222**.

13.2.7.9 Type 7 (Raise Lower Continuous) Driver Module Parameters

Type 7 (Raise Lower Continuous) Driver modules include the parameters common to all Driver modules - see **"Functionality" on page 207**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Drive Position	The current HVAC equipment item's position as a percentage of it full range as understood by the Driver module. If there is no feedback (boundless) it is estimated, otherwise it will reflect the 'Feedback' parameter.	0 to 100	0	O	P
Feedback	A positional feedback input to the Driver module. This enables the HVAC equipment item's actual position to be fed back to the Driver module. If set to a specific value, it will be ignored.	Full analogue	0	I	F
Full Scale Drive	The time in seconds taken to drive the HVAC equipment item (e.g. actuator) over its full range. Can be connected to an analogue output.	0 to 32767 0 = Not enabled	0	I	D
Reverse Operation	When enabled the behaviour of the raise and lower channels are swapped enabling the drive direction to be reversed i.e. the in-phase channel becomes the anti-phase channel and the anti-phase channel becomes the in-phase channel. The OFF situation is unchanged.	0 or 1 1 = Enabled 0 = Disabled	0	N	E
State	The current state of the module's outputs.	0, 1, 2, 3 0 = Off 1 = Stationary 2 = Lower 3 = Raise	0	O	B

13.2.8 Type 8 (Multistage Digital)



Type 8 (Multistage Digital) Driver modules enable a single input to drive a number of digital outputs in stages. Each stage corresponds to a separate output channel. This type of driver is ideal for controlling fan speed relays or boilers.

They switch a number of outputs ON and OFF as the value of the 'Source' parameter changes.

Note: The value of the 'Source' parameter is truncated; the decimal part is removed so that the value is reduced to an integer value (e.g. 3.23 becomes 3).

Note: Only the first channel is shown on the strategy symbol.

They change between stages as the input value passes through each integer up to the maximum number of stages determined by the 'Number of Stages' parameter depending on the selected switching type.

Note: The module requires consecutive outputs channels e.g. for 3 stages it will automatically take 3 consecutive outputs from the output specified by the 'In-phase Hardware' parameter. These outputs cannot be spread across I/O modules. Only the first channel is shown in the strategy symbol and IQSET dialogue box.

The way in which the outputs are turned ON/OFF is determined by the 'Switching Type' parameter which can be set to one of three modes:

Sequenced
Cascade
Binary

In addition the sequenced and cascade types can have a 'Lead Output' parameter; this defines the output stage at which the sequence starts.

Sequenced

With the sequenced switching type the outputs are energised and de-energised in sequence as the input varies, so that each output is energised and de-energised before the next output. There is a minimum time delay of 300 mS between de-energising an output and energising the next to ensure that if relays are being driven, only one relay is active.

Step	*Input Value*	*Lead Output*	*Channel*							
			1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
1	<1	1								
2	> = 1<2	1	ON							
3	> = 2<3	1		ON						
4	> = 3<4	1			ON					
5	> = 4<5	1				ON				
6	> = 5<6	1					ON			
7	> = 6<7	1						ON		
8	> = 7<8	1							ON	
9	> = 8	1								ON

As can be seen in the table above as the input value increases from 1 to 8 the outputs 1 to 8 are enabled in sequence, and as the input value reduces the appropriate output is enabled.

If the 'Lead Output' parameter is changed from '1', the channel number at which the sequence starts is offset:

Step	*Input Value*	*Lead Output*	*Channel*							
			1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
1	<1	3								
2	> = 1<2	3			ON					
3	> = 2<3	3				ON				
4	> = 3<4	3					ON			
5	> = 4<5	3						ON		

Step	*Input Value*	*Lead Output*	*Channel*							
			1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
6	> = 5<6	3							ON	
7	> = 6<7	3								ON
8	> = 7<8	3	ON							
9	> = 8	3		ON						

The above table shows the sequence when the 'Lead Output' parameter is set to '3'; the sequence now starts at channel 3.

Cascade

With the cascade switching type the number of outputs energised increases as the input value increases. There is no requirement for a minimum time delay between stages.

Step	*Input Value*	*Lead Output*	*Channel*							
			1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
1	<1	1								
2	> = 1<2	1	ON							
3	> = 2<3	1	ON	ON						
4	> = 3<4	1	ON	ON	ON					
5	> = 4<5	1	ON	ON	ON	ON				
6	> = 5<6	1	ON	ON	ON	ON	ON			
7	> = 6<7	1	ON	ON	ON	ON	ON	ON		
8	> = 7<8	1	ON	ON	ON	ON	ON	ON	ON	
9	> = 8	1	ON	ON	ON	ON	ON	ON	ON	ON

If the 'Lead Output' parameter is changed from '1', the channel number at which the sequence starts is offset:

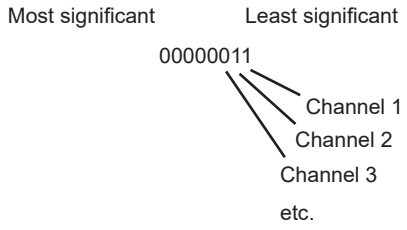
Step	*Input Value*	*Lead Output*	*Channel*							
			1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
1	<1	4								
2	> = 1<2	4				ON				
3	> = 2<3	4				ON	ON			
4	> = 3<4	4				ON	ON	ON		
5	> = 4<5	4				ON	ON	ON	ON	
6	> = 5<6	4				ON	ON	ON	ON	ON
7	> = 6<7	4	ON			ON	ON	ON	ON	ON
8	> = 7<8	4	ON	ON		ON	ON	ON	ON	ON
9	> = 8	4	ON	ON	ON	ON	ON	ON	ON	ON

The above table shows the sequence when the 'Lead Output' parameter is set to '4'; the sequence now starts at channel 4.

Binary

With the binary switching type the truncated analogue input value is converted to a binary number which defines the outputs energised. There is no requirement for a minimum time delay between stages, and the 'Lead Output' parameter is not supported for the binary switching type.

For example the input value 3 will be represented by binary value 00000011 which specifies channel 1 and 2 to be energised



A binary value of 8 bits corresponds to analogue values of 0 to 255. To fully represent this on a table would require 256 rows; the reduced table below shows the first 9 values, and then some further example values.

Step	*Input Value*	*Channel*							
		1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
1	<1								
2	> = 1<2	ON							
3	> = 2<3		ON						
4	> = 3<4	ON	ON						
5	> = 4<5			ON					
6	> = 5<6	ON		ON					
7	> = 6<7		ON	ON					
8	> = 7<8	ON	ON	ON					
9	> = 8<9				ON				
46	> = 45<46	ON		ON	ON		ON		
68	> = 67<68	ON	ON					ON	
86	> = 85<86	ON		ON		ON		ON	
102	> = 101<102	ON		ON			ON	ON	
135	> = 134<135		ON	ON					ON
159	> = 158<159		ON	ON	ON	ON			ON
180	> = 179<180	ON	ON			ON	ON		ON
201	> = 200<201				ON			ON	ON
256	> = 255	ON	ON	ON	ON	ON	ON	ON	ON

13.2.8.1 Alarms

Supported - see **“Alarms” on page 207**.

13.2.8.2 Module Disable

Supported - see **“Module Disable” on page 209**.

13.2.8.3 Module Override

Supported - see **“Module Override” on page 210**.

13.2.8.4 Hours Run

Supported - see **“Hours Run” on page 210**.

Note: The Hours Run value is the total time for which any of the outputs have been ON (i.e. ‘State’ parameter is non-zero).

13.2.8.5 Driver Invert

Not Supported.

13.2.8.6 HOA Operation

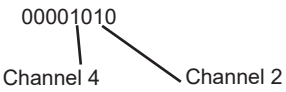
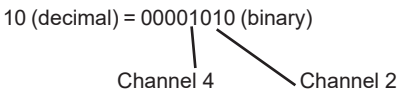
Supported - see **“Hand Off Auto (HOA) Operation” on page 211**.

13.2.8.7 COV Operation

Supported - see **“COV Operation” on page 212**.

13.2.8.8 Type 8 (Multistage Digital) Driver Module Parameters

Type 8 (Multistage Digital) Driver modules include the parameters common to all Driver modules - see **"Driver Module Common Parameters" on page 234**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Lead Output	Not binary. This defines the first output channel (relative to the In-phase Output) to be energised as the input value increases from 0 (e.g. Lead Output = 2 with In-phase Output = 2, means channel 3 will be energised first).	0 to maximum number of stages	1	I	D
Number of Stages	The number of output channels used. The channels allocated will continue sequentially from the In-Phase channel.	IQ41x = 0 to 6 IQ422 = 0 to 6 IQ4E = 0 to 8 IQ4NC/00. = 0 to 6 IQ4NC/12 = 0 to 6 IQ4NC/16/ = 0 to 8 IQ4NC/32 = 0 to 8	0	N	R
Output status	The 8 status bits which indicate which output channels are energised (corresponds to the state parameter).  	00000000 to 1111111	00000000	O	F
State	The current state of output channels as an analogue value whose binary equivalent indicates the output channels which are energised (e.g. a value of 10 indicates that channels 4 and 2 are energised).  This output is coded such that it may be connected to an Type 18 (Analogue to Digital Converter) Function module so that the energised channels can be decoded.	0 to 255	0	O	B
Switching Type	The type of sequencing.	0, 1, or 2 0 = Sequenced 1 = Cascade 2 = Binary	0	N	P

13.3 Driver Module Common Parameters

Driver modules also include the following parameters that are common to most modules: 'Disable Module', 'Label', 'Page', 'Sequence Step', 'Sequence Table', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Active Priority Level	The current priority level of the input to the modules's algorithm. Normal strategy operation writes to priority level '10'. If a higher priority value comes in from BACnet, this controls the module. It enables the strategy to detect that the module has been overridden (e.g. due to a BACnet communication). For example this could be done by checking that the Active Priority Level has not exceeded the expected level using a Type 15 (Comparator) Function module. *Note: For BACnet writes the level can be specified in the message.*	0 to 16 0 = Source unconnected and no other active input. 3 = HOA switch override. 4 = Override 9 = IC comms write 10 = Strategy 16 = BACnet 0 = highest priority	10	O	u
Alarm Delay (secs) (HOA)	v3.70 or greater. The period (in seconds) for which the HOA input must be in an alarm condition before the HOA alarm is generated. Digital, Analogue, Binary Hysteresis and Multistage Driver modules only.	0 to 178200	0	I	d1(D)
Alarm Delay (secs) (maintenance)	The period (in seconds) for which the maintenance input must be in an alarm condition before the maintenance alarm is generated. *Note: Should normally be set to '0' as the alarm is normally set by strategy which already includes a delay - see* **page 208**.	0 to 178200	0	I	m1(D)
Alarm Delay (secs) (readback)	The period (in seconds) for which the readback input must be in an alarm condition before the readback alarm is generated. *Note: Should normally be set to '0' as it is normal practice to use a readback module to set the alarm state and this already includes a delay - see* **page 207**.	0 to 178200	0	I	r1(D)
Alarm Group (HOA)	v3.70 or greater. The Alarm Group module to which the HOA alarm will be sent. Digital, Analogue, Binary Hysteresis and Multistage Driver modules only.	0 to 500	1	N	d1(G)
Alarm Group (maintenance)	The Alarm Group module to which the maintenance alarm will be sent.	0 to 500	1	N	m1(G)
Alarm Group (readback)	The Alarm Group module to which the readback alarm will be sent.	0 to 500	1	N	r1(G)
Alarm Sent (maintenance)	Indicates if the maintenance alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	m1(K)
Alarm Sent (readback)	Indicates if the readback alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	r1(K)

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm state (HOA)	v3.70 or greater. The state of the HOA alarm condition. It will be '1' if the HOA interval alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable. Digital, Analogue, Binary Hysteresis and Multistage Driver modules only.	0 or 1 1 = Alarm 0 = No alarm	0	O	d1(S)
Alarm state (maintenance)	The state of the maintenance interval alarm condition. It will be '1' if the maintenance interval alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	m1(S)
Alarm State (readback)	The state of the readback alarm condition. It will be '1' if the readback alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	r1(S)
Anti-phase Hardware Channel	Not Type 8 (Multistage Digital). The output channel on the specified I/O module used by the driver's anti-phase output. The range of channel numbers available depends on the type of IQ4 variant For type 1, 3, 4, 5, and 7 Driver modules the output channel should be digital, for a type Type 6 (Time Proportional + OR) Driver module it should be analogue and for a Type 2 (Analogue) Driver module it can be analogue or digital.	0 to 196 0 = Not connected	0	N	A
Anti-Phase Output (mV)/Anti-Phase Output	The value of the anti-phase output channel.	Type 2 (Analogue) and Type 6 (Time Proportional + OR): 0 to 1000 Other types: 0 or 1	0	O	a
BACnet Event Time Stamps	BACnet only. An array of three parameters; the times of the last transitions into alarm states: 'n1' parameter = time of last charge to 'To OffNormal' 'n2' parameter = time of last charge to 'To Fault' 'n3' parameter = time of last charge to 'To Normal' *Note: A readback alarm is considered 'OffNormal' and its transition times are recorded here but not intrinsically reported. So the Readback alarm time = n1, clear readback is n3; n2 is always '0'.* *Note: A maintenance alarm is not considered a BACnet event, so does not set these times.*	date&time; yyyy-mm-dd&Thh:mm:ss	0 0 0 0 - 0 0 - 00T00:00:00	O	n1 to n3
BACnet Priority Array	BACnet only. The 16 values in the priority array. The lowest numbered array element which is currently set up will be used for the driver's input A null signifies that the array element is release or not used.	0 or 1 or Full analogue	All null	O	o1 to o16

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Clear Alarm (HOA)	v3.70 or greater. Enables/disables reporting of the HOA clear alarm. Digital, Analogue, Binary Hysteresis and Multistage Driver modules only.	0 or 1 1 = Enabled 0 = Not enabled	1	N	d1(C)
Clear Alarm (maintenance)	Enables/disables reporting of the maintenance interval clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	m1(C)
Clear Alarm (readback)	Enables/disables reporting of the readback clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	r1(C)
Clear Alarm Delay (secs) (readback)	v3.50 or greater. The period (in seconds) for which the readback input must be cleared before the clear alarm is generated. *Note: Should normally be set to '0' as it is normal practice to use a readback module to set the alarm state and this already includes a delay - see page 207.*	0 to 178200	0	I	r1(d)
Clear Alarm Delay (secs) (HOA)	v3.70 or greater. The period (in seconds) for which the maintenance input must be cleared before the clear alarm is generated. Digital, Analogue, Binary Hysteresis and Multistage Driver modules only.	0 to 178200	0	I	d1(d)
Clear Alarm Delay (secs) (maintenance)	v3.50 or greater. The period (in seconds) for which the maintenance input must be cleared before the clear alarm is generated. *Note: Should normally be set to '0' as the alarm is normally set by strategy which already includes a delay - see page 208.*	0 to 178200	0	I	m1(d)
COV Increment	Not Type 1 (Digital). The amount by which the driver value must change before a BACnet COV notification message is sent to those subscribers in the list of subscribers who have subscribed to this parameter. The 'COV Increment' parameter must not be set to '0'.	+ Full analogue	0.25	N	C
Enabled (HOA)	Enables/disables reporting of the readback alarm. Digital, Analogue, Binary Hysteresis and Multistage Driver modules only.	0 or 1 1 = Enabled 0 = Disabled	0	I	d1(E)
Enabled (maintenance)	Enables/disables reporting of the maintenance interval alarm.	0 or 1 1 = Enabled 0 = Disabled	0	I	m1(E)
Enabled (readback)	Enables/disables reporting of the readback alarm.	0 or 1 1 = Enabled 0 = Disabled	0	I	r1(E)
Hardware Module	The address of the physical I/O module that contains the input channel providing the module's input - only apply to IQ4s that support external I/O modules, set to '0' for all other types of IQ4.	0 to 30 0 = IQ4 1 to 30 = I/O modules	0	N	i

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
HOA State	The Hand/OFF/AUTO status of the module's output if it is on an I/O module that supports Hand/Off/Auto operation. Always '0' when the module's output is on one of the IQ4's on board outputs or on an I/O module that does not support Hand/OFF/AUTO operation - see **"Hand Off Auto (HOA) Operation" on page 211**.	0, 1, 2, 3 0 = Auto 1 = Hand 2 = Off 3 = Error	0	O	h
Hours Run	Not Type 2 (Analogue). The time (in hours) for which the in-phase output (prior to inversion) has been ON. For Type 8 (Multistage Digital) see additional definition in **"Hours Run" on page 232**.	0 to 87600	0	O	H
In Alarm	The module's alarm status. It has accompanying text shown below which is shown on the driver modules overview page in the 'alarm' column. 0 = No alarm 8 = Maintenance 9 = Readback The alarm state is shown if the alarm condition has been present for longer than that alarm's delay time, and that alarm is enabled to be reported. If both alarms are present the Maintenance alarm will be reported. It will be set back to '0' immediately both the alarms are in the clear state (for longer than their alarm delay times).	0, 8, 9 0 = No alarm 8 = Maintenance alarm 9 = Readback alarm	0	O	!
In-Phase Hardware Channel	The output channel on the specified I/O module used by the driver. The range of channel numbers available depends on the type of IQ4. For Type 8 Driver modules it is the first connected (lowest number) channel. For type 1, 3, 4, 5, 7 and 8 Driver modules the output channel should be digital, for a Type 6 (Time Proportional + OR) Driver module it should be analogue and for a Type 2 (Analogue) Driver module it can be analogue or digital.	0 to 32 0 = Not connected	0	N	L
In-Phase Output (mV)/In-Phase Output	The value of the in-phase output channel. For Type 8 (Multistage Digital) Driver modules it is '1' if any of the driver's output channels are ON, and '0' if all its channels are OFF.	Type 2 (Analogue) and Type 6 (Time Proportional + OR): 0 to 1000 Other types: 0 or 1	0	O	l
Invert	No For Type 8 (Multistage Digital). Logically inverts the driver's output.	0 or 1 0 = Not inverted 1 = Inverted	0	I	I
Maintenance input	The maintenance interval alarm input to the Driver module (normally from Type 6 (Hours Run) Logic module).	0 or 1 0 = No alarm 1 = Alarm	0	S, N	s

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Maintenance	Indicates the maintenance interval alarm state. Clicking on **Maintenance** on the web page will display the maintenance interval alarm sub-module. The way in which the sub-module declares alarms and clear alarms is described in **"Alarms" on page 71**.	-	-	-	-
Module Status	The module's status	0, 1, 2, 3, 4, 5 0 = Normal (automatic control) 1 = Overridden 2 = Disabled 3 = Hand (manually set on) 4 = Off (manually set off) 5 = BACnet (BACnet in control)	0	O	_
Number of Starts	Not Type 2 (Analogue). The number of times the in-phase output (prior to inversion) has been transitioned from '0' to '1'. For Type 8 (Multistage Digital) Driver modules it is the number of transitions from the state where all the output channels are OFF to the state where any of the output channels are on (i.e. from where the 'State' parameter is '0' to where the 'State' parameter is non-zero).	0 to 10^9	0	O	N
Override Output	Overrides the Driver module's output to the override value or status set in 'Override' parameter.	0 or 1 1 = override 0 = no override	0	I	^
Override Value/Status	The value to which the module is overridden when the 'Override Output' parameter is set to '1'.	Full analogue or 0, 1	0	I	v
Power on Delay (secs)	The period (in seconds) after the IQ4 is powered up or performs a soft restart before the driver output is switched on. It can be used to stop all the plant switching on simultaneously when power is returned to the controller, so each driver should be given a different delay. During the delay analogue output channels will be held at 0 V, and digital output channels will be OFF.	0 to 1275	0	N	T
Readback input	The readback alarm input to the driver (normally from Type 3 (Readback) Logic module).	0 or 1 0 = No alarm 1 = Alarm	0	I	b
Readback	Indicates the readback alarm state. Clicking on **Readback** on the web page will display the readback interval alarm sub-module. The way in which the sub-module declares alarms and clear alarms is described in **"Driver Invert" on page 210**.				
Source	The value input to the Driver module.	0 or 1 for Type 1 (Digital) Or Full analogue	0	I	S

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Time in Alarm (secs) (HOA)	The elapsed time in seconds since the HOA alarm state commenced. Digital, Analogue, Binary Hysteresis and Multistage Driver modules only.	+ Full analogue	0	O	d1(T)
Time in Alarm (secs) (readback)	The elapsed time in seconds since the readback alarm state commenced.	+ Full analogue	0	O	r1(T)
Time in Alarm (secs) (maintenance)	The elapsed time in seconds since the maintenance alarm state commenced.	+ Full analogue	0	O	m1(T)
Type	The type of Driver module.	1, 2, 3, 4, 5, 6, 7, 8, 9 1 = Digital 2 = Analogue 3 = Time Proportional 4 = Raise Lower 5 = Binary Hysteresis 6 = Time Proportional + OR 7 = Raise Lower (continuous) 8 = Multistage	-	N	Y
Value	The current drive level - see **"Value" on page 239**.	Full analogue	0	O	V

Value

The current drive level. For Type 1 (Digital) Drivers module's the value is a digital status, for all other types of Driver modules it is an analogue value.

If the input is outside the range 0 to 100 it will have been clipped to be within the range 0 to 100 (but see negative input - **"Analogue Output Channel and Relay Module Operation" on page 221**). It includes the effects of the override function, and the HOA feedback. If the in-phase output channel is in Hand, the value is back calculated as follows:

Driver Type	*Value Type*	*Value Calculated*	
		No Invert	***Invert***
1	Digital	In-phase channel status	Anti-phase channel status
2	Analogue	Not supported	
3	Analogue	100% x In-phase channel status	100% x Anti-phase channel status
4	Analogue	If using boundless operation it is an estimate of the actuator position based on amount of time raise and lower outputs have been overridden. If not using boundless operation it is the value of the 'Feedback' parameter.	
5	Analogue	100% x In-phase channel status	100% x Anti-phase channel status
6	Analogue	100% x In-phase channel status	100% x Anti-phase channel status
7	Analogue	If using boundless operation it is an estimate of the actuator position based on amount of time raise and lower outputs have been overridden. If not using boundless operation it is the value of the 'Feedback' parameter.	
8	Analogue	Not supported	

Note: In the above table for type 3, 5, and 6 Driver module's, the In-phase/Anti-phase channel status will be either ON or OFF, so Value will either be 100% or 0%.

13.4 Driver Module Connectives

The following parameters are available for connection to other modules in the strategy:

All Driver module types:

Anti-Phase Output (mV)/ Anti-Phase Output
HOA State
Hours Run (not Type 2 (Analogue))
In-Phase Output (mV)/ In-Phase Output
Number of Starts (not Type 2 (Analogue))
Active Priority Level
Alarm State (maintenance)
Alarm State (readback)

Type 4 (Raise Lower with End Stop) and Type 7 (Raise Lower Continuous) only:

Driver Position

Type 8 (Multistage Digital) only:

State

14 FUNCTION MODULES

Module Identifier	Fn (where n is module number)
Default overview parameters	($, D)
Size in brIQs	20

14.1 Functionality

Function modules enable mathematical processes to be performed on a number of variables. There are several types of Function module each performing a different calculation - see **"Function Module Types" on page 242**. They have up to four analogue inputs, depending on the function being performed and usually one analogue output. For certain types of function the output (and/or an input) may be digital. The inputs may be either sources or specific values enabling such parameters as scaling factors to be generated by the strategy.

In general if the calculation involves a divide by zero, the module will output the largest number it can handle (normally 10^{20}).

Function modules appear on the strategy page and can either be sequence table or event driven.

Note: Type 1 (Filter) Function modules cannot be event driven.

14.1.1 Module Disable

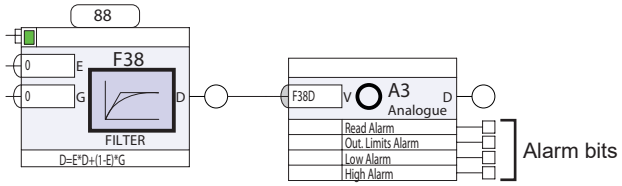
All Function modules can be disabled using the 'Disable Module' parameter - see **"Disabling Modules" on page 29**.

14.2 Function Module Alarms

A Function module with an analogue output generates alarm bits as a function of the alarm bits associated with its inputs (see **"Analogue Alarm Status Bits" on page 78**) as follows:

Module Type	*Description*
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 16, 19, 20, 23, 22, 24, 25, 26, 27, 28	The alarm bits on the inputs used are OR'ed together (summed) i.e if one input is in alarm the output will be in alarm.
11, 12 13	If inputs go into out-of-limits (OUTL) or read (READ) alarms, they are ignored, and the status of the output indicates the OR'ed function of high and low alarm bits on the remaining used inputs. If all the inputs have either out-of-limits or read alarms then the output remains unchanged (as it was before the alarm) and the output alarm bits have the OR'ed function of any of the inputs (i.e. any out-of-limits, read, high or low alarms are passed on).
14	The alarm bits from the source in use are transferred to the output.
15, 17, 18, 21, 22, 29, 30, 31, 32, 33, 34	All alarm bits are ignored.

These alarm bits can be made accessible by connecting an analogue node module to the Function module's output.

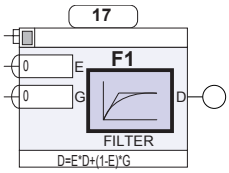


14.3 Function Module Types

There are several Function module types specified using the 'Type' parameter:

Module Type	*Description*
Type 1 (Filter)	Exponential filter with gain.
Type 2 (Rescale From)	Rescales input of 0 to 100 to user defined values and limits at these values.
Type 3 (Rescale To)	Rescales input to output of 0 to 100 and limits to 0 to 100.
Type 4 (Limit At)	Limits the output to user defined values.
Type 5 (Limit To)	Will drive the output to 0 and 100 at user defined values.
Type 6 (Log)	Logarithm base 10 of input.
Type 7 (Square Root)	Square root of input.
Type 8 (Adder/Scaler)	Adds and scales 2 inputs.
Type 9 (Multiplier)	Multiplies 3 inputs.
Type 10 (Divider)	Divides one input by another.
Type 11 (Minimum)	The minimum of up to 4 inputs.
Type 12 (Maximum)	The maximum of up to 4 inputs.
Type 13 (Average)	The average of up to 4 inputs.
Type 14 (Analogue Gate)	Digital input switches output between two analogue inputs.
Type 15 (Comparator)	Creates digital output which compares 2 inputs.
Type 16 (Enthalpy)	Calculates enthalpy value from humidity and temperature.
Type 17 (Hysteresis Band)	Creates digital output from analogue input using level and hysteresis band.
Type 18 (Analogue to Digital Converter)	Creates digital byte output from analogue input.
Type 19 (Lookup)	Scales an input according to characteristic table.
Type 20 (Adder)	Adds 4 inputs.
Type 21 (Hysteresis Limits)	Creates digital output from analogue input using upper and lower hysteresis limits.
Type 22 (Proximity)	Creates digital output which indicates if 2 analogue inputs are in defined band of each other.
Type 23 (Power)	Raises the input to a defined power.
Type 24 (Absolute)	Makes negative input positive (removes minus sign).
Type 25 (Truncate)	Removes part of input after decimal point.
Type 26 (Trigonometric)	Creates sine, cosine or tangent of input (in degrees or radians).
Type 27 (Heat Meter)	Mainly used to calculate current power of heating or cooling devices.
Type 28 (Integrator)	Integrates the input with respect to time.
Type 29 (Change of Value)	Generates a pulse on its output when the value of the 'Source' parameter changes by more than a specified amount.
Type 30 (Sample and Hold)	Sets its output to the value of the 'Source' parameter and retains that value until re-enabled regardless of the input value.
Type 31 (Psychrometrics)	Calculates the saturated vapour pressure, vapour pressure, enthalpy, moisture content and dew point based on the relative humidity, temperature and height above sea level.
Type 32 (Sliding Window)	Allows either Min, Max, Sum, Average, or Standard Deviation calculations to be applied to a specified range of data in a Plot module.
Type 33 (Differencer)	Calculates an incremental consumption from a cumulative consumption value.
Type 34 (Analogue Input Selector)	Selects one of up to eight analogue input parameters depending on the value of the 'Selector' parameter.

14.3.1 Type 1 (Filter)



Type 1 (Filter) Function modules apply an exponential filter (filter factor) to an input signal to calculate the 'Destination' parameter. This is useful where electrical noise levels are high, and an unfiltered signal would affect the operation of the system. They could also be used, in an air conditioning system where air pressure is being monitored and the pressure signal is fluctuating due to turbulence. When the module is first serviced (e.g. after a strategy download) the output starts at '0'. Type 1 (Filter) Function modules should not be event driven.

14.3.1.1 Formula

$$D_{n+1} = E \times D_n + (1 - E)\, G$$

Where:

D_{n+1} = new output
D_n = old output
E = filter factor
G = input

14.3.1.2 Filter Factor

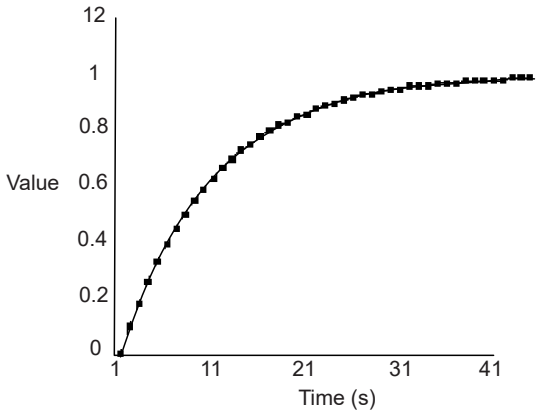
The normal way of setting the response of a simple first order filter would be to set the 'Time Constant' parameter (67% step response time). It is normally in the range 0 to 1 where '0' gives no filter action, and '1' gives an infinite filter (i.e. no change in output). The filter factor can be calculated from the desired time constant using the following equation:

$$\text{Filter factor} = \frac{-1.0}{e^{\text{Time Constant}}}$$

This gives filter factors for the various time constants as shown below:

Time constant (s)	*Filter factor*
1	0.3679
5	0.8187
10	0.9048
30	0.9672
60	0.9835

It can be seen that the filter factors get nearer to unity and critically closer together as the time constants get longer. The unit step response of the output to a filter factor of 0.9048 (i.e. time constant of 10 s) can be seen in the graph below.



It can be seen that the output reaches about 2/3rds (0.67) of its final value after 10 s (the time constant) and thereafter its slope drops off as it tends exponentially to the value of '1'.

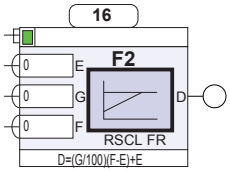
Note: When the module is first serviced e.g. after a strategy download, the module's output is set to '0'.

14.3.1.3 Type 1 (Filter) Function Module Parameters

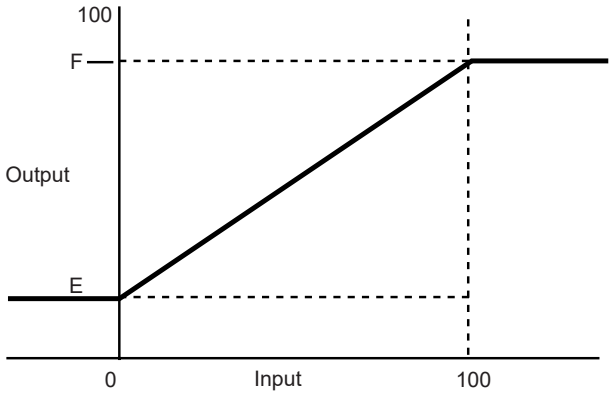
Type 1 (Filter) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies the filter factor in the calculation - see **"Filter Factor" on page 243**.	Full analogue	0	I	E
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.2 Type 2 (Rescale From)



Type 2 (Rescale From) Function modules calculate the 'Destination' parameter by rescaling the 'Source G' parameter that can be between 0 and 100 to a range between a lower limit ('Source E' parameter), and an upper limit ('Source F' parameter), this is shown by the graph.



The output, D, is limited to range E to F.

14.3.2.1 Formula

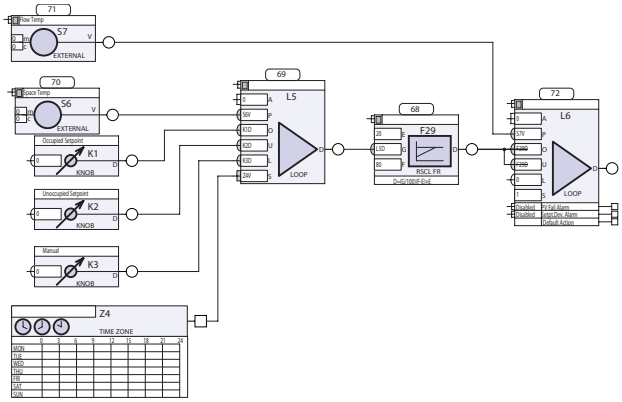
$$D = E + \frac{G(F-E)}{100}$$

Where:

D = Destination' parameter
E = 'Source E' parameter (lower limit)
F = 'Source F' parameter (upper limit)
G = 'Source G' parameter

14.3.2.2 Strategy Example

The output of L5 is fed to F29. F29 rescales the 0 to 100 input signal from L5 to a 20 to 80 output signal, where 20 represents a 0 input and 80 represents a 100 input. The rescaled signal is then used to provide the setpoint for L6.

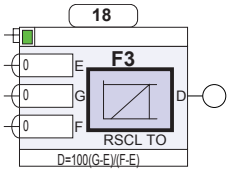


14.3.2.3 Type 2 (Rescale From) Function Module Parameters

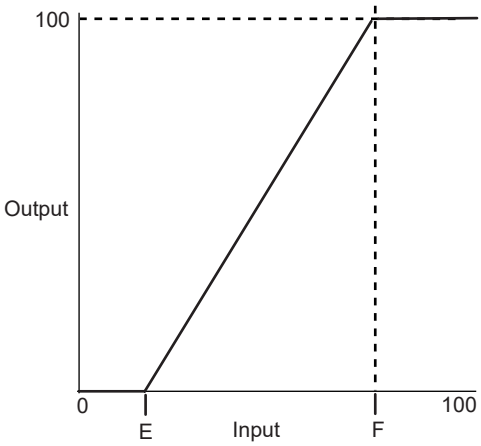
Type 2 (Rescale From) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E (lower limit) in the calculation.	Full analogue	0	I	E
Source F	Specifies F (upper limit) in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.3 Type 3 (Rescale To)



Type 3 (Rescale To) Function modules rescale the 'Source G' parameter to a percentage value between 0 and 100 and writes it to the 'Destination' parameter. As shown in the graph the 'Destination' parameter is limited to the range 0 to 100.



14.3.3.1 Formula

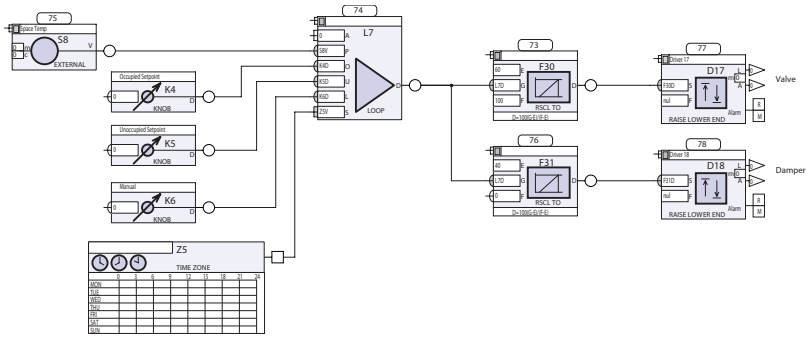
$$D = \frac{100(G-E)}{F-E}$$

Where

D = 'Destination' parameter
E = 'Source E' (Lower limit)
F = 'Source F' Upper limit
G = 'Source G' parameter

Note: If F=E, D=100.

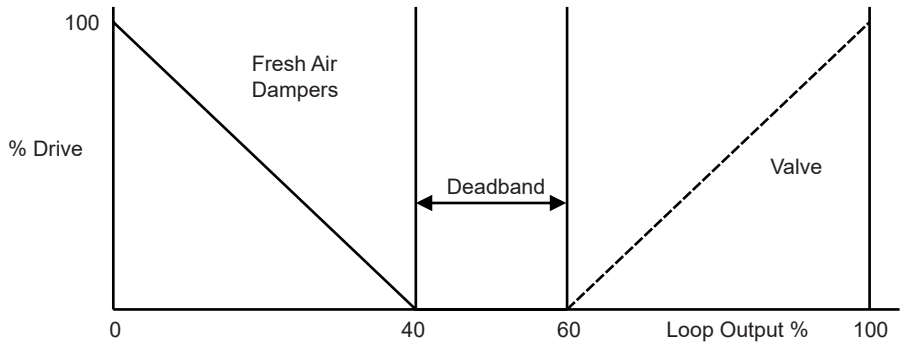
14.3.3.2 Strategy Example



In the above example, simple AHU control is achieved. S8 is monitoring the space temperature and providing an input to L7. The output from L7 is in the range 0 to 100 %, depending on the input from S8. When the output from L7 is in the range of 40 to 0 %, the output from the F31 will be in the range 0 to 100 % so that when the input to F31 rises from 0 % to 20 %, the output of F31 will fall to 50 %.

When the output from L7 reaches 40%, the dampers will be fully closed. F30 performs a similar function to F31 but when an input to F30 is at 80 % its output will be 50 % which is used by D17 to drive the valve open to 50%.

There is a 20 % difference between the lower limit of F31 (40) and the lower limit of F30 (60). This difference is known as a dead band and is incorporated so that small changes around the crossover point, where the dampers are fully closed and the valve starts to open, do not cause the valve/dampers to keep changing state over a short duration.

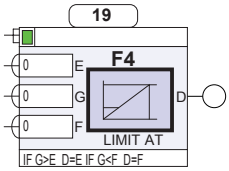


14.3.3.3 Type 3 (Rescale To) Function Module Parameters

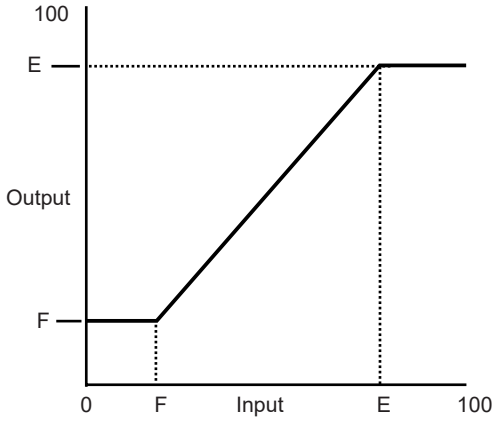
Type 3 (Rescale To) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E (lower limit) in the calculation.	Full analogue	0	I	E
Source F	Specifies F (upper limit) in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.4 Type 4 (Limit At)



Type 4 (Limit At) Function modules provide the facility for limiting the module's 'Destination' parameter between a lower limit defined by the 'Source F' parameter and an upper limit defined by the 'Source E' parameter, as shown in the graph below. For example, if there an input in the range 0 to 100 and only a limited range of the input is required i.e. 20 to 80.



14.3.4.1 Formula

If G>E then D = E
If G<F then D = F
else D = G

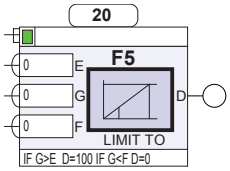
Where:
D = 'Destination' parameter
E = 'Source E' parameter (upper limit)
F = 'Source E' parameter (lower limit)
G = 'Source G' parameter

14.3.4.2 Type 4 (Limit At) Function Module Parameters

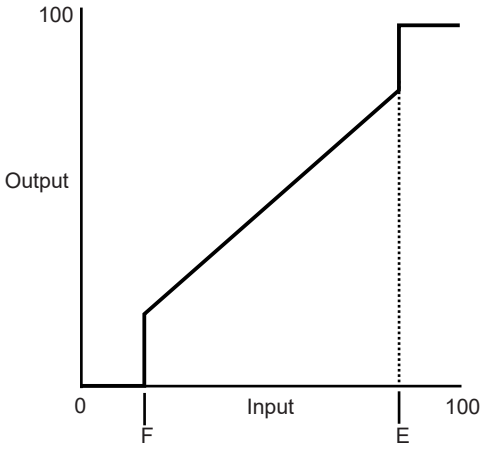
Type 4 (Limit At) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E (upper limit) in the calculation.	Full analogue	0	I	E
Source F	Specifies F (lower limit) in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.5 Type 5 (Limit To)



Type 5 (Limit To) Function modules provide the facility for limiting the 'Destination' parameter to the range of 0 to 100 % from any input range as illustrated below.



The 'Source G' is the analogue variable that is to have limits applied. The 'Source E' parameter is the upper value where, when G is greater than E, the module output goes to 100 %. The 'Source F' parameter is the lower value where, when G is less than F, the module output goes to 0 %.

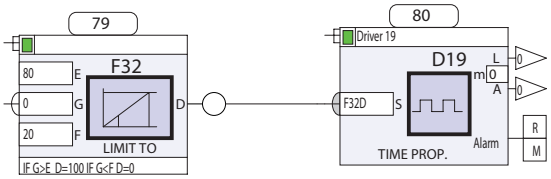
14.3.5.1 Formula

If G>E then D = 100
If G<F then D = 0
Else D = G

Where:
D = 'Destination' parameter
E = 'Source E' parameter (upper value)
F = 'Source F' parameter (lower value)
G = 'Source G' parameter

14.3.5.2 Strategy Example

Type 5 (Limit To) Function modules are often used to provide minimum ON times and minimum OFF times with time proportional drivers.



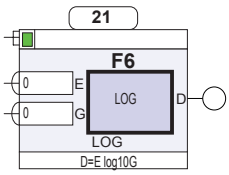
In the above example any input signal greater than 80 % will cause the time proportional driver to switch ON for 100 % of the time period, preventing short OFF periods. Inputs of less that 20 % will cause the time proportional driver to switch OFF for 100 % of the time period preventing short ON periods. For all values between E and F, the module output will correspond to the value of G.

14.3.5.3 Type 5 (Limit To) Function Module Parameters

Type 5 (Limit To) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E (upper limit) in the calculation.	Full analogue	0	I	E
Source F	Specifies F (lower limit) in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.6 Type 6 (Log)



Type 6 (Log) Function modules calculate the 'Destination' parameter as the logarithm (to base 10) of the 'Source G' parameter multiplied by the 'Source E' parameter. They can be used to provide the correct input scaling to a loop, for a logarithmic type sensor. Alternatively they could be used to provide a logarithmic scaled output to an analogue Driver module, suitable to drive a logarithmic actuator. The 'Source G' parameter must be greater than '0'; if it is '0' or negative, $\log_{10}G$ is calculated as -20.

14.3.6.1 Formula

$$D = E \log_{10} G$$

Where:

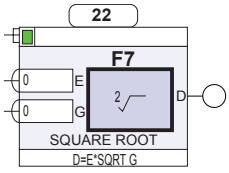
D = Destination' parameter
E = 'Source E' parameter (multiplier)
G = 'Source G' parameter

14.3.6.2 Type 6 (Log) Function Module Parameters

Type 6 (Log) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Alarms" on page 241**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E (multiplier) in the calculation.	Full analogue	0	I	E
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.7 Type 7 (Square Root)



Type 7 (Square Root) Function modules calculate the 'Destination' parameter as the square root of the 'Source G' parameter multiplied by the 'Source E' parameter. If the value of the 'Source E' parameter is set to '1', the 'Destination' parameter will be the square root of the 'Source G' parameter. If the 'Source G' parameter is negative the output is set to '0'.

14.3.7.1 Formula

$$D = E \sqrt{G}$$

Where:

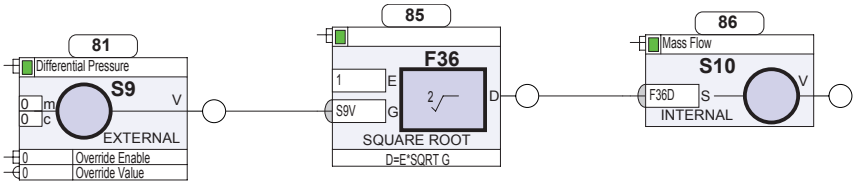
D = 'Destination' parameter
E = 'Source E' parameter (multiplier)
G = 'Source G' parameter

14.3.7.2 Strategy Example

Square roots are most commonly used when calculating the mass flow of air or water. The formula for mass flow is:

$$\text{Mass Flow} = K \sqrt{P}$$

Where K is a constant and P is the pressure differential. The above formula can be achieved by using the following strategy:



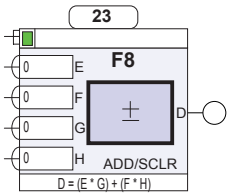
Internal sensor 10 is used to provide rescaled high and low alarm limits

14.3.7.3 Type 7 (Square Root) Function Module Parameters

Type 7 (Square Root) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.8 Type 8 (Adder/Scaler)



Type 8 (Adder/Scaler) Function modules provide the facility for adding two value, the 'Source G' and 'source H' or the ratio of inputs.

The module can be used to perform the y = mx + c function where y is the 'Destination' parameter, x is the 'Source G' parameter, m is the 'Source E' parameter, and c is 'Source H' parameter (multiplied by the 'Source F' parameter which is set to '1'). Subtraction can be achieved by setting the 'Source E' parameter to '+1' and the 'Source F' parameter to -1.

14.3.8.1 Formula

$$D = (E \times G) + (F \times H)$$

Where:

D = 'Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter
G = 'Source G' parameter
H = 'Source H' parameter

Note: If the 'Source G' and 'Source H' parameters are inputs and the 'Source E' and 'Source F' parameters are multipliers. If the 'Source E' and 'Source F' parameters specific values of '1', the 'Destination' parameter will be the sum of the 'Source G' and 'Source H' parameters.

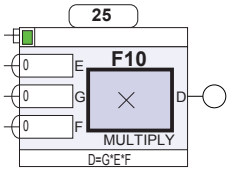
Note: If parameters are to be ignored, set them appropriately to '0' or '1'. In IQSET this is achieved by selecting the **Not Used** *check box, and the input is shown with a null value on the module symbol; however, the IQ4 treats such a parameter as having a value of '0' (i.e if the 'Source E' and 'Source F' parameters were both not used, the 'Destination' parameter would be '0' regardless of the values of the 'Source G' and 'Source H' parameters.*

14.3.8.2 Type 8 (Adder/Scaler) Function Module Parameters

Type 8 (Adder/Scaler) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G
Source H	Specifies H in the calculation.	Full analogue	0	I	E

14.3.9 Type 9 (Multiplier)



Type 9 (Multiplier) Function modules provide the facility for multiplying a value by one or two other values. It calculates the 'Destination' parameter as the 'Source G' parameter multiplied by the 'Source E' and 'Source F' parameters. If one of the parameter is specified as 1, the 'Destination' parameter will be the product of the other parameters.

14.3.9.1 Formula

D = G x E x F

Where:

D = Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter
G = 'Source G' parameter

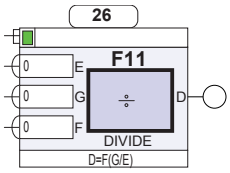
Note: Set unused parameters to '1'.

14.3.9.2 Type 9 (Multiplier) Function Module Parameters

Type 9 (Multiplier) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.10 Type 10 (Divider)



Type 10 (Divider) Function modules provide the facility for dividing a value by another value and multiplying the result by a third value. It calculates its the 'Destination' parameter by dividing the 'Source G' parameter by the 'Source E' parameter and multiplying the result by the 'Source F' parameter.

Note: The 'Destination' parameter is limited it to the largest number it can handle (i.e. to cater for divide by 0).

14.3.10.1 Formula

$$D = \frac{GxF}{E}$$

Where:

D = 'Destination' parameter
E = 'Source E' parameter (divisor)
F = 'Source F' parameter
G = 'Source G' parameter

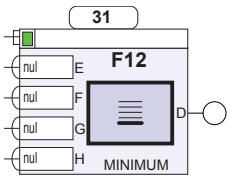
Note: Set the 'Source F' to 1 to produce G/E.

14.3.10.2 Type 10 (Divider) Function Module Parameters

Type 10 (Divider) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.11 Type 11 (Minimum)



Type 11 (Minimum) Function modules provide the facility for selecting the minimum value from 2, 3 or 4 input signals. They calculate the 'Destination' parameter as the minimum value of two, three, or four parameters 'Source E', 'Source F', 'Source G', and 'Source H'. If one of the parameters is a specific value, the module output will have a maximum limit of that value.

14.3.11.1 Formula

D = minimum value of E or F or G or H.

Where:

D = 'Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter
G = 'Source G' parameter
H = 'Source H' parameter

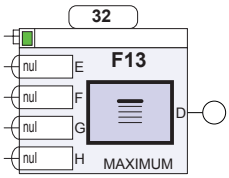
Note: Unused parameters must be left blank. In IQSET this is achieved by selecting the **Not Used** *check box, and the parameter is shown with a null value on the module symbol.*

14.3.11.2 Type 11 (Minimum) Function Module Parameters

Type 11 (Minimum) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G
Source H	Specifies H in the calculation.	Full analogue	0	I	H

14.3.12 Type 12 (Maximum)



Type 12 (Maximum) Function modules provide the facility for selecting the maximum value from 2, 3 or 4 parameters. They calculate 'Destination' parameter as the maximum value from two, three, or four parameters 'Source E', 'Source F', 'Source G', and 'Source H'. If one of the parameters is a specific value, the 'Destination' parameter will have a minimum limit of that value.

14.3.12.1 Formula

D = maximum value of E or F or G or H.

Where

D = 'Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter
G = 'Source G' parameter
H = 'Source H' parameter

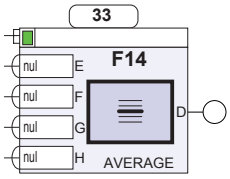
Note: Unused parameters must be left blank. In IQSET this is achieved by selecting the **Not Used** *check box, and the parameters is shown with a null value on the module symbol.*

14.3.12.2 Type 12 (Maximum) Function Module Parameters

Type 12 (Maximum) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G
Source H	Specifies H in the calculation.	Full analogue	0	I	E

14.3.13 Type 13 (Average)



Type 13 (Average) Function modules provide the facility for calculating the average value of two, three, or four parameters ‘Source E’, ‘Source F’, ‘Source G’, and ‘Source H’.

14.3.13.1 Formula

$$D = \frac{\text{Sum of used input values}}{\text{Number of used inputs}}$$

Where

D = ‘Destination’ parameter
E = ‘Source E’ parameter
F = ‘Source F’ parameter
G = ‘Source G’ parameter
H = ‘Source H’ parameter

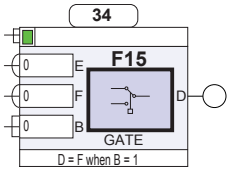
Note: Unused parameters must be left blank. In IQSET this is achieved by selecting the **Not Used** *check box, and the parameters is shown with a null value on the module symbol.*

14.3.13.2 Type 13 (Average) Function Module Parameters

Type 13 (Average) Function modules include the ‘Type’ parameter common to all function modules - see **“Function Module Common Parameters” on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G
Source H	Specifies H in the calculation.	Full analogue	0	I	H

14.3.14 Type 14 (Analogue Gate)



Type 14 (Analogue Gate) Function modules allow selection between two analogue parameters depending on the state of a digital control bit. They select the ‘Destination’ parameter from either the ‘Source E’ or ‘Source F’ parameter depending on the state of the ‘Source B’ parameter.

14.3.14.1 Formula

D = F when B = 1
D = E when B = 0

Where:

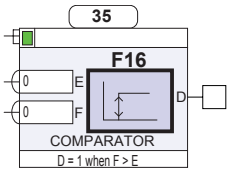
D = ‘Destination’ parameter
E = ‘Source E’ parameter
F = ‘Source F’ parameter
B = ‘Source B’ parameter

14.3.14.2 Type 14 (Analogue Gate) Function Module Parameters

Type 14 (Analogue Gate) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source B	State of control bit. Can be connected to a digital output or set to a specific value.	0 or 1	0	I	B

14.3.15 Type 15 (Comparator)



Type 15 (Comparator) Function module allows two parameters, the 'Source E' and 'Source F' parameters, to be compared. They calculate the 'Destination' parameter as '0' or 1' by comparing the 'Source E' and 'Source F' parameters. When F>E, D = 1, and when F = <E, D = 0.

14.3.15.1 Formula

D = 1 when F > E.
D = 0 when F = or < E.

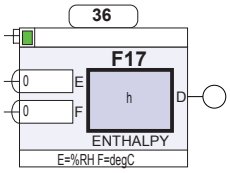
Where:
D = 'Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter

14.3.15.2 Type 15 (Comparator) Function Module Parameters

Type 15 (Comparator) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	0 or 1	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F

14.3.16 Type 16 (Enthalpy)



Type 16 (Enthalpy) Function modules calculate the 'Destination' parameter as the enthalpy value in kJ/kg of the 'Source E' and 'Source F' parameters. The 'Source E' parameter must be relative humidity and 'Source F' parameter temperature in deg C.

The specific enthalpy is a measure of the total heat content per unit mass of a gas. It can be used in an air conditioning system to calculate the positions of dampers so that the least energy is used to control the temperature and humidity of a space (i.e. to decide whether to condition the outside air or the re-circulated air).

14.3.16.1 Formula

$$D = E \times 10^{(0.0275F - 1)} + F$$

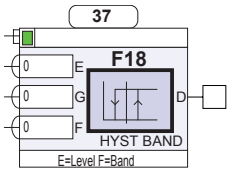
Where:
D = 'Destination' parameter (enthalpy kJ/kg)
E = 'Source E' parameter (RH%)
F = 'Source F' parameter (Temp deg C)

14.3.16.2 Type 16 (Enthalpy) Function Module Parameters

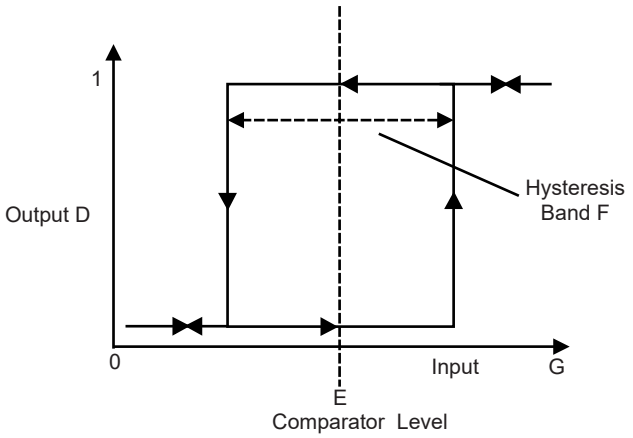
Type 16 (Enthalpy) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E (%RH) in the calculation.	Full analogue	0	I	E
Source F	Specifies F (temperature in deg C) in the calculation.	Full analogue	0	I	F

14.3.17 Type 17 (Hysteresis Band)



Type 17 (Hysteresis Band) Function modules set the 'Destination' parameter to '1' if the 'Source G' parameter rises above the value of E+F/2 and then sets the 'Destination' parameter to '0' if the 'Source G' parameter falls below E-F/2. This enables the state of a digital node to be controlled from an analogue input level. This is illustrated in the graph.



As the value of the 'Source G' parameter increases the 'Destination' parameter stays at '0' until 'Source G' parameter exceeds E+F/2 when the 'Destination' parameter = '1'. As the 'Source G' parameter drops the 'Destination' parameter stays at '1' until the input falls below E-F/2 where the 'Destination' parameter = '0'. The 'Source E' parameter defines the threshold level and the 'Source F' parameter defines the band around the level within which input changes have no effect.

14.3.17.1 Formula

D = 0 if G is below the level set by (E-F/2).
D = 1 if G is above the level set by (E + F/2).
D = unchanged, if G is in the band between levels (E-F/2) to (E+F/2).

Where:
D = 'Destination' parameter
E = 'Source E' parameter (level)
F = 'Source F' parameter (band)
G = 'Source G' parameter

When the module is first serviced (after a strategy download), if the 'Source E' parameter is within the band, then the 'Destination' parameter is '0'.

If the band is made negative so that the OFF level is greater than the ON level, the module becomes a 2 second oscillator when the input is between the levels.

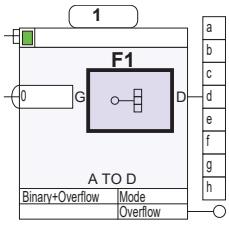
Note If the band is set to '0' then noise on the input may cause the drive to switch ON and OFF at random.

14.3.17.2 Type 17 (Hysteresis Band) Function Module Parameters

Type 17 (Hysteresis Band) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	0 or 1	0	O	D
Source E	Specifies E (level) in the calculation.	Full analogue	0	I	E
Source F	Specifies F (band) in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.18 Type 18 (Analogue to Digital Converter)



Type 18 (Analogue to Digital Converter) Function modules calculate the 'Destination' parameter by converting the value of an analogue parameter, the 'Source G' parameter to a binary value.

From v4.20 firmware or greater the way in which the binary value is calculated can be determined using the 'Mode' parameter which can be set to one of four mode:

Binary
Binary + Overflow
Bit Set
Cascade

Previous versions of firmware only support Binary mode.

Binary

Binary mode enables the conversion of 8 bit numbers (single byte). The output (D) is calculated as below:

D = Binary representation of G as 8 digital outputs

				Bit	76543210
E.g. if	G = -10	D = 0	Output byte	MSB	00000000 LSB
	G = 4	D = 4			00000100
	G = 27	D = 27			00011011
	G = 211	D = 211			11010011
	G = 255	D = 255			11111111
	G = 350	D = 255			11111111

The value of the 'Destination' parameter is limited the range 0 to 255, and is rounded to the nearest integer up (e.g. G=4.1 D=5). The 'Overflow' parameter (O) is always '0.0'.

Note: Binary mode is the only mode in pre v4.20 firmware. When upgrading strategies from pre v4.20 firmware the Type 18 (Analogue to Digital Converter) Function modules will be set to this mode to maintain compatibility.

Binary + Overflow

Binary + Overflow mode enables the conversion of multi-byte values by chaining 8 bit conversions The output (D) is calculated as the binary representation of G as 8 digital outputs in the same way as for binary mode with the addition of an overflow output (O) calculated as:

O= TRUNC (G DIV 256)

This output can be connected to the 'Source' parameter (G) of another Type 18 (Analogue to Digital Converter) Function module operating in Binary + Overflow mode - see **"Binary + Overflow Mode" on page 260**.

Note: The value of 'G' is truncated (eg G=4.1 D=4).

Bit Set

Bit set mode sets one output bit based on the value of the 'Source G' parameter. E.g. if the input is greater than or equal to 4 and less than 5 then bit 3 (d) is set. All other output bits are cleared.

If the value of the 'Source G' parameter in the range 1 to 8 the output bits are set. as below

h								1
g							1	
f						1		
e					1			
d				1				
c			1					
b		1						
a	1							
G	1	2	3	4	5	6	7	8

If the value of the 'Source G' parameter out of the range 1 to 8 all the output bits are set to 0.

Note: The value of 'G' is truncated (e.g. G=4.1 D=4).

The 'Overflow' parameter (O) is calculated as:

O = min (0,TRUNC(G – 8))

Cascade

Cascade mode sets the output bits as the input value increases.

If the value of the 'Source G' parameter in the range 1 to 8 the output bits are set. as below

h								1
g							1	1
f						1	1	1
e					1	1	1	1
d				1	1	1	1	1
c			1	1	1	1	1	1
b		1	1	1	1	1	1	1
a	1	1	1	1	1	1	1	1
G	1	2	3	4	5	6	7	8

If the value of the 'Source G' parameter out of the range 1 to 8 all the output bits are set to 1.

Note: The value of 'G' is truncated (eg G=4.1 D=4).

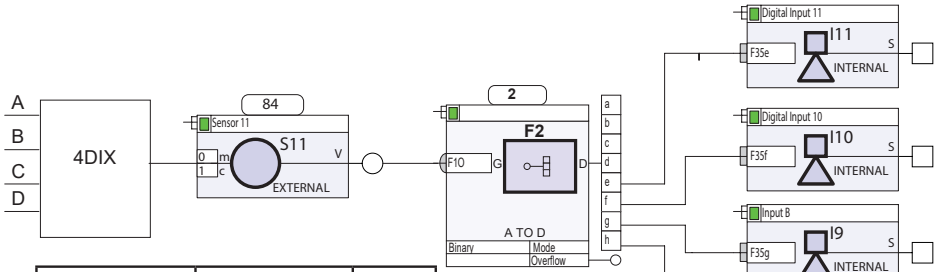
The 'Overflow' parameter (O) is calculated as:

O = min (0,TRUNC(G – 8))

14.3.18.1 Strategy Examples

14.3.18.1.1 Binary Mode

When operating in binary mode the Type 18 (Analogue to Digital Converter) Function module can be used in conjunction with a Digital Input Expander (4DIX) to expand the number of digital inputs that the IQ4 can monitor.

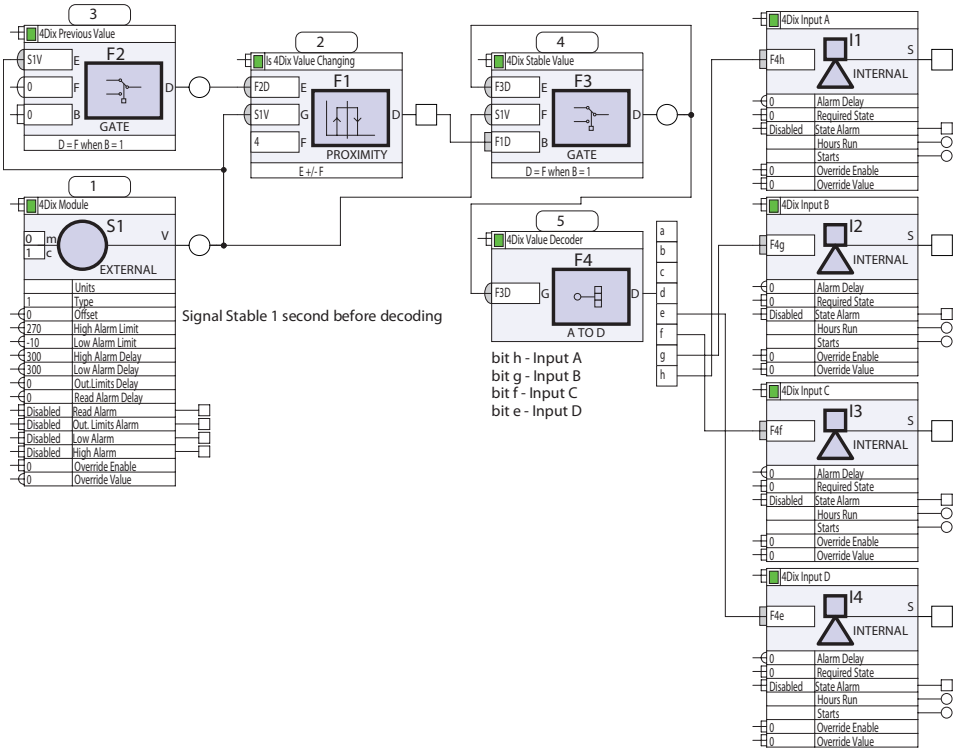


4DIX input	*Dig output*	*Bit*
A	h	7
B	g	6
C	f	5
D	e	4

In this example a single analogue input channel is used to handle four digital inputs. The binary number produced by the four inputs (A to D) produces a combined analogue signal which is input to the IQ via channel 1, scaled by sensor 11 and decoded by the Type 18 (Analogue to Digital Converter) Function module which produces 8 digital outputs. The four most significant outputs (bits 7 to 4, labelled h to e) correspond to the 4DIX inputs A to D. These digital outputs can be connected to internal digital inputs as shown, and can then be used to generate alarms.

Note: When used with a 4DIX, the sensor type must be calibrated as defined by the 4DIX/24VAC Four Digital Input Expander Module (24 Vac/dc) Data Sheet (TA200650).

Due to the response of the IQ4 sensor module it is possible for the A to D to sample the sensor output value while the 4DIX is in transition between states, resulting in incorrect outputs. In order to avoid this, the strategy has to be modified to eliminate the transient values. IQSET's 4DIX strategy block is shown in the diagram below. This strategy compares the sensor's current value with its previous value and if the value is not changing it is passed to the Type 18 (Analogue to Digital Converter) Function module where it is decoded to set internal inputs 1 to 4 to the same status as the physical input to the 4DIX. This filters unwanted changes due to the response of the input.



On sequence step 1, the S1 processes the value from the 4DIX.

On sequence step 2, F1 checks to see if the sensor's value is changing, by comparing the sensor's current value with its previous value (received from F2). If the two values are within +/- 4 of each other the module decides that the sensor value is not changing and sets its output (D) to '1'. If the two values are more than +/-4 apart the module decides that the sensor value is changing and sets its output (D) to '0'.

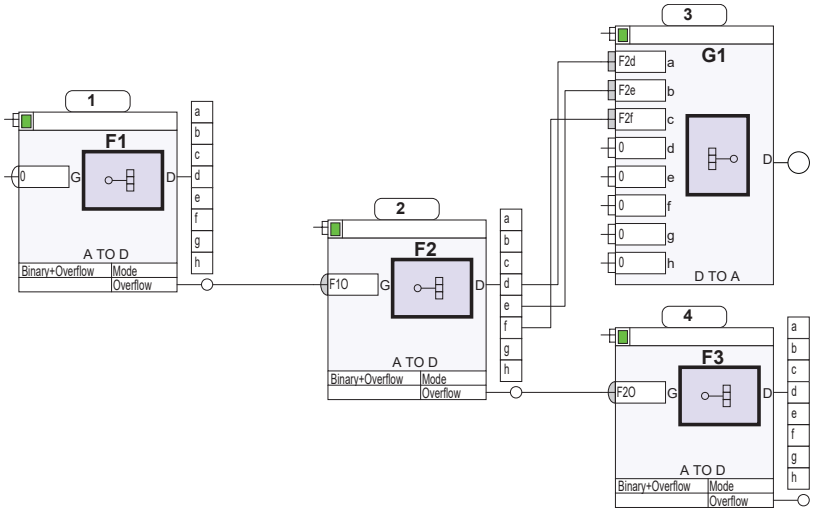
On sequence step 3, the current value of the S1 is then input by F2 which sets its output value to the new value so that it can be used in the next comparison.

On sequence step 4, F3 monitors the output of F1 along with the S1's current value. If the output of F1 is '1' the output of F3 is set to the S1's current value. If the output of F1 is '0' output of F3 is remains unchanged.

On sequence step 5, the F4 takes the output of F3 and outputs to 4 internal digital input modules as in the original strategy above.

14.3.18.1.2 Binary + Overflow Mode

The strategy below uses to Type 18 (Analogue to Digital Converter) Function modules in Binary + Overflow mode chained together to convert a 24 bit number containing mixed digital and analogue values.



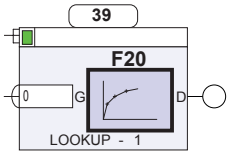
14.3.18.2 Type 18 (Analogue to Digital Converter) Function Module Parameters

Type 18 (Analogue to Digital Converter) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Bit 0	Bit 0 of the output byte.	0 or 1	0	O	a
Bit 1	Bit 1 of the output byte.	0 or 1	0	O	b
Bit 2	Bit 2 of the output byte.	0 or 1	0	O	c
Bit 3	Bit 3 of the output byte.	0 or 1	0	O	d
Bit 4	Bit 4 of the output byte.	0 or 1	0	O	e
Bit 5	Bit 5 of the output byte.	0 or 1	0	O	f
Bit 6	Bit 6 of the output byte.	0 or 1	0	O	g
Bit 7	Bit 7 of the output byte.	0 or 1	0	O	h
Mode	v4.20 or greater. The mode in which the module is operating.	0, 1, 2, 3 0 = Binary 1 = Binary + Overflow 2 = Bit set 3 = Cascade	1	N	M

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Overflow	v4.20 or greater. The value left over after the conversion, can be passed to another module, calculated as below: Binary + Overflow O = TRUNC (G DIV 256) Bit Set and Cascade modes O = min (0,TRUNC(G – 8)) : i.e. overflow cannot go negative. Binary O = 0.0	Full analogue	0	O	O
Source G	Specifies G (analogue value to be converted) in the calculation.	Full analogue	0	I	G

14.3.19 Type 19 (Lookup)



Type 19 (Lookup) Function modules scale the 'Source G' parameter according to an input/output characteristic table. The input/output characteristic table may have between 2 and 20 points which must be entered in the ascending order of the input parameter. The table is set up in a Sensor Type module with its 'Input Type' parameter set to 'Analogue Value' - see **"Sensor Type Module Parameters" on page 478**.

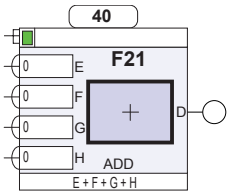
The output value of the module is calculated by linearly interpolating between the relevant 2 points. If the input value falls below or above the table range then the output value is calculated from the extrapolated straight line between the first two points or the last two points respectively. If the Sensor Type module's 'Input Type' parameter is not set to 'Analogue Value', the 'Destination' parameter is set to '0'.

14.3.19.1 Type 19 (Lookup) Function Module Parameters

Type 19 (Lookup) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Scaling Type	Sensor Type module the module is using for reference.	1 to 99	0	N	S
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.20 Type 20 (Adder)



Type 20 (Adder) Function modules calculate the 'Destination' parameter as the sum of four parameters, 'Source E', 'Source F', 'Source G' and 'Source H'.

14.3.20.1 Formula

$$D = E + F + G + H$$

Where:

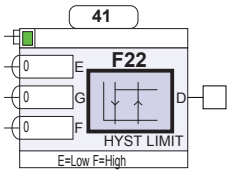
D = 'Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter
G = 'Source G' parameter
H = 'Source H' parameter

14.3.20.2 Type 20 (Adder) Function Module Parameters

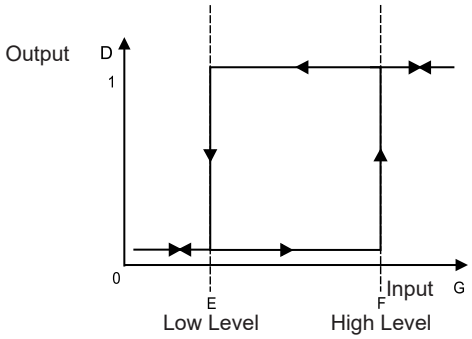
Type 20 (Adder) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G
Source H	Specifies H in the calculation.	Full analogue	0	I	H

14.3.21 Type 21 (Hysteresis Limits)



Type 21 (Hysteresis Limits) Function modules control a digital node state from an analogue input level. They set the 'Destination' parameter to '1' if the 'Source G' parameter rises above the 'Source F' parameter and then sets the 'Destination' parameter to '0' if the 'Source G' parameter falls below the 'Source E' parameter. The way the function operates can be seen in the graph below. As the 'Source G' parameter increases the 'Destination' parameter stays at '0' until the level of the 'Source F' parameter is reached when the 'Destination' parameter switches to '1'. As the 'Source G' decreases the 'Destination' parameter stays at '1' until the level of the 'Source E' parameter is reached when the 'Destination' parameter switches to '0'.



When the module is first serviced (after a strategy download), if the ' Source G' parameter is within the band the output is '0'.

Note: If the high level is set below the low level, the module becomes an oscillator with a 2 second period.

14.3.21.1 Formula

D = 0 if the G is below low limit E
D = 1 if the G is above high limit F
Dn+1 = unchanged, if the G is in the band between limits E and F.

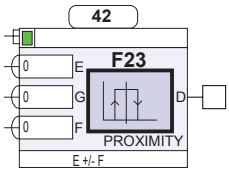
Where:
D = New 'Destination' parameter
Dn+1 = Old 'Destination' parameter
E = 'Source E' parameter (low limit)
F = 'Source F' parameter (high limit)
G = 'Source G' parameter

14.3.21.2 Type 21 (Hysteresis Limits) Function Module Parameters

Type 21 (Hysteresis Limits) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	0 or 1	0	O	D
Source E	Specifies E (low limit) in the calculation.	Full analogue	0	I	E
Source F	Specifies F (high limit) in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.22 Type 22 (Proximity)



Type 22 (Proximity) Function modules check whether an analogue value is within a defined range of another value i.e. they check whether two analogue values are approximately equal to each other. It sets its the 'Destination' parameter to '1' if G > = E-F or < = E+F.

14.3.22.1 Formula

D = 1 if |E - G| < = F
D = 0 if |E - G| > F

Where:
D = 'Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter (maximum difference - proximity)
G = 'Source G' parameter

|E - G| is the unsigned (absolute value or modulus of) difference between the 'Source E' and 'Source G' parameters.

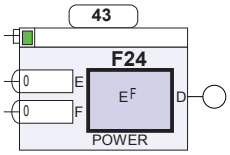
Note: The proximity, 'Source F' parameter, should not be set less than '0.1', otherwise it is be difficult to check the operation of the module as the web page's value resolution is limited to two decimal places.

14.3.22.2 Type 22 (Proximity) Function Module Parameters

Type 22 (Proximity) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	0 or 1	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F (proximity) in the calculation.	Full analogue	0	I	F
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.23 Type 23 (Power)



Type 23 (Power) Function modules will raise an parameter to the power defined by another parameter and deliver it to the 'Destination' parameter. The 'Destination' parameter is calculated as the 'Source E' parameter to the power of the 'Source F' parameter. The 'Source E' parameter is multiplied by itself F times. e.g. if F = 2, then D = E^2.

The 'Source F' parameter may be a non-integer, e.g. 2.56. This enables an anti-log function to be performed. E.g. if $\log_{10}x = 3.44$, then we can find x by:

$x = \text{antilog}(3.44) = 10^{3.44}$

So using the Type 23 (Power) Function module if the 'Source E' parameter = 10 and the 'Source F' parameter = 3.44, then:

$D = 10^{3.44}$ = antilog (3.44) = 2754

i.e. x = 2754

14.3.23.1 Formula

$D = E^F$

Where:
D = 'Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter (power)

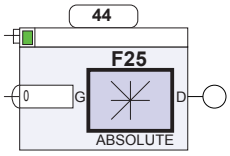
Note: If E = F = 0 then D = 1.

14.3.23.2 Type 23 (Power) Function Module Parameters

Type 23 (Power) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source E	Specifies E in the calculation.	Full analogue	0	I	E
Source F	Specifies F (power) in the calculation.	Full analogue	0	I	F

14.3.24 Type 24 (Absolute)



Type 24 (Absolute) Function modules calculate the absolute value of their input, 'Source G'. i.e. will make a negative number positive by removing the negative sign.

E.g. if G = -6, D = 6 and if G = 6, D = 6

14.3.24.1 Formula

D = |G| (absolute value of G)

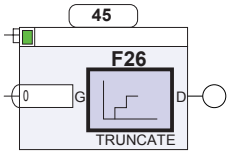
Where:
D = 'Destination' parameter
G = 'Source G' parameter

14.3.24.2 Type 24 (Absolute) Function Module Parameters

Type 24 (Absolute) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.25 Type 25 (Truncate)



Type 25 (Truncate) Function modules take the value of the 'Source G' parameter, remove any digits after the decimal point and write it to the 'Destination' parameter.

E.g. if G = 23.468, D = 23, or if G = -23.468, D = -23

14.3.25.1 Formula

D = integer value of G

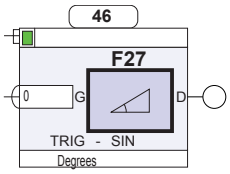
Where:
D = 'Destination' parameter
G = 'Source G' parameter

14.3.25.2 Type 25 (Truncate) Function Module Parameters

Type 25 (Truncate) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.26 Type 26 (Trigonometric)



Type 26 (Trigonometric) Function modules calculate the 'Destination' parameter as the sine, cosine, or tangent of the 'Source G' parameter (in degrees, or radians). The 'Sub Type' parameter selects the trigonometric function from either sine, cosine, or tangent. The 'Angle Type' parameter sets the units of angle to be either degrees or radians.

E.g. if 'Source G' = 90, angle type set to 'degrees', and sub type set to 'sine', the 'Destination' parameter = 1 (i.e. Sine 90°).

Note: Tan 90 is calculated as a very large number (10^{20}).

14.3.26.1 Formula

D = TRIG (G)

Where:
D = 'Destination' parameter
G = 'Source G' parameter

TRIG is a trigonometric function (either sine, cosine, or tangent).

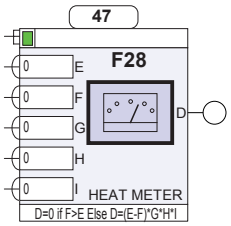
14.3.26.2 Type 26 (Trigonometric) Function Module Parameters

Type 26 (Trigonometric) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Angle Type	The units of angle used.	0 or 1 0 = Degrees 1 = Radians	0	N	R
Destination	The value of the output.	Full analogue	0	O	D

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Source G	Specifies G in the calculation.	Full analogue	0	I	G
Sub Type	The trigonometric function used by the module.	0, 1, 2 0 = sine 1 = cosine 2 = tangent	0	N	S

14.3.27 Type 27 (Heat Meter)



Type 27 (Heat Meter) Function modules enable the power used by the source to be calculated (e.g. kW). They are able to calculate the current power of heating or cooling plant.

14.3.27.1 Formula

If F > = E
D = 0
Else
D = (E - F) x G x H x I

Where
D = 'Destination' parameter
E = 'Source E' parameter
F = 'Source F' parameter
G = 'Source G' parameter
H = 'Source H' parameter
I = 'Source 1' parameter

Note: Unused parameters must be left blank. In IQSET this is achieved by selecting the **Not Used** *check box, and the parameter is shown with a null value on the module symbol.*

14.3.27.2 Examples

Note: In the following equations K = temperature degrees Kelvin (incrementally equivalent to °C)

Virtual heat metering where flow and temperatures are available

Power = (Temperature difference between supply and return) x (medium's specific heat capacity) x (mass flow)

For Heating Plant

Power kW (D) = (SupplyTemp °C (E) - Return Temp °C (F)) x (Volume Flow dm^3/s (G)) x (Density kg/dm^3 (H)) x (Specific Heat Capacity kJ/(kg.K) (I))

For Cooling Plant

Power kW (D) = (Return Temp °C (E) - SupplyTemp °C (F)) x (Volume Flow dm^3/s (G)) x (Density kg/dm^3 (H)) x (Specific Heat Capacity kJ/(kg.K) (I))

Estimating current power of a heat exchanger:

Power = (Temperature difference across Heat exchanger) x (Coil Duty) x (Valve position) /100

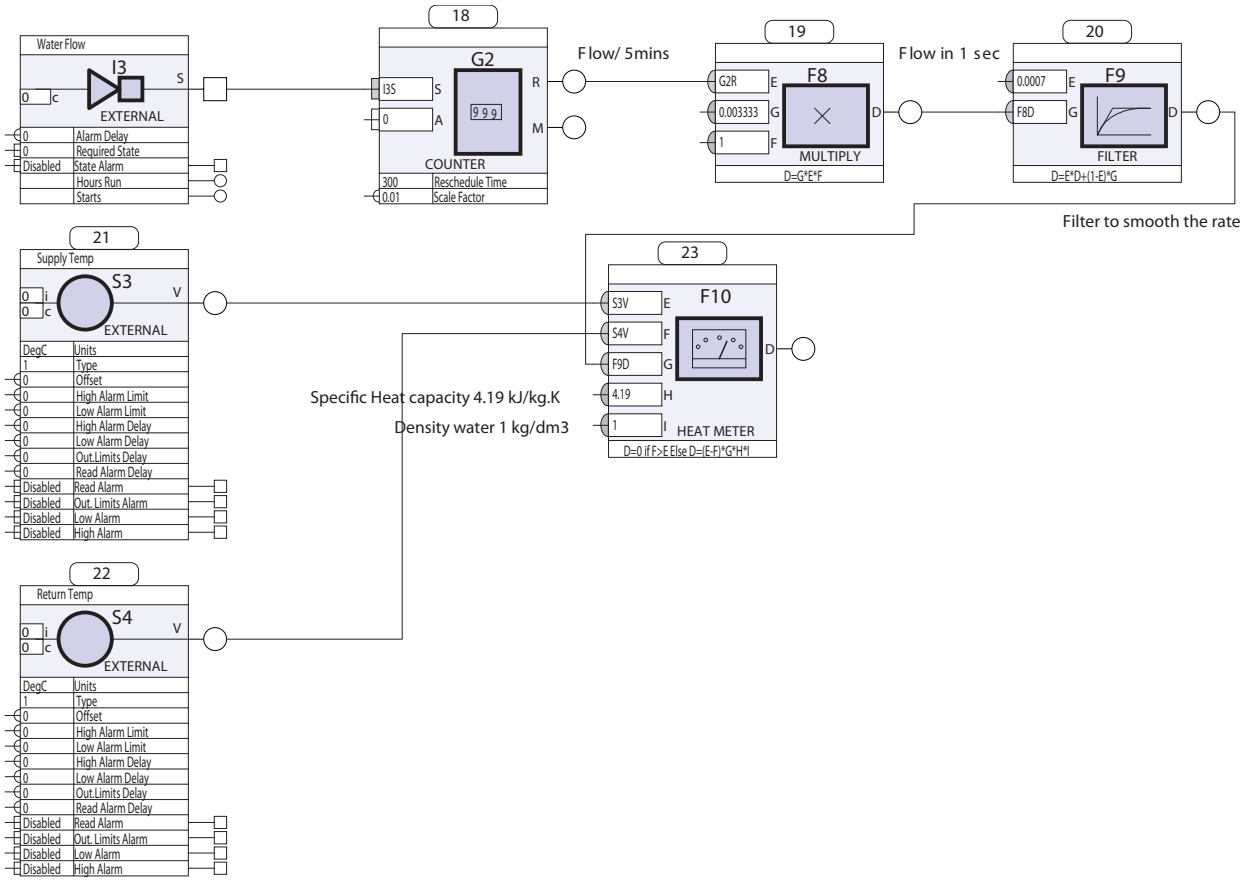
Heating

Power kW (D) = (Water Supply Temp °C (E) - Air On Temp °C (F)) x (Coil Duty kW/K (G)) x (Valve position % (H)) x (0.01 constant) (I))

Cooling

Power kW (D) = (Air On Temp °C (E) - Water Supply Temp °C (F)) x (Coil Duty kW/K (G)) x (Valve position % (H)) x (0.01 constant) (I))

The strategy below shows the heat meter module being used to calculate power (kW) from a water flow pulse input and the two temperature measurements.



The power value would then have to be integrated to produce energy consumption (kWh); this can be done using the Type 28 (Integrator) Function module - see **"Type 28 (Integrator)" on page 267**.

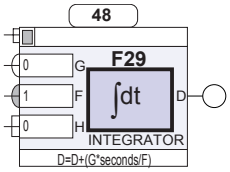
Note: Type 27 (Heat Meter) Function module functionality can also be used for other purposes (see degree day example below in **"Degree Days" on page 269**.

14.3.27.3 Type 27 (Heat Meter) Function Module Parameters

Type 27 (Heat Meter) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Input I	The value of input I.	Full analogue	0	I	I
Source E	Specifies E (upper temperature) in the calculation.	Full analogue	0	I	E
Source F	Specifies F in the calculation.	Full analogue	0	I	F
Source H	Specifies H in the calculation.	Full analogue	0	I	H
Source G	Specifies G in the calculation.	Full analogue	0	I	G

14.3.28 Type 28 (Integrator)



Type 28 (Integrator) Function modules calculate the time integral of an analogue value. A scale factor is included and the destination parameter can be reset. If a Plot module is logging the module's output it should be triggered by the same pulse as the module's 'Reset' parameter. Increases in the output that are smaller than about 1 part in 10^7 may not be shown on the web page but will still be accumulated accurately.

14.3.28.1 Formula

IF

H = 1 THEN D = 0

ELSE

$D_{n+1} = D_n + (G \times t)/F$

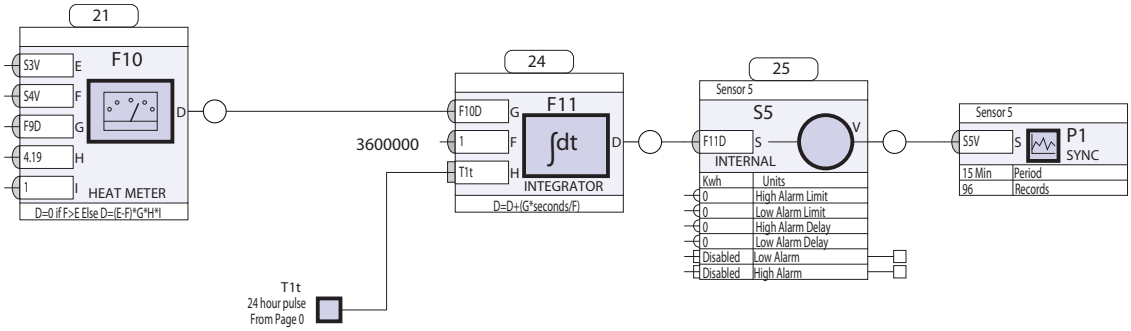
Where:

D_{n+1} = 'Destination' parameter (new output)
D_n = 'Destination' parameter (old output)
G = 'Source G' parameter
F = 'Source F' parameter (scale factor)
H = 'Source H' parameter
t = time between module servicing (secs) - automatically calculated

14.3.28.2 Examples

Integrate the power output of the heat meter module

The Type 28 (Integrator) Function module can be used as shown below to integrate the power output of the heat meter module (e.g kW) to produce the energy consumption (e.g kWh).



The Type 28 (Integrator) Function modules's scale factor, 'Source F' parameter, must be correctly set to ensure the required output. In the example the output from the heat meter module is in kW. In order to provide the output of the Type 28 (Integrator) Function module in MWh a scaling factor of 3600000 is required. The table below provides some examples of how the scale factor should be set to achieve a particular result.

Source	*Output*	*Scale factor*
Power in kW	in kWh	3600
Power in kW	in MWh	3600000
Flow per second (in litres/second)	in l (litres)	1

The Plot module's period must be the same as the time pulse used for the reset (e.g. every 24 hours in the example above).

Increases in the output smaller than 1 part in 10^7 may not be shown on the web page, but will still be accumulated accurately.

Degree Days

Degree days are used to normalise the energy consumption of a building with respect to the prevailing temperatures:

Degree days = ΣD.dΘ

Where

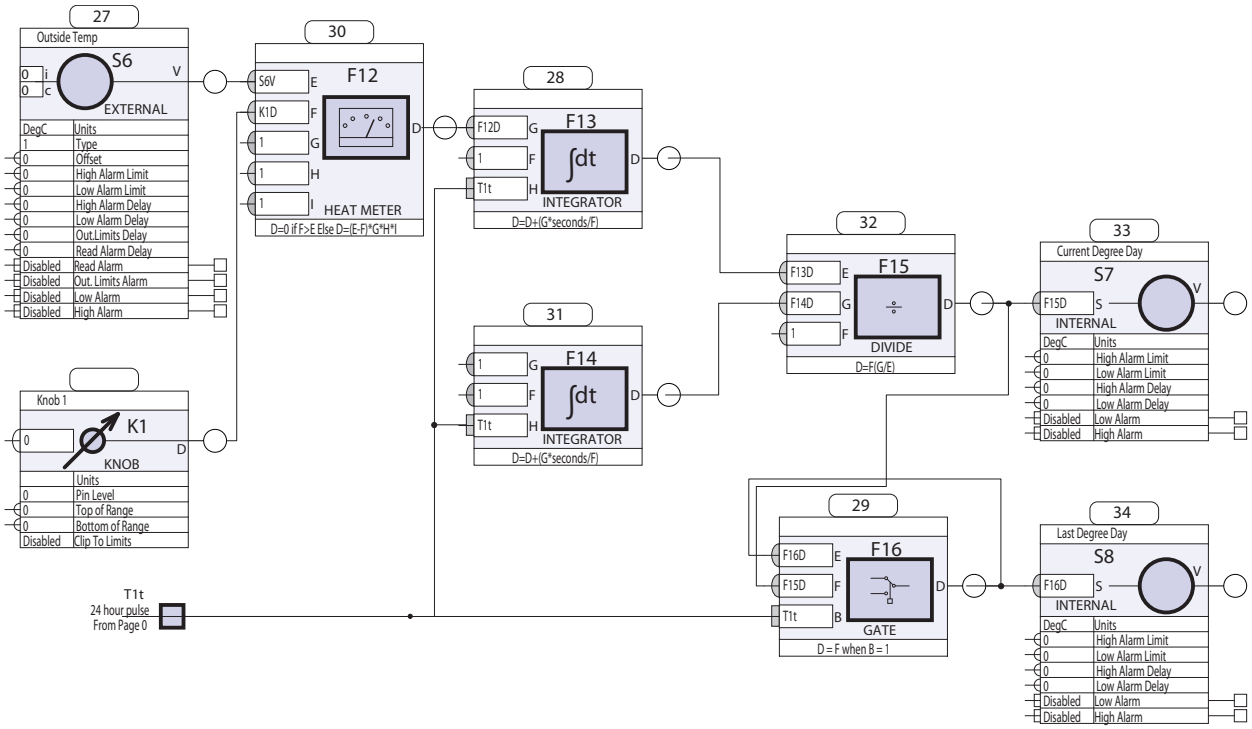
dΘ = Θdatum - Θoutside

and dΘ = >0, Θ = temperature

and D = number of days = t/(3600 x 24) where t is in secs

Degree days = Σ(t/3600 x 24).(Θdatum - Θoutside)

In the example strategy below the Type 28 (Integrator) Function module (F14) is used to calculate the amount by which the Outside Air Temperature is below the degree day datum (set on knob 1). F13 is used as a subtractor but it also will register '0' when the outside air temperature exceeds the degree day datum. F13 will accumulate the sample from F12, and F14 will accumulate the number of samples taken. F15 divides the accumulated value by the number of samples to find the average in the day so far. At midnight the average degree day for the day is latched into F16, and the integrator is reset.

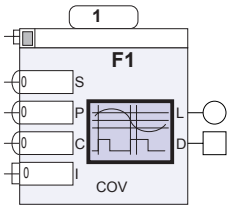


14.3.28.3 Type 28 (Integrator) Function Module Parameters

Type 28 (Integrator) Function modules include the 'Type' parameter common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The value of the output.	Full analogue	0	O	D
Source F	Specifies F (scale factor) in the calculation, the scale factor used to scale the total value before it is output.	Full analogue	0	I	F
Source G	Specifies G (value which is to be accumulated)in the calculation.	Full analogue	0	I	G
Source H	Resets the output D to '0'.	0 or 1 1 = Reset	0	I	H

14.3.29 Type 29 (Change of Value)



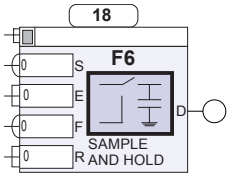
v3.30 or greater firmware only. Type 29 (Change of Value) Function modules detect changes to an analogue value. They pulse the 'Destination D' parameter for the specified pulse duration when the value of the 'Source' parameter differs from the 'Last Value' parameter by more than the 'Significant change' parameter. The 'Last Value' parameter is set when a pulse is generated. The length of the pulse can be specified using the 'Pulse Duration' parameter.

14.3.29.1 Type 29 (Change of Value) Function Module Parameters

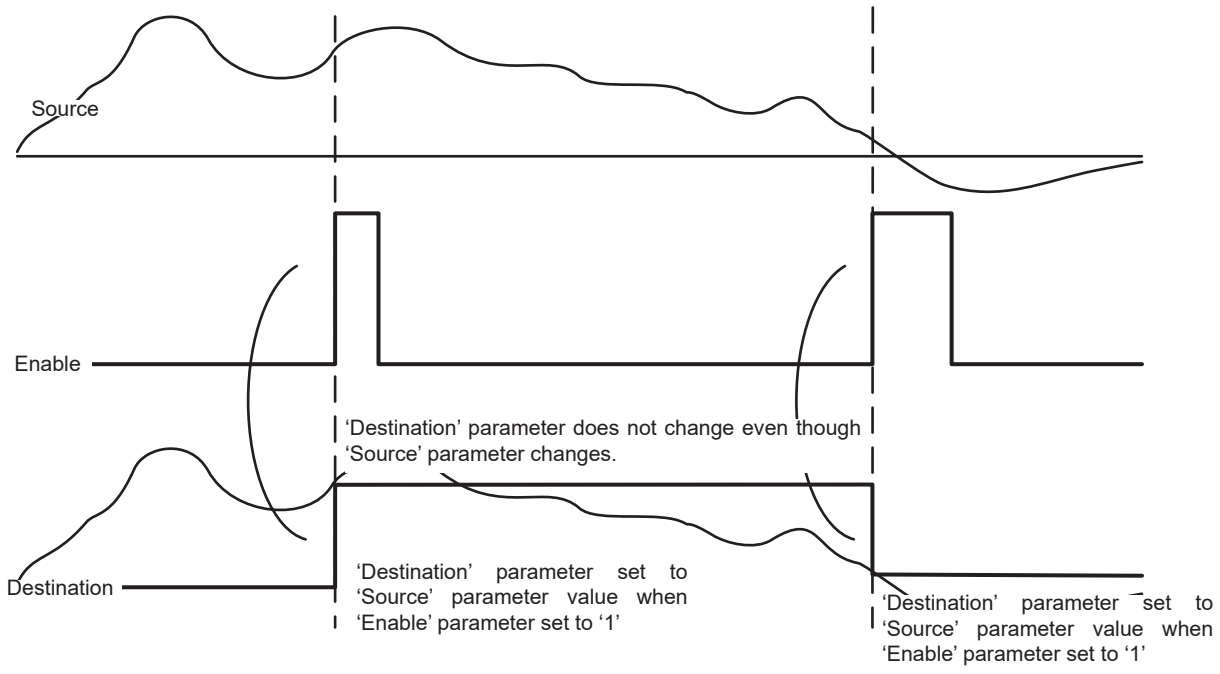
Type 29 (Change of Value) Function modules include the parameters common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination D	The module's output.	0 or 1	0	O	D
Invert Output	When set to '1' then the normal status of the 'Destination D' parameter is '1' and the output will pulse to '0'.	0 or 1 1 = Inverted 2 = Normal	0	I	I
Last Value	The value of the 'Source' parameter when the pulse was last generated. Set to the 'Source' parameter on power up.	Full Analogue	0	N	L
Pulse Duration	The duration of the output pulse, in seconds. If this is set to '0' then the pulse will last for one complete cycle of the sequence table and in the case of event based execution the OFF event will follow the ON event directly.	Full Analogue	0	N	P
Significant change	The amount the value must increase or decrease before the output is pulsed.	+ve Analogue	0	N	C
Source	The value that is to be monitored for changes.	Full Analogue	0	I	S

14.3.30 Type 30 (Sample and Hold)



v3.30 or greater firmware only. Type 30 (Sample and Hold) Function modules allow an analogue value to be retained. The modules set the 'Destination' parameter to the value of the 'Source' parameter when the 'Enable' parameter is set to '1'. The 'Destination' parameter will remain at that value until the 'Enable' parameter is set to '1' again - see diagram below:



The Destination' parameter can be set to the value of the 'Default Value' parameter by setting the 'Reset' parameter to '1'.

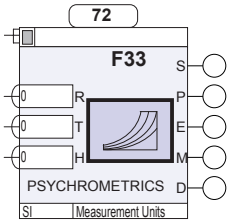
The module's status is retained when the IQ4 is powered down and when the IQ4 restarts it the module will adopt the status at power down.

14.3.30.1 Type 30 (Sample and Hold) Function Module Parameters

Type 30 (Sample and Hold) Function modules include the parameters common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Default Value	The value to be used when the 'Reset' parameter is set to '1'.	Full Analogue	0	N	F
Destination D	The module's output.	Full Analogue	0	O	D
Enable	When set to '1' sets the module's output 'D' to the value of the 'Source' parameter 'S'.	0 or 1	1	I	E
Reset	When set to '1' sets the module's output 'D' to the value specified by the 'Default Value' parameter.	0 or 1 0= no change 1 = Reset	0	I	R
Source	The value that is to be processed.	Full Analogue	0	I	S

14.3.31 Type 31 (Psychrometrics)



v3.30 or greater firmware only. Type 31 (Psychrometrics) Function modules calculate the saturated vapour pressure, vapour pressure, enthalpy, moisture content, and dew point based on the relative humidity, temperature, and height above sea level.

14.3.31.1 Units

The module provides the results of its calculations in either SI or imperial units depending on the value of the 'Units' parameter.

If the 'Units' parameter is set to '0' (SI units) all the module's outputs will be in SI units, and the module will expect its inputs to be in SI units, and if set to '1' (imperial units) all the module's outputs will be in imperial units, and the module will expect its inputs to be in imperial units

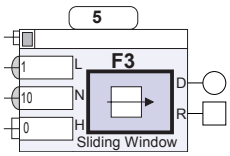
Parameter	*Units*	
	SI	*Imperial*
Saturated Vapour Pressure	kPa	psi
Vapour Pressure	kPa	psi
Enthalpy	kJ/kg	BTU/lb
Moisture content	g/kg	lb/1000lb
Dew point	degC	degF
Relative Humidity	%RH	%RH
Temperature	degC	degF
Height	meters	feet

14.3.31.2 Type 31 (Psychrometrics) Function Module Parameters

Type 31 (Psychrometrics) Function modules include the parameters common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

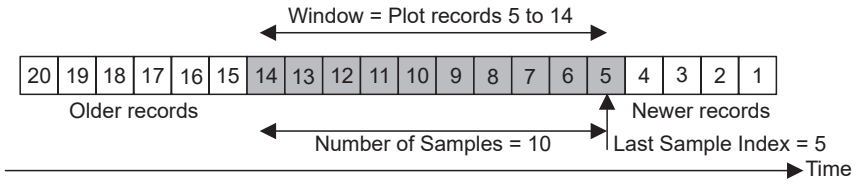
Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Dew point	The dew point (degC or degF) calculated by the module.	Full Analogue	0	O	D
Enthalpy	The Enthalpy (KJ/Kg or BTU/lb) calculated by the module.	Full Analogue	0	O	E
Height	The height above sea level (meters or feet) used by the module in its calculations.	Full Analogue	0	I	H
Moisture content	The moisture content (g/kg or lb/1000lb) calculated by the module.	Full Analogue	0	O	M
Relative Humidity	The relative humidity (% RH) used by the module in its calculations.	Full Analogue	0	I	R
Saturated Vapour Pressure	The saturated vapour pressure (kPa or psi) calculated by the module.	Full Analogue	0	O	S
Temperature	The temperature (degC or degF) used by the module in its calculations.	Full Analogue	0	I	T
Units	The type of units (SI or imperial) used by the module in its calculations and outputs	0 or 1 0 = SI 1 = Imperial	0	N	U
Vapour Pressure	The vapour pressure (kPa or psi) calculated by the module.	Full Analogue	0	O	P

14.3.32 Type 32 (Sliding Window)



v3.40 or greater firmware only. Type 32 (Sliding Window) Function modules allow either Min, Max, Sum, Average, or Population Standard Deviation calculations to be applied to a range of data in a Plot module.

The range of data is taken over a moving time window which is the number of Plot records specified by the 'Number of Samples' parameter counting back form the Plot record specified by the 'Last Sample Index' parameter:

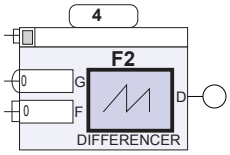


14.3.32.1 Type 32 (Sliding Window) Function Module Parameters

Type 32 (Sliding Window) Function modules include the parameters common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Data Ready	Set to '0' if there fewer samples than required stored in the Plot module, otherwise set to '1'. *Note: In* **Type 1 (Synchronised)** *Plot modules Plot records are filled in after a power outage. Therefore the Data Ready parameter will only be '0' for a period (last sample + number of samples) after the plot is first instantiated.*	0 or 1	0	O	R
Destination	The value of the output.	Double Precision	0	O	D
Function	The type of calculation that is to be performed.	0, 1, 2, 3, 4 0 – Min, 1 - Max 2 – Sum 3 – Average 4 - SD	2	N	F
Hold	If '1' the value of the Destination parameter will be retained.	0 or 1 1 = Hold 0 = No hold	0	I	H
Last Sample Index	The index of the last sample to be used in the Plot module buffer – working backwards in time. 1 is the last (most recent) in the buffer.	+Full Analogue	1	I	L
Number of Samples	The number of values from the Plot nodule that are to be used. *Note: There is a possibility of non data samples within the plot data (e.g. due to time changes, Log Status records etc). These samples will reduce the number of samples available. Therefore the 'Number of Samples' must be less than the Plot module's number of samples to allow for any non data samples stored within the plot data.*	0 to 1000	10	I	N
Plot Source	The Plot module containing the data to which the calculation is to be applied.	Full Analogue	0	I	S

14.3.33 Type 33 (Differencer)



v3.40 or greater firmware only. Type 33 (Differencer) Function modules calculate the difference between the value of the 'Source Value at Last Reset' parameter and the value of the 'Source' parameter e.g. to calculate an incremental consumption value from a cumulative consumption value. Setting the 'Reset' parameter to '1' sets the 'Source Value at Last Reset' parameter to the value of the 'Source' parameter effectively resetting the calculation.

14.3.33.1 Compensation for Input Changes

When the controller starts, e.g. after a power failure, the module checks to see if there is a big difference between the value of the 'Source' parameter and the value of the 'Last Source Value' parameter. If it finds a difference it uses the value of the 'Source Value at Last Reset' (R) parameter it in the calculation.

For large positive changes (e.g. initial acquisition of meter reading or manual adjustment to a higher value). The value of the 'Source Value at Last Reset' (R) parameter is calculated as follows:

IF
G - L > G/10
THEN
R = G

For negative changes (e.g. meter reset to 0). The value of the 'Source Value at Last Reset' (R) parameter is calculated as follows:

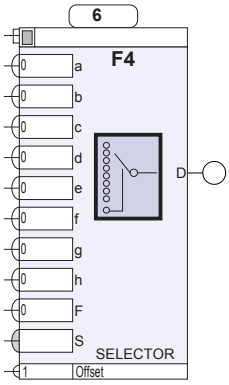
R = R - L - G

14.3.33.2 Type 33 (Differencer) Function Module Parameters

Type 33 (Differencer) Function modules include the parameters common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Difference	The difference between the value of the 'Source Value at Last Reset' parameter and the value of the 'Source' parameter.	Double Precision	0	O	D
Last Source Value	The last value of the 'Source' parameter used in the calculation. It allows compensation following input changes - see **'Compensation for Input Changes'**. *Note: It is not displayed in system browsers or on the IQ4's web pages.*	Double Precision	0	N	L
Reset	When set to '1' sets the value of the 'Source Value at Last Reset' parameter to the value of the 'Source' parameter should be linked to a parameter that changes between '0' and '1' a at a regular interval . *Note: If linked to a timing pulse* **Type 1 (Synchronised)** *Plot modules using that pulse will log the value of the 'Difference' parameter before the module is serviced. This ensures that no consumption is lost overall if the differences are totalised in an M&T package such as Trend Energy Manager.*	0 or 1	0	I	E
Source Value at Last Reset	The value of the 'Source' parameter when the 'Reset' parameter was last set to '1'.	Double Precision	0	I	R
Source	Value to be differenced.	Double Precision	0	I	G

14.3.34 Type 34 (Analogue Input Selector)



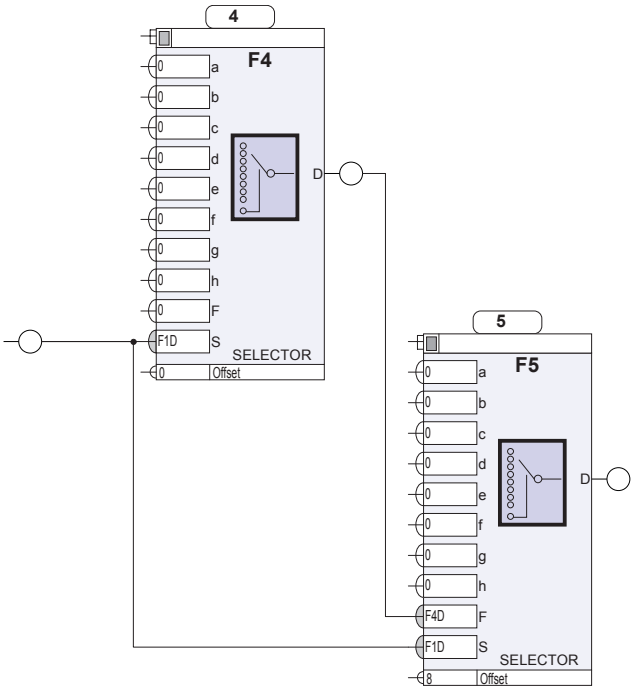
v3.40 or greater firmware only. Type 34 (Analogue Input Selector) Function modules set their 'Destination' parameter to the value of one of up to eight analogue input parameters ('Input a', 'Input b', 'Input c', 'Input d', 'Input e', 'Input f', 'Input g', or 'Input h') depending on the ROUND of the value of the 'Selector' parameter minus the value of the 'Offset' parameter providing this value is 0, 1, 2, 3,4, 5, 6 or 7.

E.g. if 'Selector' parameter = '10.4' and 'Offset' parameter = 5. This give a calculation of 10.4 - 5 = 5.4 which is rounded to '5 '

If the ROUND of the value of the 'Selector' parameter minus the value of the 'Offset' parameter is not 0, 1, 2, 3,4, 5, 6 or 7 the 'Destination' parameter is set to the value of the Default Input' parameter.

14.3.34.1 Cascade Operation

By connecting the 'destination' parameter to the 'Default Input' parameter of another Type 34 (Analogue Input Selector) Function module and setting the 'Offset' parameter of the first module to '0' and the 'Offset' parameter of the second module to '8' one analogue value can be selected from 16. If required more Type 34 (Analogue Input Selector) Function modules can be connected and their 'Offset' parameters set to '16', '24' etc allowing selection from even more analogue inputs.



14.3.34.2 Type 34 (Analogue Input Selector) Function Module Parameters

Type 34 (Analogue Input Selector) Function modules include the parameters common to all Function modules - see **"Function Module Common Parameters" on page 277**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Default Input	The value used when the result of the calculation above is not 0, 1, 2, 3,4, 5, 6 or 7.	Full Analogue	0	I	F
Destination D	The module's output.	Full Analogue	0	O	D
Input a	The value used when the result of the calculation above is '0'.	Full Analogue	0	I	a
Input b	The value used when the result of the calculation above is '1'.	Full Analogue	0	I	b

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Input c	The value used when the result of the calculation above is '2'	Full Analogue	0	I	c
Input d	The value used when the result of the calculation above is '3'	Full Analogue	0	I	d
Input e	The value used when the result of the calculation above is '4'	Full Analogue	0	I	e
Input f	The value used when the result of the calculation above is '5'	Full Analogue	0	I	f
Input g	The value used when the result of the calculation above is '6'	Full Analogue	0	I	g
Input h	The value used when the result of the calculation above is '7'	Full Analogue	0	I	h
Offset	The value subtracted from the value of the 'Selector' parameter. It allows support for base values of 0 and 1. It also supports the ability to cascade selector modules - see **'Cascade Operation'**. The default of 1 allows support for operation with BACnet multi-state objects.	Full Analogue	1	I	O
Selector	Determines which of the 8 analogue inputs is used for the 'Destination' parameter's value.	Full Analogue	0	I	S

14.4 Function Module Common Parameters

Function modules include the following parameters that are common to most modules: ‘Disable Module’, ‘Label’, ‘Page’, ‘Sequence Step’, ‘Sequence Table’, ‘X co-ordinate’ and ‘Y co-ordinate’ - see **“Common IQ4 Module Parameters” on page 165**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Type	The module type.	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34 1 = Type 1 (Filter) 2 = Type 2 (Rescale From) 3 = Type 3 (Rescale To) 4 = Type 4 (Limit At 5 = Type 5 (Limit To) 6 = Type 6 (Log) 7 = Type 7 (Square Root) 8 = Type 8 (Adder/Scaler 9 = Type 9 (Multiplier) 10 = Type 10 (Divider) 11 = Type 11 (Minimum) 12 = Type 12 (Maximum) 13 = Type 13 (Average) 14 = Type 14 (Analogue Gate) 15 = Type 15 (Comparator) 16 = Type 16 (Enthalpy) 17 = Type 17 (Hysteresis Band) 18 = Type 18 (Analogue to Digital Converter) 19 = Type 19 (Lookup) 20 = Type 20 (Adder) 21 = Type 21 (Hysteresis Limits) 22 = Type 22 (Proximity) 23 = Type 23 (Power) 24 = Type 24 (Absolute) 25 = Type 25 (Truncate) 26 = Type 26 (Trigonometric) 27 = Type 27 (Heat Meter) 28 = Type 28 (Integrator) 29 = Type 29 (Change of Value) 30 = Type 30 (Sample and Hold) 31 = Type 31 (Psychrometrics) 32 = Type 32 (Sliding Window) 33 = Type 33 (Differencer) 34 = Type 34 (Analogue Selector)	-	N	Y

14.5 Function Module Connectives

The following parameters are available as connectives (not type 18, 31):

Destination

Note: For some modules the 'Destination' has another name the parameter has a text comms id of 'D'.

For type 18 the following connectives are available:

Bit 0
Bit 1
Bit 2
Bit 3
Bit 4
Bit 5
Bit 6
Bit 7

For type 31 the following connectives are available:

Saturated Vapour Pressure
Vapour Pressure
Enthalpy
Moisture content
Dew point

15 IC COMMS MODULES

Module Identifier	Nn (where n is module number)
Default overview parameters	($, D)
Size in brIQs	19

15.1 Functionality

Inter-Controller Communications (IC comms) modules support communication between networked IQ controllers, and between IQ4s and BACnet devices. There are nine IC Comms module types to cope with differing requirements - see **"IC Comms Module Types" on page 288**.

In addition to the nine different IC Comms module types there are three variable types, specified by the 'Variable Type' parameter, used by Type 1 (To), Type 2 (Global To), Type 3 (Min), Type 4 (Max), Type 5 (Sum), and Type 6 (Average) to specify the type of value that is transferred:

- **Analogue**: A transference of value from one analogue point to another analogue point
- **Digital Byte**: A transference of status from 8 digital points to another 8 digital points
- **Digital Bit**: A transference of status from one digital point to another digital point.

Note: Type 7 (From Analogue), Type 8 (From Byte), and Type 9 (From Bit) IC comms modules do not use the 'Variable Type' parameter to specify the type of value transferred as it is an integral part of the module's functionality.

IC comms modules appear on the strategy page and do not need to be sequenced. IC comms received in this controller from another controller can be represented on the strategy page using a dummy IC comms module.

Note: IC comms should not be used across an auto-dialling connection. Their use across routers is covered by the statement in **"When Not to Build Internetworks Across Routers" on page 42**.

15.1.1 Receiving IC comms from Other Controllers

IQ4s are able to receive IC comms messages initiated by IQ1, IQ2, IQ3, IQeco and IQ4 controllers although not all IC comms types are available in the IQeco and IQ3 controllers. This is described in the table below:

Module Type	*Variable Type*	*Initiated in*				
		IQ1	*IQ2*	*IQeco*	*IQ3*	*IQ4*
Type 1 (To)	Analogue	Yes	Yes	Yes	Yes	Yes
	Digital Byte	Yes	Yes	§Yes	*Yes	Yes
	Digital Bit	Yes	Yes	Yes	Yes	Yes
Type 2 (Global To)	Analogue	Yes	Yes	Yes	Yes	Yes
	Digital Byte	Yes	Yes	§Yes	*Yes	Yes
	Digital Bit	Yes	Yes	Yes	Yes	Yes
Type 3 (Min)	Analogue	Yes	Yes	No	*Yes	Yes
Type 4 (Max)	Analogue	Yes	Yes	No	*Yes	Yes
Type 5 (Sum)	Analogue	Yes	Yes	No	*Yes	Yes
Type 6 (Average)	Analogue	Yes	Yes	No	*Yes	Yes
Type 7 (From Analogue)		Yes	Yes	Yes	†Yes	Yes
Type 8 (From Byte)		#Yes	#Yes	#Yes	†#Yes	#Yes
Type 9 (From Bit)		#Yes	#Yes	#Yes	†#Yes	#Yes

Although the IQ4 will respond to these requests, the digital byte module in the IQ4 cannot be set by the strategy. However, Type 9 (From Bit) IC comms using switch or digital input status as the item selection works.
§ Only available when sending from IQeco v2 or greater.
† Only available when sending from IQ3 v2.0 or greater.
* Only available when sending from IQ3 v1.2 or greater.

Note: IC comms with IQLs is described in the Trend LONWORKS Products Engineering Manual ().

In order to receive or reply to an IC comms message the corresponding module must have been created in the IQ4. E.g if a Type 7 (From Analogue) IC comms message requests the value of Sensor module 1 then Sensor module 1 must exist in the IQ4 for correct operation.

Note: An IC comms message addressed to a digital bit requires a digital byte module, and only one bit of the byte module will be set up.

15.1.2 Initiating IC comms to Other Devices

IQ4s are able to initiate IC comms messages to IQ1, IQ2, IQ3, IQeco, IQ4 controllers and BACnet devices although not all IC comms types are available in the IQeco and IQ3 controllers. This is described in the table below:

Module Type	*Variable Type*	*Initiated in IQ4*					
		IQ1v4.1 or greater	*IQ2*	*IQeco*	*IQ3*	*IQ4*	*BACnet Device*
Type 1 (To)	Analogue	Yes	Yes	Yes	Yes	Yes	Yes
	Digital Byte	Yes	Yes	§Yes	*Yes	Yes	No
	Digital Bit	Yes	Yes	Yes	Yes	Yes	Yes
Type 2 (Global To)	Analogue	Yes	Yes	Yes	Yes	Yes	Yes
	Digital Byte	Yes	Yes	§Yes	*Yes	Yes	No
	Digital Bit	Yes	Yes	Yes	Yes	Yes	Yes
Type 3 (Min)	Analogue	Yes	Yes	No	*Yes	Yes	No
Type 4 (Max)	Analogue	Yes	Yes	No	*Yes	Yes	No
Type 5 (Sum)	Analogue	Yes	Yes	No	*Yes	Yes	No
Type 6 (Average)	Analogue	Yes	Yes	No	*Yes	Yes	No
Type 7 (From Analogue)		Yes	Yes	Yes	†Yes	Yes	Yes
Type 8 (From Byte)		Yes	Yes	Yes	†#Yes	#Yes	No
Type 9 (From Bit)		Yes	Yes	Yes	†#Yes	#Yes	Yes

Note: IQ4s cannot initiate an IC comms messages to IQ1 controller with firmware less than 4.1.

Although the IQ4 will respond to these requests, the digital byte module in the IQ4 cannot be set by the strategy. However, Type 7 (From Analogue) IC comms using switch or digital input status as the item selection works fine.
§ Only available when sending to IQeco v2 or greater.
† Only available when sending to IQ3 v2.0 or greater.
* Only available when sending to IQ3 v1.2 or greater.

Note: IC comms with IQLs is described in the Trend LONWORKS Products Engineering Manual (TE200292).

15.1.3 Module Disable

If required the module can be disabled - see **"Disabling Modules" on page 29**.

15.1.4 Security on IC comms Writes

Trend protocol IC comms writes (where a value is being changed), i.e. all Type 1 (To) and Type 2 (Global To) IC comms messages need to be able to authorise the change in a controller that has security enabled. This is done by including the Personal Identification Number (PIN) for Type 0 User Module 1.

Controllers that are to receive these messages must contain a Type 0 User Module with a matching PIN that has an authority level high enough to allow the change, unless there are no Type 0 User Modules set up in which case the change will be permitted. If the PIN for Type 0 User Module number 1 in the initiating controller is not set up, the message will be sent without a PIN.

Note: The authority level required in the remote controller for the selected PIN must be 99 for a Type 2 (Global To) IC comms module and 95 for a Type 1 (To) IC comms module; for a change to a Knob or Switch module it must be equal to or greater than the PIN level for that particular item.

Note: For IQ4 v3.30 or greater the Type 0 User Module must have network access.

15.1.5 IC Comms/COV Subscription Limitations

From v3.40 the maximum number of IC comms modules increased from 250 to 1000, and from v3.43 the maximum number of BACnet COV subscriptions increased from 400 to 1000. These limits are shared i.e configuring a single IC Comms will reduce the number of available COV subscriptions by 1. The following criteria apply to these limits.

- 1000 IC comms/COV subscriptions are available for use over the IP network. However, only 500 are available for use over either the MS/TP or Trend Current Loop networks (shared between both networks,), the remaining 500 are still available for the IP network.
- A maximum of 300 IC comms/COV subscriptions per minute is recommended. For MS/TP and Trend Current Loop networks a maximum of 100 IC comms/COV subscriptions per minute shared between the two networks is recommended. These recommendations can be used together i.e maximum of 200 IC comms/COV subscriptions per minute on the IP network, and another 100 IC comms/COV subscriptions per minute onto the MS/TP and Trend Current Loop networks.

Note: COV subscription is not available over Trend Current Loop.

15.1.6 Retransmission Interval and Significant Change

IC comms must be used with caution because of the load on the communications involved. On a large system there may be many controllers resulting in hundreds of possible IC comms messages. Since it takes a finite time to send a message, the rate at which messages are sent becomes important, otherwise the network may become overloaded and it will become difficult to send alarms or communicate with the supervisor.

For this reason there is a retransmission interval, specified using the 'Retransmission Interval' parameter, which can be set up for each communication. It defines the time in minutes between sending the message, and should be set to be the maximum value that the system can stand.

Type 1 (To) and Type 2 (Global To): The message will be sent whenever the value of the 'Significant Change' parameter is exceeded (i.e. not just at the interval specified by the 'Retransmission Interval' parameter). If no significant change occurs, or if the significant change parameter is set to '0', the message will be sent at the interval specified by the 'Retransmission Interval' parameter. If the 'Retransmission Interval' parameter is set to '0' the message will only be sent when there is a significant change. If both the 'Retransmission Interval' and 'Significant Change' parameters are set to '0', the message will be sent at 15s intervals. Once sent the message will not be sent again for at least 15s to avoid a fluctuating value causing excessive transmission.

Type 3 (Min), Type 4 (Max), Type 5 (Sum), and Type 6 (Average): If the 'Retransmission Interval' parameter is set to '0', the message is resent after 15s. This should be avoided because of the communications overhead involved, so the interval should be set non-zero. If the message fails to return, it is resent after 2 minutes.

Note: The 'Significant Change' parameter is only available for analogue variable type messages. For analogue type variables a significant change is considered to have occurred if the 'Significant Change' parameter is exceeded, or the alarm status (ROCH) differs from the last stat value sent. For a digital variable type (bit and byte) there is a significant change if the status has changed.

Type 7 (From Analogue), Type 8 (From Byte), and Type 9 (From Bit): The minimum period between sending the messages is 15s so that if the 'Retransmission Interval' parameter is set less than this it will be sent again after 15s anyway. This should be avoided because of the communications overhead involved, so the interval should be set non-zero.

For Type 7 (From Analogue), Type 8 (From Byte), and Type 9 (From Bit) BACnet IC comms with COV Mode enabled the 'Retransmission Interval' parameter has a special function - see **"COV Service" on page 282**. If it is set to '0' it will set up the IC comms module to listen for unsolicited COV messages from the server. If it is set to non-zero it will set up a solicited COV subscription in the server; the retransmission interval becomes the lifetime of the subscription, but it will re-subscribe before the lifetime expires to keep the subscription alive.

15.1.7 COV Service

Type 7 (From Analogue), and Type 9 (From Bit) BACnet IC comms modules have the option to use the Change Of Value (COV) service. This is selected by setting the 'COV Mode' parameter to '1'. It enables the client (IQ4) to receive COV reports from the server (via the NTD module). The server sets the COV increment (used for analogue values only) so Trend have no control over this, and the server will send the report when the change in value since the last report exceeds the COV increment. For binary properties the report is sent on a change of state.

As explained in **"BACnet COV Reporting and Receiving" on page 64** COV reception can either be unsolicited or solicited.

Unsolicited COV reception is selected by setting the 'Retransmission Interval' parameter to '0'. This will cause the IQ4 to fetch the required value once, and then to 'listen' for that value being broadcast from the server. The initial fetched value will set the output, and keep the 'Failed Bit' parameter at '0'; if it fails to fetch the initial value 3 times it will set the 'Failed Bit' parameter to '1'.

Solicited COV reception is selected by setting the 'Retransmission Interval' parameter to non-zero. This will cause the IQ4 to fetch the required value once, and to send a subscription to the server with the value of the 'Retransmission Interval' parameter being used to set the subscription lifetime. The initial fetched value will set the output, and keep the 'Failed Bit' parameter at '0'; if it fails to fetch the initial value three times or if it fails two consecutive subscription requests it will set the 'Failed Bit' parameter to '1'. The IQ4 will re-subscribe before the Lifetime expires to keep the subscription going. The subscription can be cancelled by disabling either the IC comms module or COV Mode.

15.1.8 Retries and Failed Bit

When it sends an IC comms message the IQ4 will wait 13 or 15s for a reply (13s if the destination address is greater than the source address, else 15s). If no reply is received within that period, the IQ4 will set the 'Failed Bit' parameter to '1', and mark the message accordingly. If the last message was replied to, the 'Failed Bit' parameter will be set to '0'.

If a reply has not been received, the message will be sent again after:

$$120s + \Delta T$$

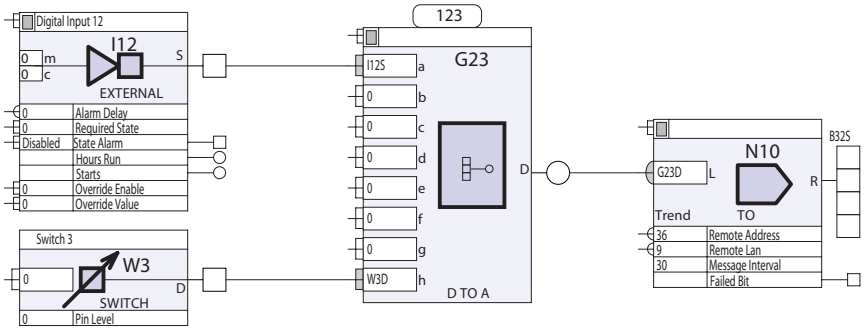
Where:

ΔT = Local Address MOD 17 + Local LAN MOD 13. Local LAN and Local Address are Address module parameters.

Note: Type 7 (From Analogue), and Type 9 (From Bit) IC comms with COV Mode enabled work slightly differently - see **"COV Service" on page 282**.

15.1.9 Generating a Digital Byte IC Comms

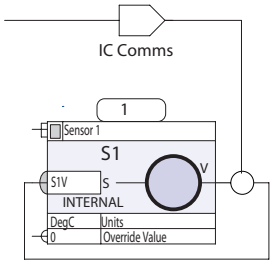
Type 1 (To) and Type 2 (Global To) Trend protocol IC comms can transfer a digital byte. The IC comms module has an analogue input; so a Type 8 (D to A) Logic module has to be used to create an analogue value from the digital bits.



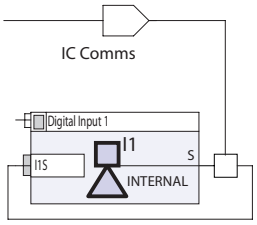
The above strategy shows two bits produced by a digital input and a switch being converted to an analogue value and then sent as a byte Data To IC comms. In a receiving IQ3 or IQ4 controller a digital byte module must be used to receive this message.

15.1.10 IC Comms Writes

Type 1 (To) and Type 2 (Global To) Trend protocol IC comms to a Sensor module uses the 'Value' parameter which is actually the Sensor module's output. This will be overwritten as the Sensor module is serviced. This problem can be overcome for Type 1 (Internal Analog) Sensor modules by connecting the output to the input.



Similarly an IC comms write to a digital input would normally use the 'Status' parameter which is actually the Digital Input module's output. This will be overwritten as the Digital Input module is serviced. This problem can be overcome for a Type 1 (Internal) Digital Input module by connecting the output to the input.



Both Type 0 (Analogue) Sensor and Type 2 (External) Digital Input modules can't be looped back, so IC comms to them will be overwritten when serviced. Type 2 (Digital) and Type 3 (Internal Digital) Sensor modules have an analogue output with a digital input so can't be looped back directly. For these reasons it is recommended not to send IC comms writes to these modules.

IC comms writes to Switch and Knob modules will not be overwritten as the parameter being written to is the module's input. Similarly Analogue Node, and Digital Byte modules which are normally accessed by absolute item selection will not be overwritten for the same reason.

IC comms writes to Driver modules can be overwritten and should not be used.

15.1.11 Power On Delay

In order to control the number of messages put on to the communications network when a whole site powers up, IC comms modules have power on delays. These delays also apply on soft restart. The delay is between 60 s and 74 s dependent on controller address:

(60+address MOD 15).

Note: The expression MOD 15 means the remainder left after dividing by 15. For example controller 20 will have a delay of:

60+20MOD 15

The remainder after dividing 20 by 15 is 5, so the delay will be:

60+5 = 65 secs

Controller 21 would then have a value of 66 seconds etc. This does not ensure unique delay times for all controller on the network, but will reduce the number of IC comms messages at start up.

15.2 BACnet IC Comms

BACnet IC comms are only available on /BAC variants and IQ4NCs. Type 1 (To), Type 2 (Global To), Type 7 (From Analogue), and Type 9 (From Bit) support BACnet IC comms which enables them to send IC comms to a BACnet device using the BACnet protocol. The 'Protocol' parameter must be set 'BACnet' to enable communication using the BACnet protocol.

Note: Type 1 (To) and Type 2 (Global To) Byte IC comms do not support BACnet protocol.

The IC comms message is actually sent to a Non-Trend Device module (NTD) in the IQ4 which forwards it to the BACnet device(s). The NTD module is allocated a LAN number (the Trend BACnet LAN), a node address on that LAN, and address attributes so that the controller selection process described on **page 284** is the same for both Trend and BACnet protocol IC comms. For a Trend protocol Type 2 (Global To) the 'Remote Lan' parameter can be set to 128 (a global message to every LAN, i.e. a global global), but this is not allowed for BACnet protocol.

In order to send a BACnet IC comms the Type 5 (BACnet IP) and Type 13 (BACnet Application) Network modules must be correctly configured - see **"BACnet IP Communications" on page 63**.

Type 1 (To), Type 2 (Global To) IC comms using BACnet protocol correspond to the BACnet write property service which has a write priority level; this enables the receiving device to decide whether to change the value or not. If the BACnet IC comms is subsequently disabled a relinquish message is sent to the receiving device so that writes to it at a lower priority level can be acted on.

Note: The property used for BACnet IC comms is always the selected object's present value.

15.3 Controller Selection

There are two methods of selecting the controller(s) to which the communication is to occur:

Absolute Controller Selection
Label Matching Controller Selection

15.3.1 Absolute Controller Selection

For Type 1 (To), Type 7 (From Analogue), Type 8 (From Byte), and Type 9 (From Bit) IC comms modules messages are directed to a particular controller defined by 'Remote Lan' and 'Remote Controller Address' parameters.

15.3.2 Label Matching Controller Selection

For type Type 2 (Global To), Type 3 (Min), Type 4 (Max), Type 5 (Sum), and Type 6 (Average) IC comms modules messages are sent to a set of controllers specified by the 'Remote Lan' and 'Attribute' parameters.

The 'Attribute' parameter specifies the attribute used to select the controller(s) containing the remote module on the LAN specified by the 'Remote Lan' parameter. For controllers to match the requirement for the IC comms message they must have at least one attribute that matches the attribute in the specified attribute of the initiating controller set up in their respective Address modules.

Note The match is performed on all attributes of the receiving controllers.

The match can be an exact match with the specified attribute, or wild cards can be used to match attributes with similar text - see **"Attribute Selection - Wildcards" on page 285**.

15.3.2.1 Attribute Selection - Wildcards

In addition to the simple matching of two text strings, special characters known as wildcards can be used to indicate that a exact match is not required, but some substitute characters are acceptable. The following wildcards are available:

*****
?
&
|
.

***** Wildcard

The * is a wildcard character that can be included in an attribute to allow any character, no character, or group of characters to match that position in the initiator's attribute e.g.

IC comms originating controller	IC comms receiving controller
Attribute **CANTEEN***	attribute **CANTEEN SOUTH**

In early versions of IQ4 firmware the controller would ignore any characters after the wildcard. From v3.21 firmware the onwards characters after the wildcard are not ignored making it the same as that of IQ2 and IQ3 v3.06 or greater.

E.g. In early versions of firmware an IC comms message sent using an attribute of 'AHU*Flow' would match an attribute of 'AHU 1 Damper' as the characters '1 Damper' after the wildcard are ignored. In version v3.21 firmware or greater it would not match as the characters '1 Damper' after the wildcard are not ignored.

An attribute of 'AHU*Damper' could be used in an IC comms message to achieve a match in the receiving controller with an attribute of 'AHU 1 Damper'.

? Wildcard

The **?** is a wildcard character that can be included in an attribute to allow any single character to match that position in the initiator's attribute (there must be a character) e.g.

IC comms originating controller	IC comms receiving controller
Attribute **CANTEEN?**	Attribute **CANTEEN1**

& Wildcard

& is a special character that performs a string AND function. When in an attribute, the message may only be delivered if the destination has two separate attributes that match the characters either side of the **&** e.g.

IC comms originating controller	IC comms receiving controller
Attribute 2 **NORTH&OUT**	Attribute 2 **NORTH**
	Attribute 3 **OUT**

| Wildcard

| is a special character that performs a string OR function. When in an attribute, the message may only be delivered if the destination has either or both of the two separate attributes that match the characters either side of the **|** e.g.

IC comms originating controller	IC comms receiving controller
Attribute 3 **NORTH|OUT**	Attribute 2 **NORTH**

. Wildcard

. is a special character that when placed in an attribute the text beyond the first fullstop is ignored and the full stop is replaced by a wildcard * e.g.

IC comms originating controller	IC comms receiving controller
Attribute 4 **FL3.LEADER**	Attribute **FL3A1**

When the originating controller sends a message, the attribute is truncated to **FL3***.

Note: The attribute will work as normal for matching with incoming text messages, e.g. the supervisor can obtain the average from this controller by requesting FL3.LEADER.

15.4 Item Selection

There are two methods of selecting the remote point to or from which the communication is to occur:

Label Matching Item Selection
Absolute Item Selection

15.4.1 Label Matching Item Selection

This form of item selection only applies Type 1 (To), Type 2 (Global To), with analogue or digital bit variable types and Type 3 (Min), Type 4 (Max), Type 5 (Sum), or Type 6 (Average) that use the Trend protocol.

The IC comms message is sent to point(s) of the correct variable type with a label that matches the label of the local point. All the relevant modules in the destination controller(s) that have a label match will act on the message.

Once the IC comms message is received by one of the relevant modules with a matching label, the message will only be processed further if the variable type (V, or S) is present in the module see table below.

Label Match Destination module	*Analogue Value (V)*	*Digital Bit Status (S)*
Sensor	Accepted	Rejected
Knob	Accepted	Rejected
Digital Input	Rejected	Accepted
Switch	Rejected	Accepted

If the correct variable type is present the value is transferred. The parameter transferred using label matching is ‘Value’ for an analogue value, and ‘Status’. The four alarm status bits are not transferred along with the analogue value; these are only transferred when the remote point is an analogue node and label matching excludes analogue nodes.

Note: For the Type 3 (Min), Type 4 (Max), Type 5 (Sum), and Type 6 (Average) IC comms the variables are always analogue. Since these functions ask for analogue values which may be contributed to from a number of modules, if one of the modules has a value which has associated Read or Out of Limits alarms then that module’s value will not contribute to the result.

Note: Only Sensor, Digital Input, Knob, and Switch modules are checked for label matching.

Note: If the local point doesn’t have a label, the IC comms will not operate.

Note: Label matching item selection should be used with discretion as there is a overhead incurred by its use due to the load involved in checking all the controller labels for a match. This may cause the IC comms to take longer.

For details of specifying labels see **“Common IQ4 Module Parameters” on page 165**. Single letters or single letters followed by a number should be avoided (e.g. S, K, W, S8, W1).

The match can be an exact match with the specified label, or wild cards can be used to match labels with similar text - see **“Item Selection - Wildcards” on page 286**.

Specifying Label Matching Item Selection

Label matching item selection is specified by setting the ‘Remote Module’ parameter to blank.

15.4.1.1 Item Selection - Wildcards

In addition to the simple matching of two text strings, special characters known as wildcards can be used to indicate that a exact match is not required, but some substitute characters are acceptable. The following wildcards are available:

*****

.

?

These wildcards work in the same way as those for used for controller selection see **“* Wildcard” on page 285**, **“? Wildcard” on page 285**, or **“. Wildcard” on page 285**.

15.4.2 Absolute Item Selection

This form of item selection applies all type of IC comms modules. It is the only method of item selection available for BACnet protocol IC comms.

Absolute item selection presents less of a system overhead than item label matching item selection and should be used in preference.

Specifying Absolute Item Selection

The remote point is specified by entering the item code for the required point in the 'Remote Module' parameter.

For Trend protocol IC comms the item code is specified in the format below:

<Module>(<Module Parameter>)

<Module> specifies the type of module using its module identifier and module number (e.g. S1 specifies sensor 1). <Module Parameter> specifies the parameter within the module using its text comms ID (e.g. V specifies the 'Value' parameter).

For BACnet protocol IC comms the item code is specified in the format below:

<object type><instance><property>

E.g. AI2PV, analogue input, 2, present value.

<object type> specifies the BACnet object type for the required point. <instance> specifies the BACnet instance for the required point. <property> specifies the BACnet property that is to be transferred.

A list of object type codes and property codes is given in **"BACnet Object/Property Codes and Units" on page 519**.

Note: If just the object type and instance are entered, the property will default to the 'Present Value'. For example entering AI2 (object type AI, analogue input, and instance 2) will cause the IC comms to communicate with AI2PV.

Care should be taken when entering or editing the 'Remote Module' parameter as errors may cause the remote controller's strategy to change; an understanding of text communications (for Trend protocol - see **"Text Communications" on page 543** and BACnet properties (for BACnet protocol IC comms) - see **"BACnet Properties Supported" on page 69** is required.

The safest way to set up the item code is to use the **Helper** dialogue box in IQSET without selecting the **Advanced** button as IQSET will ensure that the remote module matches the local module connection and restricts the parameters that can be selected to those described below:

Parameters available for Trend protocol IC comms:

Module Type	*Parameter*	*Variable Type*	*IC comms Module Type*
Analogue Node	Value	Analogue	1 and 2
Knob	Value		
Sensor	Value		
Byte	Status	Byte	
Bit	Status	Bit	
Switch	Status		
Digital Input	Status		
Analogue Node	Value	Analogue	3, 4, 5, 6, and 7
Knob	Value		
Sensor	Value		
Byte	Status	Byte	8
Bit	Status	Bit	9
Switch	Status		
Digital Input	Status		

Parameters available for BACnet protocol IC comms are shown in the table below:

Object Type	*Property*	*Variable Type*	*IC comms Module Type*
Analogue Input	Present Value	Analogue	1, 2, and 7
Analogue Value	Present Value		
Analogue Output	Present Value		
Multi-state Input	Present Value		
Multi-state Value	Present Value		
Multi-state Output	Present Value		
Binary Input	Present Value	Bit	1, 2, and 9
Binary Value	Present Value		
Binary Output	Present Value		

In IQSET this is done by clicking **Look Up** button in the IC comms module's **Parameters** dialogue box. This displays a dialogue box which enables the selection of the basic module types (object type for BACnet IC comms).

Selecting the **Advanced** button will allow other parameters to be specified. However care should be taken when doing this as errors may cause the remote controller's strategy to change.

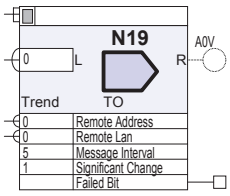
For details of using IQSET see the IQSET Manual (TE200147).

15.5 IC Comms Module Types

There are 9 types of IC comms module specified by the 'Comms Direction' parameter.

Direction Class	*Description*
Type 1 (To)	Send data to another controller.
Type 2 (Global To)	Send data to a set of controllers.
Type 3 (Min)	Request the minimum value from a set of controllers.
Type 4 (Max)	Request the maximum value from a set of controllers.
Type 5 (Sum)	Request the sum of values from a set of controllers.
Type 6 (Average)	Request the average of values from a set of controllers.
Type 7 (From Analogue)	Request analogue data from another controller.
Type 8 (From Byte)	Request digital byte data from another controller.
Type 9 (From Bit)	Request digital bit data from another controller.

15.5.1 Type 1 (To)



Type 1 (To) IC comms modules enable the IQ4 to send a value (analogue, digital byte, or digital bit) to another IQ controller.

15.5.1.1 Specifying the Remote Controller

The controller to which the value is sent is specified by the 'Remote Lan' and 'Remote Controller Address' parameters - see **"Absolute Controller Selection" on page 284**.

15.5.1.2 Specifying the Type of Value

The type of value is specified by the 'Variable Type' parameter.

15.5.1.3 Specifying the Remote Module

The location the value is to be written to can be specified absolutely or using label matching - see **"Item Selection" on page 286**.

15.5.1.4 Alarm Bit Handling

If sending an analogue value and the point in the remote controller is an analogue node the alarm status bits (ROLH), are also sent if available; they aren't sent if the remote point is not an analogue node. These bits would be exposed by the receiving analogue node.

15.5.1.5 Significant Change

The module supports significant change which enables the IC comms message to only be sent when the value of the source changes by a specified amount or there is a status change - see **"Retransmission Interval and Significant Change" on page 281**.

Note: The 'Significant Change' parameter is only available when the 'Variable Type' parameter is set to '0-Analogue'.

15.5.1.6 BACnet IC comms

BACnet IC comms are supported - see **"BACnet IC Comms" on page 284.**

15.5.1.7 COV Mode

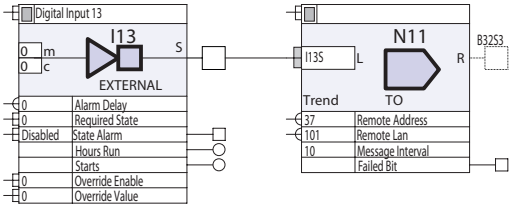
COV mode is not supported.

15.5.1.8 Examples

Data To Controller 37 on LAN 101 - digital bit 32,3

Sends the status of digital input 13 to byte 32 bit 3 in controller 37 on LAN 101 on status change and at least every 10 mins. An IQ3 or IQ4 receiving this would need byte module 32 to be created in its strategy.

The IC comms module should be set up as shown.

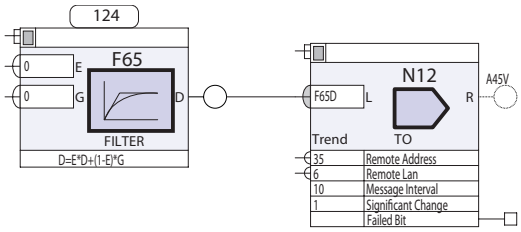


Parameter	***Value***
Protocol	Trend
Comms Direction	1 (To)
Variable Type	Digital Bit
Remote Controller Address	37
Remote Lan	101
Local Module	I13S
Remote Bit	B32S3
Retransmission interval	10

Data To Controller 35 on LAN 6 - analogue node 45

Sends the value and alarm status bits of function module 65 output to analogue node 45 in controller 35 on LAN 6 whenever the value changes by 1 or greater or for an alarm status change, and at least every 10 mins. An IQ3 or IQ4 receiving this would need Analogue Node module 45 to be created in its strategy.

The IC comms module should be set up as shown.



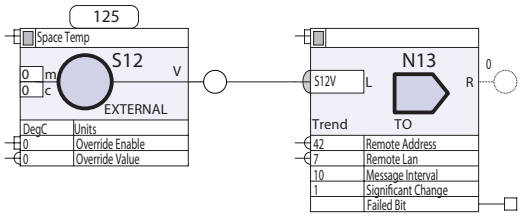
Parameter	*Value*
Protocol	Trend
Comms Direction	1 (To)
Variable Type	Analogue
Remote Controller Address	35
Remote Lan	6
Local Module	F65D
Remote Analogue	A45V
Significant Change	1.00
Retransmission interval	10

Data To Controller 42 on LAN 7 - analogue value/label item selection

Sends the value of Sensor module 12 to the value modules with a label of 'Space Temp' (i.e. sensor 12's label) in controller 42 on LAN 7 whenever the value changes by greater than 1, and at least every 10 mins.

Note that the alarm status bits are not sent as the remote point is selected by label matching and so it is not explicitly set to an analogue node.

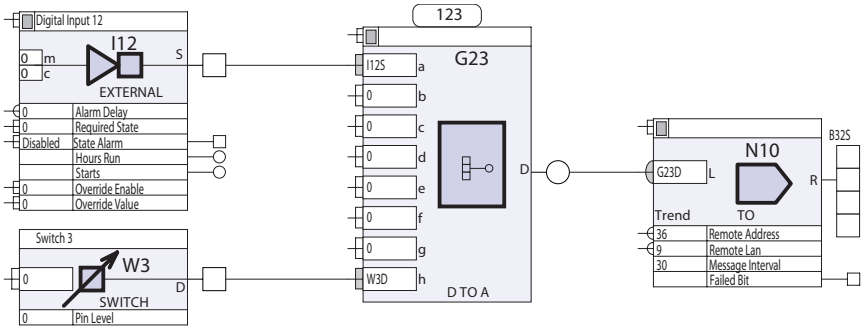
The IC comms module should be set up as shown.



Parameter	*Value*
Protocol	Trend
Direction	1 (To)
Variable Type	Analogue
Remote Controller Address	42
Remote Lan	7
Local Module	S12V
Remote Analogue	0
Significant Change	1.00
Retransmission Interval	10

Data To Controller 36 on LAN 9 - digital byte 32

Sends the status of the digital inputs to byte 32 in controller 36 on LAN 9 on status change and at least every 30 mins. An IQ3 or IQ4 receiving this would need byte module 32 to be in its strategy.

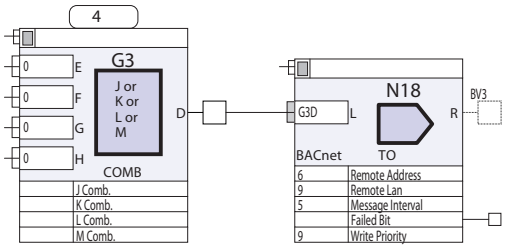


The IC comms module should be set up as shown.

Parameter	***Value***
Protocol	Trend
Comms Direction	1 (To)
Variable Type	Digital Byte
Remote Controller Address	36
Remote Lan	9
Local Module	G23D
Remote Byte	B32S
Retransmission Interval	30

BACnet Data To Controller 6 LAN 9 - binary value 3

Sends the Logic module 3 output status with a BACnet write priority of '9' to the BACnet device specified by the NTD module set up as controller 6 on LAN 9 every 5 mins.



The IC comms module should be set up as shown.

Parameter	***Value***
Protocol	BACnet
Comms Direction	1 (To)
Variable Type	Bit
Remote Controller	6
Remote Lan	9
Local Module	G3S
Remote Module	BV3
Retransmission Interval	5
Write Priority	9

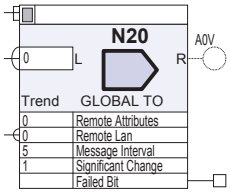
Note: An NTD module must be set up with a device address 6 on LAN 9; this module will convert the Trend address to the BACnet address.

15.5.1.9 Type 1 (To) IC comms Module Parameters

Type 1 (To) IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Last Sent	The value or status of the local module when last sent.	0 or 1 Or Full analogue	0	O	l
Protocol	The protocol bring used for the IC comms.	0 or 1 0 = Trend 1 = BACnet	0	N	P
Remote Controller Address	Address of controller being containing the Remote Module on the LAN.	1, 4 to 119 excluding 10	0	I	A
Significant Change	Analogue variable type only. The amount above which local module's value has to change before the IC comms message is sent. The IC comms may be sent more frequently according to the significant change - see **"Retransmission Interval and Significant Change" on page 281**. It should be set as large as possible to minimise communications traffic.	0 to 3276	0	N	S

15.5.2 Type 2 (Global To)



Type 2 (Global To) IC comms modules enable the IQ4 to send a value (analogue, digital byte, or digital bit) to a set of other controllers selected via attribute addressing.

15.5.2.1 Specifying the Remote Controllers

The controllers (remote controllers) to which the value is sent are identified by an address attribute; the message can be sent to a single LAN or every LAN - see **"Label Matching Controller Selection" on page 284**.

15.5.2.2 Specifying the Type of Value

The type of value is specified by the 'Variable Type' parameter.

15.5.2.3 Specifying the Remote Module

The location the value is to be written to can be specified absolutely or using label matching - see **"Item Selection" on page 286**.

15.5.2.4 Alarm Bit Handling

If sending an analogue value and the point in the remote controller is an analogue node the alarm status bits (ROLH) are also sent if available; they aren't sent if the remote point is not an analogue node. These bits would be exposed by the receiving analogue node.

15.5.2.5 Significant Change

The module supports significant change which enables the IC comms message to only be sent when the value of the source changes by a specified amount or there is a status change - see **"Retransmission Interval and Significant Change" on page 281**.

Note: The 'Significant Change' parameter is only available when the 'Variable Type' parameter is set to '0-Analogue'.

15.5.2.6 BACnet IC comms

BACnet IC comms are supported - see **"BACnet IC Comms" on page 284.**

15.5.2.7 COV Mode

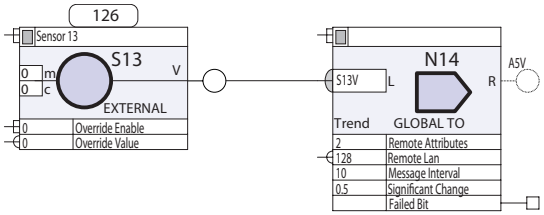
COV mode is not supported.

15.5.2.8 Examples

Global To to all LANs - analogue node 5

Transfers the analogue value and alarm status bits from S13 to analogue node 5 of any controller (with an attribute which matches attribute 2) on any LAN. This will be sent whenever the value changes by more than 0.5, or the alarm status changes and at least every 10 minutes. Since the 'Attribute 2' parameter in the Address module of this controller is set to 'XYZ Co' then this attribute would have to be in any controller that is to receive this message. An IQ3 or IQ4 receiving this would need to Analogue Node module 5 in its strategy.

The IC comms module should be set up as shown.



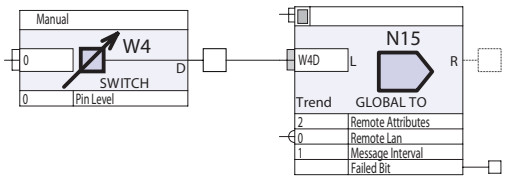
Parameter	*Value*
Protocol	Trend
Comms Direction	2 (Global To)
Variable Type	Analogue
Attribute	2
Remote Lan	128
Local Module	S13V
Remote Analogue	A5
Significant Change	0.5
Retransmission Interval	10

The Address module's 'Attribute 2' parameter must be set to 'XYZ Co'.

Global To to Local LAN - digital bit/label item selection

Transfers the status from W4 to valid modules with a label of 'Manual' (i.e. switch 4's label) in any controller with an attribute which matches the 'Attribute 2' parameter on the Local LAN ('0' specifies Local LAN). This will be sent on status change and at least every 10 minutes. Since the 'Attribute 2' parameter in the Address module of this controller is set to 'XYZ Co' then this attribute would have to be in any controller to receive this message. An IQ3 or IQ4 receiving this would need Switch module 4 to be created in its strategy.

The IC comms module should be set up as shown.



Parameter	*Value*
Protocol	Trend
Comms Direction	2 (Global To)
Variable Type	Digital Bit
Attribute	2

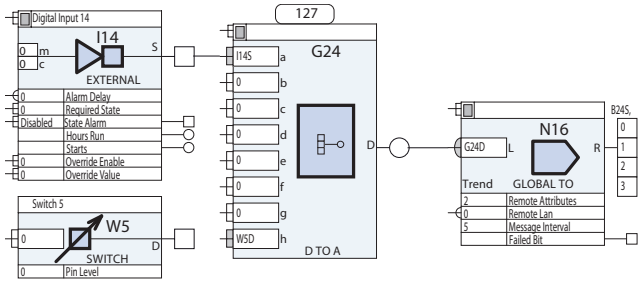
Parameter	*Value*
Remote Lan	0
Local Module	W4D
Remote Bit	0
Retransmission Interval	10

The Address module's 'Attribute 2' parameter must be set to 'XYZ Co'.

Global To to Local LAN - digital byte 24

Transfers the status of the inputs to G24 to byte 24 in any controller with an attribute which matches the 'Attribute 2' parameter on the Local LAN ('0' specifies Local LAN). Since the 'Attribute 2' parameter in the Address module of this controller is set to 'XYZ Co' then this attribute would have to be in any controller to receive this message. An IQ3 or IQ4 receiving this would need byte 24 to be created in its strategy. It will be sent on change of status or at least every 5 minutes

The IC comms module should be set up as shown.



Parameter	Value
Protocol	Trend
Comms Direction	2 (Global To)
Variable Type	Digital Byte
Attribute	2
Remote Lan	0
Local Module	G24D
Remote Byte	B24S
Retransmission Interval	5

The Address module's 'Attribute 2' parameter must be set to 'XYZ Co'.

Global Distribution of an Analogue Value - Global To all LANs - analogue/label item selection

The message is sent from this controller's S14 to any controller on any LAN (signified by 128) with an attribute which matches the 'Attribute 7' parameter which is a wildcard *, so it is sent to every controller. The remote modules are selected as any valid module with a label which matches by item label attribute using the label of S14 which is 'Outside Temp.Orig'. This will be truncated to Outside Temp* which will match with Outside Temp.Copy if set up in the receiving controllers. This will be sent when there is a change in sensor 14's value of greater than 0.5, or alarm status change, or after 10 minutes, whichever is the sooner.

The IC comms module should be set up as shown.

Parameter	*Value*
Protocol	Trend
Comms Direction	2 (Global To)
Variable Type	Analogue
Attribute	7
Remote Lan	128
Local Module	S14V
Remote address	0
Significant Change	0.5
Retransmission Interval	10

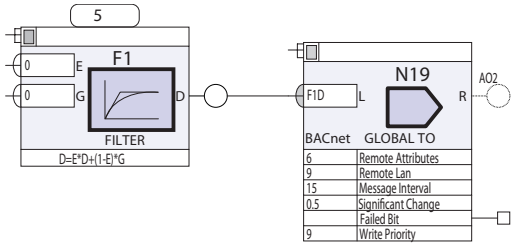
The Address module's 'Attribute 7' parameter must be set to '*'.

BACnet Global To LAN 9 - analogue output 2

This will send the F1's output value with a BACnet write priority of '9' to the BACnet analogue output 2 of the BACnet devices specified by the NTD modules set up with an attribute 'Aircon1' on LAN 9.

Note: This attribute should only be present on the BACnet NTDs (on BACnet LANs) not on controllers on Trend LANs.

It will be sent when there is a change in function module 1's output by greater than 0.5, or alarm status change, or after 15 minutes, whichever is the sooner.



The IC comms module should be set up as shown.

Parameter	*Value*
Protocol	BACnet
Comms Direction	2 (Global To)
Variable Type	Analogue
Attribute	6
Remote Lan	9
Local Module	F1D
Remote Module	AO2
Retransmission Interval	15
Significant Change	0.5
Write Priority	9

Note: An NTD module must be set up with a attribute 'Aircon 1' on LAN 9 for each BACnet device to be written; these modules will convert the Trend address to the BACnet address.

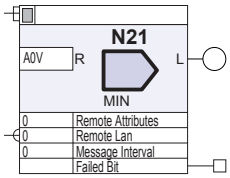
The controller's Address module's 'Attribute 6' parameter must be set to 'Aircon1'.

15.5.2.9 Type 2 (Global To) IC comms Module Parameters

Type 2 (Global To) IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Last Sent	The value or status of the local module when last sent.	0 or 1 Or Full analogue	0	O	l
Protocol	The protocol bring used for the IC comms.	0 or 1 0 = Trend 1 = BACnet	0	N	P
Significant Change	The amount above which local module's value has to change before the IC comms message is sent. The IC comms may be sent more frequently according to the significant change - see **"Retransmission Interval and Significant Change" on page 281**. It should be set as large as possible to minimise communications traffic.	0 to 3276	0	N	S

15.5.3 Type 3 (Min)



Type 3 (Min) IC comms modules request the value of a particular analogue node or sensor (remote node (R)) from all controllers that match the search criteria set up in the module, calculate the minimum value, and make it available as an output (L).

The message has to travel from one controller to the next in the set of controllers. The controllers check for an attribute match before deciding whether they should contribute to the message and will only substitute their value if it is less than the current value.

Note: If the message is interpreted by the receiving controller as being for one of its Sensor modules and the Sensor module has a Read or Out of Limits alarm bit set it will not contribute to the reply.

15.5.3.1 Specifying the Remote Controllers

The controllers (remote controllers) from which the IQ4 requests the value are identified by an address attribute; the message can be sent to a single LAN, not every LAN - see **"Label Matching Controller Selection" on page 284**.

15.5.3.2 Alarm Bit Handling

Alarm status bits are not sent.

15.5.3.3 Specifying the Remote Module

The location(s) the value is to be read from can be specified absolutely or using label matching - see **"Item Selection" on page 286**.

15.5.3.4 Significant Change

The module does not support significant change.

15.5.3.5 BACnet IC comms

BACnet IC comms are not supported.

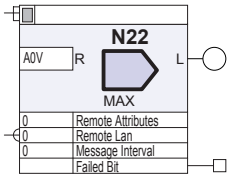
15.5.3.6 COV Mode

COV mode is not supported.

15.5.3.7 Type 3 (Min) IC comms Module Parameters

Type 3 (Min) IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**.

15.5.4 Type 4 (Max)



Type 4 (Max) IC comms modules request the value of a particular analogue node or sensor (remote node (R)) from all controllers that match the search criteria set up in the module, calculate the maximum value, and make it available as an output (L).

The message has to travel from one controller to the next in the set of controllers. The controllers check for an attribute match before deciding whether they should contribute to the message and will only substitute their value if it is greater than the current value.

Note: If the message is interpreted by the receiving controller as being for one of its Sensor modules and the Sensor module has a Read or Out of Limits alarm bit set, it will not contribute to the reply.

15.5.4.1 Specifying the Remote Controllers

The controllers (remote controllers) from which the IQ4 requests the value are identified by an address attribute; the message can be sent to a single LAN, not every LAN - see **"Label Matching Controller Selection" on page 284**.

15.5.4.2 Specifying the Remote Module

The location(s) the value is to be read from can be specified absolutely or using label matching - see **"Item Selection" on page 286**.

15.5.4.3 Alarm Bit Handling

Alarm status bits are not sent.

15.5.4.4 Significant Change

The module does not support significant change.

15.5.4.5 BACnet IC comms

BACnet IC comms are not supported.

15.5.4.6 COV Mode

COV mode are not supported.

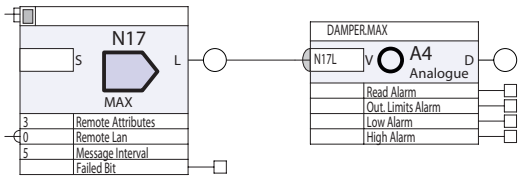
15.5.4.7 Example

Maximum value from Local LAN - label item selection

This will request the maximum value of any analogue sensor or Knob module with whose label matches with the label of Analogue Node module 4 from any controller (with an attribute which matches attribute 3) on the Local LAN ('0' specifies Local LAN) and write it into analogue node 4. This will be sent every 5 minutes.

Since this controller's Address module's 'Attribute 3' parameter 3 = 'AHU3', it will send this message to any controller with an attribute of 'AHU3'. Since the label of this controller's analogue node 4 is 'DAMPER.MAX', it will send the message to any analogue module(s) (sensor, knob, or driver) with label DAMPER*. If addressed controllers have analogue module(s) (sensor, knob, or driver) labelled DAMPER POSITION, they will contribute to the maximum.

The IC comms module should be set up as shown.



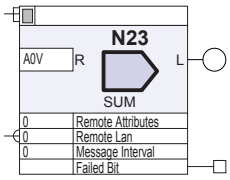
Parameter	***Value***
Comms Direction	4 (Maximum)
Variable Type	Analogue
Attribute	3
Remote Lan	0
Local Module	A4V
Remote Module	0
Retransmission Interval	5

The Address module's 'Attribute 3' parameter must be set to 'AHU3'.

15.5.4.8 Type 4 (Max) IC comms Module Parameters

Type 4 (Max) IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**.

15.5.5 Type 5 (Sum)



Type 5 (Sum) IC comms modules request the value of a particular analogue node (remote node (R)) from all controllers that match the search criteria set up in the module, calculate the sum, and make it available as an output (L).

The message has to travel from one controller to the next in the set of controllers. The controllers check for an attribute match before deciding whether they should contribute to the message and will add their value to the current value.

Note: If the message is interpreted by the receiving controller as for one of its sensors and the sensor has a Read or Out of Limits alarm bit set it will not contribute to the reply.

15.5.5.1 Specifying the Remote Controllers

The controllers (remote controllers) from which the IQ4 requests the value are identified by an address attribute; the message can be sent to a single LAN, not every LAN - see **"Label Matching Controller Selection" on page 284**.

15.5.5.2 Specifying the Remote Module

The location(s) the value is to be read from can be specified absolutely or using label matching - see **"Item Selection" on page 286**.

15.5.5.3 Alarm Bit Handling

Alarm status bits are not sent.

15.5.5.4 Significant Change

The module does not support significant change.

15.5.5.5 BACnet IC comms

BACnet IC comms are not supported.

15.5.5.6 COV Mode

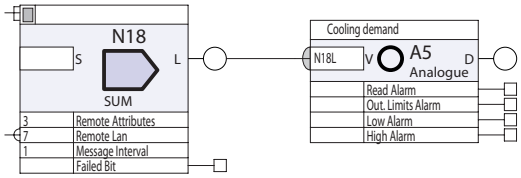
COV mode is not supported.

15.5.5.7 Example

Sum of values from LAN 7 - label item selection
This will request the sum of values from any analogue module(s) (sensor or knob) with matching label to analogue node 5 from any controller (with an attribute which matches attribute 3) on LAN 7 and write it into analogue node 5. This will be sent every minute.

Since this controller's Address module's 'Attribute 3' parameter = AHU3, it will send this message to any controller with attribute AHU3. Since the label of this controller's analogue node 5 is 'Cooling demand'. It will request the sum of the values of any analogue module(s) (sensor, or knob) with a label of 'Cooling demand'.

The IC comms module should be set up as shown.



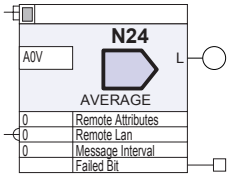
Parameter	*Value*
Comms Direction	5 (Sum)
Variable Type	Analogue
Attribute	3
Remote Lan	7
Local Module	A5V
Remote Module	0
Retransmission Interval	1

The address module's Attribute 3 must be set to 'AHU3'.

15.5.5.8 Type 5 (Sum) IC comms Module Parameters

Type 5 (Sum IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**.

15.5.6 Type 6 (Average)



Type 6 (Average) IC comms modules request the value of a particular analogue node (remote node (R)) from all controllers that match the search criteria set up in the module, calculates the average, and makes it available as an output (L).

The message has to travel from one controller to the next in the set of controllers. The controllers check for an attribute match before deciding whether they should contribute to the message and will add their value to the current value and increment the number of controllers contributing to the count. The initiating controller will calculate the average by dividing the sum by the number of controllers count on receipt of the return message.

Note: If the message is interpreted by the receiving controller as being for one of its Sensor modules and the Sensor module has a Read or Out of Limits alarm bit set it will not contribute to the reply.

15.5.6.1 Specifying the Remote Controllers

The controllers (remote controllers) from which the IQ4 requests the value are identified by an address attribute; the message can be sent to a single LAN, not every LAN - see **"Label Matching Controller Selection" on page 284**.

15.5.6.2 Specifying the Remote Module

The location the value is to be written to can be specified absolutely or using label matching - see **"Item Selection" on page 286**.

15.5.6.3 Alarm Bit Handling

Alarm status bits are not sent.

15.5.6.4 Significant Change

The module does not support significant change.

15.5.6.5 BACnet IC comms

BACnet IC comms are not supported.

15.5.6.6 COV Mode

COV mode is not supported.

15.5.6.7 Example

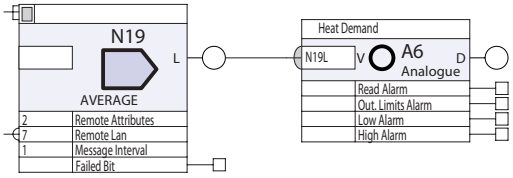
Average of values from LAN 7 - label item selection

This will request the average of values from any analogue module(s) (sensor or knob) with a label that matches with the label of analogue node 6 from any controller (with an attribute which matches attribute 2) on LAN 7 and write it into Analogue Node module 6. This will be sent every minute.

Since this controller's Address module's 'Attribute 2' parameter = 'XYZ Co', it will send this message to any controller with attribute 'XYZ Co'. Since the label of this controller's analogue node 6 is 'Heat Demand'. It will request the average of the values of any analogue module(s) (sensor, or knob) with label 'Heat Demand'.

With an average type message the returning message actually contains the sum plus the number of contributors to the sum, and the average is calculated by the IC comms module by dividing the sum by the number of contributors. The resulting average value is placed in the specified local node.

The IC comms module should be set up as shown.



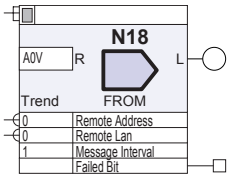
Parameter	*Value*
Comms Direction	6 (Average)
Variable Type	Analogue
Attribute	2
Remote Lan	7
Local Module	A6V
Remote Module	0
Retransmission Interval	1

The Address module's 'Attribute 3' parameter must be set to 'XYZ Co'.

15.5.6.8 Type 6 (Average) IC comms Module Parameters

Type 6 (Average) IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**.

15.5.7 Type 7 (From Analogue)



Type 7 (From Analogue) IC comms modules enable the IQ4 to request an analogue value (remote analogue) from another IQ controller and make it available as an output (L).

Note: In general it's better to use a To in the remote controller rather than a Data From in the local controller as a Data From requires 2 messages whereas a Data To only requires one, and when sending a Data To, the sender can choose to send when a significant change is exceeded whereas a Data From has to be sent at regular intervals.

15.5.7.1 Specifying the Remote Controller

The controller to which the value is sent is specified by the 'Remote Lan' and 'Remote Controller Address' parameters - see **"Absolute Controller Selection" on page 284**.

15.5.7.2 Specifying the Remote Module

The location the value is to be written to can be specified absolutely - see **"Item Selection" on page 286**.

15.5.7.3 Alarm Bit Handling

If sending an analogue value and the point in the remote controller is an analogue node the alarm status bits (ROLH) are also sent if available. They aren't sent if the remote point is not an analogue node). These bits would be exposed by the receiving analogue node.

If a reply to a Data From is not received after 3 tries (about 15s), the Failed bit and Read alarm status bit are also set.

15.5.7.4 Significant Change

The module does not support significant change.

15.5.7.5 BACnet IC comms

BACnet IC comms are supported - see **"BACnet IC Comms" on page 284.**

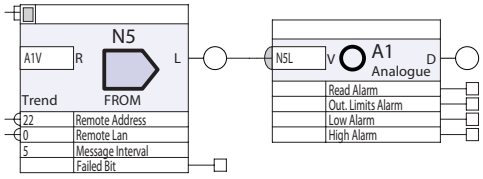
15.5.7.6 COV Mode

COV mode enables the IQ4 to receive Change Of Value (COV) reports from another BACnet device - see **"COV Service" on page 282**.

15.5.7.7 Example

Analogue Data From Controller 22 on Local LAN - analogue node 1

Requests the value and alarm status bits of analogue node 1 in controller 22 on the Local LAN every 5 mins. An IQ3 or IQ4 receiving this request would need Analogue Node module 1 to be created in it strategy.



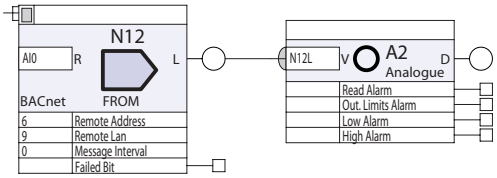
The IC comms module should be set up as shown.

Parameter	*Value*
Protocol	Trend
Comms Direction	7 (Analogue Data From)
Remote Controller Address	22

Parameter	*Value*
Remote Lan	0
Local Module	A1V
Remote Analogue	A1V
Retransmission interval	5

BACnet Data From Controller 6 LAN 9 - analogue input 0

Requests the value of analogue input 0 from the BACnet device specified by the NTD module set up as controller 6 on LAN 9 every 10 mins.



The IC comms module should be set up as shown.

Parameter	*Value*
Protocol	BACnet
Comms Direction	7 (Analogue Data From)
Remote Controller	6
Remote Lan	9
Local Module	A2V
Remote Module	AI0
Retransmission Interval	10

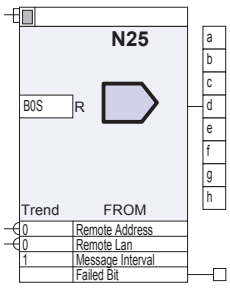
Note: A Non-Trend Device module would have to be set up for controller 6 on LAN 9; this module will convert the Trend address to the BACnet address

15.5.7.8 Type 7 (From Analogue) IC comms Module Parameters

Type 7 (From Analogue) IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
COV Mode	Enables/disables COV mode - see **"COV Service" on page 282.**	0 or 1 1 = Enabled 0 = Disabled	0	N	C
Protocol	The protocol bring used for the IC comms	0 or 1 0 = Trend 1 = BACnet	0	N	P
Remote Controller Address	Address of controller being containing the Remote Module on the LAN.	1, 4 to 119 excluding 10	0	S, N	A

15.5.8 Type 8 (From Byte)



Type 8 (From Byte) IC comms modules enable the IQ4 to request the value of a particular byte (remote byte) from another IQ controller and make it available as an output (L).

Note: In general it's better to use a To in the remote controller rather than a Data From in the local controller as a Data From requires 2 messages whereas a To only requires one, and when sending a To, the sender can choose to send when a significant change is exceeded whereas a Data From has to be sent at regular intervals.

Note: As the IQ4's digital Byte module does not have a strategy input, Type 8 (From Byte) IC comms between IQ3s and IQ4s are of little use.

15.5.8.1 Specifying the Remote Controller

The controller to which the value is sent is specified by the 'Remote Lan' and 'Remote Controller Address' parameters - see **"Absolute Controller Selection" on page 284**.

15.5.8.2 Specifying the Remote Module

The location the value is to be written to can be specified absolutely - see **"Item Selection" on page 286**.

15.5.8.3 Alarm Bit Handling

Alarm status bits are not sent.

15.5.8.4 Significant Change

The module does not support significant change.

15.5.8.5 BACnet IC comms

BACnet IC comms are not supported.

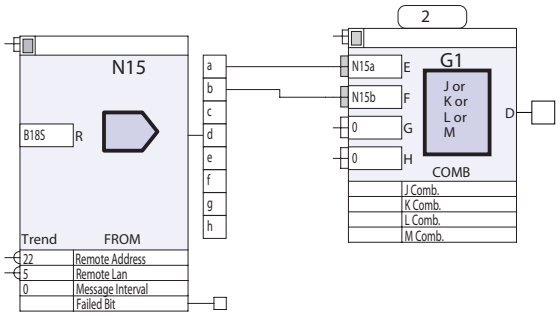
15.5.8.6 COV Mode

COV mode is not supported.

15.5.8.7 Example

Digital Byte Data From Controller 22 on LAN 5 - digital byte 18

Requests the value of controller 22 byte 18 on LAN 5 every 5 mins and makes it available as 8 output bits from the IC comms module.



The IC comms module should be set up as shown.

Parameter	***Value***
Protocol	Trend
Comms Direction	8 (Digital Byte Data From)
Remote Controller Address	22
Remote Lan	5
a	G1E
b	G1F
Remote Byte	B18S
Retransmission interval	5

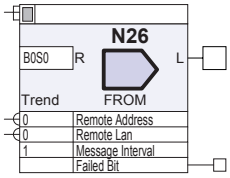
Note: If controller 22 on LAN 5 is an IQ2 this is fine, but if it is an IQ3 or IQ4 then Byte 18 in controller 22 cannot be set from the strategy. For an IQ3 or IQ4 controller 22 would have to have Digital Byte module 18 created in its strategy.

15.5.8.8 Type 8 (From Byte) IC comms Module Parameters

Type 8 (From Byte) IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
BIT0	Digital status of output bit 0.	0 or 1	0	O	a
BIT1	Digital status of output bit 1.	0 or 1	0	O	b
BIT2	Digital status of output bit 2.	0 or 1	0	O	c
BIT3	Digital status of output bit 3.	0 or 1	0	O	d
BIT4	Digital status of output bit 4.	0 or 1	0	O	e
BIT5	Digital status of output bit 5.	0 or 1	0	O	f
BIT6	Digital status of output bit 6.	0 or 1	0	O	g
BIT7	Digital status of output bit 7.	0 or 1	0	O	h
Remote Controller Address	Address of controller containing the remote module on the LAN.	1, 4 to 119 excluding 10	0	I	A

15.5.9 Type 9 (From Bit)



Type 9 (From Bit) IC comms modules enable the IQ4 to request the value of a of a particular bit (remote bit) from another IQ controller and make it available as an output (L) - see **"Initiating IC comms to Other Devices" on page 280** for compatibility.

Note: In general it's better to use a To in the remote controller rather than a Data From in the local controller as a Data From requires 2 messages whereas a To only requires one, and when sending a To, the sender can choose to send when a significant change is exceeded whereas a Data From has to be sent at regular intervals.

Note: As the IQ3's and IQ4's Digital Byte module does not have a strategy input, Type 9 (From Bit) IC comms between IQ3s and IQ4s are of little use. However when requesting Switch or Digital Input module status, they are of more use as Switch and Digital Input modules do have a strategy input.

15.5.9.1 Specifying the Remote Controller

The controller to which the value is sent is specified by the 'Remote Lan' and 'Remote Controller Address' parameters - see **"Absolute Controller Selection" on page 284**.

15.5.9.2 Specifying the Remote Module

The location the value is to be written to can be specified absolutely - see **"Item Selection" on page 286**.

15.5.9.3 Alarm Bit Handling

Alarm status bits are not sent. If a reply to a Data From is not received after 3 tries (about 15s), the Failed bit is set.

15.5.9.4 Significant Change

The module does not support significant change.

15.5.9.5 BACnet IC comms

BACnet IC comms are supported - see **"BACnet IC Comms" on page 284.**

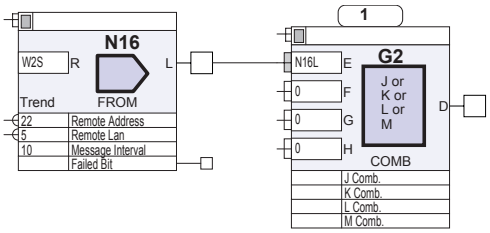
15.5.9.6 COV Mode

COV mode enables the IQ4 to receive Change Of Value (COV) reports from another BACnet device - see **"Controller Selection" on page 284**.

15.5.9.7 Example

Digital Bit Data From Controller 22 on LAN 5 - switch 2

This will request the status of controller 22's W2 on remote LAN 5 every 10 mins and make it available as a digital output.



The IC comms module should be set up as shown.

Parameter	*Value*
Protocol	Trend
Comms Direction	9 (Bit Data From)
Remote Controller Address	22
Remote Lan	5
Local Module	G2E
Remote Bit	W2S
Retransmission interval	10

Note: IQSET will offer a byte/bit item selection, and if controller 22 on LAN 5 is an IQ2 this is fine, but if it is an IQ3 or IQ4 then the byte in controller 22 cannot be set from the strategy. Also for an IQ3 or IQ4, controller 22 would have to have Digital Byte module 18 created in its strategy.

15.5.9.8 Type 9 (From Bit) IC comms Module Parameters

Type 9 (From Bit) IC comms modules include the parameters common to all IC comms modules - see **"IC Comms Modules Common Parameters" on page 305**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
COV Mode	Enables/disables COV mode - see **"COV Service" on page 282.**	0 or 1 1 = Enabled 0 = Disabled	0	N	C
Protocol	The protocol bring used for the IC comms.	0 or 1 0 = Trend 1 = BACnet	0	N	P
Remote Controller Address	Address of controller containing the Remote Module on the LAN.	1, 4 to 119 excluding 10	0	I	A

15.6 IC Comms Modules Common Parameters

IC comms modules also include the following parameters that are common to most modules: 'Disable Module', 'Label', 'Page', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Attribute	Not Types 1, 7, 8, or 9. Attribute used to select the controller(s) containing the remote module on the LAN. The selection is made by a label match between the remote attribute and one of local controller's attributes - see **"Label Matching Controller Selection" on page 284**.	1, 2, 3, 4, 5, 6, 7 1 = Identifier 2 to 7 = Attributes 2 to 7 0 = disabled	0	N	B

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
BACnet Write Priority	The priority attached to the BACnet Data To and Global To IC comms. It enables the receiving device to decide whether to change the value or not. 0 writes directly to the value, Used for points without a priority array	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 1 = highest 16 = lowest	9	N	w
Comms Direction	The type of IC comms module.	1, 2, 3, 4, 5, 6, 7, 8, 9 1 = Type 1 (To) 2 = Type 2 (Global To) 3 = Type 3 (Min) 4 = Type 4 (Max) 5 = Type 5 (Sum) 6 = Type 6 (Average) 7 = Type 7 (From Analogue) 8 = Type 8 (From Byte) 9 = Type 9 (From Bit)	-	N	D
Failed Bit	Indicates the IC comms has failed, if the time for an IC comms retry has been reached as described in **"Retries and Failed Bit" on page 282**,	0 or 1 0 = OK 1 = Failed	0	O	F
Local Module	The status or value of the local point to or from which the communication is to occur. For Type 1 (To) or Type 2 (Global To) it can be either a source or constant input; it can be connected to either an analogue output for analogue or digital byte comms, or digital output for digital bit comms. For Type 8 (From Byte) it is replaced by output BITs 0 to 7. For other Type 7 (From Analogue), Type 9 (From Bit) or Type 3 (Min), Type 4 (Max), Type 5 (Sum), and Type 6 (Average) it can be connected to an analogue input for analogue comms, or digital input for digital bit comms.	0 or 1 Or Full analogue	0	I/O	L
Remote Lan	The LAN of the controller containing the Remote Module. For Trend protocol Type 2 (Global To) it can be set to 128 which will send the IC comms message to all LANs.	0, 1, 4 to 119, 128 excluding 10 0 = Local LAN 128 = all LANs	0	I	N
Remote Module	Remote point with to or from which the communication is to occur see **"Item Selection" on page 286**. If blank the label of the local module is used for selection.	Valid module or blank	Blank	N	R
Retransmission Interval	The period (minutes) between instances of sending IC comms messages - see **"Retransmission Interval and Significant Change" on page 281** for more details.	0 to 1440	1	N	I
Time into Interval	Number of minutes since start of retransmission interval.	0 to +3276	0	O	T

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Variable Type	Not type Type 7 (From Analogue), Type 8 (From Byte), or Type 9 (From Bit). Type of variable being transferred. Type 3 (Min), Type 4 (Max), Type 5 (Sum), and Type 6 (Average) are analogue only. BACnet IC comms can only be analogue or digital bit.	0, 1, 2 0 = analogue 1 = digital byte 2 = digital bit	-	N	V

15.7 Differences Between IQ4 and IQ2 IC Comms

- The Digital Byte module in IQ4 has does not have a strategy input so the Digital Byte Data From between IQ3s and IQ4s is of little use.
- Type 1 (To), Type 2 (Global To), Type 7 (From Analogue), Type 8 (From Byte), and Type 9 (From Bit) IC comms messages can be addressed to points other than analogue nodes, or digital nodes or bytes.
- For an IQ4 to receive IC comms addressed to analogue nodes, digital nodes or bytes the analogue node or digital byte must be created in the strategy, in IQ2 they were all available by default. In IQ4 for Type 1 (To), Type 2 (Global To) and Byte communications the byte value has to be created from the digital bits by a D to A logic module or gate function modules.
- IQ2 was able to send additional status bits along with an analogue IC comms. IQ4 will only send the first 4 bits (bits 0 to 3).
- IQ2 was able to use zone labels as device attributes. This is not supported on IQ4.
- The IC comms fail bits are available as module outputs in IQ4; they were available in the digital array in IQ2.
- IQ4 is able to supports BACnet IC comms (not available in IQ2).
- IQ2 is able to select drivers by label matching. This is not supported in IQ4.

15.8 IC Comms Module Connectives

The following parameters are available as connectives for all types of IC comms modules:

Failed Bit

The following parameters are available as connectives for types 3, 4, 5, 6, 7, 8, and 9 IC comms modules:

Local module

The following parameters are available as connectives for type 8 IC comms modules:

Bit 0
Bit 1
Bit 2
Bit 3
Bit 4
Bit 5
Bit 6
Bit 7

16 INTERFACE MODULES

Module Identifier	t
Default overview parameters	$
Size in brIQs	-

16.1 Functionality

Interface modules enable the IQ4 to connected to other devices.

16.2 Interface Module Types

There are several Interface module types specified using the 'Program Type' parameter:

Module Type	*Description*
Type 1 (XNC)	/XNC variants only. Provides the ability to connect between the IQ4's strategy and the TCL application.
Type 2 (Wallbus)	Provides the ability to connect Wallbus devices to the IQ4.
Type 3 (Modbus)	/INT variants only. Provides the ability to connect Modbus devices to the IQ4.
Type 4 (M-bus)	/INT variants only. Provides the ability to connect M-bus devices to the IQ4.

16.2.1 Type 1 (XNC)

Type 1 (XNC) Interface modules are only available on /XNC variants, they are always Interface module 1 (t1). They provide the ability to connect between the IQ4's strategy and the TCL application. The /XNC functionality is described in "XNC Facility" on page 107.

Type 1 (XNC) Interface modules contain the TCL Application, the Stores and SavedVars areas, and the inputs and outputs.

Type 1 (XNC) Interface modules are not placed on the strategy page and are not sequenced.

16.2.1.1 TCL Application

This is the TCL code that has been written for the required application using TCLTool2. It obtains information by reading from the inputs, stores, non-volatile data areas and by communicating with 3rd party systems. It processes the information and updates the outputs and sends data by communications to 3rd party systems and/or Trend devices.

16.2.1.2 Inputs

Inputs enable values from the strategy to be used by the TCL application i.e. information is input to the TCL application from the strategy. They consist of up to *1400 points that can be connected to module outputs using IQSET. Each input has a link from a parameter in the strategy, a label, and a value.

The link determines the parameter in the strategy to which the input is linked e.g. if an input had a link of K1(V) it would be linked to the value of knob 1 in the strategy. The label provides a description of the link e.g. link to knob 1's value. To ensure good practice and permit the label to be access by text comms the label should not include any of the following characters such as ()^*/+-=<>\;". The input's label is not the label of the parameter to which it is linked. The value is the value of the parameter in the strategy to which the input is linked, and will change as the value in the strategy changes. Inputs are defined in the Type 1 (XNC) Interface module using TCLTool 2, and cannot be changed by the TCL application.

16.2.1.3 Outputs

Outputs enable values from the TCL application to be used by the strategy i.e. information is output from the TCL application to the strategy. They consist of up to *1400 points that can be connected to module inputs using IQSET. Each output has a link to a parameter in the strategy, a label, and a value.

The Type 1 (XNC) Interface module's outputs enable values from the TCL application to be used by the strategy i.e. information is output from the TCL application to the strategy. Each output has a link to a parameter in the strategy, a label, and a value.

The link determines the parameter in the strategy to which the output is linked e.g. if an output had a link of S1(V) it would be linked to the value of sensor 1 in the strategy. The label provides a description of the link e.g. link to sensor 1's value. To ensure good practise and permit the label to be access by text comms the label should not include any of the following characters such as ()^*/+-=<>\;". The output's label is not the label of the parameter to which it is linked. The value is the value of the parameter in the strategy to which the output is linked, and will change as the value in the strategy changes. Outputs are defined in the Type 1 (XNC) Interface module using TCLTool 2, and cannot be changed by the TCL application.

**Note: There can only be up to a total of 1400 inputs or outputs taken together (e.g. 1200 inputs and 200 outputs).*

16.2.1.4 Stores

Stores provide user configurable constants that can be used by the TCL application. There can be up to 200 stores, each has a label, and a value. Stores are defined in the Type 1 (XNC) Interface module using TCLTool 2, and cannot be changed by the TCL application. The label is used to identify the Store and is used by the TCL code to reference the Store's value. This label can be used in the TCL application as a variable name. In this case the label must not begin with a number or include spaces. To ensure good practise and permit the label to be accessed by text comms the label should not include any of the following characters such as ()^*/+-=<>\;". If the Store's value is a string it must only contain ASCII characters. The value is the constant value, and can be a string, integer, or float. They are re-initialised after power failure.

16.2.1.5 SavedVars

The SavedVars provide a location for the storage of string and numeric values that are not lost between program execution cycles, or an interruption in power that may be written to and read from by the TCL application.

There can be up to 100 SavedVars; each one contains a string and a numeric value. The string value should be used if a string needs to be available to the program between executions, it cannot contain null characters. A string containing null characters will be truncated while the numeric value should be used for integer or floating values.

Note: The numeric value of a SavedVar is stored immediately after a change, however the string value is not. The SavedVar's string values are stored in the strategy file 15 seconds after editing finishes. In some situations the editing may not finish e.g. IC comms, in this case the string values are stored to the strategy file every 10 minutes. If nothing happens to cause the string values to be stored they will be stored to the strategy file every 24 hours. Therefore in the event of a power failure it is possible that the SavedVars string values could be up to 24 hours out of date.

16.2.1.6 Communications

These enable information to be read from and written to 3rd party systems and Trend devices.

16.2.1.7 Alarm Generation

The TCL application can generate alarms using TCL's GENALARM function. There are six alarm labels that can be set up which will act as the alarm description in the alarm message.

16.2.1.8 Network Type

The 'Network' parameter determines which network the interface module will connect to.

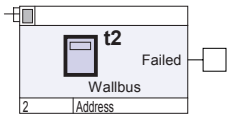
16.2.1.9 Type 1 (XNC) Interface Module Parameters

Type 1 (XNC) Interface modules include the parameters common to all Interface modules - see **"Interface Module Common Parameters" on page 328**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm Label 1	The alarm labels used for the TCL's GENALARM function.	0 to 30 character string except \ /	Blank	N	U1
Alarm Label 2					U2
Alarm Label 3					U3
Alarm Label 4					U4
Alarm Label 5					U5
Alarm Label 6					U6
Alarm Reporting Lan	The LAN number of device to which the TCL application will report software errors while being tested or debugged. Set up and cleared down by TCLTool2.	1, 4 to 119 excluding 10	Blank	N	R
Alarm Reporting OS	The device address of device to which the TCL application will report software errors while being tested or debugged. Set up and cleared down by TCLTool2.	1, 4 to 119 excluding 10	Blank	N	A
Current Line	The current line in the TCL application, only valid when the TCL application is in debug mode, or the application has stopped at this line due to an error.	1 to 1000	0	N	L
Debug Active	Indicates if TCLTool2 is running the TCL application (e.g. debugging).	0 or 1 0 = Not active 1 = Active	0	N	D
ID	The TCL application identifier generated in TCLTool2.	-	-	N	m
Input Label	The label of the Input. Each input has this parameter, 'n' is input number.	0 to 30 character string.	Null	N	In$
Input Value	The value of the Input. Each input has this parameter, n' is output number.	- full analogue to + full analogue	0	I	InV
Last Error	Number and description of last error found - see IQ4/XNC TCL Manual (TE201360). Runtime errors cause the TCL application to stop. Runtime warnings allow application to continue.	0 to 128	Blank	N	l
Number of Lines	Number of lines of code in TCL application.	1 to 1000	0	N	c
Output Label	The label of the output. Each output has this parameter, 'n' is output number.	0 to 30 character string.	Null	N	On$
Output Value	The value of the output. Each output has this parameter, n' is output number.	v3.34 or less - full analogue to + full analogue v3.40 or greater Double precision	0	O	OnV
Program State	The current state of the TCL application. Used in debugging.	1, 2, 3, 4, 5, 6, 7, 8, 9, and 10 1 = Stopped 2 = Loading 3 = Running 4 = At breakpoint 5 = Debug 6 = Error 7 = No File 8 = File not found 9 = Bad File 10 = brIQs exceeded	3	N	P

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Store-Name	The name of the Store. (‘n’ is store number 1 to 200).	0 to 30 ccharacter string.	Null		Sn$
Store-String Value	The value of the Store.	0 to 255 character string.	Null		SnV

16.2.2 Type 2 (Wallbus)



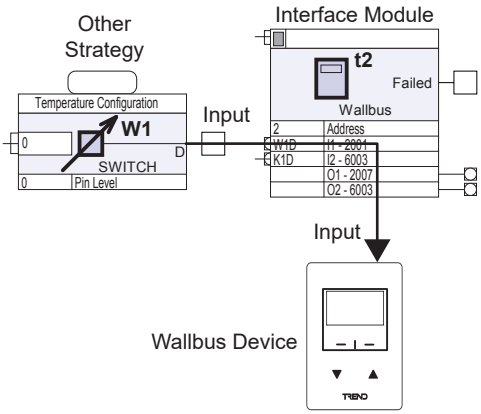
Type 2 (Wallbus) Interface modules enable a Wallbus device to be connected to the IQ4. A different instance of the Type 2 (Wallbus) Interface module is required for each Wallbus device connected to the IQ4. Type 2 (Wallbus) Interface modules appear on the strategy page and do not need to be sequenced.

16.2.2.1 Network Type

The ‘Network’ parameter determines which cable the Wallbus device is connected to. This must be set to the module number of the Type 14 (Wallbus) Network module.

16.2.2.2 Inputs

These consist of up to *1000 points that can be connected to module outputs in order to make the value in the IQ4’s strategy available to the connected Wallbus device.



In the example above W1 is connected to the Type 2 (Wallbus) Interface module’s input 1, which is linked to PV 2001 in the wallbus device. Adjusting W1 will change the value in the Wallbus device.

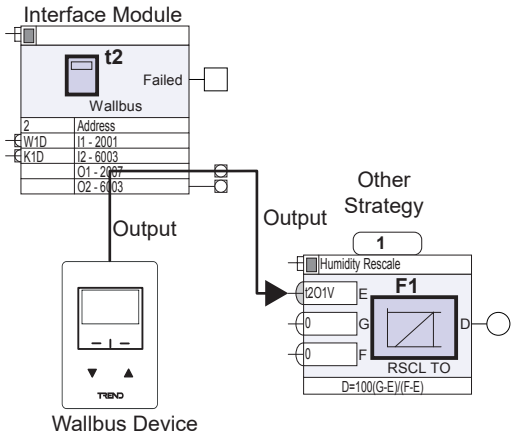
The connection of the Interface module’s input to the value it is to write to in the Wallbus device is made by connecting it to the appropriate Process Variables (PV) in the Wallbus device. The PV is defined by the ‘Inputs (PV Index)’ parameter for the input. For details of the PVs used by each Wallbus device refer to the Wallbus device’s data sheet.

The rate at which the value of the input is written to the connected PV in the Wallbus device is determined by the input’s ‘Inputs (Refresh Period)’ parameter and can be different for each input. The input will also be written to the PV when the value of the input changes by a certain amount (significant change), this amount is determined by the input’s ‘Inputs Significant change’ parameter.

**Note: There can only be up to a total of 1000 inputs or outputs taken together (e.g. 300 inputs and 700 outputs).*

16.2.2.3 Outputs

These consist of up to *1000 points that can be connected to module inputs in order to make a value from the connected Wallbus device available in the IQ4's strategy.



In the example above PV 2007 in the Wallbus device is linked to the Type 2 (Wallbus) Interface module's output one which is connected to a Function module. This allows the value from the wallbus device to be passed to the strategy in the IQ4.

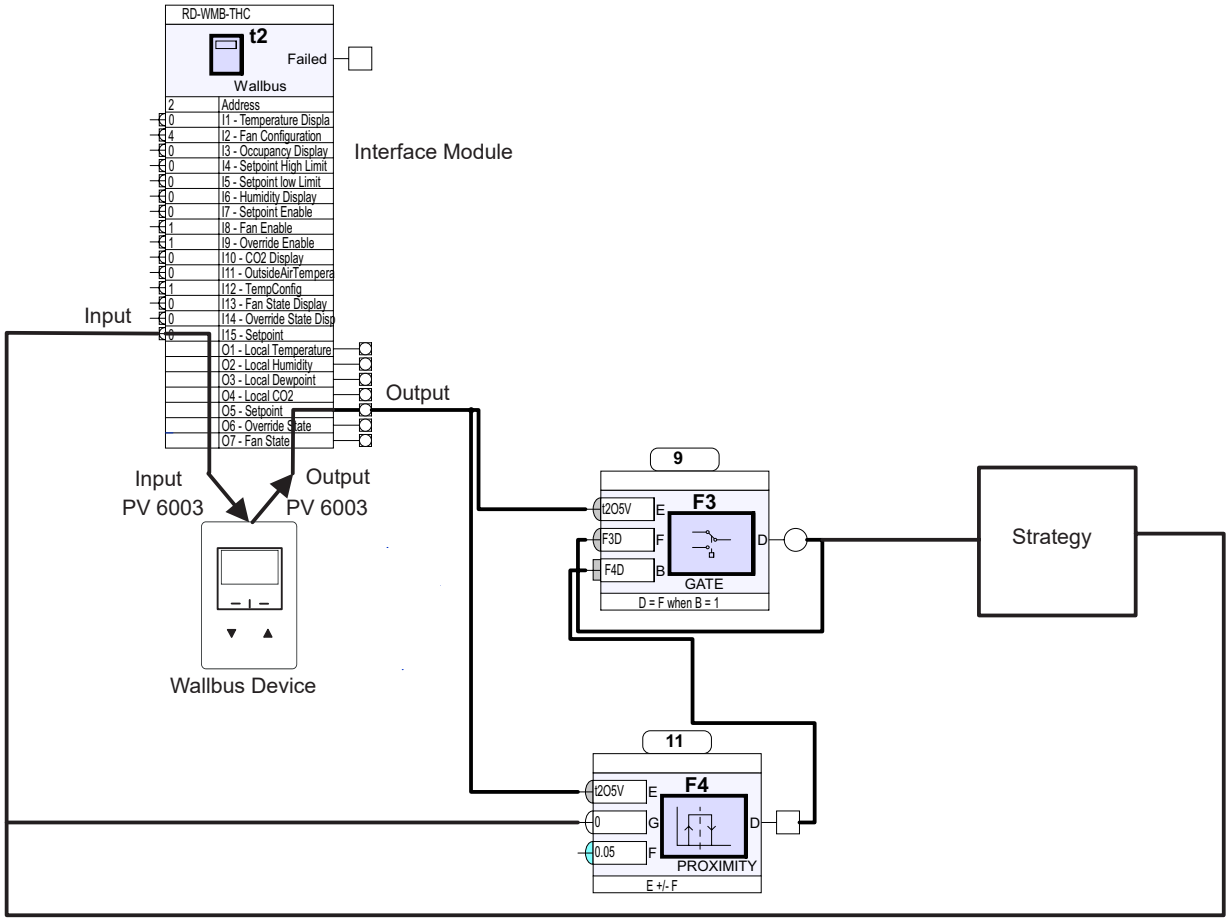
The connection of the Interface module's output to the value it is to read from in the Wallbus device is made by connecting it to the appropriate Process Variables (PV) in the Wallbus device. The PV is defined by the 'Outputs (PV Index)' parameter for the output. For details of the PVs used by each Wallbus device refer to the Wallbus device's data sheet.

If the output does not receive an update from Wallbus device for a specified period of time, specified by the output's 'Outputs Expiry Time' parameter, the value of the Interface module's output will be set to 'FL_INVALID'.

**Note: There can only be up to a total of 1000 inputs or outputs taken together (e.g. 300 inputs and 700 outputs).*

16.2.2.4 Linked Inputs/Outputs

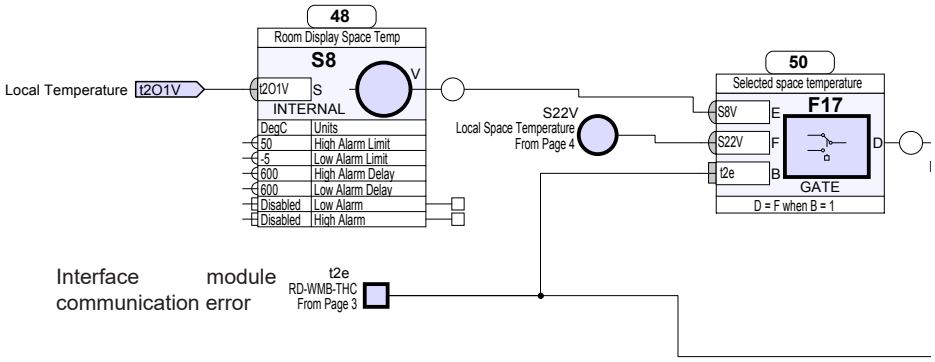
A single PV can be associated with both an input and an output to create bi-directional behaviour. If the strategy and the wallbus device can both be used to set these values, the strategy will select the last value from either source in order to be displayed and to be used further down the strategy.



In the example above PV 6003 is linked to both input 15 and output 5 enabling the value to be adjusted by both the Wallbus device and the IQ4. The setpoint value from the RD (t2O5) is connected to F3E and F4E. F4 compares the old value with the new value and if the difference in values is greater than 0.05, the new value will be switched through the gate, F3, and used in the following strategy which will produce a new value for display (t2I15).

16.2.2.5 Use of the Failed Bit

The Interface module has a failed bit which indicates that the communications to the wallbus unit have failed. This can be used to switch default values for the wallbus unit outputs (temperature, humidity, dewpoint, CO_2). This applies to wallbus sensors and displays.



In this example the output of F17 is normally the value from the wallbus device (t201V), but if the interface communications fail, the interface module error output (t2e) will cause F17 to select a separate local space sensor (S22).

16.2.2.6 Alarm Generation

Type 2 (Wallbus) Interface modules do not generate alarms.

16.2.2.7 Module Disable

Supported - see **"Disabling Modules" on page 29**.

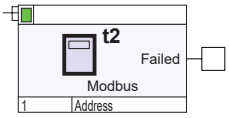
16.2.2.8 Type 2 (Wallbus) Interface Module Parameters

Type 2 (Wallbus) Interface modules include the parameters common to all Interface modules - see **"Interface Module Common Parameters" on page 328**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Address	The address determines which Wallbus device is associated with this module. A total of 14 devices may logically be attached to a single cable. Address is 4 bit (0-15). 0 is reserved, and the controller itself has an address of 1 by default, but that may be overridden in the network module. In practice the maximum number of devices fitted will be lower due to Wallbus current limitations.	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 0 and 1 = reserved	2	N	d
Device Firmware Version	Read only. The firmware version of the Wallbus device connected using the module, read from the device.	-	-	C	C
Device Profile	The name of the device profile. A profile is a specific set of parameter values that is matched to a particular function on a specific type of Wallbus device. E.g. there will be a profile for RD-WMB that defines the appropriate inputs and outputs.	-	-	N	f
Device Type	Read only. Type of the Wallbus device connected using the module, read from the device.	-	-	C	T
Disable Module	Enables/disables the module.	0 or 1 0 = Disabled 1 = Enabled	0	I	~
Filename	The name of the configuration file to be sent to Wallbus device. For non-programmable devices, leave this parameter empty.	-	-	N	F
Input Label	The label of the Input. Each input has this parameter, 'n' is input number.	0 to 30 character string.	Null	N	In$
Input Value	The value of the Input. Each input has this parameter, n' is output number.	- full analogue to + full analogue	0	I	InV
Inputs (PV Index)	The PV in the Wallbus device to which the input of the Interface module is connected. Each input has this parameter, n determines the input.	-	-	N	InN
Inputs (Refresh Period)	Defines the interval at which the Interface module writes its value to the PV in the Wallbus device. Each input has this parameter, n determines the input.	0s, 5s, 10s, 15s, 20s, 25s, 30s, 35s, 40s, 45s, 50s	-	N	InP
Inputs (Significant change)	Defines the amount by which the value of the input must change before the value is written to the PV in the Wallbus device. Each input has this parameter, n determines the input.	-	-	N	InC

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Output Label	The label of the output. Each output has this parameter, 'n' is output number.	0 to 30 character string.	Null	N	On$
Output Value	The value of the output. Each output has this parameter, n' is output number.	v3.34 or less - full analogue to + full analogue v3.40 or greater Double precision	0	O	OnV
Outputs (Expiry time)	The time, in seconds, that the value of the output is held for after the last communication from the Wallbus device. Each output has this parameter, n determines the output. '0' holds indefinitely. Any other value, nn, will cause the output's value to be set to 'FL_INVALID' nn seconds after the last update is received from the Wallbus device, where nn is the value of this parameter.	0s, 5s, 10s, 15s, 20s, 25s, 30s, 35s, 40s, 45s, 50s	-	N	OnM
Outputs (PV Index)	The PV in the Wallbus device to which the output of the Interface module is connected. Each output has this parameter, n determines the output.	-	-	N	OnN
Status	The Status of interface device. Not Present: No communication has been seen from the device since power up. Operational: The device is communicating. Not Communicating: The device was communicating since power up, but has now stopped.	0, 1, 2 0 = Not Present 1 = Operational 2= Not Communicating	0	N	t

16.2.3 Type 3 (Modbus)



/INT variants only. Type 3 (Modbus) Interface modules enable a Modbus device to be connected to the IQ4. A different instance of the Type 3 (Modbus) Interface module is required for each Modbus device connected to the IQ4. Type 3 (Modbus) Interface modules appear on the strategy page and do not need to be sequenced.

Standard strategy blocks, available in IQSET, allow easy strategy configuration using drag & drop for common meters and VSDs - see **https://partners.trendcontrols.com/partners/index.aspx#Downloads:SET** for the latest plant blocks.

16.2.3.1 Network Type

The 'Network' parameter determines which port the Modbus Interface module is using. This must be set to the module number of the Type 8 (Modbus Serial), or Type 19 (Modbus IP) Network module as required.

16.2.3.2 Inputs

These consist of up to 500 points that can be connected to module outputs in order to make the value in the IQ4's strategy available to the connected Modbus device.

16.2.3.3 Outputs

These consist of up to 500 points that can be connected to module inputs in order to make a value from the connected Modbus device available in the IQ4's strategy.

16.2.3.4 Use of the Failed Bit

The Interface module has a failed bit which indicates that the communications to the Modbus device have failed. This can be used to switch default values for the Modbus device's outputs.

16.2.3.5 Alarm Generation

Type 3 (Modbus) Interface modules do not generate alarms.

16.2.3.6 Module Disable

Supported - see **"Disabling Modules" on page 29**.

16.2.3.7 Type 3 (Modbus) Interface Module Parameters

Type 3 (M-bus) Interface modules include the parameters common to all Interface modules - see **"Interface Module Common Parameters" on page 328**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Address	The address determines which Modbus device is associated with this module.	Serial = 1 to 247 IP = 1 to 255	2	N	d
Byte Order	The byte order: Big Endian e.g. B1,B2,B3,B4 for a 32 bit integer value or B1,B2,B3,B4,B5,B6,B7,B8 for a 64 bit Integer value. Little Endian e.g. B2,B1 or B4,B3,B2,B1 or B8,B7,B6,B5,B4,B3,B2,B1 Big Endian with Word Swap e.g. B3,B4,B1,B2 Little Endian with Word Swap e.g. B2,B1,B4,B3	0, 1, 2, 3 0 - Big Endian (B1,B2,B3,B4) 1 - Little Endian (B4,B3,B2,B1) 2 - Big Endian/Word swap (B3,B4,B1,B2) 3 - Little Endian/ Word swap (B2,B1,B4,B3)	0	N	B

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Device Profile	The name of the device profile. A profile is a specific set of parameter values that is matched to a particular function on a specific type of Modbus device.	-	-	N	f
Disable Module	Enables/disables the module.	0 or 1 0 = Disabled 1 = Enabled	0	I	~
Enforce Single Write Function Codes	Specifies whether updates to registers are concatenated into multi-write function codes and will always use single write function codes.	0 or 1 0 = Disabled 1 = Enabled	0	N	w
Inputs (Data Format)	Determines how many consecutive registers are written, and how the value of I[n]V is converted to register values.	00, 01, 02, 04, 05, 07, 09, 10, 11, 12, 13 00 = single bit. A value on I[n]V >= 0.5 is converted to 1. Values < 0.5 are converted to 0. 01 = 2 byte unsigned Integer. 02 = 2 byte signed integer 04 = IEEE value. 4 byte format (float). 05 = IEEE value 8 byte format (double). 07 = BCD 2 byte integer 09 = Allows a bit mask to be applied to a 16 bit word. E.g. a bit mask retrieving bits 5 through 9 would be 0000001111100000. Bitmasks should only be defined as either a single bit or multiple consecutive bits for any given Output Instance. 10 = 4 byte unsigned integer 11 = 4 byte signed integer. 12 = 8 byte unsigned integer (64 bit) 13 = 8 byte signed Integer (64 bit)	00	N	InF
Inputs (Refresh Period)	Time in seconds between write requests. Each input has this parameter, n determines the input.	0 to 1800 0 = on change only	0	N	InP

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Inputs (Register Address)	The Modbus register address where the input's value is written. or Defines a Modbus Digital Input or Input Register address from where the input's value is written. The number of consecutive addresses written will be determined by the data format parameter for this Input. Each input has this parameter, n determines the input.	-	-	I	InN
Inputs (Register Bitmask)	Only for registers with a Data Format of 9 (Bitmask). It defines the bit(s) that will be written to when the input value changes - see **"Bitmask Values." on page 322**. When applying bit masks to data being written to the register the current value of the register must be read, and then the bit mask applied to preserve the data not affected by the bitmask. Thus only affecting the data in the bits specified. Each input has this parameter, n determines the input.	-	-	N	InM
Inputs (Scaling Factor)	The power of 10 scaling factor to apply to the register value to convert it to the required units. E.g. a scale factor of +2 would divide the value by 100, a scale factor of -2 would divide the value by 0.01. *Note: Restricted to integer values.* Each input has this parameter, n determines the input.	-20 to +20	0	N	InF
Inputs (Significant change)	The difference between the current value of I[n]V and the last transmitted value of I[n]V that will cause the new value to be sent to the Modbus device. Each input has this parameter, n determines the input.	-	-	N	InC
Inputs Base Register Address	The base value to apply to all Inputs Register Addresses. This allows register offset addressing mode to be used - see **"Scaling Factor" on page 321**.	0 to 65535	0	N	i

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Outputs (Data Format)	Determines how many consecutive registers are read, and how the values are combined to form the value of O[n]V. Each output has this parameter, n determines the output.	00, 01, 02, 04, 05, 07, 09, 10, 11, 12, 13 00 = single bit 01 = 2 byte unsigned Integer. 02 = 2 byte signed integer 04 = IEEE value. 4 byte format (float). 05 = IEEE value 8 byte format (double). 07 = BCD 2 byte integer 09 = Allows a bit mask to be applied to a 16 bit word. E.g. a bit mask retrieving bits 5 through 9 would be 0000001111100000 Bitmasks should only be defined as either a single bit or multiple consecutive bits for any given output 10 = 4 byte unsigned integer 11 = 4 byte signed integer. 12 = 8 byte unsigned integer (64 bit) 13 = 8 byte signed integer (64 bit)	-	-	OnF
Outputs (Refresh Period)	Time in seconds between read requests. Each output has this parameter, n determines the output.	0 to 1800 0 = on change only	60	N	OnP
Outputs (Register Address)	The Modbus coil or register address from where the value of O[n]V is read. The number of consecutive addresses read will be determined by the data format parameter for this output. Each output has this parameter, n determines the output.	-	-	N	OnN
Outputs (Register Bitmask)	Only for registers with a Data Format of 9 (Bitmask). It defines the bit(s) that will be extracted from the register read operation and applied to the Outputs Value - see **"Bitmask Values." on page 322**. When applying bit masks to data being written to the register the current value of the register must be read, and then the bit mask applied to preserve the data not affected by the bitmask. Thus only affecting the data in the bits specified. Each output has this parameter, n determines the input.	-	0000	N	OnM

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Outputs Base Register Address	The base value to apply to all Inputs Register Addresses. This allows register offset addressing mode to be used - see **"Scaling Factor" on page 321**.	0 to 65535	0	N	i
Outputs Register Type	Specifies whether the register can be written or just read. It is only used when the 'Register Addressing Mode' parameter is set to 1 or 2 (Extended Addressing or Extended Addressing Base 1). When set to '0' the register being addressed is a Holding Register or a Coil depending upon the output's Data Format. When set to '1' the register being addressed is a Discrete or Analogue Input depending upon the output's Data Format.	0, 1 0 = Read/Write 1 = Read Only	0	N	OnT
Outputs Scaling Factor	The power of 10 scaling factor to apply to the register value to convert it to the required units. E.g. a scale factor of +2 would multiply the value by 100, a scale factor of -2 would multiply the value by 0.01. *Note: Restricted to integer values.* Each input has this parameter, n determines the input.	-20 to +20	0	N	OnE
Register Addressing Mode	The addressing mode used by the module. When set to 1 or 2 it is combined with the 'Data Type' and 'Outputs Register Type' to determine which type of item is to be written/read. For example for a data type of single bit, and output register type of Read Only a Discrete Input register would be read. Whereas for a output register type of Read-Write a Coild would be read.	0, 1, 2 0 = Standard Modbus addressing (e.g. 40001 to 49999 for holding registers, 30001 to 39999 for analogue Input registers etc.) 1 = Extended Addressing (e.g. 0 to 65535 used to specify address. 2 = Extended Addressing Base 1. This is the same as Extended Addressing except that addresses are in the range (1 – 65536)	0	N	m

Scaling Factor

The Scaling Factor is used to specify the units that the Strategy will have for a given MODBUS Register value. So if the modbus register has the units of kWh and the Strategy requires units of Wh, then a Scaling Factor of 3 on the output would multiply the register value by 1000 to give a value of Wh on the Interface Module Output.

If the value was to be written back into the MODBUS Register, then applying a Scaling Factor of 3 to the Interface Module Input would divide the Input Value by 1000 to turn it from Wh to kWh when written to the register.

Register Offset Addressing.

By assigning a non zero value to the Base Register Address (Input or Output) every register address specified in the appropriate class (Input or Output) will have that offset added to the register address.

e.g. if the 'Inputs Register Base Address' parameter has a value of 40000, then any register address specified in the inputs will be offset by 40000, e.g. if an input[3]. registeraddress = 8, then the actual register read will be 40008.

This is useful for reading from gateways where a single Modbus UID has been used to assimilate data from multiple devices and has then been split into address ranges in the gateway. This typical in Modbus IP gateway devices that are reading from multiple Modbus RTU devices and collating the data.

This can be used in one of two ways:

- Specify a base address in the offset register, such as 40100, and all input register addresses would be values in the range 0 to 99.
- Specify a value of 100 in the offset address and all input register addresses would be start at 40000, e.g. 40000, 40001,40002 etc. This would then access registers 40100, 40101,40102 etc.

This is useful when multiple Interface modules are used with a set of pre-set register addresses, and the offset can then be set at commissioning time to allocate the modules to different devices in the Modbus Gateway.

Bitmask Values.

These are integer values that specify a bitmask to applied to the register value. They allow the specific bits to be extracted form the register value

E.g. A value that has bits 15, 0 would be OOOOOOOOOOOO, and if bit 8 was the bit of interest for the relevant input or output, the a bit mask of OOOOOOOOIOOOOOOO would need to be applied, this would be written to the Bitmask value as Hex 0x0100 or a decimal value of 256.

Similarly a Bitmask that specified bits 0 to 3 as the bits of interest would have a Bitmask value of

OOOOOOOOOOOOIIII or Hex 0x000F or Decimal Value 15.

Care needs to be taken when defining the bit field, as the order of the bits will depend upon whether the ByteOrder parameter specifies Big Endian or Little Endian formats. i.e. whether bit 15 is the left most bit or the rightmost bit.

Values that are affected by the bitmask need to be range restricted and shifted to/from the appropriate bit position to be applied to the register.

e.g. for a Input Value that applied to a single Bitmask value should be restricted to 0 or 1 (values greater than 1 should be truncated to 1) and the value shifted to the bit position that is specified by the bitmask and then written to the Modbus Register (preserving all existing bits in the register that are not specified in the bit mask).

So for an Input Value of the Interface module with a Bitmask Value of 256 (Bit 8 only), the Register Value should be read from the Modbus Device, the value should have the inverse of the bitmask 'anded' with the data to clear bit 8 and preserve all other bits, then the Input Value should be shifted up to position 8 and 'ored' with the data to be written back to the Modbus Register.

For an Output Value, a Modbus Register would be read , the Bitmask 'anded' to the data, and the value then shifted down to position 0 and written to the Outputs Value field. E.g. for a Bitmask of OOOOOIIIOOOOOOOO – Hex 0x0700 – Decimal 1792 – the value from the register would be 'anded' , to remove all data except that in bit positions 10,9 and 8 – this would then be shifted down by 8 bits to leave a Value in the range 0..7 that would be presented to the Outputs Value parameter.

16.2.3.8 Advice for Configuring MODBUS Interface Modules

16.2.3.8.1 Register Addresses as specified by the manufacturers data sheet

There does not appear to be a standard way that manufacturers assign register addresses to their data.

MODBUS has a defacto standard, which separates Coils, Discrete Inputs, Input Registers and Holding Registers into their own separate address spaces.

- Coils (digital input/outputs) are usually specified as being in the range 00001 to 09999
- Discrete Inputs (Digital Inputs) are usually specified as being in the range 10001 to 19999

- Input registers (analogue inputs) are usually specified as being in the range 30001 to 39999
- Holding Registers (analogue input/outputs) are usually specified as being in the range 40001 to 49999

However, when this information is put onto the wire, the MODBUS driver will convert the register address into a Function Code and Offset.

Different Function codes apply to different register types, and the offset is in the range 0 to 65535.

e.g. Coil 00001 will be requested using Function Code 01 with and Offset address of 0.

Digital Input 10010 will be requested using Function Code 02 and an offset address of 9.

i.e. the offset address will have one subtracted from the register address and the top digit of the register address will determine which function code to use.

Beware that some manufacturers may specify their register addresses in this format – but with the 1 already subtracted – e.g. MODBUS standard 30001 becomes 30000.

When using the "Standard" addressing mode in Trend Interface modules, this would have the 1 subtracted again when the request in sent out. Thus the wrong register would be read.

This form of register description would often be accompanied by a Hex Offset form of the address

e.g. 30010 would be offset 0A in hex. Or 30011 would be offset 0A in hex , one has the 1 subtracted , one form doesn't.

Example of a data sheet using the standard addressing model:

2819	42820	System PF	1000 = 1.000
2820	42821	Frequency	500 = 50.0
2821	42822	Phase 1 Volts	Kvp
2822	42823	Phase 1 Amps	Ki
2823	42824	Phase 1 kW	Kp
2824	42825	Phase 2 Volts	Kvp
2825	42826	Phase 2 Amps	Ki
2826	42827	Phase 2 kW	Kp
2827	42828	Phase 3 Volts	Kvp
2828	42829	Phase 3 Amps	Ki

Example of a data sheet using the hex addressing model:

Voltage	L1-N	5B00	2	0,1	V		Unsigned
Voltage	L2-N	5B02	2	0,1	V		Unsigned
Voltage	L3-N	5B04	2	0,1	V		Unsigned
Voltage	L1-L2	5B06	2	0,1	V		Unsigned
Voltage	L3-L2	5B08	2	0,1	V		Unsigned
Voltage	L1-L3	5B0A	2	0,1	V		Unsigned
Current	L1	5B0C	2	0,01	A		Unsigned
Current	L2	5B0E	2	0,01	A		Unsigned
Current	L3	5B10	2	0,01	A		Unsigned
Current	N	5B12	2	0,01	A		Unsigned

Another form of MODBUS addressing commonly used is "Extended" addressing mode. This takes the form of either a Hex or a Decimal Offset address in the range 0 to 65535 or 0 to FFFF if expressed in Hex form. This will be done commonly if the Register addresses are outside of the 01 to 9999 range that the standard mode gives for each register type.

When using this mode the data sheet should specify which register type the address is for, either by specifying the function code or the register type. This is most commonly used for addressing Holding Registers.

Example of a data sheet using the extended addressing model:

51281	0xC851	1	Phase to Phase Voltage: U12	V / 100	U16
51282	0xC852	1	Phase to Phase Voltage: U23	V / 100	U16
51283	0xC853	1	Phase to Phase Voltage: U31	V / 100	U16
51284	0xC854	1	Simple voltage : V1	V / 100	U16
51285	0xC855	1	Simple voltage : V2	V / 100	U16
51286	0xC856	1	Simple voltage : V3	V / 100	U16
51287	0xC857	1	Frequency : F	Hz / 100	U16
51288	0xC858	1	Current : I1	A / 1000	U16
51289	0xC859	1	Current : I2	A / 1000	U16
51290	0xC85A	1	Current : I3	A / 1000	U16

16.2.3.8.2 Byte Ordering of Data (especially when spanning multiple registers such as Floating Point (IEEE 754))

The Byte order for data is important to get correct along with the register address.

Most manufacturers will use Big Endian byte (most significant to least significant -MS/LS) ordering as this is the natural order that data appears on the wire, i.e. B1,B2,B3,B4 for a 4 byte data item such as an IEEE754 floating point value.

These data items span multiple registers (two registers for an 4 byte item)

Some manufacturers will have the words swapped round however, so that when specifying the register B3 & 4 are in the first register and B1 &2 are in the following register.

Address	Hex Address	Parameter
30000	0x00	Voltage V1N
30002	0x02	Voltage V2N
30004	0x04	Voltage V3N
30006	0x06	Average Voltage LN
30008	0x08	Voltage V12
30010	0x0A	Voltage V23
30012	0x0C	Voltage V31
30014	0x0E	Average Voltage LL
30016	0x10	Current I1
30018	0x12	Current I2
30020	0x14	Current I3

Note: in the example above a mixture of standard addressing and extended (Hex) addressing. However the standard addressing has been offset by -1, therefore a value of 1 would need to be added to each address if using standard mode. Not all manufacturers will however document this in their data sheet.

16.2.3.8.3 Network Communications settings

Serial RTU protocol settings will be 8 Data Bits.

Parity may or may not be applied – i.e. Parity Settings may be any of Odd,Even or None.

If Parity is set to None, some manufacturers will require two stop bits to keep the bit length of a Byte on the wire the same – i.e. 1 start + 8 data + 2 stop bits, or 1 Start + 8 Data + 1 Parity + 1 Stop bit.

Other manufacturers will not require this.

Serial ASCII protocol may only require 7 data bits, however some manufacturers may have this set at 8 data bits.

RTU and ASCII protocols cannot be mixed on the same Network segment (i.e the same network module)

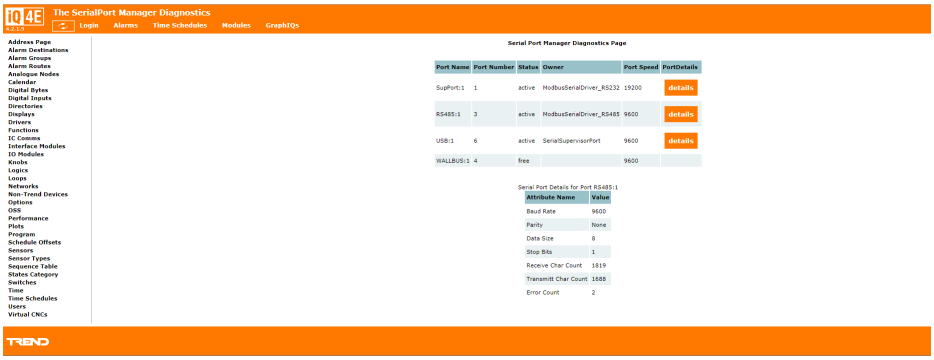
Neither can a mixture of different parity settings be mixed on the same network segment. i.e. for any given network segment – all units must be able to have the same serial settings.

16.2.3.8.4 Diagnostic Comms Captures.

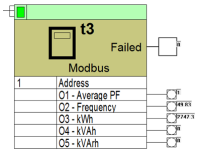
Trend provides a diagnostic overview page(Serial Port Manager Diagnostics) for both the Serial Interfaces, which will show the settings for each Serial Communication port and which Network module each port belongs to.

Trend also provides a diagnostic overview page for the Interface communications, which captures a snapshot of the last 128 modbus requests and responses. Responses are annotated if an exception response is encountered – such as unknown register address. It is also worth looking at this to determine the Endian-ness of the responses and if any WordSwaps are necessary for multi register transactions.

Example Serial Port Manager Diagnostics page



The module blow indicates values from a MODBUS device.



- Average PF = 1
- Frequency = 49.83
- KWh = 2747.3
- kVAh = 0
- kVArh = 0

The example below show the comms received for these values if configured for Big Endian with word swap comms

Interface Module 14 Log Partially Full

DateTime	Delta	Data	Exception Code
25/07/2019 11:21:56	00:00.219	Rx: 01 04 04 B4 CD 45 2B 3F 04	
25/07/2019 11:21:56	00:00.194	Tx: 01 04 00 3A 00 02 51 C6	
25/07/2019 11:21:55	00:00.218	Rx: 01 04 04 53 33 42 47 6A 5D	
25/07/2019 11:21:55	00:00.180	Tx: 01 04 00 38 00 02 F0 06	
25/07/2019 11:21:55	00:00.219	Rx: 01 04 04 00 00 3F 80 EB D4	
25/07/2019 11:21:55	00:58.969	Tx: 01 04 00 36 00 02 91 C5	
25/07/2019 11:20:56	00:00.219	Rx: 01 04 04 B4 CD 45 2B 3F 04	
25/07/2019 11:20:56	00:00.195	Tx: 01 04 00 3A 00 02 51 C6	
25/07/2019 11:20:55	00:00.218	Rx: 01 04 04 53 33 42 47 6A 5D	
25/07/2019 11:20:55	00:00.180	Tx: 01 04 00 38 00 02 F0 06	
25/07/2019 11:20:55	00:00.219	Rx: 01 04 04 00 00 3F 80 EB D4	
25/07/2019 11:20:55	00:58.969	Tx: 01 04 00 36 00 02 91 C5	

This data is in floating point format with the words swapped (B3, B4, B1, B2)

The example below show the comms received for these values if configured for Bug Endian – no word swap

Interface Module 14 Log Partially Full

DateTime	Delta	Data	Exception Code
25/07/2019 11:35:55	00:00.219	Rx: 01 04 04 45 2B B4 CD 28 15	
25/07/2019 11:35:55	00:00.195	Tx: 01 04 00 3A 00 02 51 C6	
25/07/2019 11:35:55	00:00.219	Rx: 01 04 04 42 47 53 33 23 0C	
25/07/2019 11:35:54	00:00.179	Tx: 01 04 00 38 00 02 F0 06	
25/07/2019 11:35:54	00:00.219	Rx: 01 04 04 3F 80 00 00 F6 78	
25/07/2019 11:35:54	00:58.970	Tx: 01 04 00 36 00 02 91 C5	
25/07/2019 11:34:56	00:00.219	Rx: 01 04 04 45 2B B4 CD 28 15	
25/07/2019 11:34:56	00:00.193	Tx: 01 04 00 3A 00 02 51 C6	
25/07/2019 11:34:55	00:00.219	Rx: 01 04 04 42 47 53 33 23 0C	
25/07/2019 11:34:55	00:00.179	Tx: 01 04 00 38 00 02 F0 06	
25/07/2019 11:34:55	00:00.220	Rx: 01 04 04 3F 80 00 00 F6 78	
25/07/2019 11:34:55	00:58.970	Tx: 01 04 00 36 00 02 91 C5	

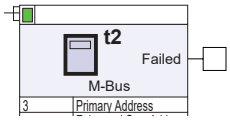
This data is in floating point format with no word swap (B1, B2, B3, B4)

Both or the examples above give the same value of 2747.3 for the KWh.

16.2.3.8.5 Cabling

It is recommend that the cable is terminated at each end using standard 120ohm 0.25 W resistor.

16.2.4 Type 4 (M-bus)



/INT variants only v4.30 or greater. Type 4 (M-bus) Interface modules enable an M-bus device to be connected to the IQ4. A different instance of the Type 4 (M-bus) Interface module is required for each M-bus device connected to the IQ4. Type 4 (M-bus) Interface modules appear on the strategy page and do not need to be sequenced.

Standard strategy blocks, available in IQSET, allow easy strategy configuration using drag & drop for common meters and VSDs - see **https://partners.trendcontrols.com/partners/index.aspx#Downloads:SET** for the latest plant blocks.

If the manufacturer's data sheet does not fully explain the registers and 3rd party program such as MB-Sheet can be used to obtain the register information from the M-bus device.

16.2.4.1 Network Type

The 'Network' parameter determines which port the M-bus Interface module is using. This must be set to the module number of the Type 15 (M-bus) Interface module.

16.2.4.2 Outputs

These consist of up to 1000 points that can be connected to module inputs in order to make a value from the connected M-bus device available in the IQ4's strategy.

16.2.4.3 Use of the Failed Bit

The Interface module has a failed bit which indicates that the communications to the M-bus device have failed.

16.2.4.4 Alarm Generation

Type 4 (M-bus) Interface modules do not generate alarms.

If the data received for an output is not valid, either because of the value received, or because there is no communication, the value of the output will be frozen and the alarm bits for the output value are set for use by any downstream modules. There is no alarm status output on the Interface module output.

If the configured index is not found the Read Alarm bit is set to TRUE. If the configured index is found but is of a non numeric data type (e.g. string) the OSL Alarm bit is set to TRUE. In both cases the value is left at the last set value.

16.2.4.5 Module Disable

Supported - see **"Disabling Modules" on page 29**.

16.2.4.6 Type 4 (M-bus) Interface Module Parameters

Type 4 (M-bus) Interface modules include the parameters common to all Interface modules - see **"Interface Module Common Parameters" on page 328**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Device Profile	The name of the device profile. A profile is a specific set of parameter values that is matched to a particular function on a specific type of device.	-	-	N	f

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Disable Module	Enables/disables the module.	0 or 1 0 = Disabled 1 = Enabled	0	I	~
ID	The identifier for the device being read.	8 digit string	-	-	i
Manufacturer Code	3-character code representing the manufacturer of the device being read.	3-character string	-	-	m
Medium	The medium the meter is monitoring e.g. Heat, Oil, Water	String	-	N	u
Outputs (Data Item Index)	Determines which of the data values returned from the device is written to O[n]V. Each output has this parameter, n determines the output.	0 to 255	-	-	OnN
Outputs Scaling Factor	The power of 10 scaling factor to apply to the register value to convert it to the required units. E.g. a scale factor of +2 would multiply the value by 100, a scale factor of -2 would multiply the value by 0.01. All values read from the meter are in a normalised form for the type of value they are reading – e.g. Wh. So if the value should be presented as KWh – the value returned from the meter would need a scaling factor of 3. *Note: Restricted to integer values.* Each input has this parameter, n determines the input.	-20 to +20	0	N	OnE
Primary Address	Address of the device to be read. 0 is for an unconfigured Interface Module and means use the Secondary Address (if configured)	0 to 250	0	N	d
Refresh Period	Time in minutes between read requests. *Note: Some battery powered meters will only respond at a minimum refresh rate* *of 2 hours (or more). Once one refresh request has been made, a further request will receive no response until the meters refresh period has elapsed.*	1 to 2400	120	N	P
Secondary Address	The serial number of device to be read.	8-decimal digit string value.	-	N	D
Version	The version number	Integer	N/A	N	v

16.3 Interface Module Common Parameters

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Failed	Indicates that comms to the interface have timed-out.	0 or 1 0 = OK 1 = timeout	0	N	e
Label	The name of the interface.	0 to 30 character string except \ /	-	N	$
Network	The network module number of the network to which the interface is to be connected. For Type 1 (XNC) Interface modules this parameter is only required when interfacing using a serial connection.	Valid module number	-	N	n
Program Type	The type of interface module.	1, 2, 3, 4 1 = XNC 2 = Wallbus 3 = Modbus 4 = Mbus	1	N	Y
Size of Modules	Size of the module including any application file (e.g. TCL application).	1 to 2000	0	N	Z
Medium	The type of medium	0 to 30 character string except \ /	-	N	u

16.4 Interface Module Connectives

The value of the outputs are available as connectives.

17 I/O MODULES

Module Identifier	in (where n is module number)
Default overview parameters	$
Size in brIQs	0

There are several types of I/O module - see **"Driver Module Types" on page 212**.

17.1 I/O Module Types

There are several I/O module types specified using the 'Sub Type' parameter:

Module Type	*Description*
Type 0 (Trend I/O)	Defines Trend I/O modules

17.1.1 Type 0 (Trend I/O)

I/O modules (strategy I/O module) specify the physical I/O modules (IQ4/IO or XCITE I/O) that are connected to an IQ4E or IQ4NC/XNC. A separate strategy I/O module must be added for each physical I/O module connected to the IQ4E to enable the channels on the physical I/O module to be connected to the appropriate Driver, Type 0 (Analogue) Sensor, Type 2 (Digital) Sensor, and Type 2 (External) Digital Input module. I/O module '0' is used for the IQ4's on-board I/O channels.

Note: External I/O modules are only available on IQ4Es but I/O module '0' exists in all IQ4s as it refers to the IQ4's on-board I/O channels.

The I/O modules do not appear on the strategy page and do not need to be sequenced.

17.1.1.1 Strategy I/O Module Creation

I/O modules can be created in one of two ways:

I/O Module Creation in IQSET
Automatic Strategy I/O Module Creation

17.1.1.1.1 I/O Module Creation in IQSET

I/O modules are configured using IQSET. Each module is given an address, module type and a 30 character label.

17.1.1.1.2 Automatic Strategy I/O Module Creation

When a module is added to the I/O bus or powered up, or if the IQ4 is powered up, the IQ4 will automatically discover the physical I/O modules connected to its I/O bus. It will attempt to match any existing strategy I/O modules to the physical I/O modules, but if these don't exist it will create I/O modules in the strategy corresponding to the physical I/O modules it has discovered.

For unaddressed IQ4/IO modules it will create strategy I/O modules for each physical I/O module (starting at I/O module 100) with the 'State' parameter set to unaddressed, and the 'Actual Type' parameter set to the type of the physical I/O module.

For IQ4/IO modules and XCITE I/O modules that are already addressed it will create strategy I/O modules for each physical I/O module with the 'Address' parameter set to the physical I/O module's address and the 'Actual Type' parameter set to the type of the physical I/O module. The module number of the strategy I/O module number is determined by the physical I/O module's address.

Note: There may be duplicate addresses on the I/O bus.

Note: Automatically created strategy I/O modules are not archived and will disappear after power cycle if the physical I/O module is no longer there. Therfore if a physical I/O module is added to the I/O bus and subsequently removed, the I/O module will be automatically created, but will only exist until the next power cycle. If an automatically created strategy I/O module is edited (e.g. by text comms) it will be archived and will persist through power cycles.

17.1.1.2 I/O Module Matching

If the IQ4E has a strategy that contains strategy I/O modules it will attempt to match the unaddressed physical I/O modules on the I/O bus with the I/O modules in the strategy.

The IQ4E compares the type of strategy I/O module with the physical I/O modules and if there is a unique match between the type of physical I/O module and the 'Configured Type' parameter of a strategy I/O module it assigns the physical I/O module to the matching strategy I/O module, giving the physical I/O module the address and label specified in the strategy I/O module.

E.g. If I/O module 1 ('Address' parameter = '1') is created in the strategy with the 'Configured Type' parameter set to 'IQ4/IO/8UI', if it is the only IQ4/IO/8UI module in the strategy, and if there is single IQ4/IO/8UI module on the I/O bus, the IQ4/IO/8UI module would be associated with strategy I/O module 1 and given the address 1 on the I/O bus.

If there is more than one physical I/O module of the same type on the I/O bus and one strategy I/O module of that type in the strategy the first physical I/O module to request an address will be addressed others will remain unaddressed.

If there is more than one physical I/O module of the same type on the I/O bus and more than one strategy I/O module of that type in the strategy the IQ4E would be unable to determine which physical I/O module an individual strategy I/O module should be associated with, and the physical I/O modules will remain unaddressed.

Any unaddressed modules must be addressed, ensuring that there are no duplicate addresses. For details of configuring the physical I/O modules - see **"Configure I/O Modules" on page 159**.

Strategy matching continues in the background and as soon as the IQ4E is able to uniquely identify an individual physical I/O module (the addressing process means there is only one unaddressed module of that type) it will match it with a strategy I/O module.

17.1.1.3 Type 0 Trend I/O Module Parameters

I/O Module modules include the following parameter that is common to most modules, 'Label', 'Page', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Actual Type	(Read only) The type of I/O module that has been read from the I/O bus by the IQ4E. This confirms that the I/O module actually exists on the I/O bus. Blank if the module has been created in IQSET but is not communicating on the I/O bus.	0 to 30 character string. Limited to the types listed by the 'Configured Type' parameter	Blank	N	A
Address	(Read only if I/O module is not present or for XCITE I/O modules). Address of module on the I/O bus. If connecting a IQ4/IO module the address is set by the strategy in the range 1 to 30. If connecting XCITE I/O module the address is set by the I/O module's address switch in the range 1 to 15 (labelled 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F). *Note: This parameter is always the same as the module number, changing it will change the module number.*	0 to 30 0 = on board I/O	N/A	N	M

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Configured Type	The type of I/O module as configured in IQSET. Blank if the module has been automatically created due to the I/O module's discovery by the IQ4 - see **"Strategy I/O Module Creation" on page 329**.	IQ4/IO/16DI IQ4/IO/8DO IQ4/IO/4DO IQ4/IO/8UIO IQ4/IO/8DI IQ4/IO/8UI IQ4/IO/8AO IQ4/IO/4UIO IQ4/IO/8DO/HOA IQ4/IO/8AO/HOA XCITE/IO/8UI XCITE/IO/4UI XCITE/IO/4UI/4AO XCITE/IO/2UI/2AO XCITE/IO/8DO XCITE/IO/4DO XCITE/IO/16DI XCITE/IO/8DI XCITE/IO/8DI/8TI XCITE/IO/8AO XCITE/IO/4AO XCITE/IO/8DO/HOA XCITE/IO/4DO/HOA	Blank	N	Y
Failed	Status of last communication between IQ4 and the physical I/O module. Set to '1' if the 'State' parameter is 'not present', 'recoverable fault', or 'unrecoverable fault'.	0 or 1 1 = Failed	0	O	F
Firmware Version	The firmware version of the I/O module discovered by communication with the module.	0 to 30 character string.	Fixed	N	C
Serial Number	The serial number of the physical I/O module, also printed on the module.	String	Factory set	N	S

<table>
<tr><th>Parameter</th><th>Description</th><th>Range</th><th>Default</th><th>Type</th><th>Text Comms</th></tr>
<tr><td>State</td><td>The communication state of the I/O module. It is first set up when the IQ4E automatically identifies the physical I/O module and then whenever the strategy initiates a communication to the module.

<table><tr><td>Not present</td><td>I/O module has never been there since power up.</td></tr><tr><td>Self testing</td><td>I/O module is performing a self text - exists briefly after I/O module power up.</td></tr><tr><td>Standby</td><td>Module is in standby - exists briefly after I/O module self test.</td></tr><tr><td>Operational</td><td>I/O module is working correctly.</td></tr><tr><td>Major Recoverable fault</td><td>I/O module has a identified a fault from which it can recover. It is still communicating with the IQ4E.</td></tr><tr><td>Major Unrecoverable fault</td><td>I/O module was communicating and has now failed to communicate.</td></tr><tr><td>Unaddressed</td><td>I/O module does not have an address on the I/O bus - only when connected to IQ4/IO modules.</td></tr><tr><td>Duplicate Address</td><td>I/O module’s address is a duplicate on the I/O network - only when connected to IQ4/IO modules.</td></tr><tr><td>Upgrade</td><td>The I/O module’s firmware is being upgraded.</td></tr></table></td><td>0, 1, 2, 3, 4, 5, 6, 7, 8
0 = Not present
1 = Self testing
2 = Standby
3 = Operational
4 = Major recoverable fault
5 = Major unrecoverable fault
6 = Unaddressed
7 = Duplicate Address
8 = Upgrade</td><td>0</td><td>N</td><td>T</td></tr>
<tr><td>Wink</td><td>v3.30 firmware or greater. When set to ‘1’ the I/O module indicates its address by flashing its I/O Bus indicator. A long flash indicates ‘10’ and a short flash indicates ‘1’ e.g. 21 would be two long flashes and 1 short flash.

When set to ‘1’ by pressing the I/O module’s Service button the I/O module will flash it’s address once. The value remains at ‘1’ and subsequent presses of the Service button will cause it to flash the address again. The value can only be set back to ‘0’ using IQSET, text comms or the IQ4’s web pages.

When set to ‘1’ using IQSET, text comms or the IQ4’s web pages the I/O module will repeatedly flash it’s address until the I/O module’s Service button is pressed or it is set to ‘0’.

Note: Set to ‘1’ when the module is unable to address itself because all free or matching slots are unavailable.</td><td>0 or 1
0 = No flash
1 = Flash</td><td>0</td><td>N</td><td>W</td></tr>
</table>

17.2 I/O Module Connectives

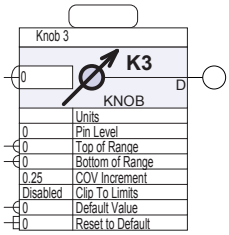
The following parameter is available for connection to other modules in the strategy:

Failed

18 KNOB MODULES

Module Identifier	Kn (where n is module number)
Default overview parameters	$, V, %
Size in brIQs	13

18.1 Functionality



Knob modules enable analogue values such as setpoints to be changed by system users within a specified range, they can be viewed and adjusted on all Trend user interfaces. The function performed by any Knob module is determined by the strategy.

Knob modules appear on the strategy page and do not need to be sequenced unless the input is connected. If a Knob module is event driven, its output should not be connected back to its input.

18.1.1 Reset Operation

v3.40 or greater. The Knob module's normal mode of operation is to set the 'Destination' parameter to the value of the 'Value' parameter. From version 3.30 the Knob module has a second mode of operation: reset to default.

The reset to default mode allows the Knob module to be set to a default value when required by the strategy. It sets the 'Destination' parameter to the default value ('Default Value' parameter) when the 'Reset to Default' parameter is set to '1'.

18.1.2 COV Operation

COV (Change Of Value) operation enables Knob modules to send a value to a list of subscribers using BACnet protocol when the value is changed by more than the value specified by the 'COV Increment' parameter.

18.1.3 Knob Adjustment Mode

Knob adjustment mode available in IQ2 controllers is not supported in IQ4. Most of the parameters, that one would want to adjust using knob adjustment mode, are available as module inputs in IQ4; a module input can be adjusted by connecting a normal knob module to it.

Parameters that are not available as inputs are normally items that one would not want to adjust from the strategy (e.g. module type).

18.1.4 Security

Each Knob modules has a configurable user level to protect its value from being changed by an unauthorised person.

If the level is left set to '0' (default), any user except the 'Guest' user can adjust the value. If the level is non-zero, there must be a Type 0 User Module set up in the IQ4 with a user level of equal to or greater than the user level set in the knob module; the user can then log on and make the change.

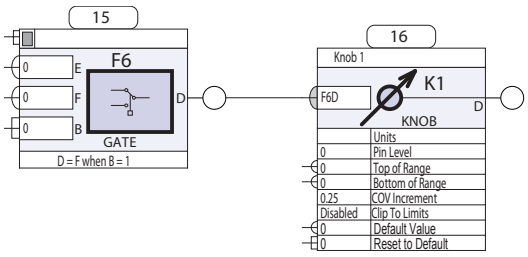
This user level is specific to a particular Knob module. Different Knob modules can have different levels. Changes to the other parameters are protected at a fixed level as defined in the Type 0 User Module see - **"Fixed User Levels" on page 511**.

18.1.5 Range

Knob modules can only be adjusted over the range defined by the 'Top Range' and 'Bottom Range' parameters.

18.1.6 Knob Sequencing

If a Knob module is connected to a source, or if its 'Top Range' and 'Bottom Range' parameters are connected, it must be placed in the sequence table. This ensures that it will be serviced in the correct sequence. The output of the module will remain for one cycle of the sequence table, so if there were to be a write to the Knob module that value would appear on the output the next time the module is serviced, remain for one sequence table cycle, and then the output would revert to its previous value.



In the above example the output of F6 is '0', so if the IC communications message were to write a value of 8 to the Knob module the value of the Knob module's output will change to 8 the next time it is serviced and remain for one sequence table cycle. The subsequent service will set the Knob module's output back to '0' (or whatever is the current value of F6's output).

If the Knob module's input is set to a constant value (i.e. unconnected), it can be left un-sequenced. It will then be serviced at intervals in the sequence table cycle.

18.2 Knob Module Parameters

Knob modules also include the following parameters that are common to most modules: 'Label', 'Page', 'Sequence Step', 'Sequence Table', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Bottom range	The minimum value to which the Knob module's 'Value' parameter can be set.	Full analogue	0	I	B
Clip to Limits	When enabled, the Knob module's 'Value' parameter is set to the value of the 'Bottom Range' parameter or the value of the 'Top Range' parameter if the Knob module is adjusted outside these limits by text comms. When this parameter is disabled text comms writes to the value outside the limits will be ignored so that the Knob module will remain at its previous value.	0 or 1 1 = enabled 0 = disabled	0	N	L
COV Increment	The amount by which the Knob module's value has to change before a BACnet COV notification is sent. The 'COV Increment' parameter must not be set to '0'.	+ Full analogue	0.25	N	C
Destination	The value of the module's output.	Full analogue	0	O	D
Default Value	v3.40 or greater. The value to which the 'Destination' parameter is set when the 'Reset to Default' parameter is set to '1'.	Full analogue	0	I	F
Reset to Default	v3.40 or greater. When set to '1' the 'Destination' parameter is set to the default value.	0 or 1 1 = reset	0	I	R
Top Range	The maximum value to which the Knob module's 'Value' parameter can be set.	Full analogue	0	I	T

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Units	The units for the Knob module's value. For compatibility with BACnet they should be chosen from the list of units in **"BACnet Object/Property Codes and Units" on page 519**. If units other than these are used, they will be transmitted over BACnet as 'no units'.	0 to 4 characters	Blank	N	%
User Level	The PIN level of the user with which the user must be logged in in order to change the Knob module's value.	0 to 99	0	N	P
Value	The Knob module's value.	Full analogue	0	O	V

18.3 Knob Module Connectives

The following parameters are available as connectives:

Destination

19 LOGIC MODULES

Module Identifier	Gn (where n is module number)
Default overview parameters	($, D)
Size in brIQs	18

19.1 Functionality

Logic modules allow outputs to be generated as a function of a number of digital values within a strategy. The resultant outputs are available for use by other modules in the strategy.

Logic modules appear on the strategy page and can be either sequence table or event driven.

Note: Type 6 (Hours Run) Logic modules should not be event driven.

19.2 Module Disable

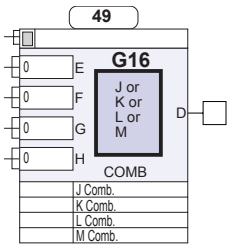
If required Logic modules can be disabled - see **"Disabling Modules" on page 29**.

19.3 Logic Module Types

There are several Logic module types specified using the 'Type' parameter:

Type	*Description*
Type 1 (Combination)	Logically combines digital inputs.
Type 2 (Timer)	Applies ON and OFF delays and a minimum on time to a single digital input.
Type 3 (Readback)	Checks if two digital inputs follow each other within a grace time of one input changing.
Type 4 (Counter)	Counts a pulse type digital input and also calculates its pulse rate.
Type 5 (Delay)	Generates an ON state if two inputs differ outside a grace time.
Type 6 (Hours Run)	Counts hours for which a digital input is on, and checks if an interval is exceeded.
Type 7 (Fan Out)	Connects a digital input to up to 6 digital outputs.
Type 8 (D to A)	Converts up to 8 digital inputs into an analogue value.
Type 9 (Change of State)	Generates a pulse on its output when the value of the 'Source' parameter changes.
Type 10 (Flip Flop)	Sets its output to '1' and retains that status until reset regardless of the input status.
Type 11 (Latch)	Sets its output to the value of the 'Source' parameter and retains that status until reset regardless of the 'Source' parameter's value.
Type 12 (Digital Input Selector)	Selects of one of up to eight digital input parameters depending on the value of the 'Selector' parameter.

19.3.1 Type 1 (Combination)



Type 1 (Combination) Logic modules calculate the 'Destination' parameter (D) as the logical combination (OR) of up to 4 logical (AND) functions (J), (K), (L), and (M).

Each logical function is itself a function of 4 digital values 'Source E', 'Source F', 'Source G', and 'Source H' parameters which are AND'ed together:

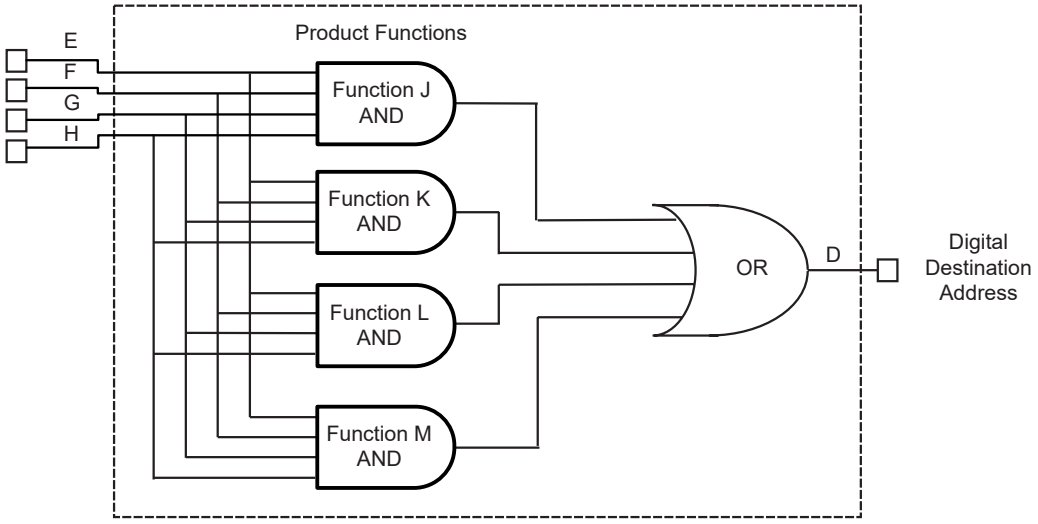
(E) AND (F) AND (G) AND (H)

An AND NOT function can also be achieved by specifying a lower case letter instead of upper case. This allows combinations of AND, and AND NOT functions within each of the logical functions (J), (K), (L), and (M).

The 'Destination' parameter is the result of all logical functions OR'ed together:

(J) OR (K) OR (L) OR (M)

This is illustrated in the diagram below:



The action of the Type 1 (Combination) Logic module is described by a boolean expression, which is a short way of showing the module's function. A Trend boolean expression uses the + symbol to represent the OR function, upper case letters to represent true state ('1') and lower case letters to represent the inverse state ('0'). E.g. If the function is described as 'E', then if the bit data at the input is '1', the output will be '1'; if, however, e is used, the output will be '0' (i.e. the inverse) when the input is '1'. Letters are written side by side (as in multiplication) to represent the AND function.

19.3.1.1 Designing a Function

A simple way to derive the boolean expression for the required logical function is to write down a conditional statement. Say, for example, we require to switch a fan ON when there is a manual override or when there is a demand greater than 20% and the building is occupied, or when there is a demand greater than 10% and building is non-occupied. The conditions we will have to monitor are:

Manual Override ON
Demand > 20%
Building Occupied
Demand > 10%
Building Non-Occupied.

We can now write a conditional statement i.e:

Manual override ON
OR Demand > 20% AND Building Occupied
OR Demand > 10% AND Building Non-occupied.

If we use the source letters for a Logic module:

Manual Override ON = E
Demand > 20% = F
Building Occupied = G
Demand > 10% = H

Now Building Non-occupied is the same as NOT Building Occupied. So the above conditional statement can now be written as:

E OR F AND G OR H AND NOT G

Using the notation described above:

E+FG+Hg

This can be configured into a Type 1 (Combination) Logic module by setting:

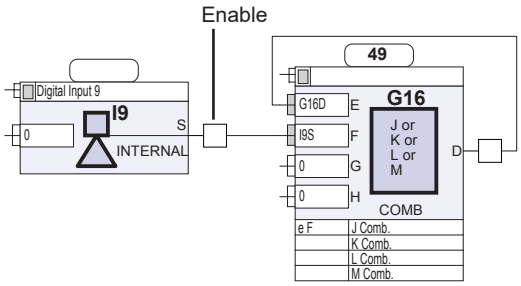
Combination J = E
Combination K = FG
Combination L = Hg

To check that the function works correctly, the rigorous approach is to create a truth table.

The above function was simple to design as the output was a function only of input, but supposing the new output state depends on the previous output state. The output must be connected back as an input i.e. feedback.

For example, we wish to make a pulse generator, i.e. we want the output to change from '0' to '1' and back again every time we service the module. We can have an enable input which will enable the pulse output if ON and disable it if OFF.

The module can be drawn:



We can write down the truth table for the function:

	E	F	D
1	0	0	0
2	0	1	1
3	1	0	0
4	1	1	0

The two columns 'E' and 'F' show all possible combinations of inputs. This is best done by counting up in binary to the number of binary digits available as inputs. The right hand column 'D', shows the output for each combination of inputs.

Thus, in row one, the output is at '0' and the module is disabled so the output stays at '0'. In row two the module is now enabled so we want the count to change i.e. D is '1'. In row three the output is '1', but the module is disabled so we want the output to be '0'. In row four the output is at '1', the module is enabled so the output must become '0'.

The next step is to write down the combinations needed for the output to be '1' and OR them together. There is only one such output in this case for row 2.

D = F AND NOT E = Fe

This is the required function.

19.3.1.2 Examples

OR Function
The logic OR function is being performed on the 'Source E' and 'Source F' parameters. Where, if the state of these parameters is '1', then the module's 'Destination' parameter will be '1'.

Combination J = E
Combination K = F
Output D = J+K = E+F

AND Function
The logic AND function is being performed on the 'Source E' and 'Source F' parameters. Where if the states of these parameters is '1', then the module output state will be '1'. If either or both input states are '0' the 'Destination' parameter will be '0'.

Combination J = EF
Output D = J = EF

NOR Function
The logic NOR function is being performed on 'Source E' and 'Source F'. In this case the output is '1' if both 'Source E' and 'Source F' are '0' or is '0' if either 'Source E' or 'Source F' is '1'; this function is E NOR F.

Combination J = ef
Output D = J = ef (NOT E AND NOT F)

NAND Function
The logic NAND function is being performed on the 'Source E' and 'Source F' parameters. In this case the 'Destination' parameter is '1' if either of these parameters are '0' or conversely it is '0' if both they are both are '1'; this function is E NAND F.

Combination J = e
Combination K = f
Output D = J+K = e+f (NOT E OR NOT F)

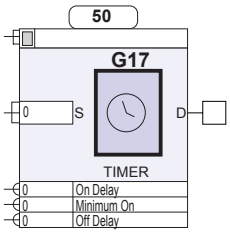
19.3.1.3 Type 1 (Combination) Logic Module Parameters

Type 1 (Combination) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Combination J	Logically combines up to 4 digital values (E), (F), (G), or (H) using AND, and AND NOT functions.	Trend boolean of EeFfGgHh,	Blank	N	J
Combination K					K
Combination L					L
Combination M					M
Source E	State of input E, one input to the logical combination.	0 or 1	0	I	E
Source F	State of input F, one input to the logical combination.				F
Source G	State of input G, one input to the logical combination.				G
Source H	State of input H, one input to the logical combination.				H

Note: The 'Source E', 'Source F', 'Source G', and 'Source H' parameters are '0' by default, but are not used in the logical combination unless they are entered into one of the combinations J, K, L, M.

19.3.2 Type 2 (Timer)



Type 2 (Timer) Logic modules can be used to provide the following timer functions:

ON delay
OFF delay
Minimum ON

ON Delay: When the 'Source' parameter is set to '1' the 'Destination' parameter is not set to '1' until the ON delay specified by the 'On Delay' parameter has elapsed. This only applies if the module's input is '1' throughout the ON delay period.

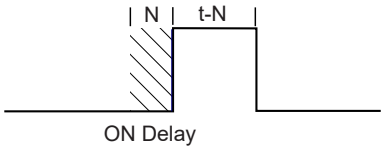
OFF Delay: When the 'Source' parameter is set to '0' the 'Destination' parameter is not set to '0' until the OFF delay specified by the 'Off Delay' parameter has elapsed. This only applies if the 'Source' parameter is '0' throughout the OFF delay period. If the minimum on time is set, the OFF delay is applied after the minimum on time has expired.

Minimum ON: When the 'Source' parameter is set to '1' the 'Destination' parameter is set to '1' and will not be set to '0' until the minimum on time, specified by the 'Minimum On' parameter has elapsed. The 'Destination' parameter will only be set to '0' after the time has elapsed. If the ON delay is also set up, the minimum on time is applied after the ON delay has expired.

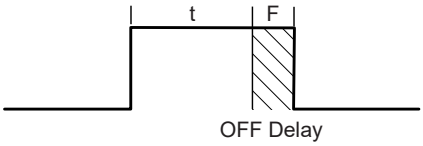
If the 'Source' parameter is set to '1' the 'Destination' parameter will not change to '1' until the ON delay time has elapsed. It will then remain at '1' for at least the length of time specified by the 'Minimum On' parameter. When the 'Source' parameter changes to '0' the 'Destination' parameter will only be set to '0' providing the minimum on time has elapsed. If an OFF delay is set, the output will stay ON until the OFF delay has elapsed.

The following diagrams shows the timing for different timer functions where 't' represents the time duration of the input pulse.

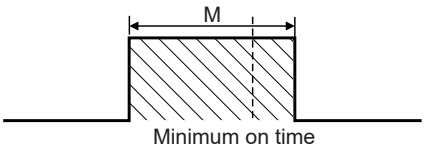
Output D with on delay (N): Duration of 'Destination' parameter is t-N



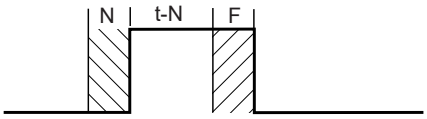
Output D with off delay (F): Duration of 'Destination' parameter is t+F



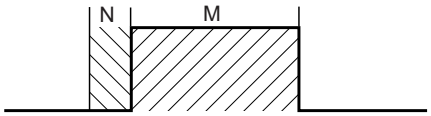
Output D with minimum on time (M) where M>t: Duration of 'Destination' parameter is max(M,t)



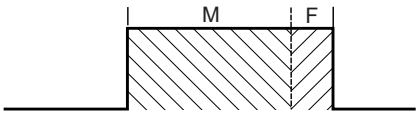
Output D with on delay (N), and off delay (F): Duration of 'Destination' parameter is t-N+F



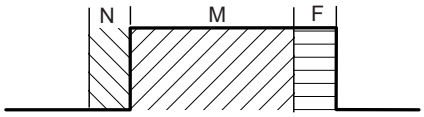
Output D with on delay (N) and minimum on time (M): Duration of 'Destination' parameter is max(M, t-N)



Output D with minimum on time (M), and off delay (F): Duration of 'Destination' parameter is max(M, t)+F



Output D with on delay (N), minimum on time (M), and off delay (F): Duration of 'Destination' parameter is max(M, t-N)+F



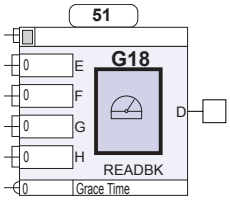
The time range for each of these timer functions (ON delay, minimum on time, OFF delay) is 0 to 32767 seconds (about 9 hours). The maximum output time ON is thus 65534 seconds (assuming t<M, or t-N<M) and is achieved by the M+F or N+M+F combinations shown above.

19.3.2.1 Type 2 (Timer) Logic Modules Parameters

Type 2 (Timer) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Minimum On	The minimum time (seconds) for which the output will be held ON.	0 to 32767	0	I	M
Off Delay	The delay (seconds) between input switching OFF and output switching OFF.	0 to 32767	0	I	F
On Delay	The delay (seconds) between input switching ON and output switching ON.	0 to 32767	0	I	N
Source	State of the input being subjected to timer functions.	0 or 1	0	I	S

19.3.3 Type 3 (Readback)



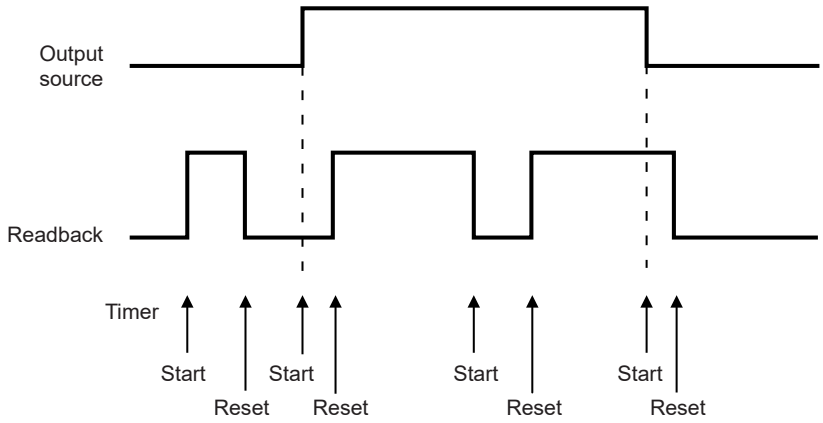
Type 3 (Readback) Logic modules provide the facility of comparing an output of a Driver module to the HVAC equipment with the a feedback input from that equipment (e.g. pump drive output with flow switch input). The 'Destination' parameter is set to '1' if the two signals are different for more than the grace time.

They have two sets of inputs, i.e. two equipment drives and two readback inputs. Normally, only one equipment drive and one readback input are used. If the second drive and readback are not used they should not be connected in IQSET.

Considering one pair of inputs, they are compared when the module is serviced and if either of the two inputs differ then the grace timer is started and will be reset back to '0' if the two inputs become the same. If the two inputs remain different for the whole of the grace time, then the module output will be set.

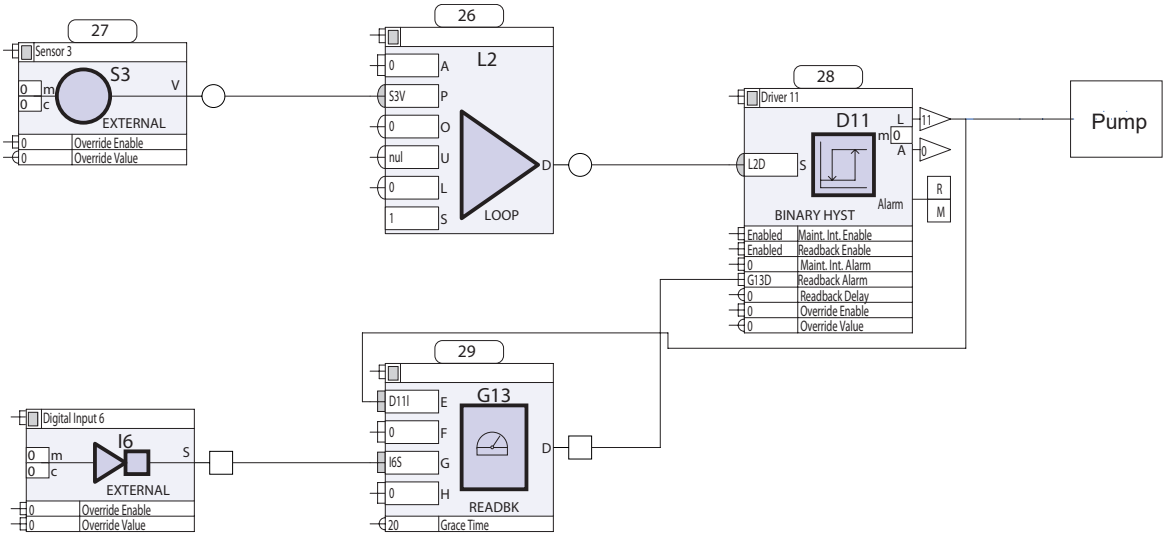
The output is normally connected to the 'Readback Alarm' parameter of the Driver module being monitored.

The effect of this is that the HVAC equipment being controlled and monitored has the grace time in which to switch ON or OFF, without producing an alarm, and also if the equipment switches into the wrong state at any time, it must be in that faulty condition for the whole grace time before an alarm is generated. This will filter out short term differences.

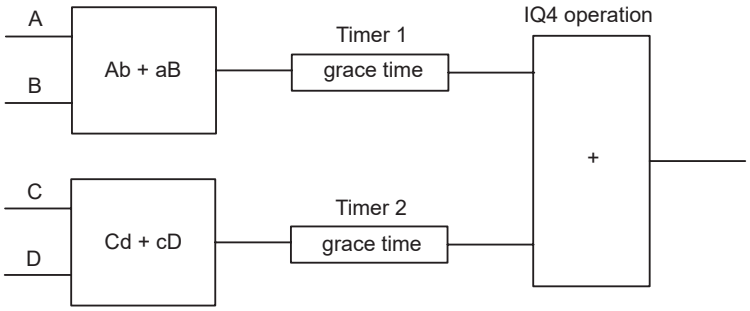


In the above diagram the grace timer is started when the inputs differ and reset when the inputs are the same. If the period from start to reset ever extends the grace time then the module output will be set, which will cause a driver readback alarm to be generated if the output is connected to a driver readback input.

The strategy diagram below shows the Type 3 (Readback) Logic module being used to generate the readback alarm for a Driver module.



If both pairs of inputs are used the two pairs work as described above independently of each other, each pair having a separate timer as shown below.



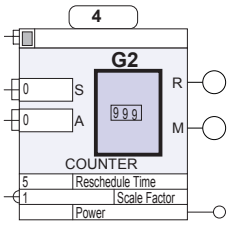
The outputs from the two timers are OR'd to produce the module's output (Destination).

19.3.3.1 Type 3 (Readback) Logic Module Parameters

Type 3 (Readback) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Source E	State of 'Source E', the 1st drive input to be compared to 'Source G', 1st readback input.	0 or 1	0	I	E
Source F	State of 'Source F', the 2nd drive input to be compared to 'Source H', 2nd readback input.	0 or 1	0	I	F
Source G	State of 'Source G', the 1st readback input to be compared to 'Source E', 1st drive input.	0 or 1	0	I	G
Source H	State of 'Source H', the 2nd readback input to be compared to 'Source F', 2nd drive input. Can be connected to a digital output.	0 or 1	0	I	H
Grace Time	Value of grace time, (seconds) time for which inputs can be different before output switches ON.	0 to 32767	0	I	T

19.3.4 Type 4 (Counter)



Type 4 (Counter) Logic modules provide the facility for counting pulse type signals such as those provided by gas and electricity meters.

They count the pulses on the input and multiply this count by the scaling factor to produce the cumulative output.

They produce a rate output which is calculated by the change in count over the reschedule time.

They produce a power output in either kWh or Btu/h depending on the units of the 'Cumulative Count' parameter.

19.3.4.1 Input Monitoring

If the module is sequence table driven, and the input is from an internal digital value (rather than from a Type 2 (External) digital input module), then it will only be monitored when the module is sequenced. In order to be monitored it must be ON at least for the period between sequencing thus the maximum frequency is:

$$\frac{\text{number of times module is entered in the sequence table}}{\text{2 x cycle time (1s)}}$$

E.g. If the module is entered twice in the sequence table, maximum frequency is 2/2 , i.e. 1 pulse per second.

19.3.4.2 Cumulative Output

The value of the 'Cumulative Output' parameter is calculated by multiplying the number of pulses received by the scaling factor. The scaling factor specified by the 'Scale Factor' parameter, it defines the value of each pulse i.e. it scales the counter.

From v3.30 firmware the 'Cumulative Output' parameter is a double precision value.

19.3.4.3 Rate

The value of the 'Rate' parameter calculated as the change in cumulative count over the reschedule time (i.e. it takes into account the scaling factor). The reschedule time specified by the 'Reschedule Time' parameter which defines the period over which the rate is measured and the count is increased. At the end of the reschedule period, the rate output is changed and the rate value (i.e. change in cumulative count) is added to the cumulative output.

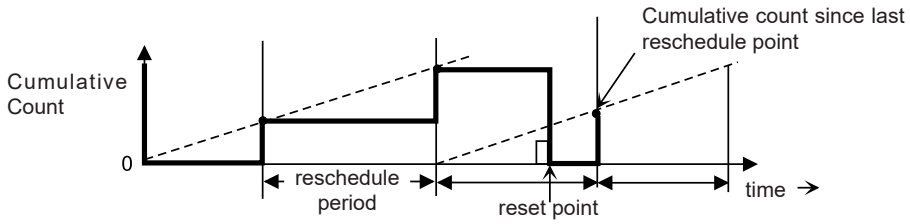
19.3.4.4 Power (v3.30 or greater)

The value of the 'Power' parameter calculated in either kWh or Btu/h depending on the units of the 'Cumulative Count' parameter as below:

Power = (Rate*3600)/ Reschedule Time

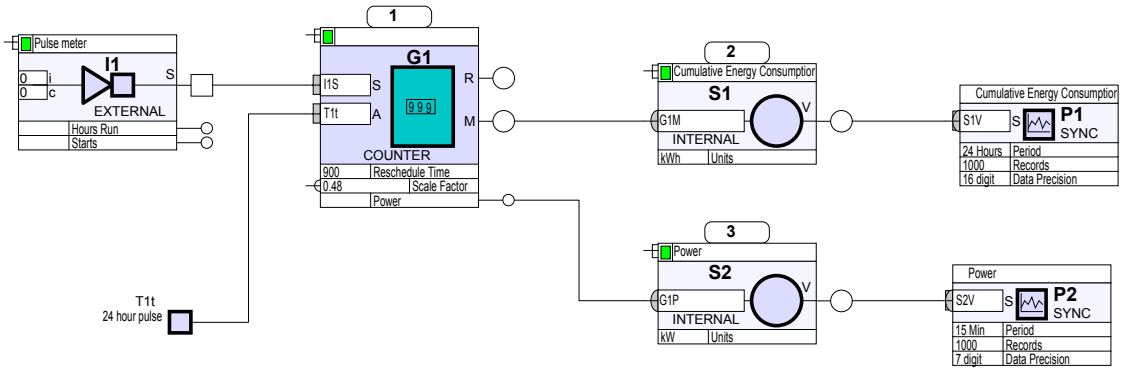
19.3.4.5 Resetting

The 'Reset' parameter resets the count to '0'. If the 'Reset' parameter is '1' when the module is serviced, whether or not the reschedule period has just ended, the 'Cumulative Output' parameter is set to '0'. If this occurs in the middle of a reschedule period, the change in count will be added to the 'Cumulative Output' parameter at the end of the reschedule period. The effect will be that the cumulative count goes to '0' when the module is serviced, and the 'Cumulative Output' parameter shows the change in count since the last reschedule point at the end of the reschedule period. The rate thus remains unaffected.



Thus in general at the end of a reschedule period the 'Cumulative Output' parameter will not be '0' even if the 'Reset' parameter is '1' at that time, because the value of the change in count since the last reschedule time is not '0'. This ensures that pulses counted during a reset period are not lost, so that the total count between resets would agree with the count of another device (e.g. an electricity meter).

19.3.4.6 Example



In the above example, the pulses from an electricity meter are being used to calculate the cumulative energy used each day, and the average power over every 15 minutes.

The Type 4 (Counter) Logic module's 'Scale Factor' parameter is 0.48 as each pulse is worth 0.48 kWh. The cumulative count will then be calculated in kWh and is reset at midnight each day. S1 is logged at midnight each day to give a 1000 day log.

The counter's reschedule time is set to 900 s (15 min) so that the rate would be the kWh consumed in 15 min.

The power output is given by:

Power = (Rate*3600)/ Reschedule Time = (Rate*3600)/900 = Rate*4
So (kWh/(15 min))*4 = kWh/h = kW i.e the correct units for power

This is then monitored by S2 which is logged at 15 minute intervals, so that the power graph will show fluctuations every 15 minutes for a 250 day log.

Note: Although the counter module reschedule time and the plot module period are not synchronised the strategy will still work.

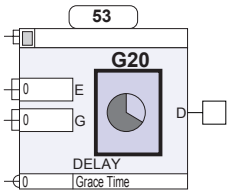
19.3.4.7 Type 4 (Counter) Logic Module Parameters

Type 4 (Counter) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Cumulative Count	v3.30 or greater. The number of pulses the module has counted since it was last reset.	Full analogue	0	N	c
Cumulative Output	The cumulative output calculated by the module.	Double Precision	0	O	M
Last Cumulative Count	v3.30 or greater. The value of the Cumulative Count when the module was last serviced.	Full analogue	0	N	l
Power	v3.30 or greater. The power in either kW or Btu/hour.	Full analogue	0	O	P
Rate	The rate calculated by the module.	Full analogue	0	O	R

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Reschedule Time	Value of time (seconds) between calculating rate and updating cumulative output. The selection of the reschedule time is a compromise between: The longer the time period, the more pulses are counted and the less the effect of, failure to count one pulse in a period but counting it in the next instead, has on the value. The shorter the time period, the more the rate can follow the actual fluctuations of the pulses. With a long period of 1 hour, the rate is calculated over that period and will only change once per hour.	0 to 1275	5	N	T
Reset	Resets the count to '0'.	0 or 1 1 = Reset	0	I	A
Scale Factor	The input used to define the value of each pulse.	Double Precision	1	I	F
Source	The digital input being counted.	0 or 1	0	I	S

19.3.5 Type 5 (Delay)



Type 5 (Delay) Logic modules are designed specifically for IQ1 and early IQ2 controllers to enable a delay to be used in the item alarm reporting mechanism (i.e. for strategy modules - digital inputs, loops, sensors, drivers). IQ4s have 'Alarm Delay' parameters integrated into these modules. In IQ4 it can be used in a similar way to the Type 3 (Readback) Logic module.

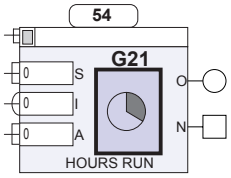
They calculate the 'Destination' parameter as '0' or '1' by comparing two inputs ('Source E' and 'Source G' parameters). If they are different for longer than the grace time, specified by the 'Grace Time' parameter the 'Destination' parameter will be set to '1'. If they are the same, 'Destination' parameter will be '0'.

19.3.5.1 Type 5 (Delay) Logic Module Parameters

Type 5 (Delay) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Source E	The state of the 1st input to be compared with 2nd input G.	0 or 1	0	I	E
Source G	The state of the 2nd input to be compared with 1st input E.	0 or 1	0	I	G
Grace Time	The time (seconds) for which the inputs can be different before the output switches ON.	0 to 32767	0	I	A

19.3.6 Type 6 (Hours Run)



Type 6 (Hours Run) Logic modules accumulate the time for which the module's 'Source' parameter is set to '1' and make it available in the 'Hours Run Output' parameter. They can be used to:

- Facilitate strategies which require the usage of HVAC equipment (e.g. pumps, boilers) to be cycled upon the basis of hours run.
- Provide maintenance interval checking.

The 'Hours Run Output' parameter is calculated as the length of time in hours for which the 'Source' parameter has been set to '1'.

The value of the 'Hours Run Output' parameter is compared with the value of the 'Interval' parameter and if the value of the 'Hours Run Output' parameter is greater than the value of the 'Interval parameter the 'Interval Limit Alarm' parameter is set to '1' (interval exceeded).

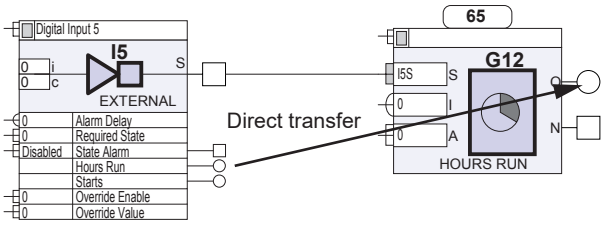
When the 'Reset' parameter is set to '1', the 'Hours Run Output' parameter is set to '0' and since the hours run of 0 cannot now exceed the interval, the 'Interval Limit Alarm' parameter is also set to '0'.

Type 6 (Hours Run) Logic modules should always use the sequence table, and must not be event driven.

19.3.6.1 External Digital Inputs and Drivers

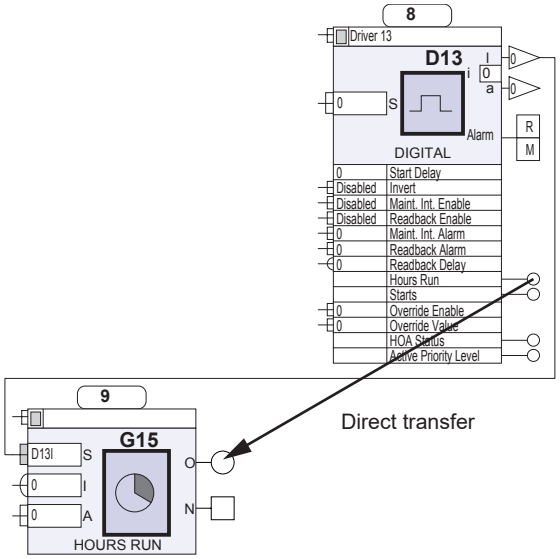
The Type 6 (Hours Run) Logic module has a special hours run function which operates when it is connected to a Type 2 (External) Digital Input module or to a Driver module's output (not analogue or raise/lower).

If a Type 6 (Hours Run) Logic module is connected directly to a Type 2 (External) Digital Input as shown the value of the Digital Input module's 'Hours Run' parameter is transferred directly to the Type 6 (Hours Run) Logic module's 'Hours Run' parameter without scaling when the module is serviced.



If the Type 6 (Hours Run) Logic module is reset this will reset both Type 6 (Hours Run) Logic module's 'Hours Run' parameter and the Digital Input module's 'Hours Run' parameter. Similarly if the Digital Input module's 'Hours Run' parameter is changed, this will be copied to the Type 6 (Hours Run) Logic module's 'Hours Run' parameter.

If a Type 6 (Hours Run) Logic module is connected directly to a Driver module's output (not analogue and raise/lower) as shown, the Driver module's 'Hours Run' parameter is transferred directly to the Type 6 (Hours Run) Logic module's 'Hours Run' parameter without scaling when the module is serviced.



If the Type 6 (Hours Run) Logic module is reset this will reset both the Type 6 (Hours Run) Logic module's 'Hours Run' parameter and the Driver module's 'Hours Run' parameter. Similarly if the Driver module's 'Hours Run' parameter is changed, this will be copied to the Type 6 (Hours Run) Logic module's 'Hours Run' parameter.

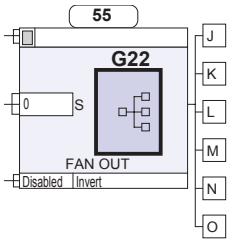
These special relationships exist mainly for compatibility with IQ2 series controllers although they also enable the hours run to be reset, and the interval exceeded to be generated.

19.3.6.2 Type 6 (Hours Run) Logic Module Parameters

Type 6 (Hours Run) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Hours Run Output	The value of accumulated hours run.	Full analogue	0	O	O
Hours Run	Same value as Hours Run output.	Full analogue	0	-	-
Interval Limit Alarm	The state of the interval exceeded output.	0 or 1 1 = interval exceeded	0	O	N
Interval	The maximum number of hours run before an alarm is generated.	Full analogue	0	I	I
Reset	Resets the count of hours and alarm condition	0 or 1 1 = Reset	0	I	A
Source	The state of the module input whose hours run is being accumulated.	0 or 1	0	I	S

19.3.7 Type 7 (Fan Out)



Type 7 (Fan Out) Logic modules enable a digital value to be connected to up to 6 other digital values. The state of the input is passed to the output bits when the module is serviced. There is an invert parameter which allows the inverse state of the input to be passed to all the outputs.

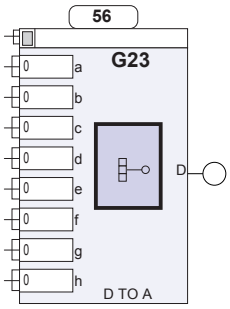
They are intended to provide compatibility with IQ2 strategies and have little application in IQ4s.

19.3.7.1 Type 7 (Fan Out) Logic Modules Parameters

Type 7 (Fan Out) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Output J	The state of output J dictated by the state of the source and invert inputs.	0 or 1	0	O	J
Output K	The state of output K dictated by the state of the source and invert inputs.	0 or 1	0	O	K
Outputs L	The state of output L dictated by the state of the source and invert inputs.	0 or 1	0	O	L
Outputs M	The state of output M dictated by the state of the source and invert inputs.	0 or 1	0	O	M
Outputs N	The state of output N dictated by the state of the source and invert inputs.	0 or 1	0	O	N
Outputs O	The state of output O dictated by the state of the source and invert inputs..	0 or 1	0	O	O
Source	The state of source input being fanned out.	0 or 1	0	I	S
Invert	Specifies whether the state of the outputs are the same, or opposite to the state of the input.	0 or 1 1 = Inverted	0	I	I

19.3.8 Type 8 (D to A)



Type 8 (D to A) Logic modules convert up to eight digital bits into one analogue value which represents the binary value of the input bits.

Each input bit has a binary weighting so that the complete set of eight digital inputs is treated as a binary number and the output is the analogue value of this binary number. This gives a unique analogue value for every input combination. The inputs are defined as below:

a Bit 0 = 1
b Bit 1 = 2
c Bit 2 = 4
d Bit 3 = 8
e Bit 4 = 16
f Bit 5 = 32
g Bit 6 = 64
h Bit 7 = 128

Where:

output = 1x(a) + 2x(b) + 4x(c) + 8x(d) + 16x(e) + 32x(f) + 64x(g) + 128x(h)

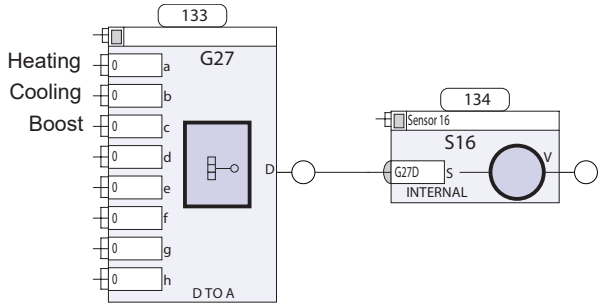
Thus if inputs a, d, and f are set to '1' and the others set to '0',

a + d+ f = Bit 0 + Bit 3 + Bit 5 = 1 + 8 + 32 = 41

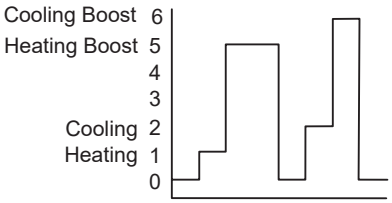
i.e binary 00101001 = decimal 41

This module could be used to generate a single analogue value for display on a supervisor page which will give an encoded value to report the HVAC equipment status.

This example shows the three states, heating, cooling and boost which have the values 1, 2, 4 respectively. The output is monitored by internal sensor S16 which can be logged.

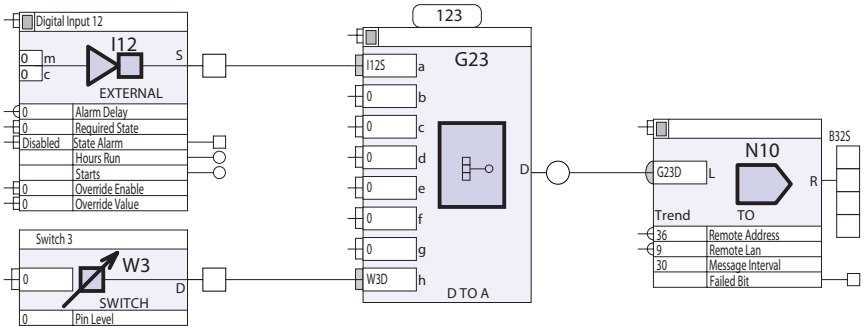


An example graph is shown below.



It could also be used to encode a number of digital values into a single analogue value to be sent to another controller by an IC communications message. The digital bits can then be decoded in the other controller using a Type 18 (Analogue to Digital Converter) Function module.

This type of application is used to encode digital bits for a digital byte Data To type IC communications message as shown:

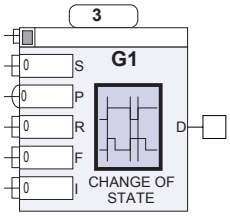


19.3.8.1 Type 8 (D to A) Logic Module Parameters

Type 8 (D to A) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

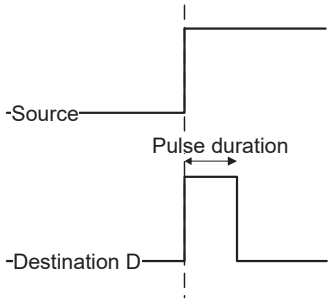
Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Bit 0	The state of 1st input which adds 1 to the output.	0 or 1	0	I	a
Bit 1	The state of 2nd input which adds 2 to the output.	0 or 1	0	I	b
Bit 2	The state of 3rd input which adds 4 to the output.	0 or 1	0	I	c
Bit 3	The state of 4th input which adds 8 to the output.	0 or 1	0	I	d
Bit 4	The state of 5th input which adds 16 to the output.	0 or 1	0	I	e
Bit 5	The state of 6th input which adds 32 to the output.	0 or 1	0	I	f
Bit 6	The state of 7th input which adds 64 to the output.	0 or 1	0	I	g
Bit 7	The state of 8th input which adds 128 to the output.	0 or 1	0	I	h

19.3.9 Type 9 (Change of State)

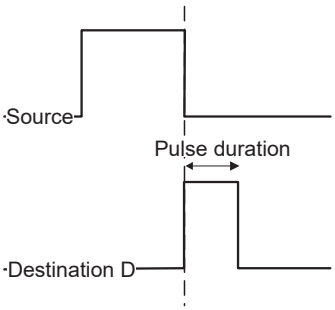


v3.30 or greater firmware only. Type 9 (Change of State) Logic modules pulse the 'Destination D' parameter when the value of the 'Source' parameter changes. The pulse can be initiated on either the rising edge of the change, falling edge of the change or both.

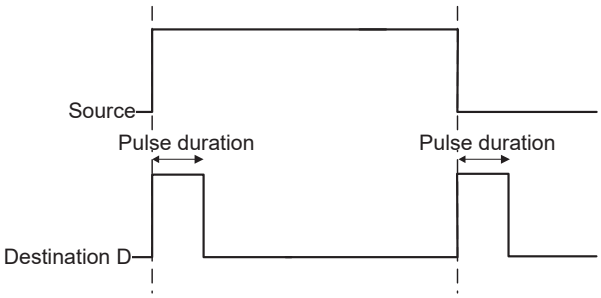
Rising Edge Change



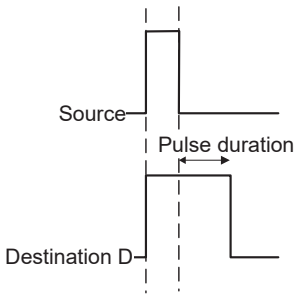
Falling Edge Change



Rising and Falling Edge Change



If a change of status occurs during a pulse the pulse is generated again in effect increasing the length of the pulse.

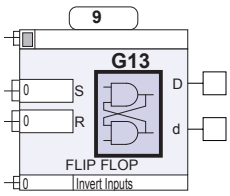


19.3.9.1 Type 9 (Change of State) Logic Module Parameters

Type 9 (Change of State) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

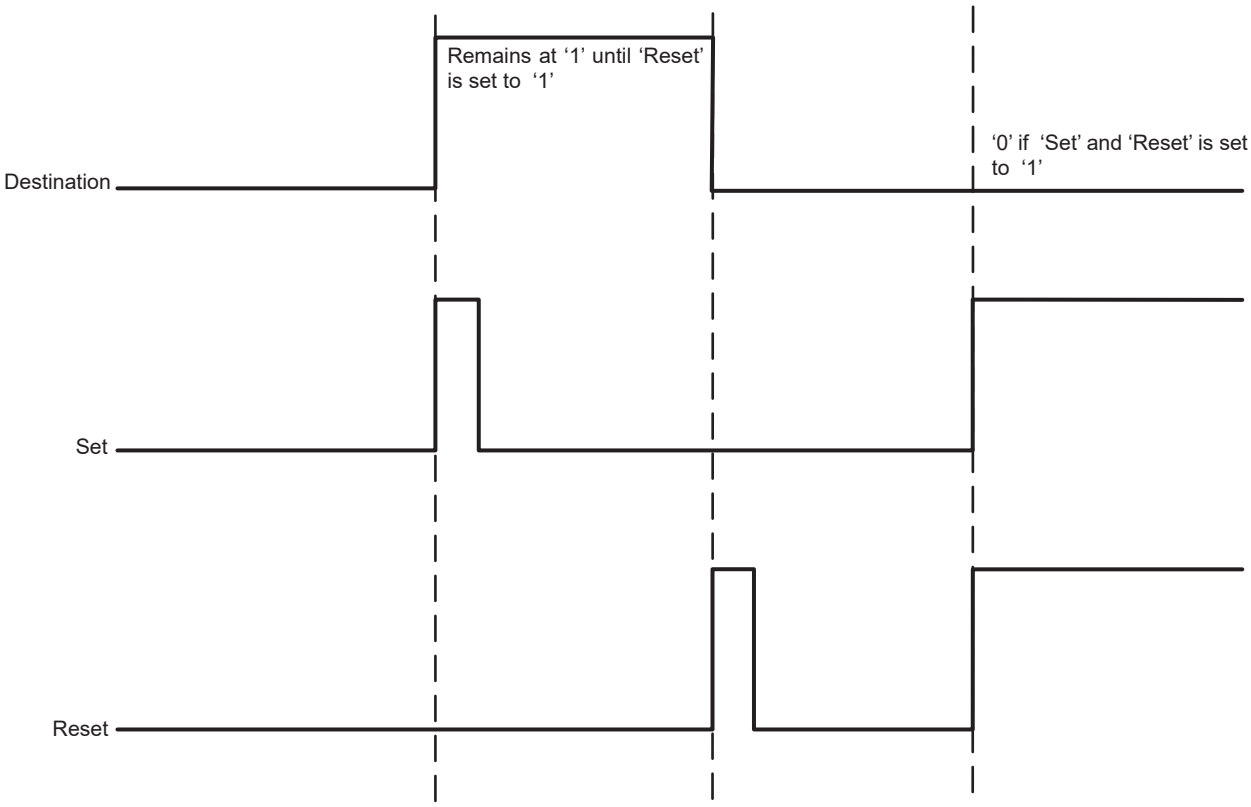
Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Invert Output	When set to '1' then the normal status of the 'Destination D' parameter is '1' and the output pulse is set to '0'.	0 or 1 1 = Inverted 0 = Normal	0	O	I
Pulse Duration	The duration of the output pulse, in seconds. If this is set to '0' then the pulse will last for one complete cycle of the sequence table and in the case of event based execution the OFF event follows the ON event directly.	+Analogue	0	I	P
Source	The value that is to be monitored for changes.	0 or 1	0	I	S
Rising Edge Enable	When set to '1' pulses the module's 'Destination D' parameter on the rising edge of a change of value.	0 or 1	1	I	R
Falling Edge Enable	When set to '1' pulses the module's 'Destination D' parameter on the falling edge of a change of value.	0 or 1	1	I	F

19.3.10 Type 10 (Flip Flop)



v3.30 or greater firmware only. Type 10 (Flip Flop) Logic modules set the 'Destination' parameter (D) to '1' when the 'Set' parameter is set to '1' and will keep it set to '1' until the 'Reset' parameter is set to '1'. If both the 'Set' and 'Reset' parameters are '1' the 'Destination' parameter (D) is set to '0'. The 'destination' parameter (d) is always set to the inverse of the 'Destination' parameter (D). The operation can be inverted by setting the 'Invert inputs' parameter to '1'.

The modules operation is shown below:



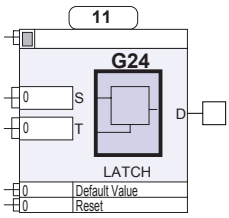
The module's status is retained when the IQ4 is powered down and when the IQ4 restarts it the module will adopt the status at power down.

19.3.10.1 Type 10 (Flip Flop) Logic Module Parameters

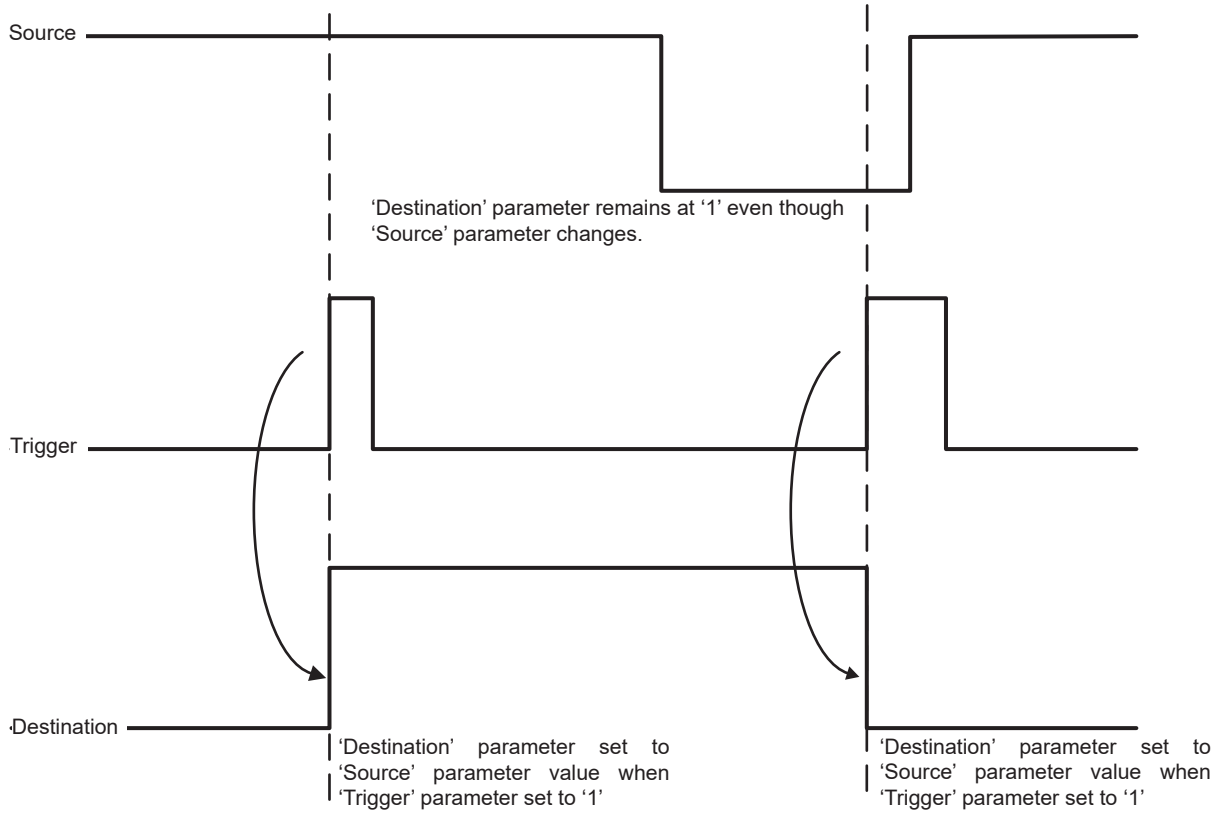
Type 10 (Flip Flop) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination d	The inverse of the Destination D' parameter.	0 or 1	1	O	d
Invert inputs	Specifies whether the state of the outputs of the module is inverted.	0 or 1 0 = Normal 1 = Inverted	0	N	P
Reset	Resets the module's output 'D' to '0'.	0 or 1 0= no change 1 = Reset	0	I	R
Set	When set to '1' sets the modules output 'D' to '1'.	0 or 1	0	I	S

19.3.11 Type 11 (Latch)



v3.30 or greater firmware only. Type 11 (Latch) Logic modules allow a digital state to be stored when a trigger input is set. The 'Destination' parameter (D) is set to the value of the 'Source' parameter when the 'Trigger' parameter is set to '1'. The Destination' parameter will remain at that value until the 'Trigger' parameter is set to '1' again - see diagram below:



The Destination' parameter (D) can be set to the value of the 'Default Value' parameter by setting the 'Reset' parameter to '1'.

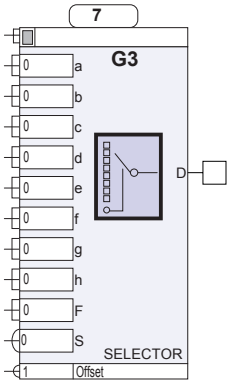
The module's status is retained when the IQ4 is powered down and when the IQ4 restarts it the module will adopt the status at power down.

19.3.11.1 Type 11 (Latch) Logic Module Parameters

Type 11 (Latch) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Default Value	The value to be used when the 'Reset' parameter is set to '1'.	0 or 1	0	N	F
Reset	When set to '1' sets the module's 'Destination D' parameter to the value specified by the 'Default Value' parameter.	0 or 1 0 = no change 1 = Reset	0	I	R
Source	The value that is to be processed.	0 or 1	0	I	S
Trigger	When set to '1' sets the module's output 'D' to the value of the 'Source' parameter.	0 or 1	1	I	T

19.3.12 Type 12 (Digital Input Selector)



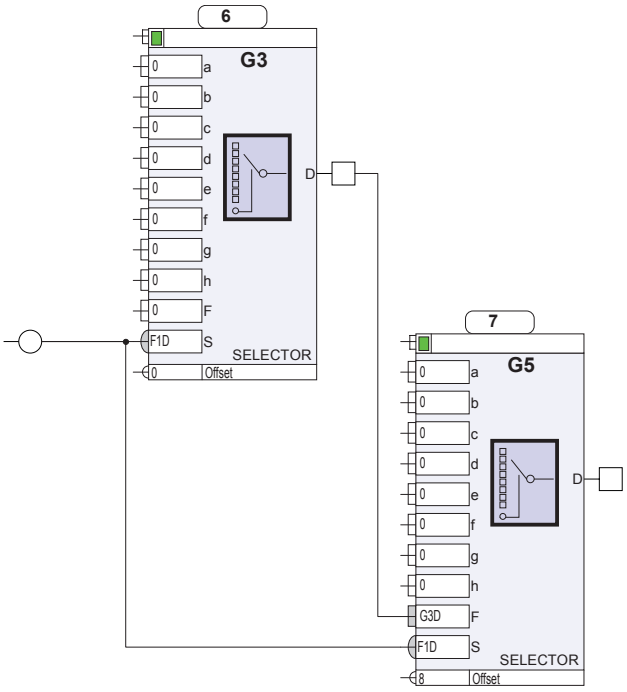
v3.40 or greater firmware only. Type 12 (Digital Input Selector) Logic modules set their 'Destination' parameter to the value of one of up to eight digital input parameters ('Input a', 'Input b', 'Input c', 'Input d', 'Input e', 'Input f', 'Input g', or 'Input h') depending on the ROUND of the value of the 'Selector' parameter minus the value of the 'Offset' parameter providing this value is 0, 1, 2, 3,4, 5, 6 or 7.

E.g. if 'Selector' parameter = '10.4' and 'Offset' parameter = 5. This give a calculation of 10.4 - 5 = 5.4 which is rounded to '5'.

If the ROUND of the value of the 'Selector' parameter minus the value of the 'Offset' parameter is not 0, 1, 2, 3,4, 5, 6 or 7 the 'Destination' parameter is set to the value of the 'Default Input' parameter.

19.3.12.1 Cascade Operation

By connecting the 'Destination' parameter to the 'Default Input' parameter of another Type 12 (Digital Input Selector) Logic module and setting the 'Offset' parameter of the first module to '0' and the 'Offset' parameter of the second module to '8' one digital value can be selected from 16. If required more Type 12 (Digital Input Selector) Logic modules can be connected and their 'Offset' parameters set to '16', '24' etc allowing selection from even more digital inputs.



19.3.12.2 Type 12 (Digital Input Selector) Logic Module Parameters

Type 12 (Digital Input Selector) Logic modules include the parameters common to all Logic modules - see **"Logic Module Common Parameters" on page 357**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Default Input	The value used when the result of the calculation above is not 0, 1, 2, 3,4, 5, 6 or 7.	0 or 1	0	I	F
Input a	The value used when the result of the calculation above is '0'	0 or 1	0	I	a
Input b	The value used when the result of the calculation above is '1'	0 or 1	0	I	b
Input c	The value used when the result of the calculation above is '2'	0 or 1	0	I	c

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Input d	The value used when the result of the calculation above is '3'	0 or 1	0	I	d
Input e	The value used when the result of the calculation above is '4'	0 or 1	0	I	e
Input f	The value used when the result of the calculation above is '5'	0 or 1	0	I	f
Input g	The value used when the result of the calculation above is '6'	0 or 1	0	I	g
Input h	The value used when the result of the calculation above is '7'	0 or 1	0	I	h
Offset	The value subtracted form the value of the 'Selector' parameter. It allows support for base values of 0 and 1. It also supports the ability to cascade selector modules - see **'Cascade Operation'**. The default of 1 allows support for operation with BACnet multi-state objects.	Full Analogue	1	I	O
Selector	Determines which of the 8 digital inputs is used for the 'Destination' parameter's value.	Full Analogue	0	I	S

19.4 Logic Module Common Parameters

Logic modules include the following parameters that are common to most modules: 'Disable Module', 'Label', 'Page', 'Sequence Step', 'Sequence Table', 'Page', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination /Destination D	Not type 4, 6, or 7. The module output is digital for all module types except type 8 which is analogue.	0 or 1 Or Full analogue	0	O	D
Type	The type of Logic module	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	1 = Type 1 (Combination) 2 = Type 2 (Timer) 3 = Type 3 (Readback) 4 = Type 4 (Counter) 5 = Type 5 (Delay) 6 = Type 6 (Hours Run) 7 = Type 7 (Fan Out) 8 = Type 8 (D to A) 9 = Type 9 (Change of State) 10 = Type 10 (Flip Flop) 11 = Type 11 (Latch) 12 = Type 12 (Digital Input Selector)	N	Y

19.5 Logic Module Connectives

The following parameters are available as connectives:

Type 1, 2, 3, 5, 8, 9, 10, 11, 12

Destination/Destination D

Type 4

Cumulative Output
Rate

Type 6:

Interval Limit Alarm
Hours Run Output

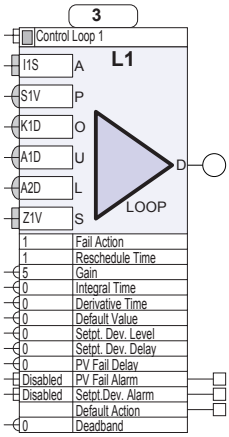
Type 7:

J
K
L
M
N
O

20 LOOP MODULES

Module Identifier	Ln (where n is module number)
Default overview parameters	$, D
Size in brIQs	55

20.1 Functionality



Loop modules implement a proportional, integral, and derivative (PID) control algorithm suited to the majority of HVAC control applications.

Two setpoint inputs are provided in order to support switching between occupied an non occupied setpoints.

Proportional gain, integral time and derivative time constants may be adjusted or driven from a connected module in the strategy.

They perform the mathematical calculations that determine, the value of the 'Destination' parameter based on present, and past values of measured variables. They compare the measured value with the current setpoint. The difference between the measured value and the current setpoint is the error. The module then calculates the value of its output to reduce the error using the 'Proportional Gain', 'Integral Time', and 'Derivative Time' parameters and the previous output value.

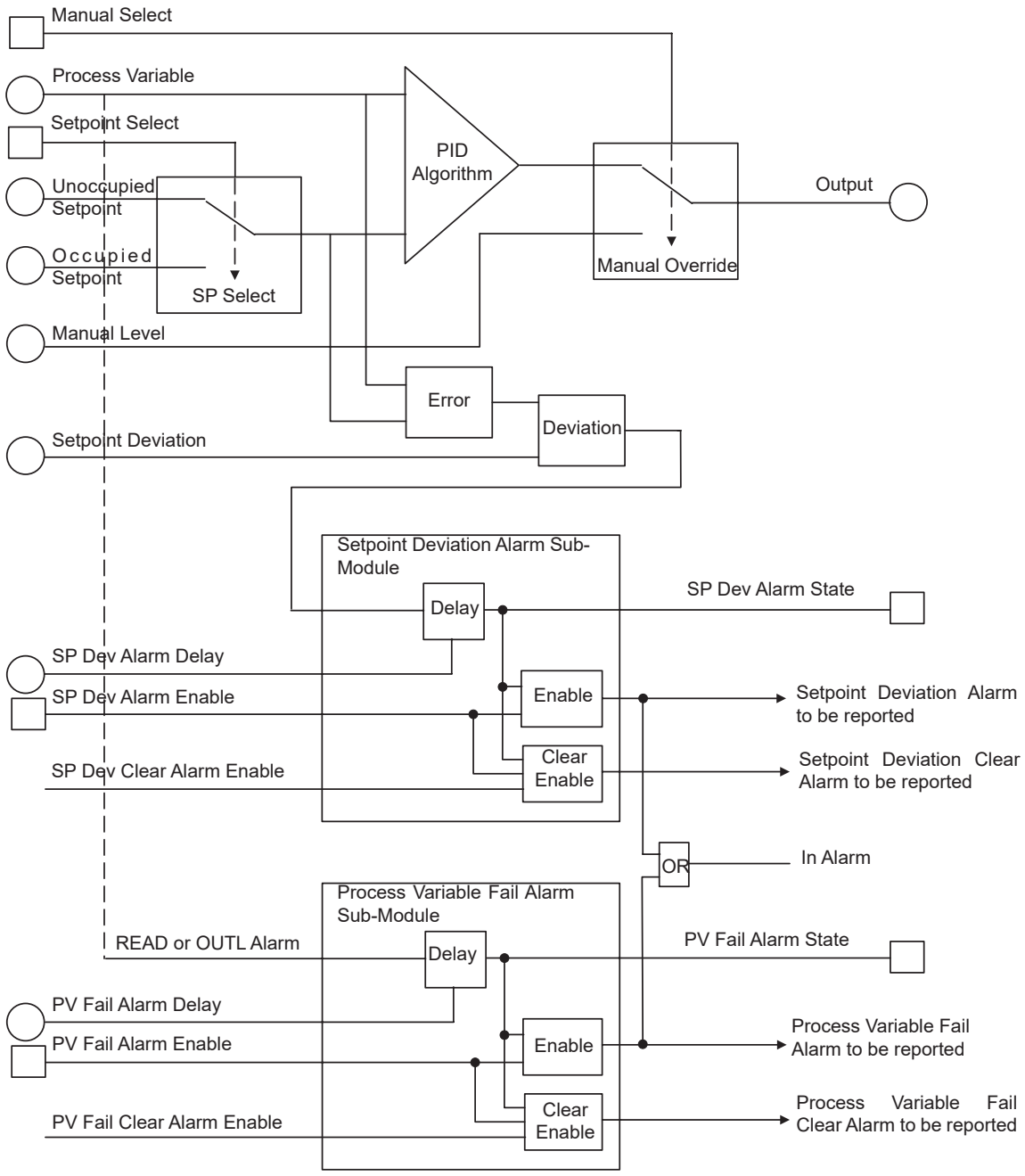
Loop modules appear on the strategy page and must be sequenced, they cannot be event driven, and must not be multiplexed.

Loop modules provide the following types of closed loop control - see **"Closed Loop Control" on page 361**:

Proportional Control
Proportional and Integral Control
Proportional + Integral + Derivative (PID) Control
Loop Integral Only Action

20.1.1 Loop Module Schematic

This diagram below shows how the various parameters described below inter-operate.

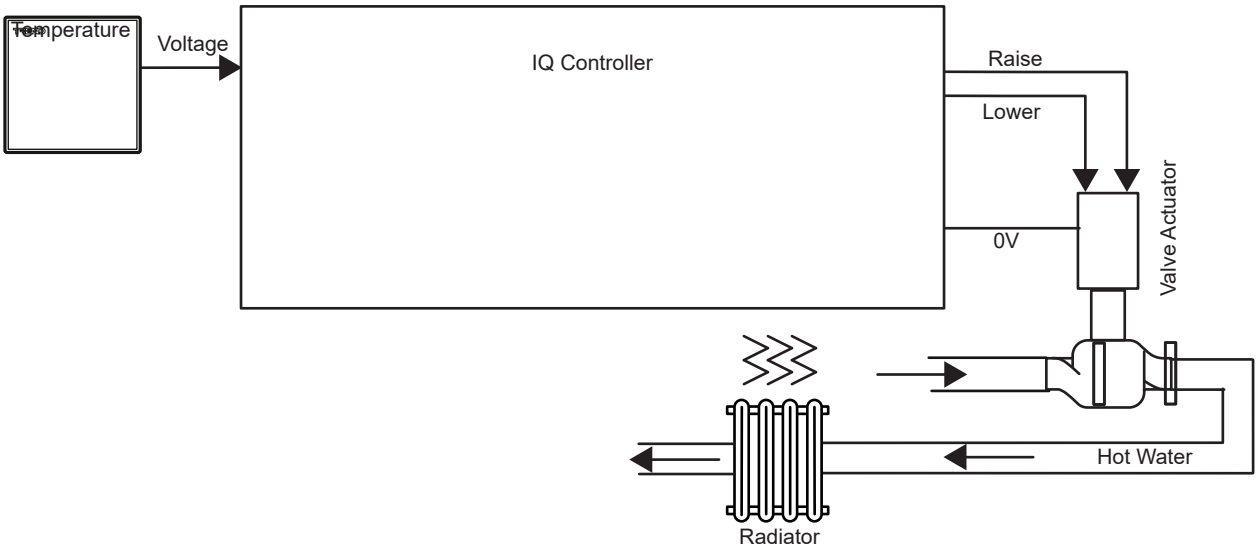


The ‘Setpoint Select’ parameter specifies the required setpoint from the either unoccupied or occupied setpoint. The loop algorithm compares the process variable with the selected setpoint and calculates the Loop module’s output. The ‘Manual Select’ parameter can override the Loop module’s output to the manual level.

If the error between the process variable and the selected setpoint is greater than the setpoint deviation level the alarm condition is passed to the Setpoint Deviation Alarm sub-module. If it exists longer than the delay time the SP Dev Alarm State output is generated. If the alarm is Enabled the alarm is reported and the In Alarm output is set non zero. When the alarm clears the clear alarm will be reported if the alarm is Enabled and the Clear Enable is also set.

If the Process Variable input has either a Read or Outside Limits alarm status accompanying the value, the alarm condition is passed to the Process Variable Fail Alarm sub-module. If it exists longer than the Delay time the PV Fail Alarm State output is generated. If the alarm is Enabled the alarm is reported and the In Alarm output is set non zero. When the alarm clears for longer than its alarm delay the clear alarm will be reported if the alarm is Enabled and the Clear Enable is also set.

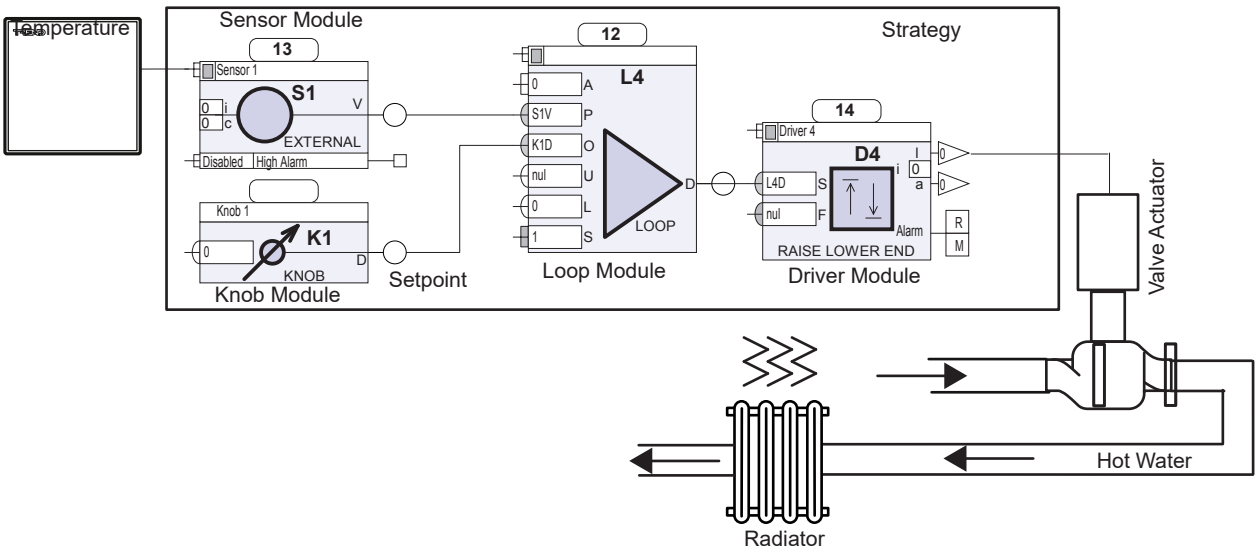
20.1.2 Closed Loop Control



In the above diagram, the sensor is measuring the temperature of an air space. The output of the sensor will be a signal which varies in a defined way with the temperature.

The controller will decide from the temperature measurement whether more, less, or sufficient heat is being supplied and either raise or lower the valve actuator by energising the relevant relay on the output side.

The valve will then move, the hot water flow will change and then so will the room temperature by way of the radiator.



The process by which the controller decides what the output level should be for a given input is described by the strategy diagram as shown above. In the strategy diagram the software modules are represented by fixed shapes.

In the above strategy the signal from the sensor is converted into a number which the sensor module will scale into degrees centigrade. The sensor module also checks whether the sensor reading is within alarm limits. The value from the sensor is the loop process variable. The Loop module compares the process variable with the required level or setpoint. The setpoint is the required room temperature in deg C. In the general case the process variable and the setpoint are in the same engineering units as the controlled variable.

The difference between the process variable and the setpoint is known as the error. In this example the error is the setpoint minus the process variable.

Error = setpoint (O) - Process variable
E = O - P

The Loop module will generate an output which is designed to reduce the error (i.e. move the process variable towards the setpoint). This output will be proportional to the error (the bigger the error, the bigger the output) and may also increase as the error persists. The proportional output is set by the 'Gain' parameter, and the amount to overcome a persistent error is set by the 'Integral' parameter.

If the process variable temperature is below the setpoint, a heating loop such as this will produce an output greater than 0. The Loop module output is always measured in percent as this is a general ratio term which can be applied to any units being measured or controlled.

Generally the Loop module's output is a demand (here a heating demand), and 0% requires 0 heating whereas 100% requires maximum heating. In this range 0-100% various items of HVAC equipment may be sequenced i.e fans on at 20%, a damper from 30% to 60%, and an electric heater on at 90%.

In this example, however, we are controlling a valve so that 0% means valve closed, 100% means valve fully open and any intermediate percentage level corresponds to a valve position.

Since the temperature is below setpoint our loop will have some output, say 30% which will be passed to the driver module. In this case the Raise/Lower driver module used knows the valve position. If the valve is positioned at 20% then it needs to raise the valve position by 10%. This means it will pulse the raise line for 10% of the full drive time.

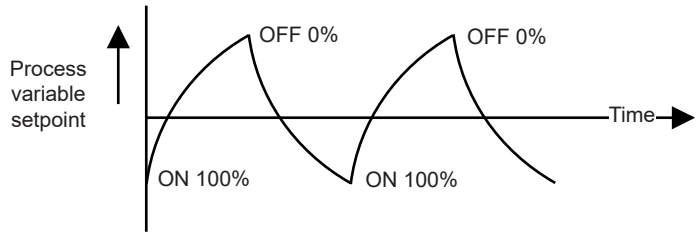
The valve will then be 30% open, more water will flow through the pipe, the radiator will emit more heat, and the air space temperature will rise.

When the temperature sensor is next monitored by the loop module it may have reduced the error, so the loop output may fall, the valve will close a bit and reduce the rate of heating.

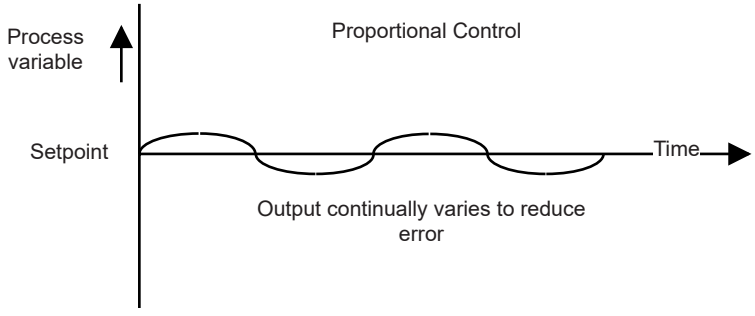
This will produce a continual process of measure, calculate, output, change process, measure, calculate etc. This is a closed loop. The Loop Module is the prime module in controlling the process variable, in this case the room temperature.

20.1.2.1 Proportional Control

With a simple thermostatic control system, the heating is either ON (i.e. 100%) or OFF (i.e. 0%). With the thermostat set to 20 deg C it may bring the heating on as the temperature falls below 19 deg C and then switch it off as the temperature rises above 20 deg C. This produces a sawtooth graph of temperature against time.



If a sequenced heater is used so that for a large error two heating elements are switched on and for a smaller error only one element is selected, then we can now select 100%, 50% and 0% heating. If we add more heating stages we approach a condition where we only select the number of elements needed at a particular level. This is approaching proportional control where the number of elements is proportional to the error. This results in a smooth curve about the setpoint.



For proportional control, the Loop module's output is proportional to the difference between the setpoint and the measured value. This difference is known as the error:

Error	=	Occupied setpoint - process variable
E	=	O - P

As the measured value increases towards the setpoint value, the error decreases and the Loop module's output is reduced in proportion to the error. Conversely, as the measured value decreases away from the setpoint value, the error increases and the loop output is increased in proportion to the error.

Since the Loop module's output is not allowed to go negative, it is 0 for negative errors.

The reason that the process variable overshoots the setpoint is that having selected 0 output level, it takes some time for the plant to respond (e.g. radiator to cool down). During this time the temperature may raise above the setpoint.

The Loop module's output is not allowed to go above 100%, so once this is reached any further increase in error will have no effect on the output.

Thus, the Loop modules has a defined output band called the proportional band over which the output is proportional to the error. When the measured value equals the setpoint, the loop module's output will be 0. As the error increases, the output will increase linearly until the output reaches 100 percent which is at the end of the proportional band where the output will remain constant until the error decreases. The relationship between error and loop output is known as Gain.

proportional loop	o/p = Gain x Error	Where
from above	o/p = G x (O - P)	

Where:

O = Occupied Setpoint
P = Process Variable
G = Gain

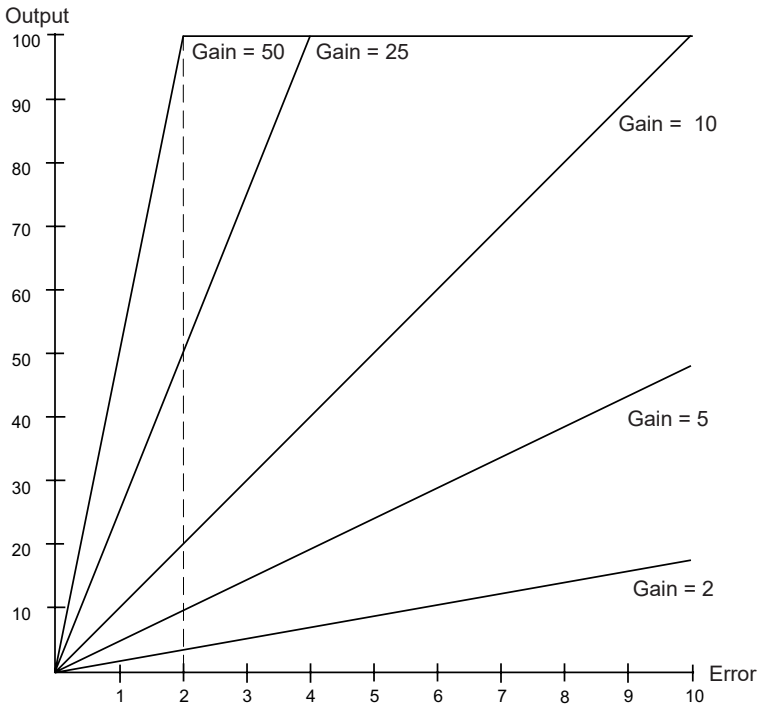
This gain is expressed in percent per engineering unit of error.

With a gain of 10 and an error of 2, the output will be 20%.

The maximum output will be reached when the error reaches 100/10 i.e. 10 deg C. Thus, the proportional band runs from 0 deg C error to 10 deg C error. The proportional band is thus 10 deg C and in general the proportional band can be calculated from :

proportional band = 100/Gain engineering units

The graph below shows the relationship of output to error for various gains.



Note:

Error = Setpoint - Measured Value
Output = Error x Gain

20.1.2.1.1 Cooling Control

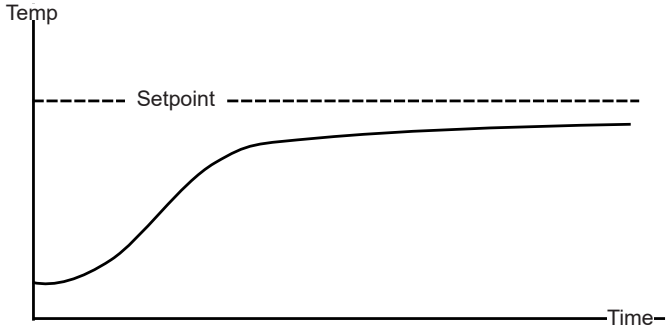
In the case of cooling control, the Loop module's output must start to increase when the process variable is above the setpoint (when the error is negative) because:

- The loop output cannot go negative
- The loop output cannot go above 100%
- Loop output = G x (O - P) within range 0 -100%

The loop gain must be set negative so that the Loop module's output becomes positive for a negative error. So for a cooling loop the gain is set negative.

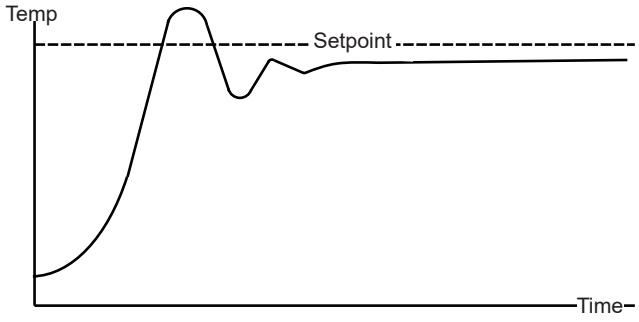
20.1.2.1.2 Effect of Changing Gain

The choice of loop gain is critical to the performance of the system and the Loop module's response to changes in process variable. The graph below shows the response of a system to an increase in setpoint (e.g. from non occupied to occupied) with insufficient gain.



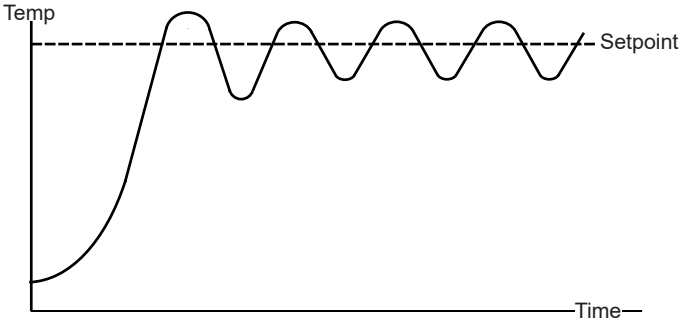
The temperature response is a very slow rise which drifts up to a steady level well below the setpoint (overdamped).

If the gain is increased there comes a point, as shown below, where the second oscillation is a quarter of the amplitude of the first. This is optimum loop gain for fastest response.



The temperature now rises rapidly, overheats and falls below the setpoint. It then rises again to a level which is 1/4 the height of the first peak above the final level. This is called 1/4 wave damping.

If the gain is further increased, as it increases the overshoot increases, more peaks appear and their magnitude increases (Damped Oscillation sub-critical) until a point is reached where the output oscillates about the setpoint.



When each peak is the same height as the previous; the critical stability limit has been reached. If the gain is further increased, the succeeding peaks will increase (super-critical) until the Loop module's output swings between 0% and 100%.

Note: Even the optimum gain produces a final steady level that is below the setpoint. This is generally due to some heat input being required to maintain the building at temperature, and the proportional control settling with a steady state error (or offset) where heat input is equal to heat loss. If the heat loss changes, the offset will also change. To overcome this, an integral term is added to the proportional output to form a proportional and integral control (PI) loop.

20.1.2.1.3 Example

If the building requires 20% of its output to keep the room at the temperature of 20 deg C on a particular day, then with a gain of 10, the error will be given by:

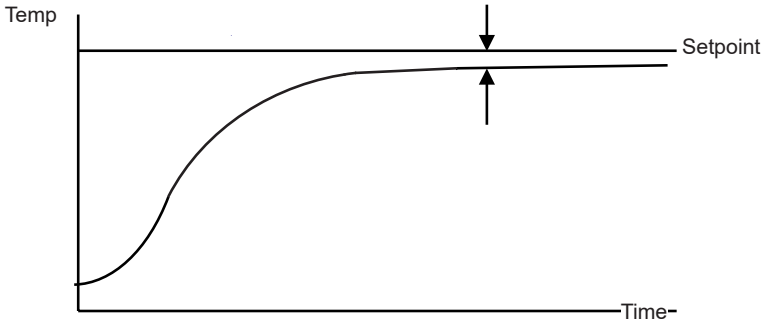
o/p = Gain x Error
therefore Error = o/p/Gain
= 20/10 = 2°C

Increasing the gain to 20 would decrease the error to 1°C but may affect stability as explained above.

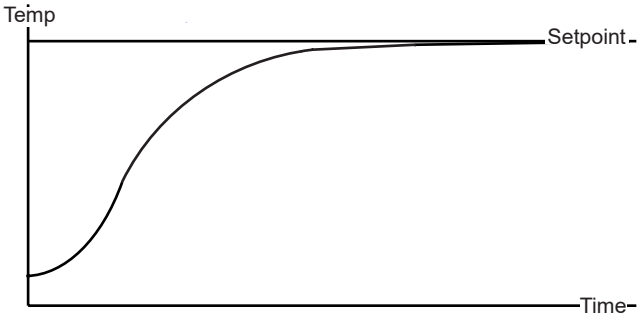
20.1.2.2 Proportional and Integral Control

The reason for adding the integral term to the proportional control is to eliminate the steady state error inherent in proportional only control. The integral action gradually increases the output to reduce the error so that the steady state condition (heat loss = heat input) now occurs at the setpoint value. This is in effect resetting (or moving) the proportional band to reduce the error.

The graph below shows the steady state offset.



Adding the integral term reduces the offset to '0'.



The integral term, integrates (accumulates) the error over a time period and adds this accumulated sum to the loop output. The integral time constant is defined as the time for which the loop output would have added to it an amount equal to the product of the error and the proportional gain. Therefore, the shorter the integral time, the greater the integral action, and the longer the integral time, the smaller the integral action. The integral time constant is expressed in minutes and for a loop with unit gain and unit error, the output will change by one unit (i.e. gain x error) over the integral time period.

Loop output = Gain x Error + integral term

The integral term can be seen in configuration mode and is added as above if the integral time (I) is non-zero.

$$\text{integral term} = \int \frac{G}{I}(E.dt)$$

Where:

G = proportional gain constant
E = error
I = integral time constant

Thus:

$$\text{Loop o/p} = GE + \int \frac{G}{I}(E.dt)$$

Just as increasing the gain produces instability, so does decreasing the integral time constant (increasing integral action) and selecting the correct integral time is critical.

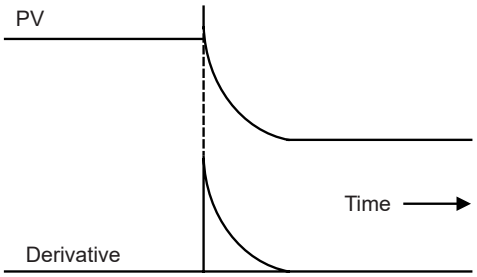
20.1.2.3 Proportional + Integral + Derivative (PID) Control

Generally buildings respond slowly to temperature change and the integral action can compensate for load changes reasonably well, but for the output to change, an error must exist. The system response is dependent on the size of the error and not on how rapidly the error is building up. To control this situation a derivative term is sometimes added to the Proportional and Integral (PI) Control, to form a three term PID controller. The derivative term looks at the rate of change of the error and adds to the output to resist the change. PID control is sometimes known as rate control.

PID control is a very effective technique, if the load on a control loop is changing rapidly. Derivative control is enabled by setting up a derivative time constant.

The derivative term is proportional to the rate of change of the error and gives an output if the error is building up rapidly. It similarly reduces the output if the error is falling rapidly. The derivative term is predictive looking at what might happen, the integral looks at what has happened, and the proportional terms looks at what is happening. The derivative term responds immediately to a change in process variable whereas both integral and proportional term require that an error builds up.

As the process variable (Pv) starts to fall, its initial rate of change is high so the derivative will rise to a maximum, then as the Pv rate of fall reduces, the derivative will start to drop until, when the Pv is steady again, the derivative is 0.



Loop o/p = Gain x Error + integral - derivative

The derivative term is shown subtracted because it is calculated from the rate of change of the process variable not the rate of change of the error. It can be seen on the web page and is subtracted as above if the derivative time constant (T) is non-zero.

$$\text{derivative term} = T\frac{d(Pv)}{dt}$$

Where:

d(Pv) means rate of change of process variable with respect to time.
dt (where T is in minutes and dPv/dt is rate of change per minute)

Notice the bigger the derivative time constant, the bigger the derivative effect (i.e. unlike integral time constant).

$$\text{Loop o/p} = GE + \int \frac{G}{I}(E.dt) \quad - G\,T\frac{d(Pv)}{dt}$$

The derivative used in the IQ is filtered by taking the average of the current value and last value.

new Pv difference = (current Pv difference - last Pv difference)/2

The derivative term cannot be greater than +/-25%

20.1.2.4 Loop Integral Only Action

Integral only control is used to deal with fast changing HVAC equipment (e.g. light level, flow, and pressure control).

If the loop gain is set to zero, the loop adopts integral only loop action. The gain is forced to 0 if the Integral Only check box is selected in IQSET.

For integral only action, the integral time is the time taken to give a percentage output change equal to the current error (e.g. if the current error of a flow control loop is 5 l/s and it has an integral time of 1 minute, after 1 minute the output will change by 5%).

Integral times of around 3 minutes have been used for VAV damper control; shorter times may be used for lighting control.

20.1.3 Setpoint Selection

Each Loop module has two setpoints. These provide two different levels of control, for example, one for periods when the building is occupied and the other for periods when the building is unoccupied.

The setpoint to which the Loop module controls depends on the value of the 'Setpoint Select' parameter. It is a digital input to the Loop module which switches between setpoints. When set to '1', the occupied setpoint is used and when set to '0' the unoccupied setpoint is used. Its value is usually derived from an OSS module. However, this it may be derived from other parts of the strategy.

If the unoccupied setpoint is set to 'not used' in IQSET, the Loop module is turned off during non occupation: during which the module's output is set to '0', the 'Integral' and 'Derivative' parts are also set to '0'.

20.1.4 Module Disable

Loop modules can be disabled - see **"IQ4 Modules" on page 165**. The 'Disable Module' parameter will stop the Loop module being serviced. Its internal states and output will be maintained at their values prior to disabling.

20.1.5 Manual Override

The output of the Loop module can be manually set to a specified level (defined by the 'Manual Level' parameter) using the 'Manual Select' parameter. If the 'Manual Select' parameter is '0', then the Loop module is in auto and it will control to the specified setpoint. If the 'Manual Select' parameter is set to '1', then the output is set to the manual level and the variable, setpoint etc., are ignored.

When overridden, the integral value will be set to a value equal to the difference between the manual level and the sum of the proportional and derivative terms so that when it is set back to automatic operation, the output will initially be at the level it was when set to manual. The integral function will then operate normally by gradually changing the loop output to minimise the control error. This facility is called pre-loading the integral or sometimes preconditioning the loop, and results in a 'bump free' transfer from manual to closed loop control.

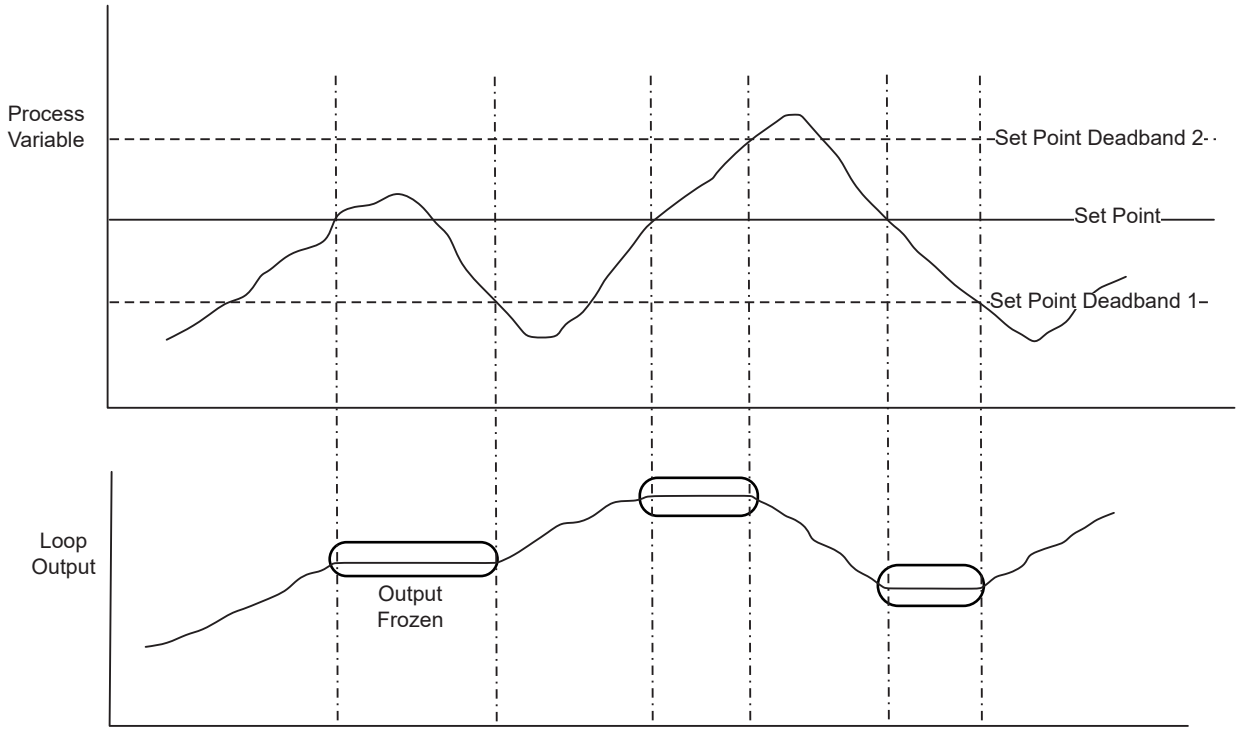
Note: When in manual mode, the Loop module's output may be outside the normal range of 0 to 100 %. Also while in manual mode, the module's alarm outputs, and the destination node alarm status are cleared.

The manual override facility is useful when commissioning a system, in that the correct operation of all the output devices can be verified in a methodical manner. The 'Manual Select' parameter's value can be set by strategy which enables strategies to be written which use the Loop module's 'Manual Select' parameter to effect an automatic override or similar function.

20.1.6 Deadband

The deadband (specified by the 'Dead Band' parameter) reduces the effect of noise and disturbances so that actuator wear is reduced. When the process variable achieves the setpoint then the Loop module's output is frozen until the absolute error becomes greater than deadband/2. While the output is frozen the integral term is back-calculated in order to give a bump free transfer when the output is unfrozen.

The ‘Dead Band’ parameter defines the width of the deadband in the engineering units of the process variable and is placed symmetrically about the setpoint as illustrated below.



20.1.7 Alarms

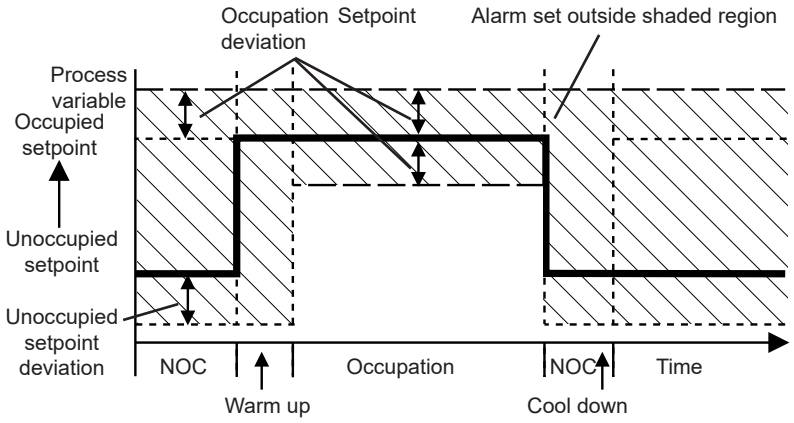
Loop modules can report two alarms:

Setpoint Deviation Alarm
Process Variable Fail Alarm

20.1.7.1 Setpoint Deviation Alarm

The setpoint deviation alarm (SDEV) is generated if the Loop module cannot maintain the process variable within the specified limits i.e when the difference between the occupied setpoint and the value of the process variable is too large. This difference is specified by the ‘Setpoint Deviation’ parameter.

During occupation the alarm will be generated when the value of the ‘Process Variable’ parameter is above or below the occupied setpoint by more than the ‘Setpoint Deviation’ parameter’s value. During non-occupation the alarm will be generated when the value of the ‘Process Variable’ parameter is above the occupied setpoint by more than the ‘Setpoint Deviation’ parameter’s value, and below the unoccupied setpoint by more than the ‘Setpoint Deviation’ parameter’s value. This is shown below:



The process of alarm generation is described fully in **“Item Alarm Generation” on page 77**.

Note: If the setpoint selection is not optimum start (i.e. is set to OTP), then the warm-up time is effectively reduced to '0' so that a setpoint deviation alarm is likely to occur at the start of occupation. This could be overcome by disabling the alarm with Type 2 (Timer) Logic modules or by making the deviation limit larger than the difference between the setpoints.

Note: The same alarm is generated for either a positive or negative deviation.

20.1.7.2 Process Variable Fail Alarm

The process variable alarm (PVFL) is generated if process variable input has a read error or out-of-limits alarm associated with it.

Note: The sensor alarm conditions are passed through any intervening function modules so, even if the sensor has been rescaled or combined with other values in some way, if the original sensor reading was in READ or OUTL (out-of-limit) alarm, the resultant analogue value will reflect this condition - see **"Function Module Alarms" on page 241**.

When the alarm occurs the Loop module ignores the value of the process variable and takes a pre-defined action (specified by the Sensor Fail parameter).

The 'Default Value' parameter defines the value used under the loop fail condition. Its actual meaning depends on the sensor fail action selected.

Sensor Fail Action	***Description***
Default PV	The Loop module uses the value of the 'Default Value' parameter as a fixed value for the process variable and calculates the output accordingly. This is usually used on single heating loops where a default reading might be set to, say 15 deg C on a Loop module with an occupied setpoint of 20 deg C and a non occupied setpoint of 10 deg C. In this case, although the loop would not control the heating, when there is a sensor failure it would produce 100% heating during occupied times and 0% heating outside occupation. When the alarm condition clears, normal control resumes.
Default OP	The Loop module uses the value of the 'Default Value' parameter as a fixed value for its output. This value can be set anywhere from 0-100% and would ensure some safe level of heating (or cooling). For example, one might choose 66% output to run two boilers out of three in the event of a sensor failure. Where 33% represents one boiler, 66% represents two boilers and 99% represents three boilers. When the alarm condition clears, normal control resumes.
Stay at Default OP	This is the same as 'Default OP' except the Loop module does not return to normal control automatically. Where this condition is present the Default Action digital output is set. To return the Loop module to normal control it is necessary to set the 'Manual Select' parameter to '1' and then back to '0'. This form of alarm action would be used where it is necessary for the engineer to oversee the return of the plant to normal running.
Freeze OP	The Loop module's output is held at its last output level before the alarm condition arose. This method is not recommended since although it works well for short-lived failures (as the HVAC equipment continues without any major disruption), if the failure happens outside occupation and continues for some time, then the loop would most likely be held at zero output, and no heating would be achieved the next day. When the alarm condition clears, normal control resumes.
No Action	The Loop module ignores the alarm status on the process variable and continues to attempt to control using the its value. *Note: For a READ alarm the last correctly received sensor value will remain in the sensor register, whereas for an out-of-limit alarm (OUTL) the scaled value corresponding to the actual reading received will be put into the sensor register even though this is beyond the normal range.*

The Loop module's 'PV Fail Alarm' parameter can be used by other modules to perform functions associated with the alarm condition.

20.2 Loop Tuning

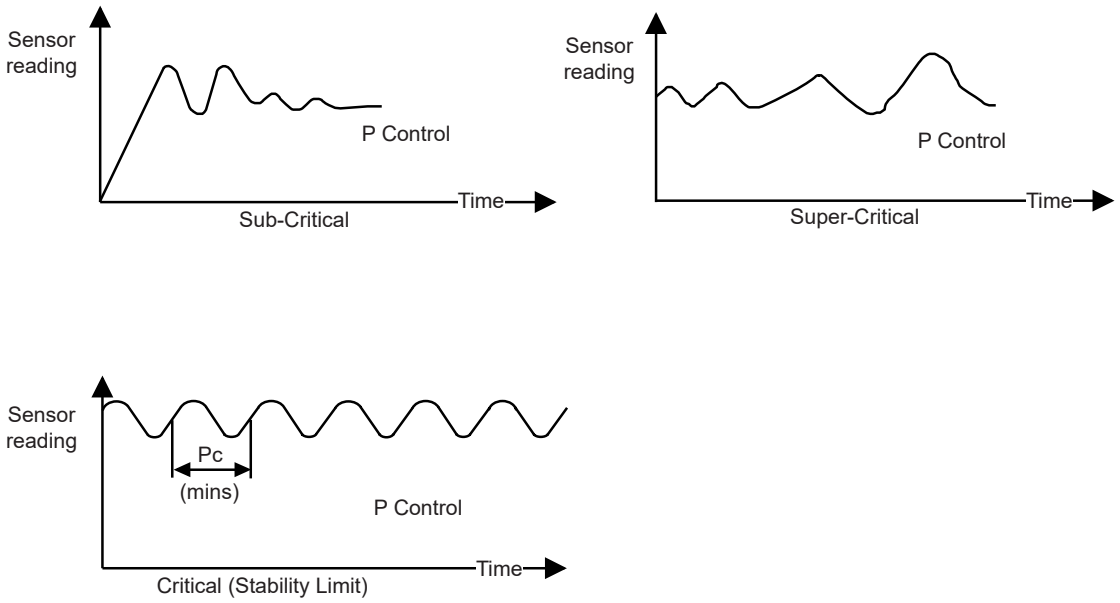
Tuning a control loop for optimum response is dependent upon what is meant by the term, optimum response. For example, the most important consideration may be to reach a setpoint as quickly as possible, albeit that an overshoot occurs. On other occasions, an overshoot is not acceptable, but a slower rate of rise can be tolerated. Consequently, tuning methods provide only a first order approximation to optimum tuning.

IQSET's Loop Tuner Applet is able to tune most loops. It will disable the Loop module and drive the output directly to observe the system response before suggesting the values of parameters required. The following describes manual methods that may be used if the Loop Tuner Applet proves unsatisfactory.

The rule of thumb of Ziegler and Nichols is by far the simplest procedure for such purposes, although it may give an excess of overshoot. There are two methods, one is based on the step-input response pattern of a process and the other utilises information obtained at the stability limit of a process under proportional control.

20.2.1 The Stability Limit Method

- Set the integral time and derivative time to 0.
- Increase the proportional gain of the loop in stages (say 5, 10, 15, 20 etc.) waiting each time to see the control result. When the loop is at critical stability limit, note the proportional gain Gc and the period of one complete oscillation Pc (minutes).



Gc is the proportional gain at stability limit when the integral and derivative time are set to 0.

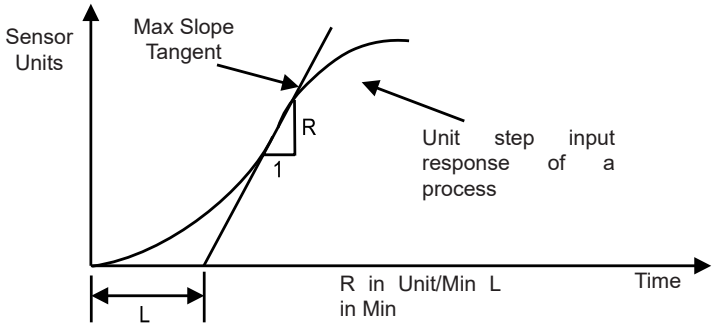
Go = 0.5Gc for P control
Go = 0.45Gc, Io = 0.83Pc for PI control
Go = 0.6Gc, Io = 0.5Pc, To = 0.125Pc for PID control

Go = optimum gain
Io = optimum integral time constant
To = optimum derivative time constant

20.2.2 The Step Method

This method avoids the necessity of putting the Loop module into oscillation

- Set the Loop module to manual at 0 output and allow conditions to stabilise.
- Increase the manual output by a step S (say 10%) and plot the response as follows:



Where:

R is the maximum slope of a tangent drawn to the unit-step input response of the process to be controlled
L is the time at which the tangent intersects the time axis. S is the change in manual output in percent.

Ziegler and Nichols recommended the following parameters for calculating optimum gain:

for P control Go = S/RL
for PI control Go = 0.9 x S/RL, Io = 3.3L
for PID control Go = 1.2 x S/RL, Io = 2L, To = 0.5L

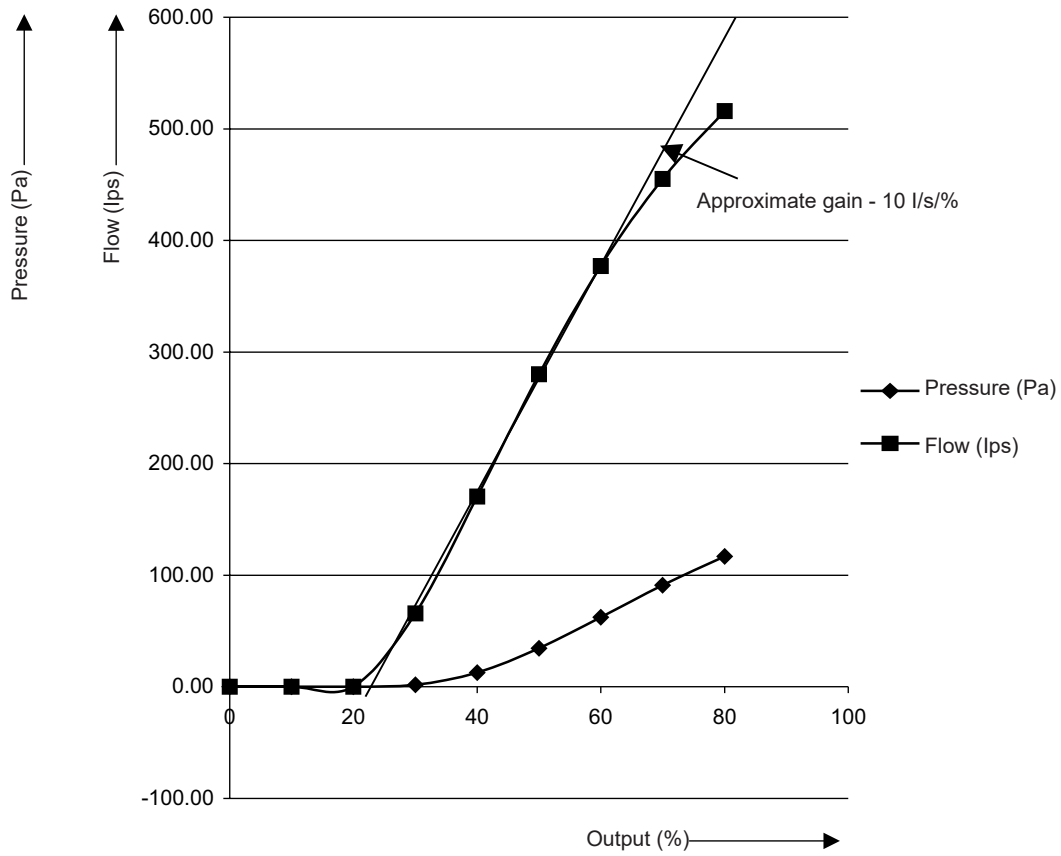
In these calculations gain is given in % per °C.

20.2.3 Tuning an Integral Only Loop

The integral should be tuned to match the effective system ‘gain’. This can be found graphically. For the example of a damper control loop:

- Manually set output level to 30% (e.g set the loop’s manual address bit to one and set the manual level).
- Measure airflow (process variable in lps, normally derived from a pressure sensor)
- Take further airflow readings at manual levels of 40%, 50%, 60%, 70%

- Plot graph and draw estimated straight line over linear part



Calculate slope of straight line: slope = 'gain' (F, lps/%)

The integral time is calculated from:

I = F x T

Where
I = integral time
F = plant gain
T = required response time constant

For a response time constant, T, of 20 s:

I = 10 x 20 = 200 secs (i.e 3.3 minutes)

For a response time constant, T, of 10 s I would be 1.7 minutes

Return output to automatic control by setting the 'Manual Select' parameter back to 0 (automatic)

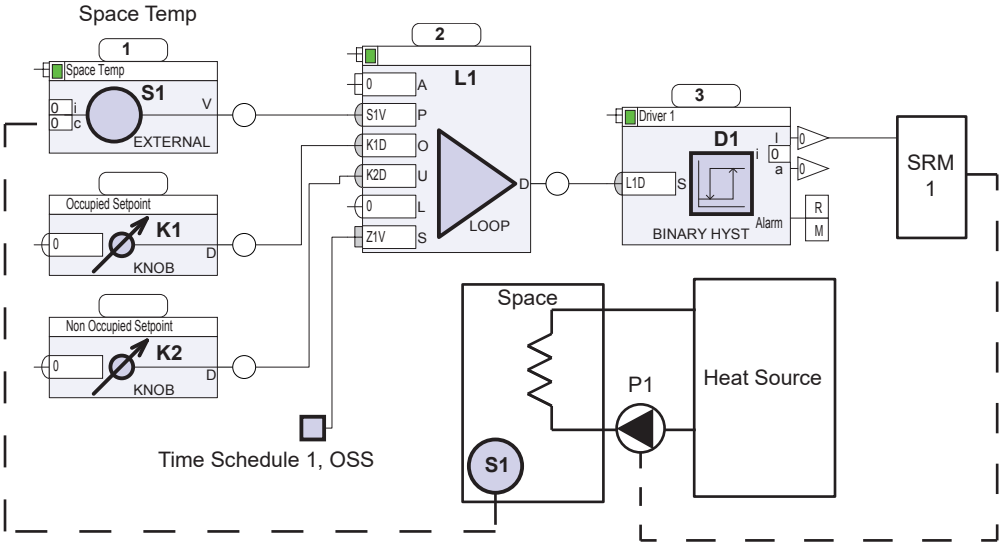
Finally check the closed loop response by introducing a step change in flow demand. If response overshoots then increase integral time, if response is sluggish, decrease integral time.

20.3 Loop Examples

The following examples are intended to illustrate in a simple way some of the techniques that can be used to control a basic radiator based central heating system. This example is used because everybody is familiar with the type of plant and can therefore concentrate on the benefits or otherwise of the types of control proposed.

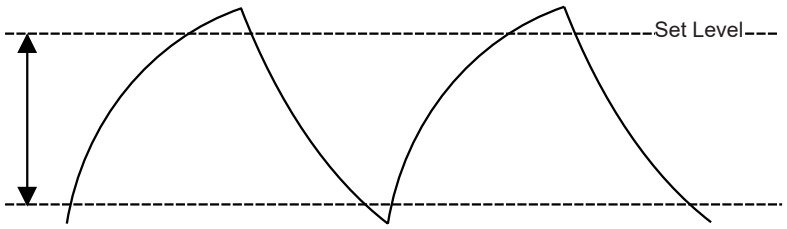
20.3.1 On/Off Pump Control

This is the simplest control that can be applied. The Loop module is set to have a high proportional gain, no integral and no derivative terms. When the sensor reading is higher than the setpoint, the Loop module's output will be '0'. When lower, it will be '100'. The output signal is fed to a simple binary hysteresis driver, such that an input signal greater than '99' will switch the output on and a signal of less than 1 will switch it off.



The Loop module acts just like a thermostat and has a distinct switch-on point and switch-off point. The extent of the deadband between these two points is set up by varying the module's 'Gain' parameter.

The output swing needed to switch the pump from ON to OFF or vice versa is 100. Therefore for a loop-gain of 100 the deadband would be 1 deg C; for a gain of 200 the deadband would be 0.5 deg C, etc. The graph of space temperature will be of the form:

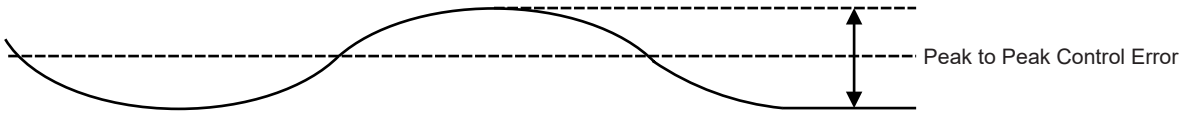


20.3.2 Time Proportional Control of Pump

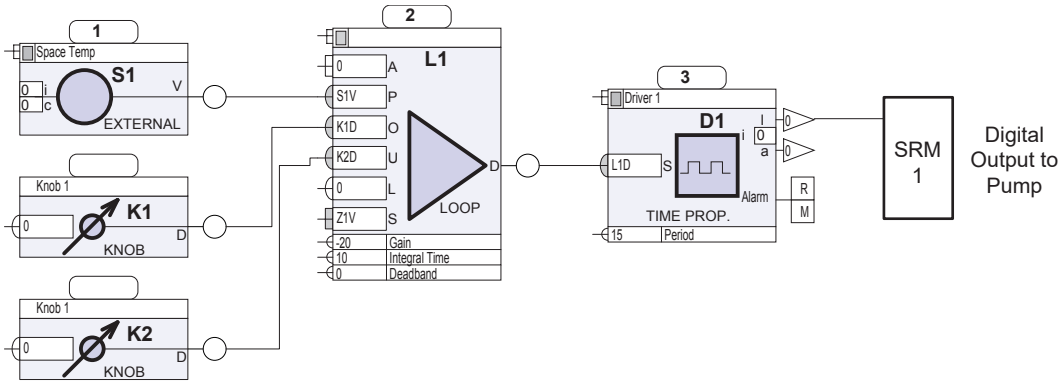
This uses the same HVAC equipment connections as in the previous example, but drives the pump in a pseudo continuous manner by using a time proportioning driver. In this case a time period is chosen for the Driver module to define the frequency at which the pump will be switched. The period chosen must be a compromise between the best level of control and the level of acceptable wear on the pump motor and contactor. In general the time constant of the heated space will be such that a period of 15 to 30 minutes will give very acceptable results. The time proportioning driver switches the pump on for a percentage of the period dependent upon the value of the input signal, which in this case is the Loop module's output. If the period is less than one third of the natural time constant of the space being controlled, control will appear to be continuous.

The Loop module's output will be continuously changing to match the measured temperature to the set temperature. In turn the pump run time will be varied in a proportional manner to match the required heat load.

The temperature graph of space temperature will be of the form:



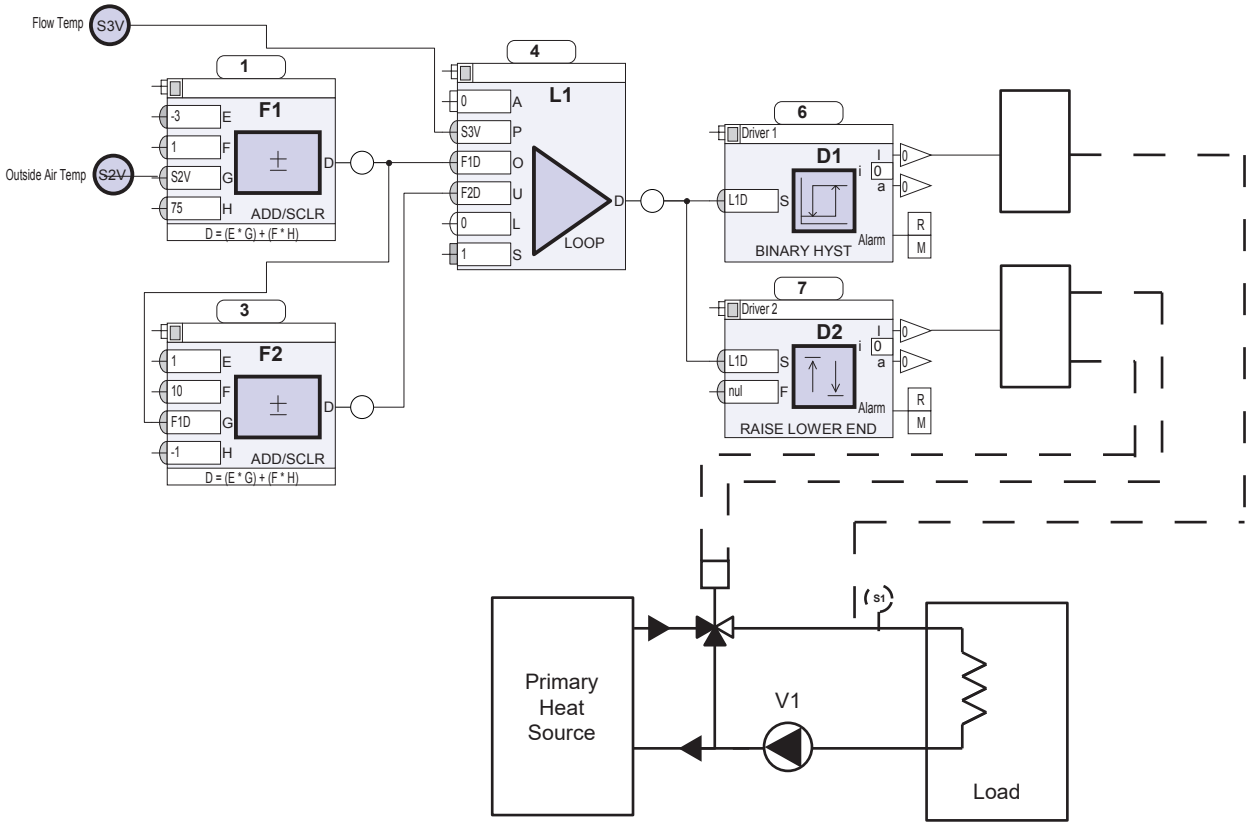
Installed systems using this method of control generally show peak error of less than 0.5 degC.



20.3.3 Weather Compensated Control

This is a traditional way of controlling radiator heated buildings where obtaining a representative space temperature reading is difficult. The flow temperature of the water passing through the radiator is regulated by a three-way valve. The temperature of this water is adjusted as a function of the outside temperature.

This is achieved by reading the outside temperature (Sensor 2) and using a Function module to calculate the required flow temperature setting. The characteristic required is that at the lowest design temperature, the flow temperature is set to its maximum and it reduces by approximately 30 deg C for every 10 deg C rise in outside air temperature. The aim is to match the heat input to the building losses so both the slope and offset of the characteristic need to be modified by experience.

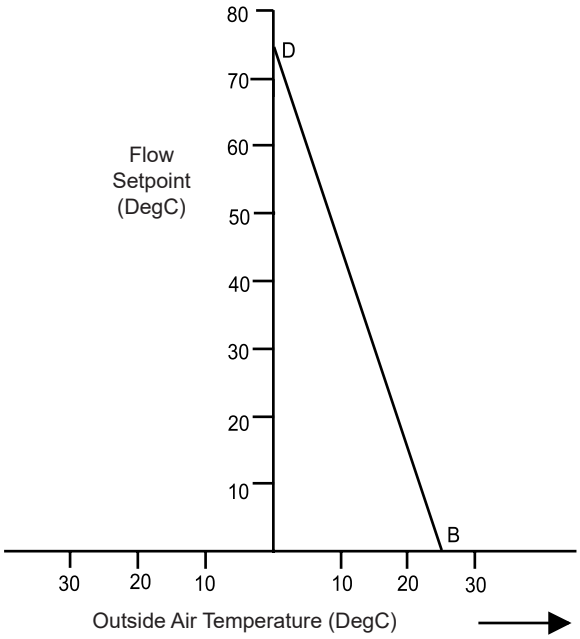


This could be designed as follows using the general equation for a straight line:

$$y = mx + c$$

Where:

m = slope i.e. change in y per change in x.
c = offset i.e. value of y when x = 0



For the required response shown the above Flow temp setpoint = y and Outside Air Temperature (OAT) = x i.e:

Flow temp setpoint = m (OAT) + c

Where:

m = D/B.

D = change in flow temp = -75 deg C (this is negative; it falls as OAT increases).
B = change in OAT = 25 deg C

Thus:

m = -75/25 = - 3 deg C flow temp/deg OAT

and

c = offset - value of flow temp when OAT = 0 = 75 degC

Thus our general equation becomes:

y = -3x + 75

Where:

y = flow temp S.P. (deg C)
x = OAT (deg C).

The general equation for an Type 8 (Adder/Scaler) Function module is:

D = E x G + F x H

Comparing the two equations:

D = y, E = -3, G = x, H = 75 and set F = 1

then:

D = -3G + 75

Which is the equation for Function module F1 in strategy diagram.

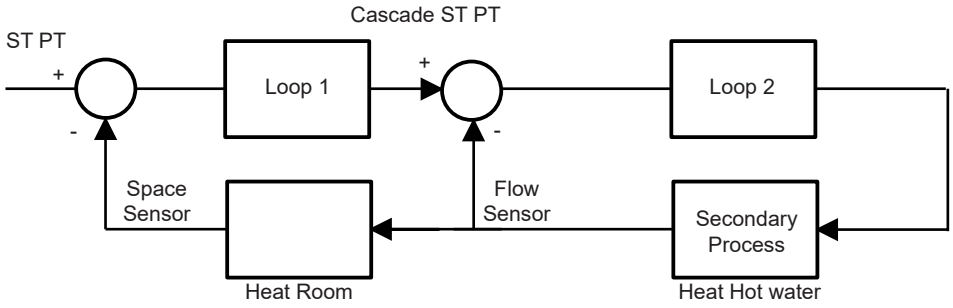
The output from the Function module is fed to the Loop module's occupied setpoint. The output of the Loop module controls the modulation of the valve. The measured value for the loop is of course the flow temperature which will be controlled to match the setpoint.

The diagram shows a second Function module F2 which is used to subtract a value from the occupied setpoint and pass the result to the non-occupied setpoint. This will reduce the flow temperature when the building is unoccupied.

A practical weather compensated strategy would be slightly more complicated if it was going to use optimum start control. For this to work effectively, the flow temperature would need to be set to a maximum during warm-up time.

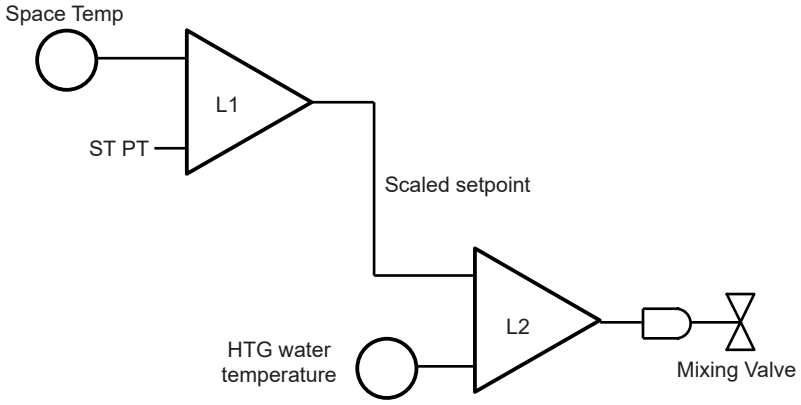
20.3.4 Cascade Control

A cascade control system consists of two connected processes, a primary process and a secondary process. These two processes are connected physically.



The above example shows the primary process of heating a room requiring a secondary process of heating hot water as a supply to the room heater.

The control is performed by applying a setpoint which is the required level for the space sensor. The first Loop module then generates an output which is a heat demand level. This can be used to produce a setpoint which is the required level for the hot water flow temperature sensor. This setpoint is a cascaded setpoint and is applied to the secondary or cascaded loop which actually controls the secondary process (heating the water). The strategy is shown below:

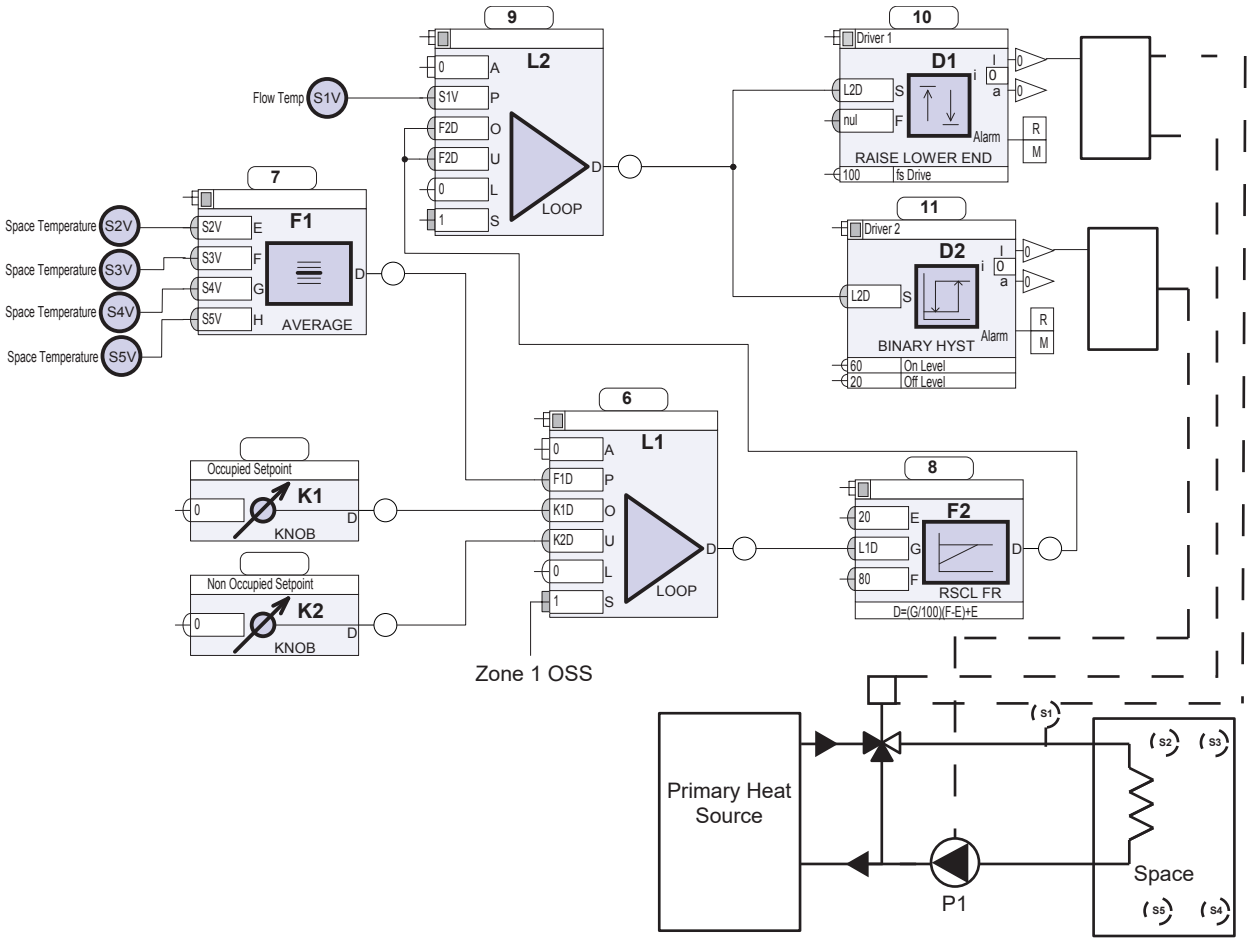


The reasons for using cascade control are:

- Disturbances arising within the secondary part of the process are corrected before they have much effect on the primary variable.
- The system is capable of responding much more quickly to set point changes than a single loop system.
- It allows the flow of the mass of energy from the secondary process to be controlled safely within prescribed limits e.g. in the example below the flow temperature setpoint would probably have setpoint limits of 20 deg C to 80 deg C.

20.3.5 Flow Temperature Cascade Control

This example shown below is of standard cascade control which is often used with variable flow temperature heating systems and with AHU control. Measurements of the space temperature are taken from sensors S2, S3, S4 and S5 and are averaged by Function module F1, (F1 could be used to take the maximum or minimum of sensors S2, S3, S4 and S5 as required). The output of module F1 is then used as the process variable by Loop 1 which has space temperature setpoints, occupied and unoccupied selected by the Setpoint Select input (Time Schedule 1 OSS).



L1 controls the primary process and produces a heat demand which is rescaled from 0 to 100 by F2 to a range from setpoint lower which is the minimum acceptable flow temperature (20 deg C) to setpoint upper which is maximum acceptable flow temperature (80 deg C). The output of F2 is used as a setpoint for L2.

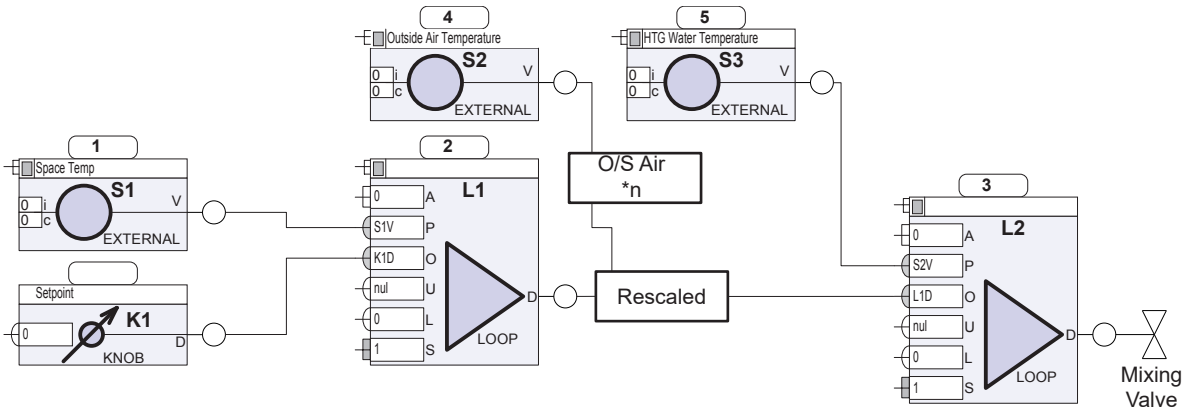
S1 measures the flow temperature and provides the process variable signal for L2. L2 is the main Loop module controlling hot water temperature by modulating valves/dampers and switching, pump/fans ON and OFF to maintain the setpoint fed to it via S2, S3, S4, S5, F1, L1 and F2. The setpoint for L2 is derived from the space temperature using L1.

Note: Loop 2 has its occupied and unoccupied setpoints connected together so that its 'Setpoint Select' parameter would have no effect. The change in level with respect to occupation is performed by L1.

20.3.6 Feed Forward Control

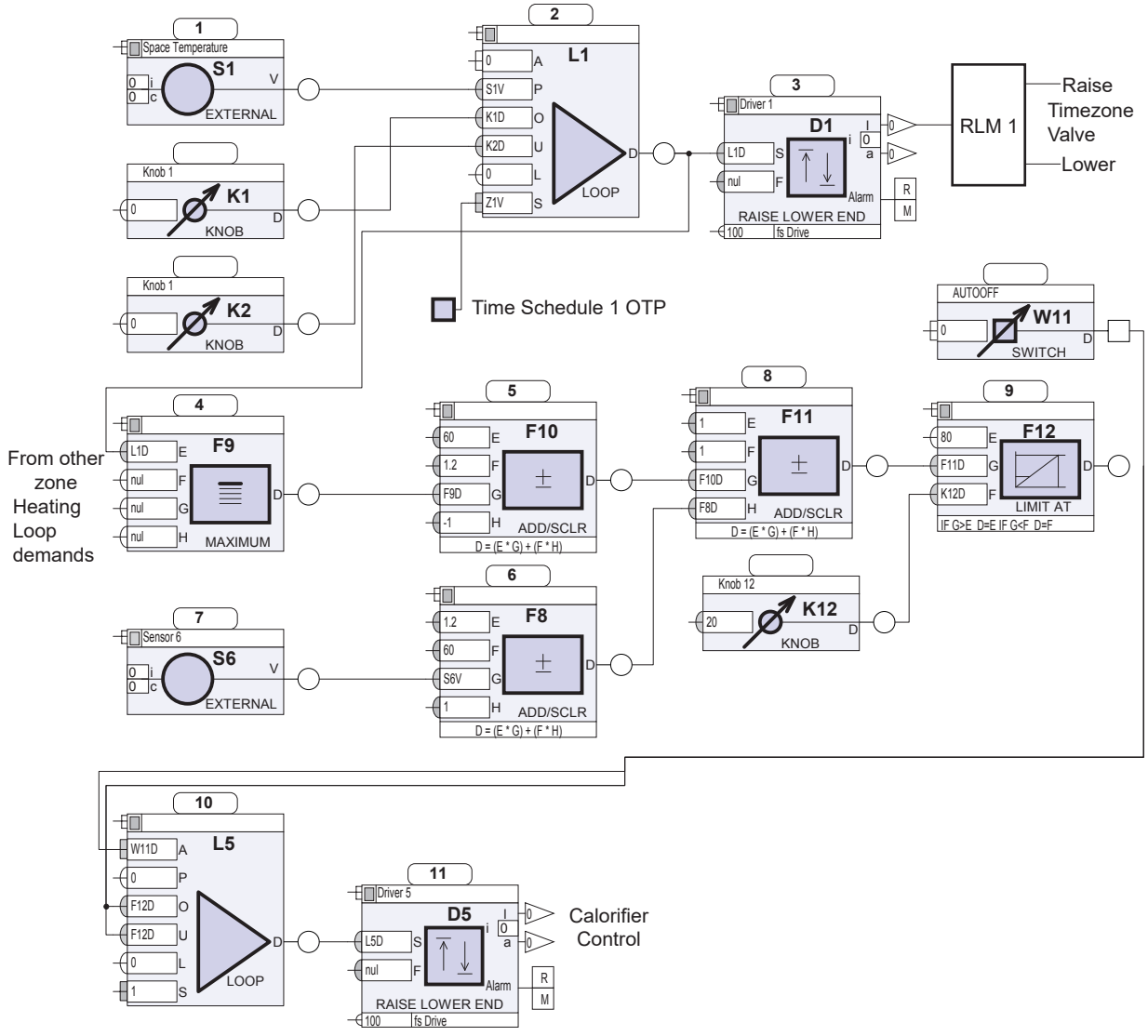
It is possible to combine the advantages of cascade control with the compensation for outside air temperature.

Outside air temperature compensation is in fact anticipating that the space temperature will drop if the heating is not increased. This is achieved by feed forward control where the outside air temperature drop is added into the Loop module's output to increase heat demand, and this is used to trim the setpoint.



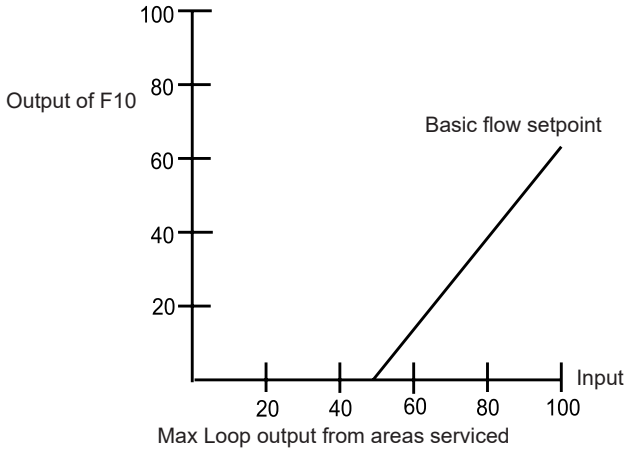
Feed forward control changes the secondary Loop modules setpoint before the primary variable detects the change (which would normally be after some time lag).

20.3.7 Cascade Control with Outside Air Temperature Compensation



The example shown in the strategy diagram above actually has four heating zones each with their own separate control valves controlled by four loops. Only L1 is shown but the other three zones are similar. The Loop modules' outputs for the four zones are examined by F9 which will calculate the maximum.

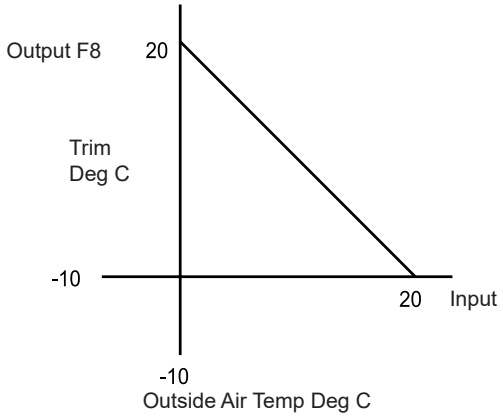
The basic flow setpoint is derived from this maximum loop output by F10 which is set up to perform the function shown in the graph below.



Thus when G = 60/1.2 = 50, the output = 0 and it rises to G = 1.2 x 100 - 60 = 60 when the input = 100.

This is then added to a value derived from the outside air temperature by module F8.

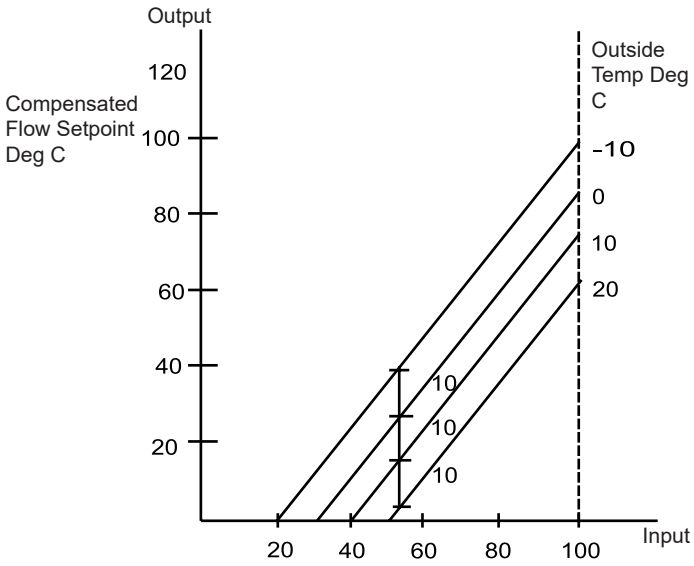
This outside air temperature (OAT) trim is calculated by module F8 which has the equation:- D = -G+20



This trim is generated as follows:

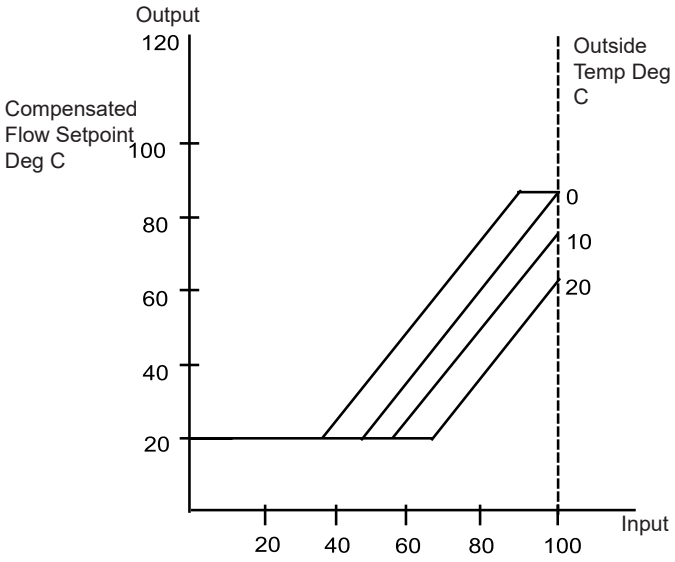
when OAT = 30, trim = -30 + 20 = -10
when OAT = -10, trim = +10 + 20 = 30

The function of F11 is to add the trim to the cascaded setpoint and this can be seen from the family of straight lines shown below.



The trim will be 0 at OAT of 20 deg C, and thus the graph at 20 deg C is the same as that shown previously for the output of F10 above. As the OAT decreases in 10 deg C, steps the setpoint will have 10 deg C added to it.

This basic setpoint is then limited by F12 so that it cannot rise above 80 deg C. The lower limit is set by Knob 12 (node 232) which will normally be 20 deg C. The outside compensated setpoint fed to the calorifier loop L5 will then vary as follows:-



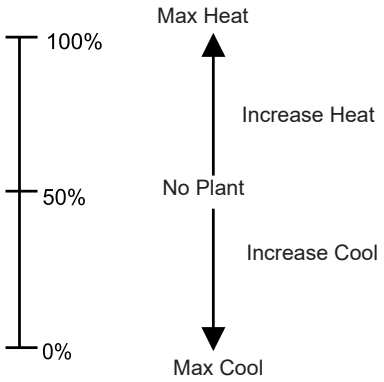
Note: L5 has an auto/manual control on its 'Manual Select' parameter. If this is set to '1' then the Loop module's output will go to 0% as specified by its 'Manual Level' parameter.

20.3.8 Heating and Cooling

Loop modules are set to cooling loops by setting the proportional gain to be negative. Thus the Loop modules operate in heating mode only or cooling mode only. When HVAC equipment such as dampers which span both the heating and cooling regimes are controlled by the Trend system, a way must be found to change from heating to cooling without driving heating and cooling simultaneously.

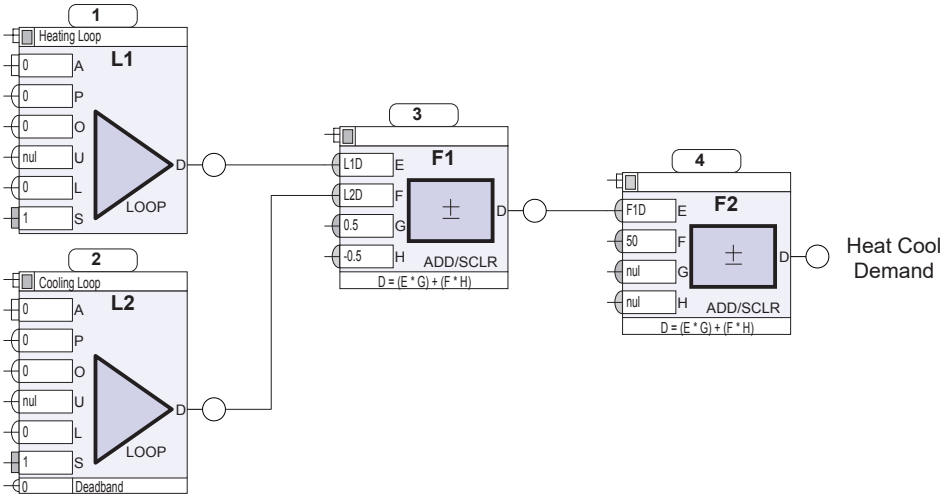
This can be achieved by having a continuous variable where 0% is full cooling and 100% is full heating and 50% selects both OFF. It can be achieved by adding half the output of the heating loop (0 -100) to minus half the output of the cooling loop (0 -100) and since the output must be 50 when both are 0, an input of 50 is added.

Heat/Cool Demand = 0.5 x Heat demand -0.5 x Cool demand + 50.

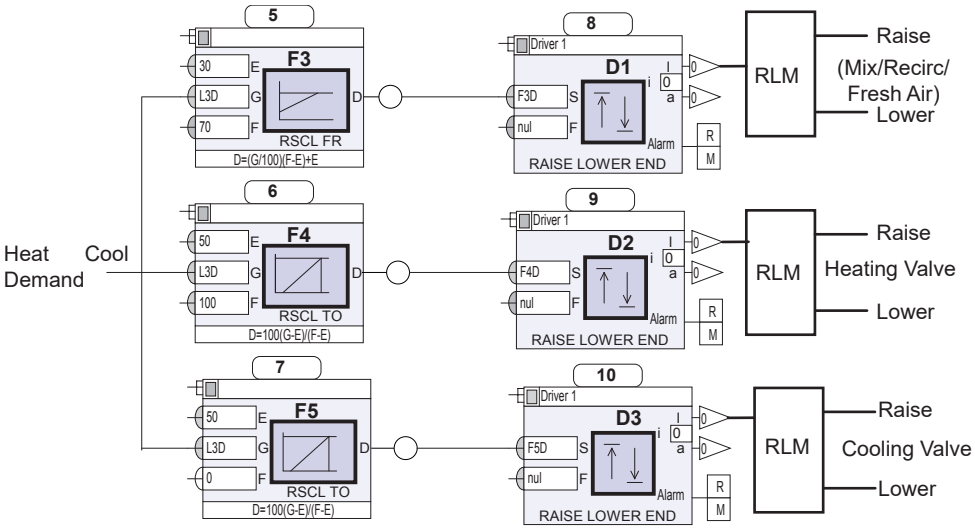


Having two separate Loop modules has the following benefits:

- If the primary aim of the system is to save energy, then it is desirable to hold off the cooling plant for as long as possible. This to a lesser extent also holds true for the heating plant. This is simply achieved by ensuring a deadband between the heating and cooling setpoints, e.g. heating could be set to 19 deg C, cooling 22 deg C.
- Better control can be achieved by virtue of being able to separately 'tune' the heating and cooling regimes.
- Because of the way the signals are combined to form one output control signal, there will not be a situation where the plant items will be fighting one another, i.e. heating and cooling plant on together.



This can be achieved by two Function modules as shown above. This combined demand can then drive dampers about a centre point and separate heat and cool plant as below:



This strategy will set the damper midway under conditions of zero demand. For maximum heat demand the dampers will be at 70% recirculation with the heating valve fully open. For maximum cooling the recirculation dampers will be at 30%, (70% fresh air), with the cooling valve fully open.

20.4 Loop Module Parameters

Loop modules include the following parameters that are common to most modules: 'Label', Disable Module', 'Page', 'Sequence Step', 'Sequence Table', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text comms*
Alarm Delay (secs) (PV Fail)	The period (in seconds) loop input must be invalid before a PV Fail alarm is generated.	0 to 172800	0	I	f1(D)
Alarm Delay (secs) (SP Deviation)	The period (in seconds) the input must be not at the required setpoint before a Setpoint Deviation alarm is generated.	0 to 172800	0	I	s1(D)
Alarm Group (PV Fail)	The Alarm Group module to which the PV Fail alarm will be sent.	0 to 500	0	N	f1(G)
Alarm Group (SP Deviation)	The Alarm Group module to which the Setpoint Deviation alarm will be sent.	0 to 500	0	N	s1(G)
Alarm Sent (PV Fail)	Indicates if the PV fail alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	f1(K)
Alarm Sent (SP Deviation)	Indicates if the Setpoint Deviation alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	s1(K)
Alarm State (PV Fail)	Indicates if alarm condition has been present for longer than the delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No Alarm	0	O	f1(S)
Alarm State (SP Deviation)	Indicates if the alarm condition has been present for longer than the delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No Alarm	0	O	s1(S)
Clear Alarm (PV Fail)	Enables/disables reporting of the PV Fail clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	f1(C)
Clear Alarm (SP Deviation)	Enables/disables reporting of the Setpoint Deviation clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	s1(C)

Parameter	*Description*	*Range*	*Default*	*Type*	*Text comms*
Clear Alarm Delay (secs) (PV Fail)	v3.50 or greater. The period (in seconds) the PV Fail alarm must be cleared before the clear alarm is generated.	0 to 172800	0	I	f1(d)
Clear Alarm Delay (secs) (SP Deviation)	v3.50 or greater. The period (in seconds) the Setpoint Deviation alarm must be cleared before the clear alarm is generated.	0 to 172800	0	I	s1(d)
Deadband	The area around the setpoint where loop output will be frozen once process variable reaches setpoint (i.e. loop output is frozen from this point until absolute error reaches deadband/2).	+ Full analogue	0	N	B
Default Action	Indicates whether the loop fail action type 2 has been taken. Once set to '1' it remain in that condition until it is reset by cycling the 'Manual Override' parameter.	0 or 1 1 = Action taken	0	O	a
Default Value	Default value used by fail action. The default value used when the loop fail action is taken. For loop fail action type 1 it should be set to a value in engineering units (e.g degC). For loop fail actions type 1 and 2 it should be set to a value in per cent. It is not used for loop fail actions type 3 and 4.	Type 0 fail action Full analogue Type 1 or 2 fail action 0 to 100	0	I	E
Derivative Time (mins)	Amount to multiply the gain times rate of change of error (e.g. degC/min) to produce derivative part of the output.	Full analogue 0 = No derivative action	0	I	T
Derivative	Derivative part of output.	-25 to +25	0	O	d
Enabled (PV Fail)	Enables/disables reporting of the PV Fail alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	f1(E)
Enabled (SP Deviation)	Enables/disables reporting of the SP Deviation alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	s1(E)
Gain	The proportional gain of the loop; the output has a component equal to the gain multiplied by the error.	Full analogue 0 = Loop is integral only	0	I	G
In Alarm	Indicates the specified alarm condition has been present for longer than its delay time, and the alarm is enabled. If both alarms are present the SP Deviation alarm will be reported. It will be set back to '0' immediately both the alarms are in the clear state (for longer than their alarm delay times).	0, 6, 7 0 = No alarm 6 = SP Deviation alarm 7 = PV Fail alarm	0	O	!
Integral Time (mins)	The time after which the integral part (an accumulation of gain times error) would be added to the output.	Full analogue 0 = no integral action	0	I	I
Integral	Integral part of output.	-100 to +100	0	O	i
Manual Level	The level to which loop output is driven if the Manual Select parameter is set to '1'.	Full analogue	0	I	L
Manual Select	Switches the loop into manual override.	0 or 1 0 = automatic 1 = Manual Override	0	I	A
Occupied Setpoint	Setpoint used when the 'Setpoint Select 'parameter is set to '1'.	Full analogue	0	I	O
Output	The value of the module's output.	0 to 100	0	O	D
Process Variable	The measured value being controlled.	Full analogue	0	I	P

Parameter	*Description*	*Range*	*Default*	*Type*	*Text comms*
PV Fail	Indicates the state of the PV Fail alarm. Clicking on PV Fail on the web page will display the PV Fail alarm sub-module. The way in which the sub-module declares alarms and clear alarms is described in **"Process Variable Fail Alarm" on page 370**.				
Reschedule Time	Interval between loop output calculations - see **"Reschedule Time" on page 385**	0 to 600s 0 = Loop disabled	0	I	R
Sensor Fail	The action taken when the loop detects a PV Fail alarm - see **"Process Variable Fail Alarm" on page 370**.	0, 1, 2, 3, 4 0 = Default PV 1 = Default OP 2 = Stay at Default 3 = Freeze OP 4 = No action	0	N	F
Setpoint Deviation	The maximum error allowed before a SP Deviation alarm is generated. *Note: The maximum error can be greater during non-occupation - see* **"Setpoint Deviation Alarm" on page 369**.	Full analogue	0	I	V
Setpoint Select	Specifies whether the module controls to the occupied or unoccupied setpoint. *Note: If set '0' and the with the unoccupied Setpoint set to 'nul' the loop is turned off.*	0 or 1 0 = Unoccupied setpoint 1 = Occupied setpoint	0	I	S
SP Deviation	Indicates the state of the SP Deviation alarm. Clicking on SP Deviation on the web page will display the SP Deviation alarm sub-module. The way in which the sub-module declares alarms and clear alarms is described in **"Setpoint Deviation Alarm" on page 369**.				
Time in Alarm (secs) (PV Fail)	The elapsed time in seconds since the PV Fail alarm state commenced.	+Full analogue	0	O	f1(T)
Time in Alarm (secs) (SP Deviation)	The elapsed time in seconds since the Setpoint Deviation alarm state commenced	+Full analogue	0	O	s1(T)
Unoccupied Setpoint	The setpoint used when the 'Setpoint Select' parameter is set to '0'. If set to 'null', it is never used.	Full analogue	0	I	U

Reschedule Time

The 'Reschedule Time' parameter defines the interval between loop output calculations. Increasing the reschedule time will have the effect of decreasing the frequency at which the output to the plant changes in response to sensor changes. This may be done to avoid 'hunting' of motorised valves.

If the integral time constant is non-zero, every time the module is processed an amount is added to the output which is a fraction of the level which would be added over the integral time, defined by the ratio of the reschedule time to the integral time.

For example:

if error = 2 deg C
gain = 20
reschedule time = 1 min
integral time = 10 mins

the amount added after 10 minutes would be:

change in integral term = Gain x Error = 20 x 2 = 40%

therefore

amount added in 1 min = (1/10) x 40 = 4%

i.e. in general amount added = R x G x E/I

This can be derived from the general equation:

Loop output = GE + $\int \frac{G}{I}(E.dt)$

Change in loop output due to integral = $\int \frac{G}{I}(E.dt)$

Using the example above for a fixed error of 2 deg C for 1 minute

Change in loop output due to integral = 20 x 2 x 1/10
Change in loop output = 4%

To achieve optimum control the Loop module's reschedule time should generally be less than or equal to the plant response time divided by 10. The plant response is normally several minutes, whereas loop reschedule time is in seconds (minimum 1s).

The sequence cycle time is also 1s, so if the reschedule time is set to its minimum of 1s, the Loop module's output will be recalculated every sequence table cycle.

If the reschedule time is 5s the Loop module will be recalculated every 5 cycles.

20.5 Loop Module Connectives

The following parameters are available as connectives:

Destination

21 NON-TREND DEVICE MODULES

Module Identifier	fn (where n is module number)
Default overview parameters	-
Size in brIQs	30

21.1 Functionality

Non-Trend Device (NTD) modules enable devices using communications protocols other than the Trend protocol (currently only BACnet) to be addressed using Trend LAN and device address. This allows IC communications with such devices. They are only available in /BAC variants and IQ4NCs.

21.2 NTD Module Parameters

Parameter	*Description*	*Range*	*Default*	*Type*	Text Comms
Address	Node address of this NTD on a BACnet LAN. Must be unique on the LAN.	0, 4 to 119 excluding 10	0	N	L
Attribute 2	Labels used with the 'Identifier' parameter to help identify the BACnet device. The seven attributes ('Identifier' plus 6 others) are used in IC communications to identify a controller or group of controllers with which this IQ4 wishes to communicate. The following characters & | * ? . are reserved for special functions Single letters should also be avoided (S, K, W etc). For further details see **"Attribute Selection - Wildcards" on page 285**.	0 to 30 character string except \ /	Blank	N	F
Attribute 3					G
Attribute 4					H
Attribute 5					I
Attribute 6					J
Attribute 7					K
Device instance	The BACnet address of the BACnet device on the BACnet system. It should be set up by preference, but if this is not specified (set to '0') both 'MAC address' and 'Network Number' parameters must be specified. It must be set up if the remote device is using automatic IP addressing - see **"BACnet with Automatic IP Addressing" on page 68**.	0 to 4194302	0	N	d
Identifier	A label which describes the BACnet device. It is also used as attribute 1 for IC communications.	0 to 30 character string except \ /	Blank	N	D
LAN	LAN number of this NTD's BACnet LAN. *Note: Must be a BACnet LAN in IQSET.*	1, 4 to 119 excluding 10	0	N	N
MAC address	The BACnet MAC address of the BACnet device on the BACnet network.	Dependent on network type	Blank	N	a
Network Number	BACnet network number of BACnet device.	0 to 65534	0	N	n
Type	Specifies the type of NTD module.	1 1 = BACnet	1	N	Y

21.3 NTD Module Connectives

None

22 NETWORK MODULES

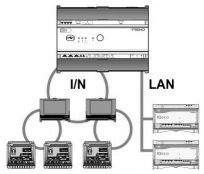
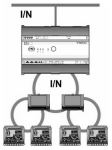
Module Identifier	nn (where second n is module number)
Default overview parameters	-
Size in brIQs	0

22.1 Functionality

Network modules are used to store the parameters specific to the type of physical networks being used. There are five types - see **"Network Module Types" on page 390**.

22.1.1 IQ4NC Communication Mode Selection

As explained in **"IQ4NC Communication Modes" on page 59** the IQ4NC can be configured to operate in one of four communication modes. The preferred method of configuring these modes is using the Network modules' overview web page. The required mode is selected by clicking on the required icon on the web page; a fifth icon enables manual configuration.

Icon	*Description*
	**Ethernet to MS/TP:** This is the default and configures the communication mode for communication between an internetwork on Ethernet and a LAN on MS/TP.
	**Trend Internetwork to MS/TP:** Configures the communication mode for communication between an internetwork on a Trend current loop network and a LAN on MS/TP.
	**Ethernet to Trend LAN:** Configures the communication mode for communication between an internetwork on Ethernet and LAN on a Trend current loop network.
	**Ethernet to Trend Internetwork:** Configures the communication mode for communication between an internetwork on Ethernet and internetwork on a Trend current loop network. In this mode the 'Virtual Local Lan' parameter is added. This is the Local LAN for the IQ4NC when Ethernet and Trend current loop are both internetworks.
	**Manual:** Enables any of the key parameters to be set up from the Network modules' summary page. Care must be taken when using this setting. Certain configurations are invalid and either cannot be selected, or will be ignored when the changes are committed.

Selecting any of the above configurations will set the Network modules' parameters as required for the selected configuration. These parameters are shown in the table on the Network modules' summary web page. The 'Trend Mode' parameter indicates which networks are Internetwork (IWRK), LAN, and not used for Trend Communications (NONE). Those cells not greyed out enable settings to be made from the summary page. The 'Trend Mode' parameter for each Network module can also be edited on the Network module web page but may be invalidated and not accepted by the IQ4NC.

Note: A change to 'Trend Mode' or 'Lan Number' parameters causes a general reset.

The state of each Network modules' 'Disable Module' parameter is also shown on the summary page. It can also be seen and edited in each Network module's page. If a Network module is disabled, that particular communication type is disabled.

Note: The IQ4NC communication mode configuration can also be set up using IQSET, but downloading a new network mode setting as part of the strategy file can cause the IQ4NC to lockout of its communications with IQSET, so the web page method is recommended.

22.1.1.1 Manually Configuring an IQ4NC Communication Mode

Manual configuration of the IQ4NC communications mode enables configurations not available directly from the summary page to be set up.

Care must be taken when setting up a configuration manually. Certain configurations are invalid and either cannot be selected, or will be ignored when the changes are committed.

22.2 Network Module Types

There are a number of Network module types specified using the 'Type' parameter:

Module Type	*Description*
Type 1 (IQ LAN)	Stores the parameters specific to the Trend current loop network.
Type 4 (Ethernet IP)	Stores the parameters specific to the Ethernet network.
Type 5 (BACnet IP)	Stores the parameters specific to the BACnet IP network.
Type 8 (Modbus Serial)	Stores the parameters specific to the Modbus Serial network.
Type 12 (BACnet MSTP)	Stores the parameters specific to the BACnet MS/TP network.
Type 13 (BACnet Application)	Stores the parameters specific to the BACnet network.
Type 14 (Wallbus)	Stores the parameters specific to the Wallbus network.
Type 15 (M-bus)	Stores the parameters specific to the M-bus network.
Type 16 (Trend Client)	Stores the parameters for the IQ4's USB Local Engineering Port and RS232 Local Supervisor Port.
Type 17 (XNC Serial)	Stores the parameters for the IQ4/XNC's Serial port.
Type 19 (Modbus IP)	Stores the parameters specific to the Modbus IP network

Note: The Network modules available are dependent on the IQ4 variant.

Network modules do not appear on the strategy page, and cannot be sequenced.

22.2.1 Type 1 (IQ LAN)

Type 1 (IQ LAN) Network modules are only available on /LAN variants or IQ4NCs. They are always Network module 2 (n2). They store the parameters specific to the Trend current loop network.

22.2.1.1 Baud Rate

The baud rate of communication on over the Trend current loop network can be automatically detected or set manually. It is recommended that the baud rate be automatically detected ('Autobaud' parameter set to '1'), but in the following circumstances it may preferable to set the baud rate manually:

- If the LAN consists of all IQ4s, then one IQ4 should have its baud rate set to the required rate manually.
- If the IQ4 has problems detecting the baud rate, indicated by the LAN OK indicator flashing for longer than 30s.

22.2.1.2 Trend Mode

The 'Trend Mode' parameter specifies the type of Trend communication that can be used over the Trend current loop network.

IQ4/LAN Variants

The Trend current loop network can only operate as a Trend LAN, therefore the 'Trend Mode' parameter is always set to '1' (LAN) and cannot be changed.

IQ4NCs

The Trend current loop network can operate as a LAN, or internetwork therefore the 'Trend Mode' parameter can be set to '1' (LAN), '2' (internetwork) or '0' (none). When set to '1' the Trend current loop network connected to the IQ4NC is a LAN, when set to '2' the Trend current loop connected to the IQ4NC is a Trend internetwork.

Setting the 'Trend Mode' parameter to '0' will prevent all Trend communications over the Trend current loop.

22.2.1.3 Module Disable

Disabling the module will prevent ALL communication over the Trend current loop network.

Note: If the Trend current loop network is enabled an IQ4NC is still able to be part of an internetwork over Ethernet, but /LAN variants cannot be part of a LAN or internetwork on Ethernet. However access to the IQ4's web pages is possible.

22.2.1.4 Alarm Handling

The module generates network alarms for the Trend current loop network.

IQ4/LAN Variants

Alarms are sent to Alarm Group module 0 which means that the alarms are logged, but not sent.

IQ4NCs

Alarms can be sent to any Alarm Group module allowing them to be sent as specified by the alarm handling strategy configured in the IQ4. IQ4NC controllers can also be used for INC alarm forwarding by the IQ4NC - see **"Alarm forwarding from a Current Loop LAN (e.g. EINC Replacement)" on page 93**.

22.2.1.5 Type 1 (IQ LAN) Network Module Parameters

Type 1 (IQ LAN) Network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm Group	The Alarm Group module to which all Trend current loop network alarms are sent - **"Alarm Handling" on page 391**. *Note: For all IQ4s except IQ4NC it is fixed at '0'.*	0 to max 0 = Logged but not sent	0	N	g
Autobaud	Enables/disables automatic baud rate detection.	0 or 1 1 = Autobaud 0 = Manual Baud	1	N	A
Baud Rate	The baud rate on the Trend current loop. If the 'Autobaud' parameter is set to '1', the baud rate is set automatically. If the 'Autobaud' parameter is set to '0' it must be set up manually. All devices on Trend current loop network must have same baud rate.	0, 1, 2, 3, 4 0 = 1k2 1 = 4k8 2 = 9k6 3 = 19k2 4 = 38k4	Set by auto baud	N	b
Disable Module	Enables/disables the Trend current loop network.	0 or 1 0 = Enabled 1 = Disabled	0	N	~
LAN Label	Label for the Trend current loop LAN.	0 to 30 character string except \ /	null	N	k
Lan Number	/LAN variants: Specifies the IQ4's LAN number. It cannot be changed from this module, it can only be changed in the Address module. IQ4NC: Specifies the LAN number of the LAN operating over the Trend current loop network if 'Trend Mode' parameter is '1'. If the 'Trend Mode' parameter is '2' or '0' the LAN number does not apply. It is stored in the Address module and copied into the Network module. If required it can be changed from the Network module's web page. It must be unique on the site.	1, 4 to 119 excluding 10	-	N	N

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Trend Mode	Specifies the type of Trend communication that can be used over the Trend current loop network. /LAN variants: it will always be '1' and is greyed out so cannot be changed. IQ4NC: it can be '2', '1', or '0'. '0' means there are no Trend Communications over the Trend current loop network. See **"Trend Mode" on page 390** for more details. *Note: The 'Trend Mode' parameter is set up from the Network modules' overview web page. The Network module's web page allows the change to be made, but will be invalidated by the IQ4.*	0, 1, 2 0 = None 1 = Lan 2 = Internetwork	1	N	J

Note: The 'Disable Module' and 'Baud Rate' parameters can be prevented from being included in the strategy file for downloading. This enables a strategy to be used by several controllers - **"Parameter Download Exclusions" on page 136**.

Note: Changes to the 'Trend Mode' or 'Lan Number' parameters cause a general reset.

22.2.2 Type 4 (Ethernet IP)

Type 4 (Ethernet IP) network modules are available on all IQ4s, they are always Network module 1 (n1). They store the parameters specific to the Ethernet network.

22.2.2.1 Module Disable

The module cannot be disabled.

22.2.2.2 Trend Mode

The 'Trend Mode' parameter specifies the type of Trend communication that can be used over the Ethernet network.

IQ4s
The Ethernet network can only operate as a LAN, therefore the 'Trend Mode' parameter is always set to '1' (LAN) and cannot be changed.

Note: For /LAN variants it is set to '0' (None) as the Type 1 (IQ LAN) Network module will be set to '1' (LAN), and only one Network module can be set to '1' (LAN).

IQ4NCs
The Ethernet network can operate as a LAN, or internetwork therefore the 'Trend Mode' parameter can be set to '1' (LAN), '2' (internetwork) or '0' (none). When set to '1' the Ethernet network connected to the IQ4 is a LAN, when set to '2' the Ethernet network connected to the IQ4 is a Trend internetwork.

Setting the 'Trend Mode' parameter to 'None' will prevent all Trend communications over the Ethernet network. However it is possible to access the IQ4's web pages, and to use BACnet protocol over the Ethernet network where applicable i.e. /BAC variants.

22.2.2.3 Alarm Handling

The module generates IP network alarms for the Ethernet network.

IQ4s
Alarms are sent to Alarm Group module 0 which means they are logged but not sent.

/LAN variants

/LAN variants do not generate Ethernet alarms unless the Type 1 (IQ LAN) network module is disabled in which case they are processed as described above.

IQ4NCs

The alarms can be sent to any Alarm Group module allowing them to be sent as specified by the its alarm handling strategy. IQ4NCs can also be used for INC alarm forwarding - see **"Alarm forwarding from a Current Loop LAN (e.g. EINC Replacement)" on page 93**.

22.2.2.4 IP Addressing Mode

The IP addressing mode, specified by the 'IP Addressing Mode' parameter, determines how the IP addressing details ('IP Address', 'Subnet Mask', 'Router 1', 'WINS server', and 'DNS Server' parameters) are set up.

If the 'IP Addressing Mode' parameter is set to '1' (Obtain Automatically) the parameters will be set up by a DHCP server. If there is no DHCP server the IQ4 will go into link local mode - see **"Link/Local Operation" on page 51**. If the 'IP Addressing Mode' parameter is set to '2' (Enter manually) the parameters must be manually specified.

For more details about IP Addressing Mode see **"IP Addressing" on page 50**.

22.2.2.5 Type 4 (Ethernet IP) Network Module Parameters

Type 4 (Ethernet IP) network modules include the 'Type' parameter common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

<table>
<tr><th>Parameter</th><th>Description</th><th>Range</th><th>Default</th><th>Type</th><th>Text Comms</th></tr>
<tr><td>Alarm Group</td><td>The Alarm Group module to which all Ethernet IP network alarms are sent - "Alarm Handling" on page 392.

Note: For all IQ4s except IQ4NC it is fixed at '0'.</td><td>0 to max
0 = Logged but not sent</td><td>0</td><td>N</td><td>g</td></tr>
<tr><td>DNS Servers 1 to 4</td><td>IP address(es) of DNS server(s). One must be set up if using host names across routers. If the 'IP Addressing Mode' parameter is set to '1' it will be set up automatically.</td><td>Valid IP address</td><td>All four set to 0.0.0.0</td><td>N</td><td>N</td></tr>
<tr><td>HTTPS Certificate Name</td><td>v3.30 or greater. The name of the IQ4's security certificate. Defined by the certificate's common name.</td><td>0 to 253 character string.</td><td>No certificate</td><td>N</td><td>c</td></tr>
<tr><td>HTTPS Certificate Expiry</td><td>v3.30 or greater. The date and time of the IQ4's security certificate will expire.</td><td>dd/mm/yyyy hh:mm:ss</td><td>No certificate</td><td>N</td><td>t</td></tr>
<tr><td>Email Password</td><td>The password required by the email server that the IQ4 is to use to send email. Only set up if required by server.</td><td>0 to 50 characters</td><td>null</td><td>N</td><td>A</td></tr>
<tr><td>Email Port</td><td>v3.30 or greater only. The TCP port used by the email server. Set automatically depending on the 'Email Security' parameter but can be changed.

<table><tr><th>'Email Security' Setting</th><th>Email Port</th></tr><tr><td>0 (No security)</td><td>25</td></tr><tr><td>1 (SSL)</td><td>465</td></tr><tr><td>2 (STARTTLS)</td><td>587</td></tr><tr><td>3 (Auto Secure)</td><td>0</td></tr></table></td><td>0 to 65535</td><td>25</td><td>N</td><td>O</td></tr>
</table>

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Email Security	v3.30 or greater only. The security protocol used by the email server.	0, 1, 2, 3 0 = No security 1 = SSL 2 = STARTTLS 3 = Auto Secure	3	N	q
Email Server Address	The IP address, internet domain name, or host name of the SMTP (Simple Mail Transfer Protocol) server to be used for sending email type alarms. It must be set up if using an email type alarm destination module. If using an internet domain name as the email server address, a DNS server must be set up.	IP address or hostname (1 to 15 characters) or internet domain name	Null	N	L
Email Min TLS version	The minimum version for outgoing connections to SMTP servers.	0, 1, 2, 3 0 = TLS v1.0 1 = TLS v1.1 2 = TLS v1.2 3 = TLS v1.3	2	N	-
Email Username	The user name required by the email server that the IQ4 is to use to send email. Only set up if required by server.	Up to 63 characters	Null	N	o
Ethernet MAC	The IQ4's Media Access Control (MAC) address; a globally unique address programmed into the Ethernet device within the IQ4.	Valid MAC address (read only)	Factory set	N	M
Hostname	The IQ4's host name. It must start with 'a' to 'z' or 'A' to 'Z' and cannot end with '-' or '_'.	0 to 15 character string.	'Trend' plus the last six digits of the MAC Address e.g. TREND_00_14_D0	N	$
Ignore Email Certificate	v3.30 or greater only. Causes IQ4 to ignore the validity of the SSL security certificate. It should normally be disabled, but can be enabled if the IQ4 has a problem authenticating the certificate. *Note: the connection will remain encrypted, but there is no way for the controller to prove it is communicating with the correct server.*	0 or 1 0 = Disabled 1 = Enabled	0	N	j
IP Addressing Mode	Specifies whether IP addressing settings are set up automatically or manually - see **"Trend Mode" on page 392**	1, 2 1 = Obtain Automatically 2 = Enter manually	1	N	P

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
IP Address	IP address of IQ4. If the 'IP Addressing Mode' parameter is set to '1' it will be set up automatically. ▪ 127.x.x.x, reserved for loop back. ▪ The non masked part of the address cannot be all 1's or 0's e.g. if address is 1.2.x.x and subnet mask is 255.255.0.0 then x.x. cannot be either 255.255 or 0.0. ▪ Addresses 224.0.0.0 to 239.255.255.255, reserved for multicast. ▪ Addresses 240.0.0.0 to 247.255.255.255, reserved for experimentation. ▪ Further addresses may not be suitable for a particular system; check with the IT department responsible for the network.	Valid IP address	Manual 128.1.1.3 Link/local start at 169.254.0.0	N	I
LAN Label	User-friendly label for the LAN.	0 to 30 character string except \ /	-	N	k
Lan Number	IQ4s: Specifies the IQ4's LAN number. It cannot be changed from this module, it can only be changed in the Address module. IQ4NCs: Specifies the LAN number of the LAN operating over the Ethernet network if the 'Trend Mode' parameter is '1'. If the 'Trend Mode' parameter is '2' or '0' the LAN number does not apply. It is stored in the Address module and copied into the Network module. If required it can be changed from the Network module's web page. It must be unique on the site.	1, 4 to 119 excluding 10	-	N	N
HTTPS Redirect	v3.30 or greater only. Redirects any web server request made to the HTTP web server to the HTTPS web server.	0 or 1 1 = redirect	0	N	n
Remote Device 1 to 20	The IP address or host name of the devices the other side of a router with which the IQ4 is to construct a Trend network on Ethernet - see **"Building an Internetwork on Ethernet Across Routers" on page 41**. There can be up to 20 different devices specified by the 'Remote Device' parameters 1 to 20. *Note: Host names must be used if the 'IP Addressing Mode' parameter is set to '1'.*	Valid IP address	All 20 set to 0.0.0.0	N	b
Remote Device Subnet 1 to 20	The subnet masks of the remote devices whose IP addresses are specified by the 'Remote Device' 1 to 20 parameters.	Valid subnet	All 20 set to 0.0.0.0	N	s
Router 1	The IP address of router (sometimes referred to as the default router) to which messages are sent if IP address is not on the local subnet. It should be set to the IP address of a router on the same subnet as the IQ4. If automatic addressing is being used it will be set up automatically.	Valid IP Address	255.255.255.255	N	R

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Secure Access Port	IQ4 v3.60 or greater. The port number used by the IQ4 for the SSH communications required for secure vCNC connections. It should only be changed if the default is not suitable, i.e. already being used on the network for something else.	1 to 65535	22	N	T
Secure Web Server Port Number	v3.30 or greater only. The port number used by the IQ4 for its HTTPS web server. *Note: When HTTPS is first enabled the IQ4 will automatically generate a self signed certificate, however this is not suitable for long term use, and a full trusted certificate should be created as soon as possible - see* **"Configure SSL Certificates (if required)" on page 136**.	0 to 65535 0 = HTTPS web server OFF	0	N	e
Standalone Mode	Enables/disables stand alone mode - see **"Stand Alone IQ4" on page 42**.	0 or 1 0 = normal 1 = stand alone mode	0	N	a
Subnet Mask	The mask used on IP address to produce subnet. If the 'IP Addressing Mode' parameter is set to '1' it will be set up automatically. Trend devices must be on the same subnet in order to construct a LAN or internetwork. *Note: Some subnet masks may not be suitable for a particular system; check with the IT department responsible for the network.*	Valid subnet mask	255.255.0.0	N	S
Trend Mode	Specifies the type of Trend communication that can be used over the Ethernet network. IQ4s: Always '1' and is greyed out so cannot be changed. /LAN variants: Can be set to '1' providing the Type 1 (IQ4 LAN) Network module is disabled. IQ4NCs: Can be '2', '1', or '0'. '0' means there are no Trend Communications over the Ethernet network. See **"Trend Mode" on page 390** for details. *Note: The 'Trend Mode' parameter is set up from the Network modules' overview web page. The Network module web page allows the change to be made, but will be invalidated by the IQ4.*	0, 1, 2 0: None 1: Lan 2: Internetwork	1	N	J
UDP Port	The User Datagram Protocol (UDP) port defines the IP port used by the IQ4. It is used to build internetworks and LANs with other Trend Ethernet network devices. All devices on the same system must use the same UDP port. Devices can be set up with different UDP ports to create completely separate systems on the same subnet.	0 to 65535	57612	N	E

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Web Server Port Number	The Ethernet port number used by the IQ4 for its HTTP web server.	0 to 65535 0 = HTTP web server OFF	80	N	w
WINS Servers 1 to 4	IP address(es) of WINS server(s). One must be set up if using host names across routers. If 'Obtain automatically' it will be set up automatically.	Valid IP address	All four set to 0.0.0.0	N	W
XML Web Services	Specifies whether the IQ4's XML Web Services is enabled and the type of security used. For basic IQ authentication the Application key (obtained from an engineered Type 0 User Module) must be included in XML write messages. For write authentication security is the same as for basic IQ authentication, but the Application key is sent in encrypted form. For read/write authentication security is the same as for basic IQ authentication, but the Application key is sent in encrypted form and is required for both read and write messages. The XML Web Services are described **"Alarm Handling" on page 391** and **"XML Syntax" on page 527**.	0, 1, 2, 3 0 = XML interface off 1 = Basic IQ authentication 2 = Write authentication 3 = Read/Write authentication	0	N	u

Note: The 'Addressing Mode', 'Subnet Mask', 'Default Router', 'UDP Port', 'Remote Device' 1 to 20 and 'Remote Device Subnet' 1 to 20, 'Hostname', 'WINS Server 1 to 4', 'DNS Server 1 to 4', 'Disable Module' and 'Baud Rate' parameters can be prevented from being included in the strategy file for downloading. This enables a strategy to be used by several controllers - **"Parameter Download Exclusions" on page 136**.

Note: A change to the 'IP Address', 'Subnet Mask', 'UDP', 'Router 1', 'Trend Mode' or 'Lan Number' parameters causes a general reset.

22.2.3 Type 5 (BACnet IP)

Type 5 (BACnet IP) Network modules are only available on /BAC variants or IQ4NC controllers, they are always Network module 3 (n3). They store the parameters specific to the BACnet IP network.

22.2.3.1 Module Disable

Disabling the module will prevent ALL communication over the BACnet IP network.

22.2.3.2 Trend Mode

Type 5 (BACnet IP) Network modules do not support the Trend Mode as Trend communication over BACnet IP is not supported, communications use BACnet protocol only.

22.2.3.3 Alarm Handling

Type 5 (BACnet IP) Network modules do not generate any Trend network alarms as Trend communication is not supported over BACnet IP.

22.2.3.4 Type 5 (BACnet IP) Network Module Parameters

Type 5 (BACnet IP) network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
BBMD	Specifies whether or not the IQ4 acts as a BACnet Broadcast Management Device (BBMD). There must be one BBMD on each BACnet IP segment if broadcast messages are to cross routers. By default the IQ4 does not act as a BBMD but it can if the BACnet segment does not have one. If BBMD is enabled the BDT Remote IP Address, BDT Remote Subnet and BDT UDP Port parameters must be set up. *Note: The BBMD functionality will not work with devices which use automatic IP addressing, and under such conditions should left disabled - see* **"BACnet with Automatic IP Addressing" on page 68**.	0 or 1 0 = Disabled 1 = Enabled	0	N	e
BDT Remote IP Address	List of up to 32 IP addresses. Each item in list is the IP address of a BBMD device the other side of a router which is able to forward broadcast messages.	Valid IP address	0.0.0.0	N	bn(B)b
BDT Remote Subnet	List of up to 32 subnet masks corresponding to the IP addresses of the devices specified in the 'BDT Remote IP Address' parameter. Should be set to 255.255.255.255.	Valid subnet mask	0.0.0.0	N	bn(S)
BDT Remote UDP	List of up to 32 UDP ports corresponding to the devices specified in the 'BDT Remote IP Address' parameter.	0 to 65535	47808	N	bn(E)
Disable Module	Enables/disables the BACnet IP network.	0 or 1 0 = Enabled 1 = Disabled	0	N	~
Network Number	The IQ4's BACnet network number. The default should be satisfactory, but if the network number is already in use it should be changed.	1 to 65534	1	N	n
UDP Port	The UDP port used to send messages to other BACnet devices. All BACnet devices on the same BACnet segment must use the same UDP port. The default is normally acceptable but if two sets of BACnet devices are required to operate on the same subnet it should be changed.	0 to 65535	47808	N	E

Note: 'BDT Remote IP Address', 'BDT Remote Subnet' and 'BDT Remote UDP' parameters do not appear on the web page, and must be set up using IQSET.

22.2.4 Type 8 (Modbus Serial)

Type 8 (Modbus Serial) Network modules are only available on v4.20 firmware or greater /INT variants. They store the parameters specific to the Modbus serial connection.

22.2.4.1 Trend Mode

Type 8 (Modbus Serial) Network modules do not support the Trend Mode as Trend communication over Modbus is not supported.

22.2.4.2 Module Disable

The module cannot be disabled.

22.2.4.3 Alarm Handling

Type 8 (Modbus Serial) Network modules do not generate any alarms.

22.2.4.4 Type 8 (Modbus Serial) Network Module Parameters

Type 8 (Modbus Serial) network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Baud Rate	The baud rate of the Modbus.	0, 1, 2, 3, 4, 5, 6, 7 0 = 1200 1 = 2400 2 = 4800 3 = 9600 4 = 19200 5 = 38400 6 = 57600 7 = 115200	4	N	b
Communication Channel Status	Read only – Indicates the status of the selected port Unavailable – the channel is in use by another Resource Available – the channel is available In Use – the channel is being used by this Network Module.	0, 1, 2 0 = Unavailable 1 = Available 2 = In Use	0	N	t
Communication Channel	The port used for the Modbus communications.	0, 1, 2 0 = Not Used 1 = RS232 2 = RS485	0	N	v
Interframe Pause	The pause in ms between frames sent by the controller on the Modbus Network.	500	0	N	i
Modbus Communication Mode	The communication mode used by the Modbus connection. It should be set to the value required by the other devices (slave device(s)) on the Modbus RTU and ASCII slaves should not be mixed on the same network port.	0, 1 0 = Modbus RTU 1 = Modbus ASCII	1	N	m
Parity	The parity required for the Modbus.	0, 1, 2 0- None 1 - Odd 2 - Even	2	N	P
Re-tries	The number of times to attempt a Modbus request before a communication error is reported.	0, 1, 2, 3	3	N	r
Timeout	The maximum wait time in ms for a response frame from a Modbus request.	0 to 65535	50	N	T

22.2.5 Type 12 (BACnet MSTP)

Type 12 (BACnet MSTP) Network modules are only available on IQ4NC controllers, they are always Network module 5 (n5). They store the parameters specific to the BACnet MS/TP network.

22.2.5.1 Trend Mode

The 'Trend Mode' parameter specifies the type of Trend communication that can be used over the BACnet MS/TP network.

Type 12 (BACnet MSTP) Network modules can only operate as a Trend LAN, therefore the 'Trend Mode' parameter is set to '1' (LAN) by default.

Setting the 'Trend Mode' parameter to '0' (None) prevents all Trend communications over the BACnet MS/TP network.

22.2.5.2 Alarm Handling

The module generates network alarms for the BACnet MS/TP network and sends them to the Alarm Group module specified by the 'Alarm Group' parameter.

Alarms can be sent to any Alarm Group module allowing them to be sent as specified by the alarm handling strategy configured in the controller. IQ4NC controllers can also be used for INC alarm forwarding - see **"Alarm forwarding from a Current Loop LAN (e.g. EINC Replacement)" on page 93**.

22.2.5.3 Module Disable

Disabling the module will prevent ALL communication over the BACnet MS/TP network.

22.2.5.4 Type 12 (BACnet MSTP) Network Module Parameters

Type 12 (BACnet MSTP) network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm Group	The Alarm Group module to which all MS/TP network alarms are sent - see **"Alarm Handling Strategy" on page 79**.	0 to max 0 = Not sent	0	N	g
Baud Rate	The baud rate on the MS/TP trunk. Set to match other devices on the trunk. 76k8 recommended.	0, 1, 2, 3, 4 0 = Disabled 1 = 9k6 2 = 19k2 3 = 38k4 4 = 76k8	4	N	b
Disable Module	Enables/disables the MS/TP network.	0 or 1 0 = Enabled 1 = Disabled	0	N	~
LAN Label	User-friendly label for the LAN.	0 to 30 character string except \ /	-	N	k
Lan Number	Specifies the LAN number of the LAN operating over the MS/TP network if the 'Trend Mode' parameter is '1'. It is stored in the Address module and copied into the Network module. It must be unique on the site.	1, 4 to 119 excluding 10	9	N	N
MAC Address	The IQ4's address on the MS/TP network - set to default unless the MAC address must be set up manually.	11 to 119	0	N	M
Manual MAC address	Enables the MAC address to be set manually.	0 or 1 0 = Automatic 1 = Manual	0	N	L

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Manual Network Number	Specifies whether the network number is calculated automatically or set up manually.	0 or 1 0 = Automatic 1 = Manual	0	N	l
Max Info Frames	The maximum number of frames the IQ4 can send on the MS/TP trunk per rotation of the MS/TP token. Do not change unless instructed by Technical Support.	0 to 20	20	N	f
Max Master	Maximum address polled to create network on MS/TP trunk. Do not change unless instructed by Technical Support.	0 to 127	127	N	O
Network Number	The BACnet network number for the MS/TP trunk. Set to be same as the 'Lan Number' parameter unless the 'Manual Network Number' parameter is set to '1'.	1 to 65534	LAN number	N	n
Service Class	Specifies whether messages on the MS/TP network require a low level acknowledgement.	0, 1, 2 0 = Unconfirmed (no acknowledgement required) 1 = Not Used (treated the same as '0' above) 2 = Confirmed (acknowledgement required)	0	N	C
Trend Mode	The type of Trend communication that can be used over the BACnet MS/TP network. See **"Trend Mode" on page 400** for more details. *Note: The Trend Mode is set up from the Network modules' overview web page. The Network module web page allows the change to be made, but will be invalidated by the IQ4.*	0 or 1 0: None 1: Lan	1	N	J

Note: A change to the 'Trend Mode' or 'LAN Number' parameters causes a general reset.

22.2.6 Type 13 (BACnet Application)

Type 13 (BACnet Application) Network modules are only available on /BAC variants and IQ4NC controllers, they are always Network module 6 (n6). They store the parameters specific to the BACnet IP network communications.

22.2.6.1 Object Name Prefixes

The IQ4 offers the option of adding the object name prefix to the BACnet object name property (BP_OBJECT_NAME), or to not include it by enabling or disabling 'Object Name Prefixes' parameter.

When enabled the BP_OBJECT_NAME would consist of the item reference prefix and the item label, and if disabled, the object name just consists of the label. E.g for Knob 1 (equivalent to BACnet analogue value object), if the label is 'Occ Setpoint',

if enabled, object name = 'K1-Occ Setpoint'
if disabled, object name = 'Occ Setpoint'

By default the object name prefixes are disabled, but if an earlier firmware version is upgraded to v2.3 or greater the prefixes will be enabled by default.

22.2.6.2 BTL Compliance

The IQ4's certification only applies when the strategy is configured correctly - see **"Creating a BTL Compliant Strategy" on page 117**.

22.2.6.3 BACnet Backup/Restore

IQ4's BACnet backup and restore complies with BIBB DM-BR-B,DM-RD-B. It enables a client to backup or restore an IQ4 over BACnet using standard BACnet commands.

BACnet is an insecure protocol, therefore in the interests of security Trend has disabled certain BACnet services by default, these can be enabled if required - see **"BACnet Communications" on page 63**.

22.2.6.4 Cold & Warm Restart Service

The Cold & Warm Restart Service complies with BIBB DM-BR-B,DM-RD-B.

WARMSTART command: This command archives the strategy to save the latest changes and then deletes and restores the strategy without resetting the controller. All plot values are lost, as the strategy is restored to the same state as when it was loaded for the first time.

COLDSTART command: This command archives the Strategy to save the latest changes and then resets the processor as if power has just been applied.

To use the WARMSTART and COLDSTART commands the BACnet Services Password must be supplied.

BACnet is an insecure protocol, therefore in the interests of security Trend has disabled certain BACnet services by default, these can be enabled if required - see **"BACnet Communications" on page 63**.

22.2.6.5 Writes over BACnet

See **"Writes over BACnet" on page 68**.

22.2.6.6 Create/Delete Service

See **"Create/Delete Service" on page 68**.

22.2.6.7 Module Disable

The module cannot be disabled.

22.2.6.8 Trend Mode

Type 13 (BACnet Application) Network modules do not support the Trend Mode as Trend communication over BACnet IP is not supported, communications use BACnet protocol only.

22.2.6.9 Alarm Handling

Type 13 (BACnet Application) Network modules do not generate any alarms.

22.2.6.10 Type 13 (BACnet Application) Network Module Parameters

Type 13 (BACnet Application) Network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Backup Failure Timeout	v3.50 or greater. The time – in seconds – that the IQ4 must wait before unilaterally ending the backup or restore procedure.	0 to 65535	30	N	f

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Backup_ and_ Restore_ state	v3.50 or greater. The current status of the BACnet backup/restore.	0, 1, 2, 3, 4, 5, 6 0 = Idle 1 = preparing for backup 2 = preparing for restore 3 = performing a backup 4 = performing a restore 5 = backup failure 6 = restore failure	0	N	t
BACnet Backup Preparation Time	v3.50 or greater. The amount of time in seconds that the IQ4 might remain unresponsive after the response to a start backup procedure has been sent.	0 to 65535	0	N	u
BACnet Backup/ Restore	v3.50 or greater. Enables/Disables the IQ4's BACnet Backup and Restore service.	0 or 1 1 = Enabled 0 = Disabled	0	N	b
BACnet Services Password	v3.50 or greater. The password required to authorise use of the IQ4's BACnet services.	8 to 20 character string	-	N	k
BACnet Writable Properties	v3.50 or greater. Enables/Disables the IQ4's BACnet Writable Properties service..	0 or 1 1 = Enabled 0 = Disabled	0	N	w
BO State Count And Active Time Properties	v3.50 or greater. Specifies whether the Change_Of_ State_Count and Elapsed_Active_Time properties on the BACnet AO & BO Objects are enabled. Must be disabled if a BTL compliant strategy is required.	0 or 1 0 = Enabled 1 = Disabled	0	N	c
BTL Compliance	Not in v3.50 or greater. For earlier versions specifies whether the controller's BTL compliance is enabled. When enabled the controller's schedule offset modules are read only over BACnet. Changes are still possible when access is over the Trend network. In order for the controller to be BTL compliant this option must be enabled i.e. Schedule Offset modules read only over BACnet.	0 or 1 1 = Enabled 0 = Disabled	1	N	c
Cold & Warm Restart Service	v3.50 or greater. Enables/Disables the IQ4's BACnet Cold & Warm Restart Service.	0 or 1 1 = Enabled 0 = Disabled	0	N	r
Create/ Delete Service	v3.50 or greater. Enables/Disables the IQ4's BACnet Object Create/Delete service. *Note: Only BACnet Calendar Objects may be created/deleted using this service. All other BACnet objects must be configured using IQSET.*	0 or 1 1 = Enabled 0 = Disabled	0	N	C

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Device instance	The IQ4's address on the BACnet network. It is unique on the BACnet network - see **"BACnet with Automatic IP Addressing" on page 68**. The automatically calculated value should normally be acceptable, but if that device instance is already in use on the BACnet system it will be necessary to set the value manually.	1 to 4194303	(LAN number x 1000) + Local address	N	i
Last_ Restore_ Time	v3.50 or greater. The last time the strategy was restored using a BACnet restore.	00:00:00 to 23:59:59	00:00:00	N	m
Manual device instance	Specifies whether the 'Device instance' parameter is calculated automatically or set up manually.	0 or 1 0 = Automatic 1 = Manual	0	N	A
Object Name Prefixes	Enables/disables item type prefix from BACnet object names. If enabled prefix is added to label - see **"Object Name Prefixes" on page 401**.	0 or 1 0 = Disabled 1 = Enabled	1	N	j
Send I-Am at Startup	Specifies whether the IQ4 will send an 'I am' message at startup to notify other BACnet devices of its presence on the BACnet network. If set to '1' the IQ4 will inform supervisors and tools of its presence on power up (e.g. IQSET would add it to its System View of the BACnet network). If set to '0' the IQ4 will not send the message and the IQ4 will only be discovered by software tools by polling.	0 or 1 0 = Message not sent 1 = Message sent	0	N	S

22.2.7 Type 14 (Wallbus)

Type 14 (Wallbus) Network modules are only available on v3.30 firmware or greater. They are always Network module 12 (n12). They store the parameters specific to the Wallbus network.

22.2.7.1 Trend Mode

Type 14 (Wallbus) Network modules do not support the Trend Mode as Trend communication over Wallbus is not supported.

22.2.7.2 Module Disable

Disabling the module will prevent ALL communication over the Wallbus network.

22.2.7.3 Alarm Handling

Type 14 (Wallbus) Network modules do not generate any alarms.

22.2.7.4 Type 14 (Wallbus) Network Module Parameters

Type 14 (Wallbus) Network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Communication Channel Status	The status of the selected communication channel. Unavailable – the channel is in use by another Resource Available – the channel is available In Use – the channel is being used by this Network Module.	0, 1, 2 0 = Unavailable 1 = Available 2 = In Use	0	N	t
Communication Channel	The communication channel used by the Wallbus network, always '4'.	4	4	N	v
Disable Module	Enables/disables the Wallbus network.	0 or 1 0 = Enabled 1 = Disabled	0	N	~
Local Address	Read only . The address of the IQ4 on the Wallbus network.	1	1	N	a

22.2.8 Type 15 (M-bus)

Type 15 (M-bus) Network modules are only available on /INT variants only v4.30 or greater, they store the parameters specific to the M-bus connection.

22.2.8.1 Trend Mode

Type 15 (M-bus) Network modules do not support the Trend Mode as Trend communication over M-bus is not supported.

22.2.8.2 Module Disable

Disabling the module will prevent ALL communication over the M-bus network.

22.2.8.3 Alarm Handling

Type 15 (M-bus) Network modules do not generate any alarms.

22.2.8.4 Type 15 (M-bus) Network Module Parameters

Type 15 (M-bus) Network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Baud Rate	The baud rate of the M-bus.	0, 1, 2, 3, 4, 5, 6, 7 0 = 300 1 = 600 2 = 1200 3 = 2400 4 = 4800 5 = 9600 6 = 19200 7 = 38400	0	N	b
Communication Channel Status	The status of the selected communication channel. Unavailable – the channel is in use by another Resource Available – the channel is available In Use – the channel is being used by this Network Module.	0, 1, 2 0 = Unavailable 1 = Available 2 = In Use .	0	N	t
Communication Channel	The port used for the M-bus communications.	0, 1, 2 0 = Not Used 1 = RS232 2 = RS485	0	N	v
Disable Module	Enables/disables the Wallbus network.	0 or 1 0 = Enabled 1 = Disabled	0	N	~
Interframe Pause	The pause in ms between frames sent by the controller on the M-bus network.	0 to 500	100	N	i
Retries	The number of times to attempt a M-bus request before a communication error is reported.	0, 1, 2, 3	3	N	r
Timeout	The maximum wait time in seconds for a response frame from a M-bus request.	0 to 65535	10	N	T

22.2.9 Type 16 (Trend Client)

Type 16 (Trend Client) Network modules are only available on v3.30 firmware or greater.. They are used by the IQ4 to communicate with its USB Local Engineering Port and RS232 Local Supervisor Port, They are always Network modules 10 and 11 (n10 and n11). n10 is used for the USB Local Engineering Port and n11 is used for the RS232 Local Supervisor Port.

Because they are used by the IQ4 they are created as required and cannot be edited. Both n10 and n11 are present in all variants of IQ4, however n11 is not present in /XNC variants but will be created if the RS232 Local Supervisor Port is to be used as a supervisor port. If the port is to be used by the XNC to interface with a 3rd party system a Type 17 (XNC Serial) Network module should used.

22.2.9.1 Trend Mode

Not supported.

22.2.9.2 Alarm Handling

Type 16 (Trend Display) Network modules do not generate any alarms.

22.2.9.3 Module Disable

The module cannot be disabled.

22.2.9.4 Type 16 (Trend Client) Network Module Parameters

Type 16 (Trend Client) Network modules include the ‘Label’ and ‘Type’ parameters common to all Network modules - see **“Network Module Common Parameters” on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Communication Channel	The communication channel used by the module.	0 or 6 1 = Serial 6 = USB	n11=1 n10=6	N	v
Communication Channel Status	The status of the selected communication channel.	0, 1, 2 0 = Unavailable, the channel is in use by another Resource 1 = Available, the channel is available 2 = In Use – the channel is being used by this module.	0	N	t

22.2.10 Type 17 (XNC Serial)

Type 17 (XNC Serial) Network modules are only available on /XNC variants. They store the parameters specific to the Serial port used by the interface.

22.2.10.1 Trend Mode

Not supported

22.2.10.2 Alarm Handling

Type 17 (XNC Serial) Network modules do not generate any alarms.

22.2.10.3 Module Disable

The module cannot be disabled.

22.2.10.4 Type 17 (XNC Serial) Network Module Parameters

Type 17 (XNC Serial) Network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Communication Channel	The Communication Channel used by the module.	1 or 3 1 = RS232 3 = RS485	1	N	v
Communication Channel Mode	The type of serial interface the port is to use. It is automatically set to match the hardware of the port to which it is connected.	RS232 or RS485	-	N	m
Communication Channel Status	The status of the selected communication channel.	0, 1, 2 0 = Unavailable, the channel is in use by another Resource 1 = Available, the channel is available 2 = In Use – the channel is being used by this module.	0	N	t

22.2.11 Type 19 (Modbus IP)

Type 19 (Modbus IP) Network modules are only available on v4.20 firmware or greater /INT variants. They store the parameters specific to the Modbus IP connection.

A Type 19 (Modbus IP) Network module is required for each Modbus device the IQ4 is to communicate with.

Note: A maximum of 10 Type 19 (Modbus IP) Network modules can be added to the controller's strategy.

22.2.11.1 Trend Mode

Type 19 (Modbus IP) Network modules do not support the Trend Mode as Trend communication over Modbus is not supported.

22.2.11.2 Module Disable

Disabling the module will prevent ALL communication over the Wallbus network.

22.2.11.3 Alarm Handling

Type 19 (Modbus IP) Network modules do not generate any alarms.

22.2.11.4 Type 19 (Modbus IP) Network Module Parameters

Type 19 (Modbus IP) network modules include the 'Label' and 'Type' parameters common to all Network modules - see **"Network Module Common Parameters" on page 409**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Connection Status	The connection status	0 , 1 0 = Disconnected 1 = Connected	0	N	t
IP Address or Hostname	The IP address or host name of the modbus device.	Valid IP address or host name	0.0.0.0	N	I
TCP Port	The TCP port used for the Modbus communications.	0 to 65535	502	N	E

22.2.12 Network Module Common Parameters

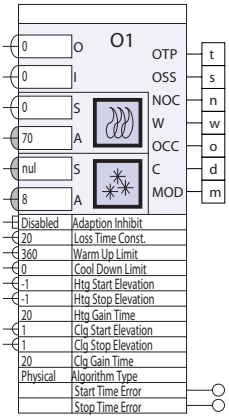
Network modules include the following parameter that common to most modules: 'Label' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Type	The type of Network module.	1, 4, 5, 8, 12, 13, 14, 16, 17, 19 1 = IQ Lan 4 = Ethernet IP 5 = BACnet IP 9 = Modbus Serial 12 = BACnet MSTP 13 = BACnet Application 14 = Wallbus 15 = Mbus 16 = Trend Client 17 = XNC Serial 19 = Modbus IP	-	N	Y

23 OPTIMUM START STOP MODULES

Module Identifier	On (where n is module number)
Default overview parameters	$,S
Size in brIQs	50

23.1 Functionality



Optimum Start Stop (OSS) modules provide a self-adaptive routine that learns the characteristics of the building by monitoring and comparing the outside air temperature against the inside air temperature, and the heating/cooling medium temperature. The module derives a dynamic thermal (Physical) model of the building which it uses to calculate the time that the heating/cooling should be switched on before the occupation of the building to achieve the desired temperature by the start of occupation (the optimum start time). It also calculates when the heating/cooling can be turned off before the end of occupation without diverging from comfortable conditions until after the end of occupation (the optimum stop time).

Note: OSS modules can be used with any item of plant that has similar characteristics to a building.

OSS Modules appear on the strategy page but do not need to be sequenced as they operate directly from the Real Time Clock (RTC).

By default OSS modules use the times provided by the equivalent Time Schedule module (i.e OSS module 1 uses the times from time Schedule module 1), these times are referred to as the Occupation Time Profile (OTP). For IQ4s with v3.30 or greater the Time Schedule module providing these times can be changed - see **"23.1.9 Connection to Time Schedule Module" on page 422**.

They can operate in heating, cooling, heating and cooling, or neither (i.e. OTP only) modes. The following parameters are used by both heating and cooling modes:

Outside Temperature
Inside Temperature
Warmup Limit
Cooldown Limit
Loss Time Constant

There is a sub-module for each of the modes, heating and cooling. Each sub-module has the following parameters which are set up separately for heating and cooling:

OSS Setpoint
Medium Temperature
Start Elevation
Stop Elevation
Gain Time Constant
Average Medium Temperature (set up by the algorithm, read only to the user)

The OSS modules have 2 types of algorithms:

Physical Algorithm
Statistical Algorithm

23.1.1 Physical Algorithm

This is the traditional Trend OSS algorithm, the only available algorithm prior to v3.0 firmware. It is based on a dynamic thermal building model. If the actual thermal response of the building is similar to this model then it gives good results. However, the responses of some buildings differ enough to give significant start/stop time errors.

It is the default algorithm type, and should remain selected if using an old strategy that has provided successful OSS times for the Time Schedule module involved, or if the this algorithm successfully models the Time Schedule involved.

The OSS module can exist in one of the three states indicated by the 'State' parameter.

0 = Passive (no heating/cooling)
1 = Heating
2 = Cooling

These can be analysed by starting with Time Schedule module's non-occupied state and the OSS module in the 'Passive' state.

23.1.1.1 Non Occupied

When in the Time Schedule module is in the non occupied state the OSS module decides whether it is in heating mode or cooling mode. It will only be in cooling mode if the 'Cooling OSS Setpoint' parameter's value is non-zero and the inside temperature is greater than the cooling target (i.e. Cooling OSS setpoint + Cooling Start Elevation).

The OSS module will look for the next OTP start time in the Time Schedule module, if none exists today, it looks for one tomorrow; if none exists tomorrow it looks no further. If the next time is tomorrow it will reference it by adding 24 hours i.e. 32:00 hours is 8 am tomorrow. The time is referenced to the previous midnight.

It will then use the outside temperature, inside temperature, medium temperature, last adapted gain coefficient and average medium temperature to calculate the 'warm up period' required to achieve the required temperature + start elevation. The 'warm up period' is then limited by the Warmup Limit and then the optimum start is calculated as:

optimum start time = OTP start time - warm up period

During non-occupation the **Optimum On Time** shown is the next optimum start time and the Optimum Off Time is the last optimum stop time.

23.1.1.2 Warm-Up

As soon as the current time equals the optimum start time it will change into the warm-up phase. During this phase, the OSS module will adapt the appropriate Gain Time Constant. It will adapt by using:-

New time constant = 0.8 x old time constant + 0.2 x new estimate.

This adapts the values smoothly and holds some historical information i.e. the trend in weather or seasonal variation. The gain time constant will not update if:

- Heating/cooling is for less than 10 minutes.
- The change in inside temperature during warm up was less than 1 deg C.
- The difference between inside and medium temperature is too low.

Note: The gain time constant will not adapt after a period of four hours following the start of warm-up.

During the warm-up the **Optimum On Time** shown is the last optimum start time and the **Optimum Off Time** is the last optimum stop time.

23.1.1.3 Occupied

During occupation the OSS module will calculate the optimum stop time. It does this by looking to the next OTP stop time. This could be during the day ahead as for the start time. At any time it will decide whether it is in heating or cooling mode and calculate the cool-down time accordingly. It performs this by measuring the inside temperature, the outside temperature, and using yesterday's adapted loss time constant to calculate the cool-down time to occupied setpoint + stop elevation. It will limit the cool-down time to the Cooldown Limit.

At any time the optimum stop time is the latest of OTP time less heating cool-down time or OTP time less cooling cool-down time.

During the occupation phase the **Optimum On Time** shown is the last optimum start time and the **Optimum Off Time** is the next optimum stop time as calculated above.

23.1.1.4 Cool-down

During this phase, if the predicted inside temperature at the end of occupation is more than 1 deg C below the optimum stop target (1 deg C above for cooling) then it will revert to OTP times.

It will not adapt the loss time constant if:

- Heating or cooling resumes less than 10 minutes after switching it off.
- The inside temperature changes by less than $^1/_2$ deg C during cooling period.
- The difference between inside and outside temperatures was too low.

Note: The loss time constant will not adapt after three hours.

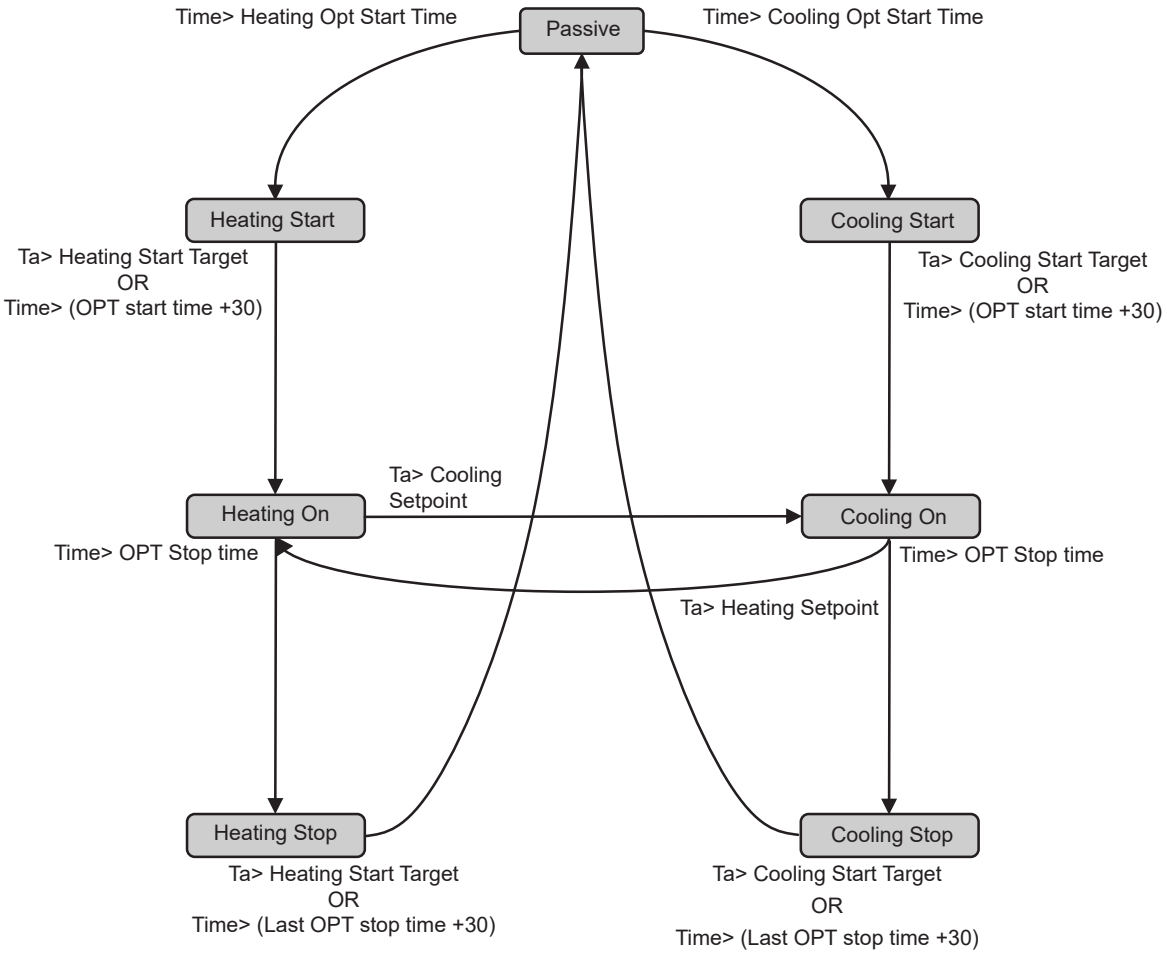
The loss time constant will use the same method of adaption as the gain time constant i.e. new time constant = 0.8 x old time constant + 0.2 x new estimate.

During cool-down the **Optimum On Time** shown is the last optimum start time and the **Optimum Off Time** is the last optimum stop time.

23.1.2 Statistical Algorithm

The statistical algorithm (which is available from v3.0 firmware) uses a statistical method to model the relationship between the measured temperatures and the start and stop times. This reduces errors that may occur due to any mismatch between the actual building and the physical model. It should be used instead of the physical algorithm in all cases other than those described on **page 411**..

The statistical algorithm will run through a number of states as shown below. The transition into and out of the states depends on the current time, and the relationship between the inside temperature and the target temperature(s).



The condition(s) for exiting a state is given adjacent to the arrow leaving the state. Other transitions between states exist (e.g if time is re synchronised or Time Schedule times are changed) but have been left off the diagram for simplicity.

Note: It is possible to transfer from heating mode to cooling mode and vice versa during the Heating On or Cooling On states (assuming both heating and cooling modes are enabled).

23.1.2.1 Passive State

During the Passive state, the OSS will calculate the optimum start time. It has to decide whether to optimise for heating or cooling (assuming both are enabled):

> IF inside temperature < heating start target temperature, THEN it will calculate heating optimum start time
> ELSE IF inside temperature > cooling start target temperature, THEN it will calculate cooling opt. start time
> ELSE IF inside temperature between these targets THEN continue monitoring.

Once the time exceeds the calculated heating or cooling optimum start it will save its start data (for future use) and transition to the Heating Start or Cooling Start state (switch on heating/cooling).

23.1.2.2 Heating Start State

During the Heating Start state it will check:

> IF the inside temperature > heating start target temperature
> OR IF Time > (OTP start time + 30 mins)
> THEN update the statistical model and transition to Heating On state

The second condition to change state means that OSS will exit this state after 30 mins regardless of whether the target temperature has been reached.

23.1.2.3 Heating On State

During the Heating On phase it will calculate the heating optimum stop time and check:

> IF time > optimum stop time it will record heating stop data and transition to the Heating Stop state

It will also check:

> IF inside temperature > Cooling setpoint it will transition to Cooling On state

23.1.2.4 Heating Stop State

During the Heating Stop state it will check:

> IF inside temperature < optimum stop target temperature
> OR IF time > (OTP stop time + 30 minutes)
> THEN update the statistical model and transition to Passive state

The second condition to change state means that OSS will exit this state after 30 mins regardless of whether the target temperature has been reached.

23.1.2.5 Cooling Start State

During the Cooling Start state it will check:

> IF the inside temperature < cooling start target temperature
> OR IF Time > (OTP start time + 30 mins)
> THEN update the statistical model and transition to Cooling On state

The second condition to change state means that OSS will exit this state after 30 mins regardless of whether the target temperature has been reached.

23.1.2.6 Cooling On State

During the Cooling On phase it will calculate the cooling optimum stop time and check:

IF time > cooling optimum stop time it will record cooling stop data and transition to the Cooling Stop state

It will also check:

IF inside temperature < Heating setpoint it will transition to Heating On state

23.1.2.7 Cooling Stop State

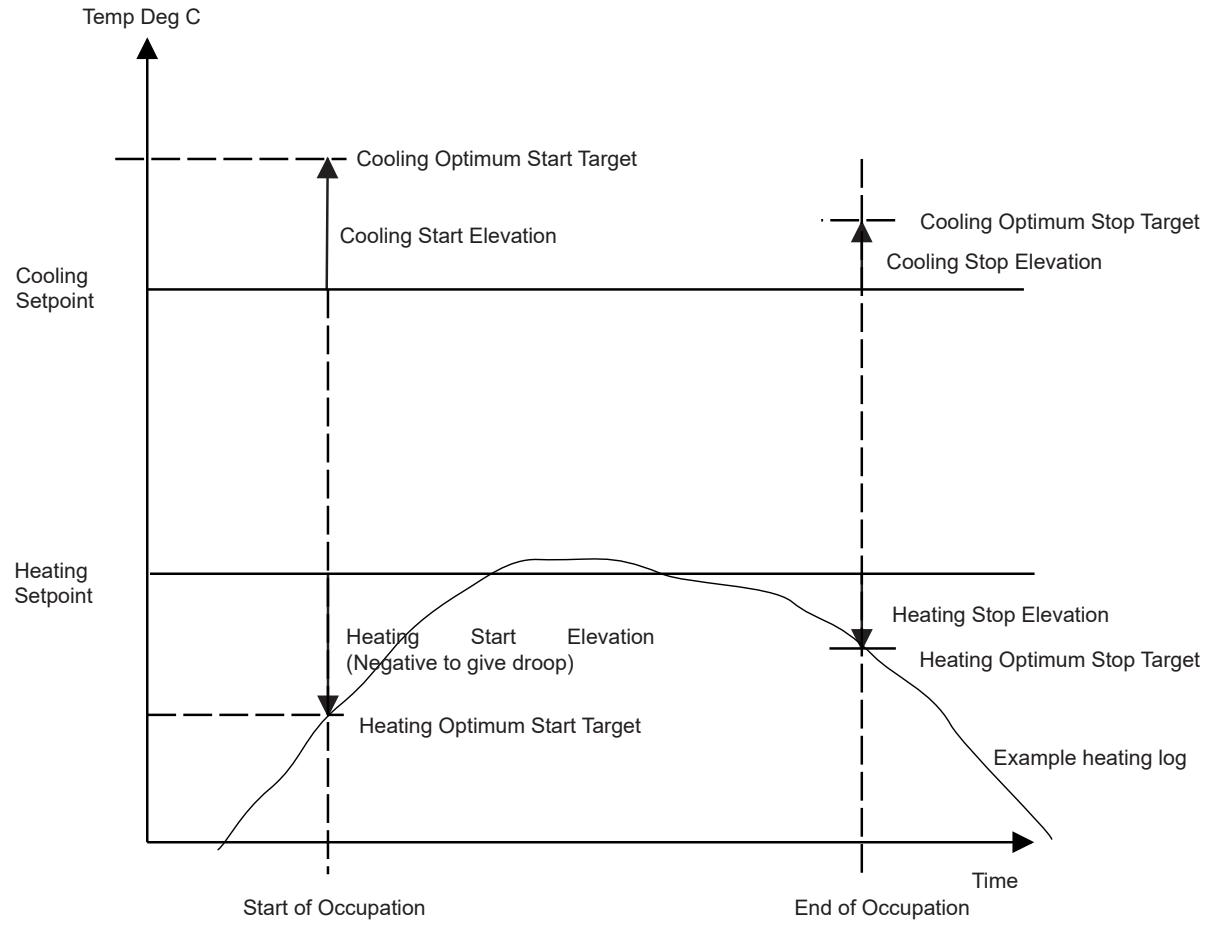
During the Cooling Stop state it will check:

IF inside temperature < cooling optimum stop target temperature
OR IF time > (last OTP stop time + 30 minutes)
THEN update the statistical model and transition to Passive state

The second condition to change state means that OSS will exit this state after 30 mins regardless of whether the target temperature has been reached.

23.1.3 Target Temperatures

Each mode (heating and cooling) is given a Start Elevation and Stop Elevation. This value is added to the mode's OSS Setpoint value in order to determine the target temperature for optimum start and optimum stop. Generally, heating elevations will be set negative to give a droop below the heating setpoint (and the cooling elevations will be set positive). If both heating and cooling are used, the cooling start target should always be above the heating start target. The same rule applies to stop targets. This is illustrated in the diagram.



23.1.4 Adapted Data

The ‘Loss Time Constant’ and ‘Gain Time Constant’ parameters are automatically adjusted every time sufficient data is gathered from operation of the plant, this is known as adapted data.

They should initially be set up with reasonable values so that acceptable OSS times may be obtained without a long initial learning period. Both time constants may be estimated by using logs of the building temperature taken on a typical day. The ‘Loss Time Constant’ parameter is used by both heating and cooling modes, but there is a separate ‘Gain Time Constant’ parameter for each mode.

23.1.4.1 Adapted Data for Heating

If OSS is being used for heating then the ‘Loss Time Constant’ parameter used is also used for cooling, but a different ‘Gain Time Constant’ parameter (‘OSS Heating Module - Gain Time Constant’ parameter) is used

23.1.4.1.1 Initial Loss Time Constant Value

The Loss Times constant is the same for heating and cooling. The initial value of the Loss Time Constant’ parameter can be calculated using the following formulas or estimated.

Calculating the initial value of the ‘Loss Time Constant’ parameter (Calculated using heating)
For the physical algorithm:

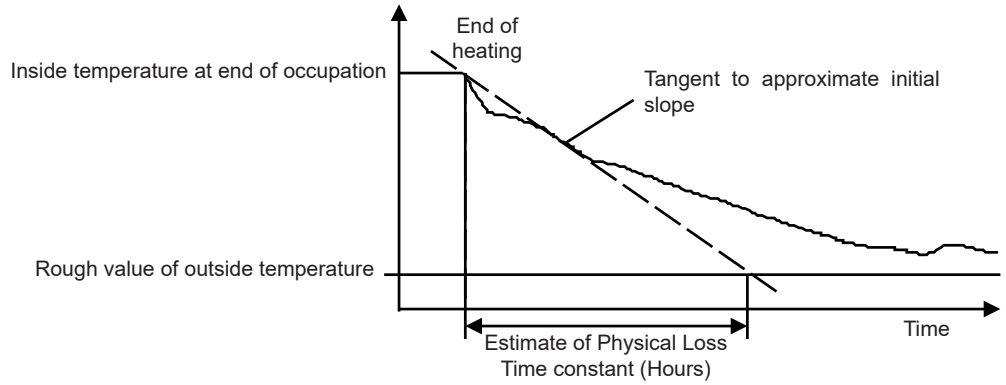
Loss Time Constant = (Inside temp - Outside temp)/Initial rate of temp loss in ^{o}C/hour

For the statistical algorithm:

Loss Time Constant = 1/Initial rate of temp loss in ^{o}C/hour

Note: The physical algorithm’s ‘Loss Time Constant’ parameter’s value will probably be between 5 hours for a very poorly insulated building and 30 hours for a very well insulated building. When adapted, it will tend to be shorter in winter than in summer as the OSS adapts to seasonal variation.

Estimating the initial value of the ‘Loss Time Constant’ parameter (from heating situations)
The initial value of the ‘Loss Time Constant’ parameter may be estimated by using logs of the inside and outside temperature taken with no heating or cooling applied. This is done from the graph of inside temperature against time by estimating the time where the tangent to the initial cool-down curve reaches the outside temperature. For the physical algorithm the time constant is then the difference between this time and the end of heating. For the statistical algorithm it is the same time difference, but divided by the difference between inside and outside temperatures.



23.1.4.1.2 Initial OSS Heating Module - Gain Time Constant Value

The initial value of the 'OSS Heating Module - Gain Time Constant' parameter can be calculated using the following formulas or estimated.

Calculating the initial value of the 'OSS Heating Module - Gain Time Constant' parameter
For the physical algorithm:

> OSS Heating Module Gain Time Constant = (Medium Temp - Initial Temp)/Initial rate of rise of inside temp in °C/hour.

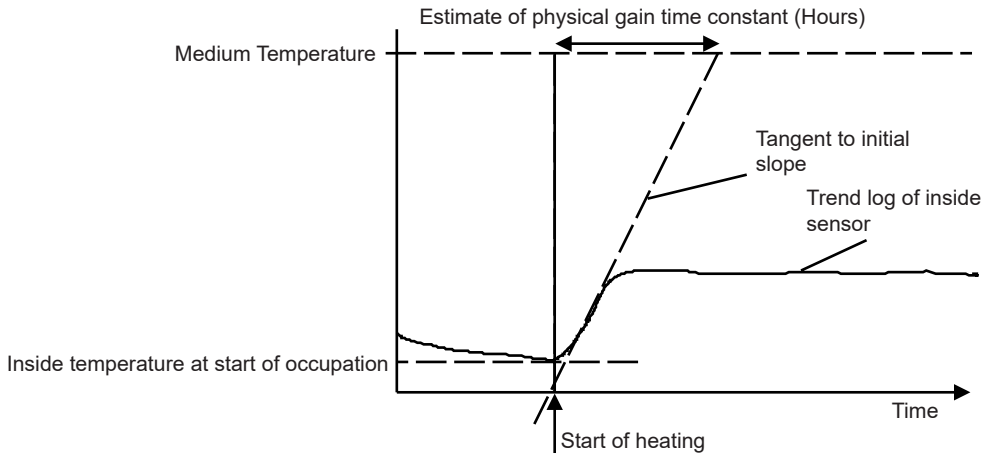
For the statistical algorithm:

> OSS Heating Module Gain Time Constant = 1/Initial rate of rise of inside temp in °C/hour

Note: The physical algorithm's 'OSS Heating Module - Gain Time Constant' parameter's value will be shorter on a building which is well insulated and longer on a poorly insulated building. The values will be in the approximate range from 5 hours (well heated) to 50 hours (poorly heated).

Estimating the initial value of the 'OSS Heating Module - Gain Time Constant' parameter
The initial value of the 'OSS Heating Module - Gain Time Constant' parameter can be estimated by using logs of the inside temperature and medium temperature during the period immediately after the heating has been applied.

This is done from the graph of inside temperature against time by estimating the time where the tangent to the initial warm-up curve reaches the heating medium temperature. For the physical algorithm the time constant is then the difference between this time and the start of heating. For the statistical algorithm it is the same time difference, but divided by the difference between inside and outside temperatures.



23.1.4.2 Adapted Data for Cooling

If OSS is being used for cooling then a different 'Gain Time Constant' parameter ('OSS Cooling Module - Gain Time Constant(hours)' parameter) is used for cooling. The 'Loss Time Constant' parameter is the same for heating and cooling and is entered once for both modes. Estimating and calculating these time constants using cooling mode parameters is very similar to the heating mode method, but the signs of the temperature and slopes change.

23.1.4.2.1 Initial Loss Time Constant Value

The Loss Time Constant is the same for heating and cooling, and is entered once for both modes. However, for cooling negative values are used as shown below:

The initial value of the Loss Time Constant' parameter can be calculated using the following formulas or estimated.

Calculating the initial value of the 'Loss Time Constant' parameter (Calculated using cooling)
For the physical algorithm:

> Loss Time Constant = (Inside temp - Outside temp)/Initial rate of temp loss in °C/hour

E.g. Inside Temp = 18
Outside Temp = 23
Rate of temp loss = -1 °C/hr (i.e. negative loss = warming up)

Loss time constant = (18-23)/-1 = 5 hrs

For the statistical algorithm:

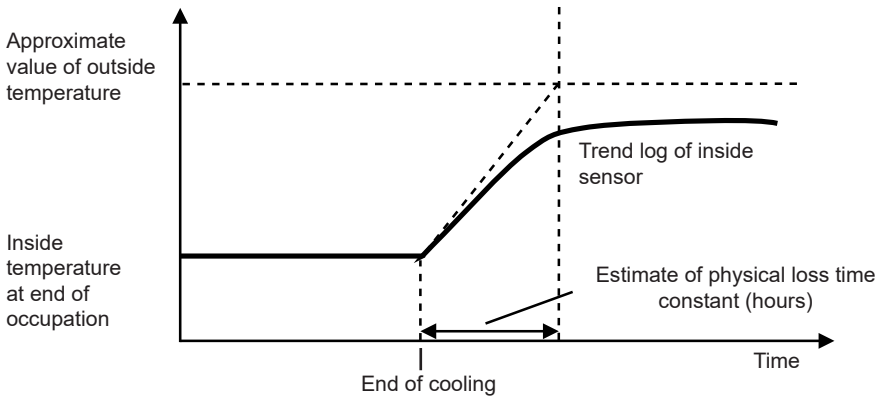
Loss Time Constant = 1/Initial rate of temp change in °C/hour

E.g. Rate of change = 1 °C/hr

Loss time constant = 1/1 = 1hr/°C

Estimating the initial value of the 'Loss Time Constant' parameter (from cooling situations)

The estimate from the graph is done in a similar way to heating, but in this case space warms up after cooling is switched off, so the physical time constant is the time between the end of cooling and the time when the tangent to the initial warmup curve reaches the outside temperature, and the statistical one is this time divided by the difference between inside and outside temperatures.



23.1.4.3 Initial OSS Cooling Module - Gain Time Constant(hours) Value

The initial value of the 'OSS Cooling Module - Gain Time Constant(hours)' parameter can be calculated using the following formulas or estimated.

Calculating the initial value of the 'OSS Cooling Module - Gain Time Constant(hours)' parameter

For the physical algorithm:

OSS Cooling Module - Gain Time Constant(hours) = (Medium Temp - Initial Temp)/Initial rate of rise of inside temp in °C/hour.

E.g. Medium temp = 5 °C
Initial temp = 25 °C
Initial rate of rise = -4 °C/hour (i.e. negative + cooling down)

Gain time constant = (5-25)/-4 = -20/-4 = 5 hours

For the statistical algorithm:

Gain Time Constant = 1/Initial rate of change of inside temp in °C/hour

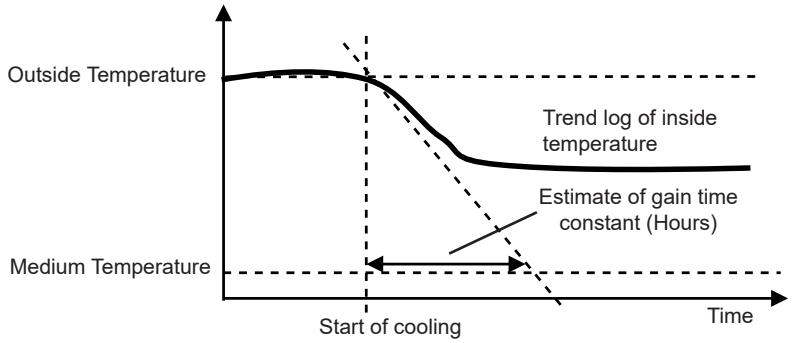
E.g. Initial rate of rise = 4 °C/hour
Gain time constant = 1/4 = 0.25 hours/°C

Estimating the initial value of the 'OSS Cooling Module - Gain Time Constant(hours)' parameter

The initial value of the 'OSS Cooling Module - Gain Time Constant(hours)' parameter can be estimated using logs of the inside temperature and medium temperature immediately after the heating/cooling has been applied.

This is done from the graph of inside temperature against time by estimating the time where the tangent to the initial cool down curve reaches the cooling medium temperature. For the physical algorithm the time constant is then the difference between this time and the start of cooling.

For the statistical algorithm it is the same time difference, but divided by the difference between inside and outside temperatures.



23.1.5 Heating and Cooling Loops in the Zone

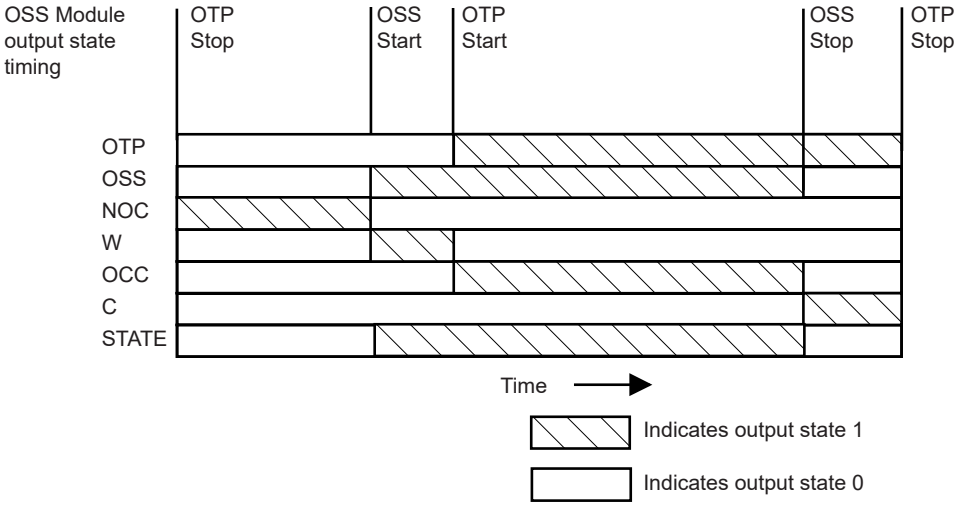
The setpoint select inputs to the Loop modules controlling the heating and cooling in the zone, should be sourced from the OSS module's 'OSS State' parameter. This is very important, for if the zone's inside temperature is strongly influenced by heating or cooling which is not under OSS control, then the Loss and Gain Time Constants may be incorrectly learnt, resulting in incorrect optimum start and stop times.

23.1.6 OSS State Outputs

The OSS module has seven outputs:

OTP State
OSS State
Non Occupied State
Warmup State
Occupied State
Cooldown Sate
Mode

The 'Mode' parameter indicates whether the OSS module is operating in heating or cooling mode. The other outputs are set to '1' or '0' depending on the time in relation to the OSS cycle. They are available for use within all schedules allocated to this zone without making external connections, but can be used in the strategy (e.g. to switch loop setpoints). This is illustrated in the diagram.



The 'State' parameter shows the current state of the OSS algorithm; these states differ between the two algorithms.

For the physical algorithm this will indicate either:

'Heating' or 'Cooling' between the OSS Start and OSS Stop times, and
'Passive' outside these times.

For the statistical algorithm this will indicate:

'Heating Start' between OSS start and OTP start (warmup for heating - W)
'Heating On' between OTP between the start of occupation and the end of occupation (occupied for heating - OCC)
'Heating Stop' between optimum stop time and the end of occupation time (cooldown for heating - C)
'Cooling Start' between OSS start and OTP start (warmup for cooling - W)
'Cooling On' between OTP between the start of occupation and the end of occupation (occupied for cooling - OCC)
'Cooling Stop' between optimum stop time and the end of occupation time. (cooldown for cooling - C)
'Passive' outside these times (NOC).

23.1.7 Time Errors

There are two time errors:

Start Time Error, and
Stop Time Error

They are calculated by the OSS module and are available as outputs for connection to other modules and enable the efficiency of the OSS function to be monitored.

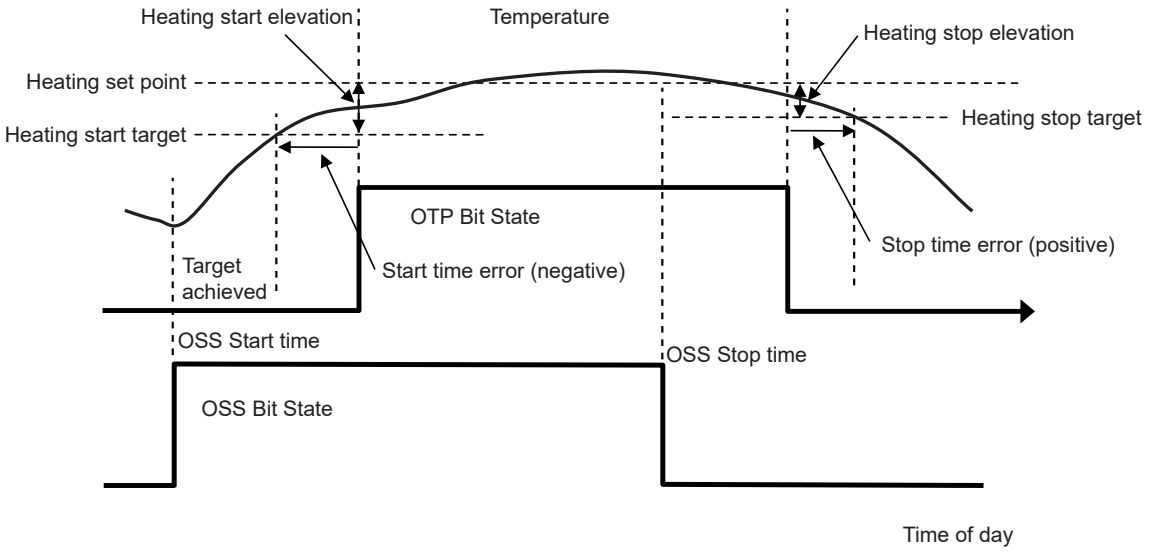
Start Time Error: This is the difference between the required start time and the achieved start time when the space temperature reached the required level. An early achieved start time will give a negative error.

Start Time Error = time start target temperature was achieved - start of occupation time (OTP)

Stop Time Error: This is the difference between the required stop time and the achieved stop time when the space temperature reached the required level. An early achieved stop time will give a negative error.

Stop Time Error = time stop target temperature was achieved - end of occupation time (OTP)

The diagram shows the heating start target (setpoint plus negative start elevation) being reached early (before OTP start), and hence giving a negative start error. The heating stop target (setpoint plus negative stop elevation) is reached late (after OTP stop), and hence giving a positive stop error.

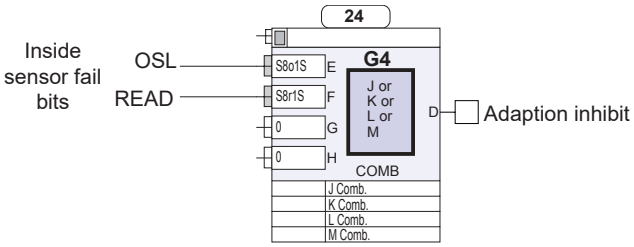


23.1.8 OSS Should be in Control

The OSS State output must bring on the heating loops which are in control of the inside temperature measurement specified. If some other action controls these loops then this will affect the calculation of the loss and gain time constants and cause incorrect optimum start and stop times to be calculated.

One cause of this is if either the Outside Temperature sensor or the Inside Temperature sensor has alarm bits set, or if either sensor is removed.

The alarm bits can be used to stop OSS adapting during sensor failure by using the OSS Adaption Inhibit input.

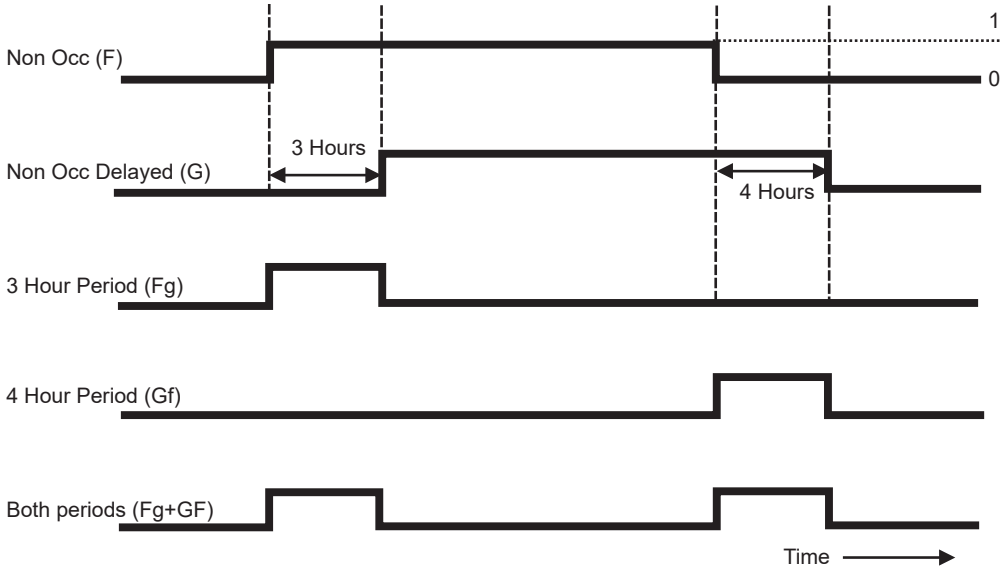


This can also be done if an override such as a loop manual override or a caretaker override overrides the OSS time action. In these cases the override bit can be connected directly to the Adaption Inhibit.

If the override or sensor fail is shorter than the period over which the OSS Module adapts, then it will start to adapt during the remaining period as the sensor changes.

The logic clamp should thus be held for 4 hours at the start of warm-up and 3 hours at start of non occupation (because, the Gain Time Constant does not adapt after a period of four hours, and the Loss Time Constant does not adapt after a period of three hours). Additional logic is required to do this.

We can create these time periods by delaying the NON OCC rising edge for 3 hours and the NON OCC falling edge (i.e. start of warm-up) for 4 hours e.g. by using a logic module with an on delay and an off delay.

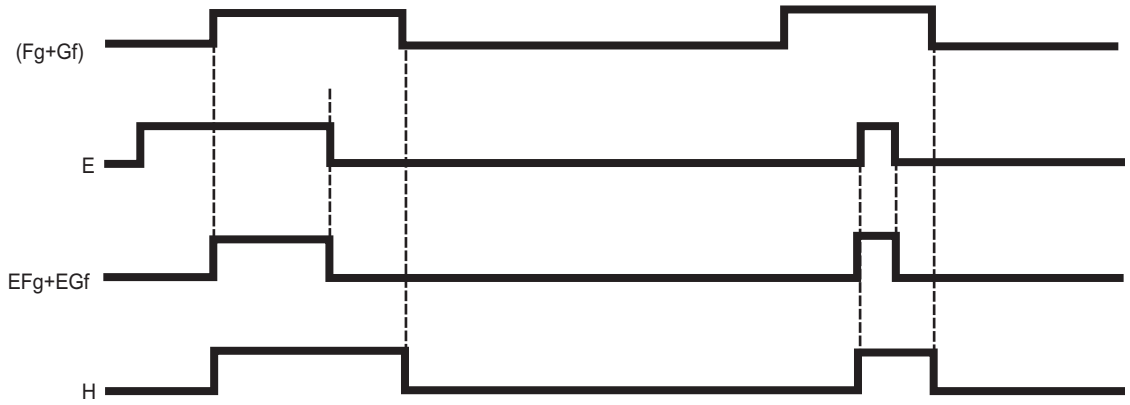


We call the NON-OCC bit (**F**) and the delayed NON OCC bit (**G**). The three hour period is when F is there, and G is not, F AND NOT G (Fg), and the four hour period is when G is there and F is not, G AND NOT F (**Gf**). To operate the logic clamp the correct condition must be used, that is, Fg + Gf which is the last time graph above.

If the fault/override occurs during periods we want the gate to clamp, and we want it to stay clamped until the end of that period.

If we call the fault/override bit E, we want to clamp if :

E AND (Fg+Gf)
= E(Fg+Gf)
= EFg+EGf



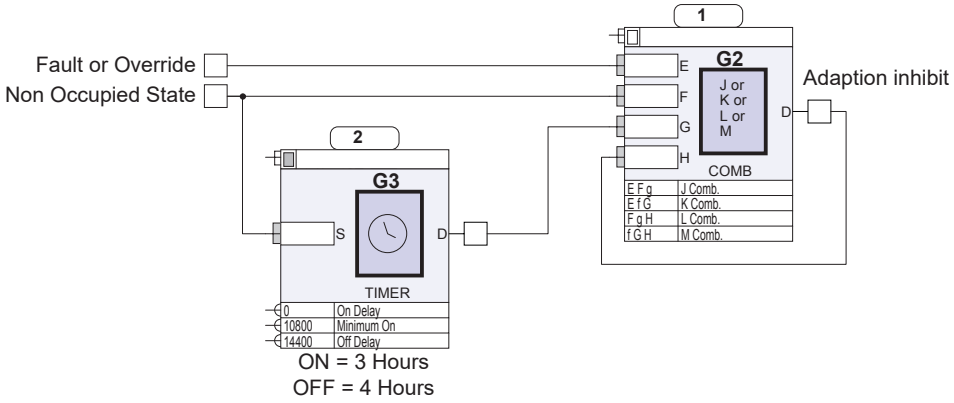
Now we want it to stay clamped until the end of the period as shown by the graph H above. This can be done by feeding back the output so that we hold the clamp on for the rest of the period:

I.e. for H AND (Fg + Gf)
= H(Fg + Gf)
= HFg + HGf

Our full expression for H now becomes:

H = EFg + EGf + HFg + HGf

This strategy can be achieved as below:



23.1.9 Connection to Time Schedule Module

IQ4s with v3.30 or greater firmware. The Time Schedule module used by OSS modules to determine occupation can be specified using the 'Time Schedule' parameter. This allows multiple OSS modules to use the same Time Schedule module. For earlier versions of firmware the Time Schedule module is determined by the OSS module number, this behaviour can be replicated by setting the 'Time Schedule' parameter to '0'.

23.2 Differences Between IQ4 and IQ2 OSS Modules

- Statistical algorithm was not present in IQ2.
- If there was not a valid medium temperature reading, the Medium Address was set to '0', and a value entered into the Average Medium Temperature in IQ2, whereas in IQ4 the user sets up the Medium Temperature as a constant (and lets the OSS module calculate the Average Medium Temperature).
- In IQ2 the heating mode is disabled by setting the Heating OSS Setpoint to '0' whereas in IQ4 it's disabled by nulling the OSS Setpoint (similar for cooling mode).

23.3 OSS Module Parameters

OSS modules include the following parameters that are common to most modules: 'Label', 'Page', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Adaption Inhibit	This will stop the OSS Module adapting if set to '1'.	0 or 1 1 = ON 0 = OFF	0	I	H
Algorithm Type	The OSS algorithm being used.	0 or 1 0 = Physical 1 = Statistical	0	N	Y
Cooldown Limit	The maximum limit of the cool down time (minutes). It should be set to the maximum length of time prior to the end of occupation that the optimum stop times may occur. If set to '0' it will effectively disable optimum stop.	0 to 1439	0	I	D
Cooldown State	The cooldown state - Set to '1' between optimum stop time and the end of occupation time.	0 or 1	0	O	d
Inside Temp	The value of the inside temperature. It should be connected to the inside temperature sensor. If several inside temperature sensors are fitted, a function (e.g. minimum) of these values may be used, in which case, it will be connected to the function module output.	0 to 100	0	I	I
Loss Time Constant	The Loss Time Constant (hours) of the system being optimised. Adapted by OSS.	0 to 100	0	I	L
Mode	The mode in which the OSS module has decided it is operating.	0 or 1 0 = heating 1 = cooling	0	O	m
Non Occupied State	The non occupied state - Set to '1' between occupation stop time and optimum start time.	0 or 1	0	O	n
Occupied State	The occupied state - Set to '1' between start of occupation and optimum stop time.	0 or 1	0	O	o
Optimum Off Time	The current calculated optimum off time.	00:00 to 48:00	00:00	O	F
Optimum On Time	The current calculated optimum on time.	00:00 to 48:00	00:00	O	N
OSS Cooling Module - Average Medium Temp	The temperature of the cooling medium at the end of the warmup phase. Adapted by OSS. It is set to the OSS Cooling Module - Medium Temp after the first warm up.	Full analogue	0	N	c1(M)
OSS Cooling Module - Gain Time Constant	The Cooling Gain Time Constant of the system being optimised. Adapted by OSS. Measured in hours for the physical algorithm and hours/°C for the statistical algorithm.	-100 to +100	0	N	c1(G)
OSS Cooling Module - Medium Temp	The temperature of the cooling medium. It should be an estimate of the average medium temperature during the warmup period and should be entered as a specific value or can be connected to the appropriate medium's temperature control loop setpoint knob. *Note: By default IQSET sets the value to 8°C.*	Full analogue	0	I	c1(A)

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
OSS Cooling Module - OSS Setpoint	The setpoint used by the Loop module being switched into occupation by the OSS state output. If set to null, it disables OSS cooling mode operating. *Note that to disable a mode the setpoint must be nulled - not set to '0'; this is done by selecting the 'Not Used' box in IQSET, or deleting the value to leave a blank on the web page. If this is not done the mode will operate using a setpoint of 0.*	Full analogue	0	I	c1(S)
OSS Cooling Module - Start Elevation	Amount by which temperature can differ from setpoint at end of warmup. Set positive for cooling.	-100 to +100	0	I	c1(T)
OSS Cooling Module - Stop Elevation	Amount by which temperature can differ from setpoint at beginning of cool down. Set positive for cooling.	-100 to +100	0	I	c1(P)
OSS Heating Module - Average Medium Temp	The temperature of the heating medium at the end of the warmup phase. Adapted by OSS. It is set to the OSS Heating Module - Medium Temp after the first warm up.	Full analogue	0	N	h1(M)
OSS Heating Module - Gain Time Constant	The Heating Gain Time Constant of the system being optimised. Adapted by OSS. Measured in hours for the physical algorithm and hours/°C for the statistical algorithm.	-100 to +100	0	N	h1(G)
OSS Heating Module - Medium Temp	The temperature of the heating medium. It should be an estimate of the average medium temperature during the warmup period and should be entered as a specific value or can be connected to the appropriate medium's temperature control loop setpoint knob. *Note: By default IQSET sets the value to 70°C .*	Full analogue	0	I	h1(A)
OSS Heating Module - OSS Setpoint	The setpoint used by the loop module being switched into occupation by the OSS state output. If set to null, it disables OSS heating mode operating. *Note that to disable a mode the setpoint must be nulled - not set to '0'; this is done by selecting the 'Not Used' box in IQSET, or deleting the value to leave a blank on the web page. If this is not done the mode will operate using a setpoint of 0.*	Full analogue	0	I	h1(S)
OSS Heating Module - Start Elevation	Amount by which temperature can differ from setpoint at end of warmup. Set negative for heating (i.e. droop). Can be connected to an analogue output.	-100 to +100	0	I	h1(T)
OSS Heating Module - Stop Elevation	Amount by which temperature can differ from setpoint at beginning of cool down. Set negative for heating (i.e. droop). Can be connected to an analogue output.	-100 to +100	0	I	h1(P)
OSS State	The OSS occupation state - Set to '1' between start of warmup (optimum start time) and start of cooldown (optimum stop time). It can be used for switching the loop into occupation.	0 or 1	0	O	s

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
OTP State	The OTP occupation state - Set to '1' between the start of occupation and the end of occupation.	0 or 1	0	O	t
Outside Temp	The value of the outside temperature. It should be connected to the outside temperature sensor. If several outside temperature sensors are fitted, a function (e.g. average) of these values may be used, in which case, it will be connected to the function module output.	0 to 100	0	I	O
Start Time Error	The difference in minutes between the required start time and the start time achieved by OSS. A negative value means OSS start was early - see **"Time Errors" on page 420**.	Full analogue	0	O	E
State	The current state of operation of the OSS module:	For physical algorithm: 0, 1, 2 0 = Passive 1 = Heating 2 = Cooling For statistical algorithm: 0, 1, 2, 3, 4, 5, 6 0 = Passive, 1 = Heating Start 2 = Heating On 3 = Heating Stop 4 = Cooling Start 5 = Cooling On 6 = Cooling Stop	0	O	S
Stop Time Error	The difference in minutes between the required stop time and the stop time achieved by OSS. A negative value means OSS stop was early - see **"Time Errors" on page 420**.	Full analogue	0	O	e
Time Schedule	v3.30 or greater firmware. Time Schedule module used by this OSS module to determine occupation status. *Note: In previous firmware versions the Time Schedule module used is determined by OSS module number. This can be replicated by setting this parameter to '0'.*	Valid module number. 0 = use OSS module number	OSS module number	I	Z
Warmup Limit	The maximum limit of the warm up time (minutes). It should be set to the maximum length of time prior to the start of occupation that the optimum start times may occur. If set to '0' it will effectively disable optimum start.	0 to 1439	0	I	W
Warmup State	The warmup state - Set to '1' between OSS start and OTP start.	0 or 1	0	O	w

23.4 Optimum Start Stop Connectives

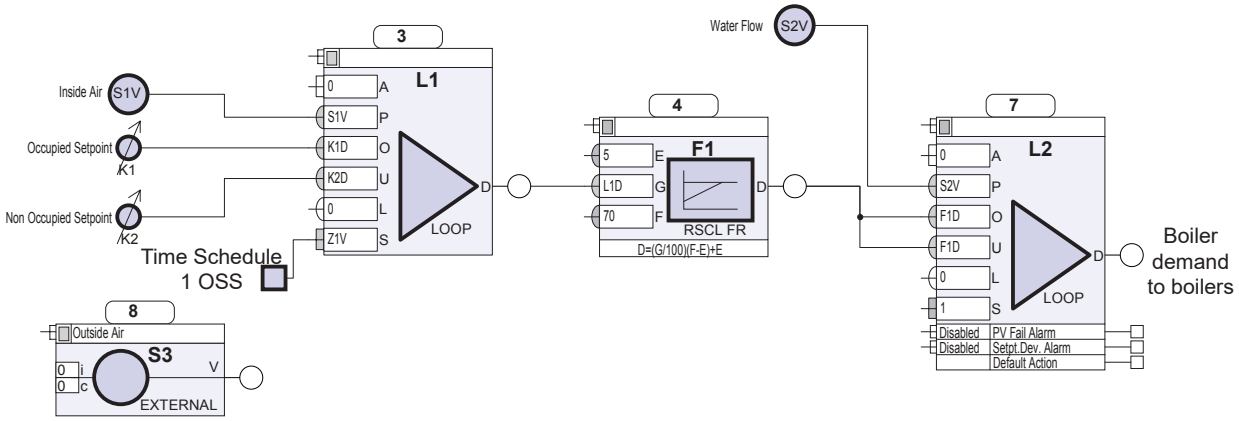
The following parameters are available as connectives:

Cooldown State
Mode
OSS State
Non Occupied State
Occupied State
OTP State
Warmup State
Start Time Error
Stop Time Error

23.5 OSS Examples

23.5.1 Example 1 Zone 1

A heating system using boilers and radiators. Heating control is by way of a cascaded loop as follows:



The OSS module is set up as follows (heating only):

Outside Temperature is connected to S3.
Inside Temperature is connected to S1.
Heating Medium Temperature is set to a constant of 60 deg C.
Heating OSS Setpoint is connected to Knob 1.

The building cools down at 1.2 deg C/hour when the inside temperature is at 20 deg C and outside temperature is at 5 deg C.

estimate of Physical Loss Time Constant = (20-5)/1.2 = 12.5 hours.
estimate of Statistical Loss Time Constant = 1/1.2 = 0.0833 hours/deg C

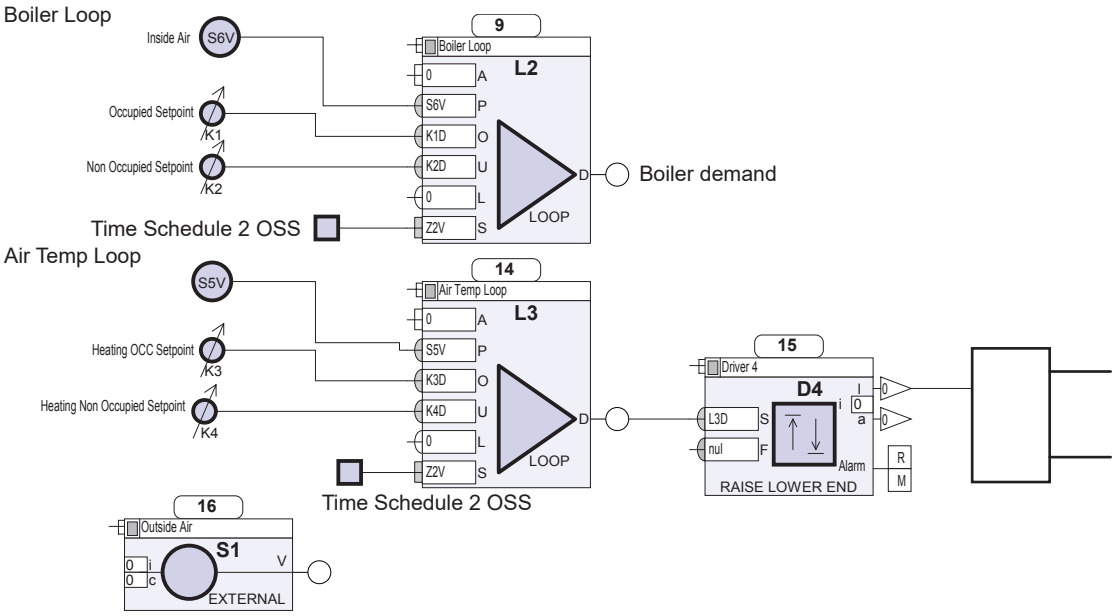
The heating medium temperature = 60 deg C.
The building warms up at 3 deg C/hour when the inside temperature is 15 deg C.

estimate of Physical Heating Gain Time Constant = (60-15)/3 = 15 hours
estimate of Statistical Heating Gain Time Constant = 1/3 = 0.33 hours/deg C

Heating Start Elevation = -2 deg C
Heating Stop Elevation = -1 deg C
Warm Up Limit = 300 s (5 hours)
Cool Down Limit = 60 s (1 hour)

23.5.2 Example 2 Zone 2 (heat/cool)

Heating:
The boiler provides constant flow temperature to the zone with a three-way valve controlling air temperature.



Outside Temperature is connected to S2

Inside Temperature is connected to S5

Heating Medium Temperature can be connected to K1 since this sets the heating medium temperature during OSS.

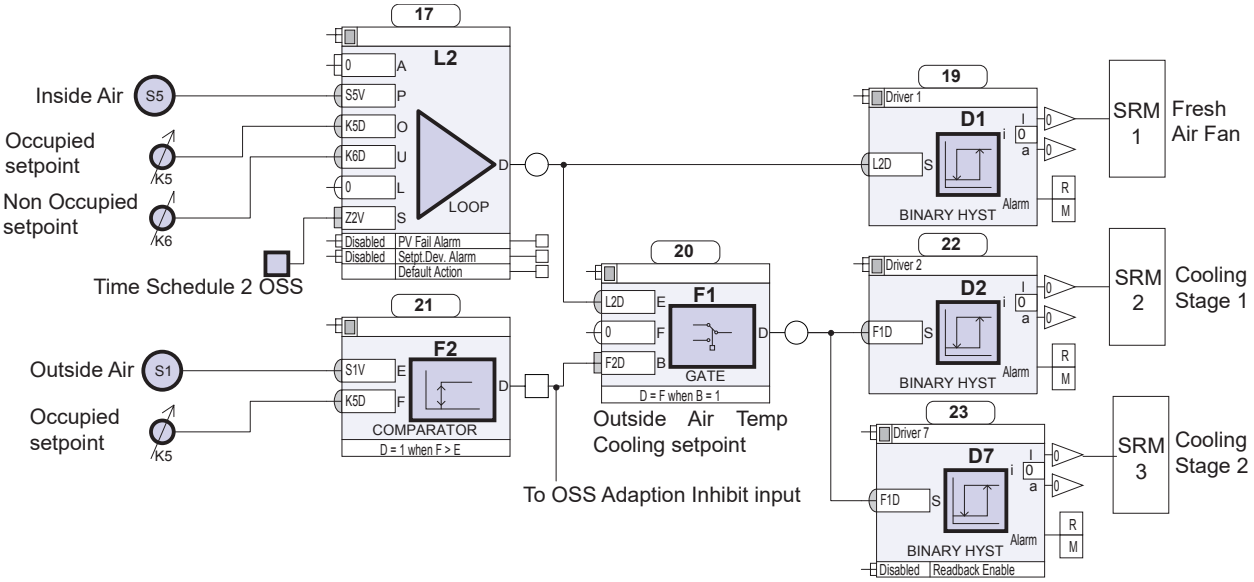
Heating OSS Setpoint is connected to K3

Heating Start Elevation = -2 deg C

Heating Stop Elevation = -1 deg C

Note: Estimates for Loss Time Constant and Heating Gain Time Constant will have to be entered.

Cooling:
Cooling is provided in three steps; full fresh air, cooling stage 1, cooling stage 2. However, in order to save energy when fresh air alone is sufficient for cooling, the compressor is inhibited if the outside air temperature is less than the cooling occupation setpoint.



Cooling Medium Temperature sensor is set to a specific value with an estimated value of the cooling medium temperature (5 deg C).

Cooling Setpoint is connected to Knob 5

Cooling Start Elevation = 1 deg C
Cooling Stop Elevation = 3 deg C

Since the cooling mode uses the same 'Loss Time Constant' parameter as the heating mode it does not need to be entered again, but a separate Cooling Gain Time Constant must be entered.

Note: The compressor is inhibited if the outside air is less than the cooling setpoint, so under these conditions the OSS module's adaption must be stopped. This can be done by connecting the F2 comparator output to the OSS Adaption Inhibit input.

23.5.3 Example 3 Electric Heating

When electric heating is used, the rate of heat input to the building is not dependent on the inside temperature but is fixed in that electric heaters are either ON giving maximum heat input or OFF giving no heat input.

The best method to use when setting up the OSS module for electric heating is to set the Heating Medium Temperature to a comparatively high value, i.e. 100 deg C. The Heating Gain Time Constant can be estimated

24 OPTIONS MODULE

Module Identifier	Vn (where n is module number)
Default overview parameters	-
Size in brIQs	0

24.1 Functionality

The Options module contains data about the options included in the IQ4. It only appears on web pages. The following options are covered:

XNC interface (/XNC)
Trend current loop network (/LAN , IQ4NC or /XNC)
BACnet communications (/BAC or IQ4NC)
XML Web Services (fitted as standard)
INC (IQ4NC only)
MSTP LAN (IQ4NC)

24.2 Options Module Parameters

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Card ID	Board type *Note: Each of these is shown as a board even though the option may be achieved by populating a section of the main board.*	1 to 4 1 = Trend current loop LAN board 2 = not used 3 = not used 4 = MSTP LAN Card	0	N	I
Description	Long description of option	0 to 100 character string.	0	N	D
Title	Short title for the option	0 to 25 character string.	Blank	N	T
Type	The type of option. Card is a hardware option (e.g. MSTP or Trend LAN). Feature is a firmware feature supported by the IQ4 (e.g. BACnet)	1, 2 1 = card 2 = feature	0	N	Y
Version	Version. For feature options only (i.e. BACnet).	“issx.yz mmm d yyyy”	0	N	V

25 PERFORMANCE MODULE

Module Identifier	q1
Default overview parameters	-
Size in brIQs	0

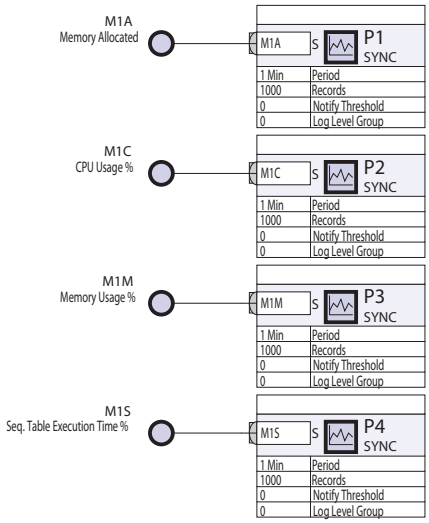
25.1 Functionality

The Performance module makes the IQ4's performance data available to the strategy, e.g. for its parameters to be plotted.

It does not appear on the strategy page and is not sequenced. It is automatically run periodically to collect performance data.

25.1.1 IQSET Device/Diagnostic Plots Feature

IQSET facilitates the plotting of the four performance parameters by allowing strategy with Plot modules connected to each of the Performance module's connectives the to be easily added.



25.2 Performance Module Parameters

The Performance module includes the following parameter common to most modules: 'Label' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Comments*	*Range*	*Default*	*Type*	*Text comms*
CPU Usage (%)	The current CPU usage as a percentage.	Full analogue	0	O	C
Memory Allocated (bytes)	The current total memory usage in bytes.	Full analogue	0	O	A
Memory Usage (%)	The current total memory usage as a percentage.	Full analogue	0	O	M
Sequence Table Execution (%)	The sequence table execution time as a percentage of the full CPU time usage.	Full analogue	0	O	S

25.3 Performance Module Connectives

The following parameters are available as connectives:

Memory allocated (bytes)
CPU Usage(%)
Memory Usage(%)
Sequence Table Execution(%)

26 PAGE MODULES

Module Identifier	pn (where n is module number)
Default overview parameters	-
Size in brIQs	4

26.1 Functionality

Page modules contain details about the strategy page. Only the label is sent to the IQ4 in the strategy file. This is so that the strategy pages may be backwards engineered from the IQ4 complete with their page labels.

Page modules do not appear on the strategy page and are not sequenced.

The web page for the Page module is not accessible from the module overview page. The module's overview web page can be accessed directly by using the page's URL which is in the format:

<controller address>/p.htm

<controller address> is the URL required to access the IQ4.

Individual Page modules can be accessed using the page number in the URL in the format:

<controller address>/p<n>.htm

<controller address> is the URL required to access the IQ4. <n> is the page number.

26.2 Page Module Parameters

Page modules include the following parameter common to most modules: 'Label' - see **"Common IQ4 Module Parameters" on page 165**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Parent	Not used	-		-	R

27 PLOT MODULES

Module Identifier	Pn (where n is module number)
Default overview parameters	($)
Size in brIQs	12

Note: Plot modules also use plot memory in addition to brIQs - see **"Plot Memory" on page 444**.

27.1 Functionality

Plot modules enable a connectable output value or status to be recorded at defined times in order to produce a graph or table of values. There are several types of plot module - see **"Plot Module Types" on page 436**.

Plot modules appear on the strategy page and are not sequenced as they operate directly from the real time clock. The actual timing varies with the type of Plot module.

27.1.1 BACnet Compliance

Only Type 3 (Periodic) (with default settings) should be used if the strategy is required to comply with the BACnet standard.

27.1.2 Alarms

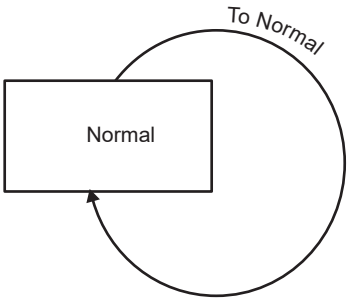
Plot modules generate BBUF item alarms and send them to the Alarm Group specified by the 'Alarm Group' parameter.

Note: There is no clear alarm for the BBUF alarm.

27.1.2.1 Plot Module BACnet Alarm State Transitions

The Plot module's BBUF alarm is a Normal state and there are no Fault or off Normal states, so the only transitions between states are from Normal to Normal and is considered an 'event' rather than an alarm.

The transition from no alarm present into the BBUF alarm state will generate a to Normal transition.



The BACnet alarm state transitions are explained in **"Alarm State Transitions" on page 65**.

27.1.3 Stop When Full

Type 2 (Triggered), and Type 3 (Periodic) and Type 4 (COV) Plot modules have the stop when full feature which, when enabled, causes logging of values to stop when the Plot module is full (i.e the number of values recorded reaches the specified number of records). When disabled the values will wrap as for Type 1 (Synchronised) Plot modules.

Recording actually stops when the Plot module reaches the number of records minus one. At this point the 'Enable' parameter is set to '0', the last plot record is used to record the status as 'Disabled', and the 'Log Full' parameter is set to '1'.

Note: The 'Log Full' parameter will be set to '1' whenever the Record Count = Number of Records, regardless of the value of the 'Stop When Full' parameter.

Note: Clearing of the 'Enable' parameter is a BACnet requirement, therefore for compliance with the BACnet standard, the 'Enable' parameter must be left unconnected.

27.1.4 Threshold Notification

The threshold notification feature generates a Buffer ready Event (BBUF) alarm when the number of records in the Plot module since the last notification (or since the beginning of logging if no notification has occurred) reaches the notification threshold, and resets the 'Records Since Last Notification' parameter before starting counting again. This enables a supervisor to record the values stored by the Plot module before they are overwritten.

The alarm is sent to the Alarm Group module specified by the Plot module's 'Alarm Group' parameter and processed according to the alarm handling strategy.

27.1.5 Module Disable

Recording can be enabled or disabled using the 'Enable' parameter. When the recording is disabled a record is kept in the module with the appropriate status and time stamp.

27.1.6 Double Precision (v3.30 or greater)

Plot modules can record both double and single precision values depending on the value of the 'Data Precision' parameter. If configured to record double precision values, the module will record the values in double precision providing that value is a double precision value; if the value being recorded is a single precision value it will be recorded as single precision.

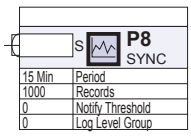
Note: Recording double precision values will use more memory - see **"27.5 Plot Memory" on page 444***. Not all modules provide double precision values.*

27.2 Plot Module Types

There are different Plot module types specified using the 'Timing Type' parameter:

Module Type	*Description*
Type 1 (Synchronised)	Type 1 (Synchronised) Plot modules are the traditional Trend plot, they can log up to 1000 records of the value at fixed intervals (from 1s up to 24 hours).
Type 2 (Triggered)	Type 2 (Triggered) Plot modules log the value when the trigger input changes from '0' to '1'. This gives a recording of events at a specific time that could otherwise be missed by Type 1 (Synchronised) or Type 3 (Periodic) Plot modules, furthermore triggered plotting maximises use of plot memory as it only records when required.
Type 3 (Periodic)	Type 3 (Periodic) Plot modules log the value at the interval defined by the 'Interval' parameter which is an input; so the interval may change from time to time. This can save memory as more data can be recorded when important events are occurring, and less when there is nothing significant to record.
Type 4 (COV)	Type 4 (COV) Plot modules only record a value when the value differs from the last logged value by more than the 'COV Increment' parameter.

27.2.1 Type 1 (Synchronised)



Type 1 (Synchronised) Plot modules record values at a defined interval (Period).

The following periods are available:

Period	*Duration for Maximum for 1000 Records*
1s	16 m 40 s
1m	16 h 40 m
5m	3 days 11 h 20 m
10m	6 days 22 h 40 m
15m	10 days 10 h
20m	13 days 21 h 20 m
30m	20 days 20 h

Period	*Duration for Maximum for 1000 Records*
1h	41 days 16 h
6h	250 days
24h	1000 days

Note: These durations assume all the records are for samples taken, but some records may be used to record other events (e.g. time synchronisation) - see **"Plot Records" on page 446**.

The normal period used for system monitoring is the 15 minutes, the faster periods are used for fast moving systems or control loop tuning, and the hourly period for such functions as energy targeting or degree days.

Note: There is nothing to stop the same value being logged by different Plot modules at different periods.

27.2.1.1 Threshold Notification

Supported - see **"Threshold Notification" on page 436**.

27.2.1.2 Power Failure Operation

When the IQ4's power is restored after power failure, the recordings that have been missed are filled in with the last value taken prior to the power failure. This results in a flat response for the period of the power failure. The status of each missed plot record is set to invalid.

27.2.1.3 Stop When Full

Not supported.

27.2.1.4 Alarm Handling

Supported - see **"Alarm Handling Strategy" on page 79**.

27.2.1.5 Double Precision

Supported - see **"Double Precision (v3.30 or greater)" on page 436**.

27.2.1.6 BACnet Compliance

Not compliant.

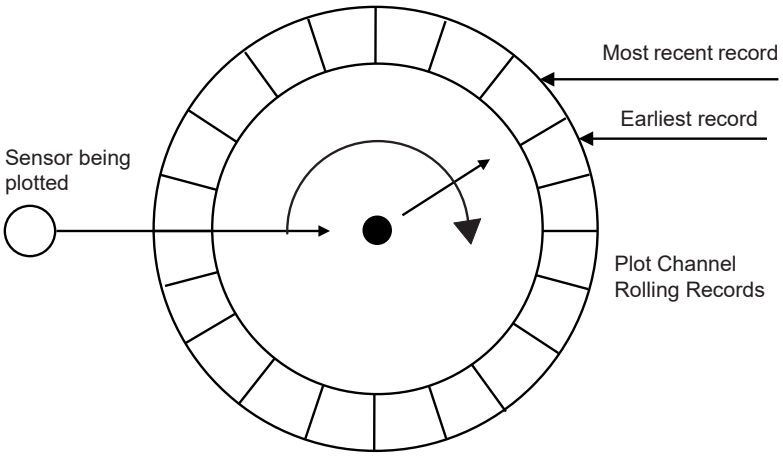
27.2.1.7 Module Disable

Not supported

27.2.1.8 Number of Records

Type 1 (Synchronised) Plot modules have a user definable number of records. When the specified number of records is reached the recordings wrap round. This means that after the number of records specified have been recorded the earliest record is overwritten so that subsequently, the full set of most recent records is always available.

The module keeps a marker on its most recent record and when it receives a request for data, the values are sent from this marker backward along with the time of the most recent record, enabling a graph of the values in chronological order to be constructed.

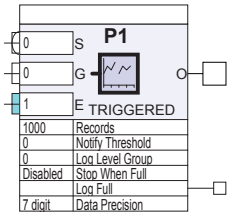


27.2.1.9 Type 1 (Synchronised) Plot Modules Parameters

Type 1 (Synchronised) Plot modules include the parameters common to all Plot modules - see **"Plot Module Common Parameters" on page 442**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Period	The interval between records: *Note: For the 1s compact plot timing inaccuracy - see* **"Effect of Time Synchronisation on Type 1 Plot Modules" on page 445**.	0, 1, 2, 3, 4, 5, 6, 7, 8, 9 0 = 1 hour 1 = 15 minutes 2 = 24 hours 3 = 1 minute 4 = 5 minutes 5 = 10 minutes 6 = 20 minutes 7 = 30 minutes 8 = 6 hours 9 = 1 second	0	N	P

27.2.2 Type 2 (Triggered)



Type 2 (Triggered) Plot modules only record a value when triggered. The value is recorded when the 'Trigger Input' parameter changes from '0' to '1'. They are useful in strategies because they allow recordings to occur asynchronously to the time pulses, for example:

- On status change (e.g. for recording digital values that change infrequently)
- On change of value (e.g. a user adjustable setpoint).

Recordings are not taken regularly; each record also contains a time stamp.

Note: A trigger that occurs less than one second after the previous trigger will be ignored.

Note: If the 'Trigger Input' parameter is connected to a time pulse, then it is effectively a synchronised plot, but also has Stop when Full functionality.

27.2.2.1 Threshold Notification

Supported - see **"Threshold Notification" on page 436**.

27.2.2.2 Stop When Full

Supported - **"Stop When Full" on page 435**.

27.2.2.3 Alarm Handling

Supported - see **"Alarm Handling Strategy" on page 79**.

27.2.2.4 Double Precision

Supported - see **"Double Precision (v3.30 or greater)" on page 436**.

27.2.2.5 BACnet Compliance

Not compliant.

27.2.2.6 Module Disable

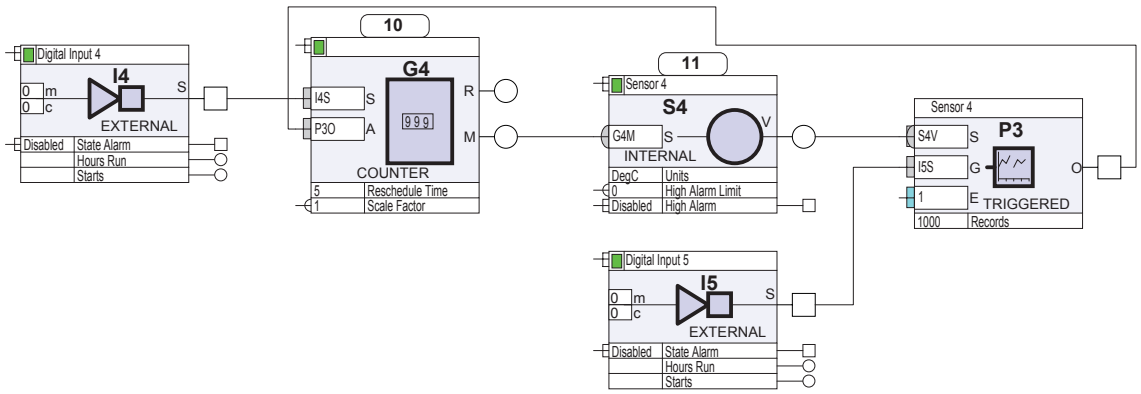
Supported - see **"Module Disable" on page 436**.

27.2.2.7 Number of Records

Type 2 (Triggered) Plot modules have a user definable number of records. They can either wrap records as for Type 1 (Synchronised) Plot modules - see **"Number of Records" on page 438**, or stop recording when the specified number of records is reached - **"Stop When Full" on page 435**.

27.2.2.8 Event Driven Strategy

The rising trigger causes the Plot module to record the value before modules that use the state of the 'Trigger Input' parameter to zero down values. Because this sequence cannot be guaranteed with event based modules, the 'Trigger Output' parameter is available for connection to subsequent modules in the strategy ensuring that they do not change the value before it has been recorded.



In the above strategy the meter pulse cumulative count is monitored by a sensor (S4). When the maximum demand occurs the triggered plot (P3) is triggered to take a reading of the sensor, and the trigger output (P3O) then resets the counter ready for the next maximum demand period.

27.2.2.9 Module Disable

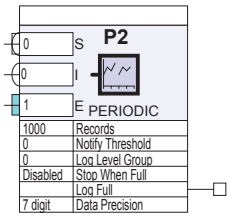
Supported - see **"Module Disable" on page 436**.

27.2.2.10 Parameters

Type 2 (Triggered) Plot modules include the parameters common to all Plot modules - see **"Plot Module Common Parameters" on page 442**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Trigger Input	The value that triggers recording. Recording occurs when the value changes from '0' to '1' not when it changes from 1 to 0.	0 or 1	0	I	G
Trigger Output	The state of the 'Trigger Input' parameter passed to this output for use in the strategy. It ensures that the record has been taken before any 'downstream' modules act on the trigger state.	0 or 1	0	O	O

27.2.3 Type 3 (Periodic)



Type 3 (Periodic) Plot modules record values at an interval specified by the 'Interval' parameter. This is different from Type 1 (Synchronised) Plot modules because the 'Interval' parameter can be connected to other modules in the strategy allowing the interval to be changed. This allows more data to be recorded when important events are occurring, and less when there is nothing significant to record which can save memory.

Note: The sampling time is not synchronised with the timing pulses, and starts when the 'Enable' parameter is set to '1'.

Recordings are not taken regularly; each record also contains a time stamp.

27.2.3.1 Threshold Notification

Supported - see **"Threshold Notification" on page 436**.

27.2.3.2 Stop When Full

Supported - see **"Stop When Full" on page 435**.

27.2.3.3 Alarm Handling

Supported - see **"Alarm Handling Strategy" on page 79**.

27.2.3.4 Double Precision

Supported - see **"Double Precision (v3.30 or greater)" on page 436**.

27.2.3.5 BACnet Compliance

Compliant.

27.2.3.6 Module Disable

Supported - see **"Module Disable" on page 436**.

27.2.3.7 Number of Records

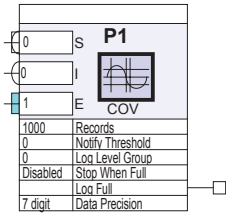
Type 3 (Periodic) Plot modules have a user definable number of records. They can either wrap records as for Type 1 (Synchronised) Plot modules - see **"Number of Records" on page 438**, or stop recording when the specified number of records is reached - **"Stop When Full" on page 435**.

27.2.3.8 Type 3 (Periodic) Plot Module Parameters

Type 3 (Periodic) Plot modules include the parameters common to all Plot modules - see **"Plot Module Common Parameters" on page 442**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Interval	The interval (seconds) between recorded values.	0 to 86400 (24 hours)	0	I	I

27.2.4 Type 4 (COV)



IQ4 v3.30 firmware or greater. Type 4 (COV) Plot modules only record a value when the value differs from the last logged value by more than the 'COV Increment' parameter. If connected to a digital value the 'COV Increment' parameter is not used and the value will be recorded whenever it changes. The Type 4 (COV) Plot module will record a value when the IQ4 is restarted or when the module is enabled to provide an initial value to which the COV can be applied.

27.2.4.1 Threshold Notification

Supported - see **"Threshold Notification" on page 436**.

27.2.4.2 Stop When Full

Supported - **"Stop When Full" on page 435**.

27.2.4.3 Alarm Handling

Supported - see **"Alarm Handling Strategy" on page 79**.

27.2.5 Double Precision

Supported - see **"Double Precision (v3.30 or greater)" on page 436**.

27.2.5.1 BACnet Compliance

Not compliant.

27.2.5.2 Module Disable

Supported - see **"Module Disable" on page 436**.

27.2.5.3 Number of Records

Type 4 (COV) Plot modules have a user definable number of records. They can either wrap records as for Type 1 (Synchronised) Plot modules - see **"Number of Records" on page 438**, or stop recording when full the specified number of records is reached - **"Stop When Full" on page 435**.

27.2.5.4 Module Disable

Supported - see **"Module Disable" on page 436**.

27.2.5.5 Type 4 (COV) Plot Module Parameters

Type 4 (COV) Plot modules include the parameters common to all Plot modules - see **"Plot Module Common Parameters" on page 442**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
COV Increment	The amount by which 'Source' parameter can differ from the last recorded value before the module will record another value. When linked to a digital value this parameter is ignored. The 'COV Increment' parameter must not be set to '0'.	+ Full analogue	0	N	I

27.3 Plot Module Common Parameters

Plot Modules include the following parameters that are common to most modules: 'Label', 'Page', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm Group	The Alarm Group module to which the BBUF alarm is sent.	0 to 500	0	N	g
Data Precision	v3.30 or greater. The precision of the recorded value - see **"Double Precision (v3.30 or greater)" on page 436**.	0 or 1 0 = Single precision 1 = Double precision	0	N	D
Last Notify Record	The value of the 'Total Record Count' parameter when the last BBUF alarm was sent.	0 to 2^{32}-1	0	O	l
Enable	Not Type 1 (Synchronised). Enables/disables the logging. The module will stop recording values when set to '0', and start recording when set to '1'. *Note: For BACnet compliance this should not be connected.*	0 or 1 0 = Disabled 1 = Enabled	1	I	E
Log Full	Not Type 1 (Synchronised). Indicates whether the log is full i.e. the number of records has reached the value specified by the 'Number of Records' parameter.	0 or 1 1 = Log full	0	O	F
Notification Threshold	The number of records to occur since the last BBUF alarm was sent before the next one will be sent (e.g. if set to 200 the alarm would be sent every 200 values i.e. 200, 400, 600 etc).	0 to 2^{32}-1	0	N	n
Number of Records	The maximum number of records for this plot.	0 to 1000 0 disables the module	1000	N	N
Record Count	The total number of records taken since the module started recording values. It will start at '0' and increase up to the number of records.	0 to number of records	0	O	R
Records Since Last Notification	The number of values recorded since the last BBUF alarm was sent.	0 to 2^{32}-1	0	N	r
Source	The status or value to be recorded.	0 or 1 Or Full analogue	0	I	S

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Stop When Full	Not Type 1 (Synchronised). Specifies whether logging will stop when the log is full - see **"Alarms" on page 435**.	0 or 1 0 = wrap around when full 1 = stop when full	0	N	W
Time of Last Notification	The date and time the last BBUF alarm was sent.	date&time;yyyy-mm-dd&Thh:mm:ss	0	O	b
Time of Last Record	The date and time of the last record taken.	date&time;yyyy-mm-dd&Thh:mm:ss	0	O	T
Timing Type	The type of Plot module.	1, 2, 3, 4 1 = Synchronised 2 = Triggered 3 = Periodic 4 = COV	-	N	Y
Total Record Count	Unique number to identify record It will start at 0 after a strategy download and will increase up to 2^{32}-1, when it will wrap back to '1'.	0 to 2^{32}-1	0	O	C

27.4 Access to Plots

The information recorded by Plot modules can be obtained using precision or compact protocols. Precision protocols are more accurate than compact protocols but take a longer time to transmit.

The way in which precision and compact protocols work depends on the combination of controller and supervisor being used. Some Trend products enable the protocol used to retrieve the recorded values to be specified.

Traditional Compact and Precision Protocols: These protocols are provided to allow compatibility with products that do not support the new plot protocols.

The precision protocol sends each value as 6 ASCII characters (e.g. 999999, 9999.9, .00001).

The compact protocol sends each value as a percentage of maximum value minus the minimum value (i.e. maximum error of 1%); these integer percentage values are each coded and communicated as one ASCII character, so can be sent approximately 6 times faster than using the precision protocol.

The values are sent back with the start time and period so that the time points are calculated in the supervisor. With the compact protocol the start time is in minutes so that the timing for the 1 second compact plot will be inaccurate, and attempts to join subsequent plots together may result in gaps or overlaps.

Note: Use of the precision protocol is not supported for Sensor modules above 99. Use of the compact protocol is not supported for Sensor modules above sensor 256. Plot modules for sensors above 99 cannot be viewed from 963, and Sensor modules above 256 can only be viewed from the IQ4 web pages.

New Protocols: With these new plot protocols the problems with 1 second logging inaccuracy, and with sensor numbers are solved so the plots for all Sensor modules (and other types of modules) can be viewed.

27.4.1 New Compact Protocols

These can only be used for Type 1 (Synchronised) Plot modules. The compact protocol can be either single or double precision.

Single precision is as for the traditional compact protocol. The values are sent as a percentage of maximum value minus the minimum value (i.e. maximum error of 1%); these integer percentage values are each coded and communicated as one ASCII character.

The values are sent back with the start time and period so that the time points are calculated in the supervisor. However, the start time is in seconds so that the timing for the new 1 second compact plot is accurate and joining subsequent plots together is OK.

Double precision plots use the same algorithm as for the single precision plots, but use two ASCII characters to code the percentage value and thus have a maximum error of 0.01% (i.e. 0 to 9999). They take approximately twice the time to communicate.

27.4.2 New Text Comms Protocols

These must be used for Type 2 (Triggered), Type 3 (Periodic) Plot, and Type 4 (COV) Plot modules, but may also be used for Type 1 (Synchronised) Plot modules.

The new text communications protocols have a selectable number of decimal places (0 to 9). The 963 v3.1 or greater allows the precision to be selected by a setting in the INI file. It defaults to 2 decimal places.

Note: Increasing the number of decimal places will increase the number of digits communicated, and hence slow down the retrieval of plot data by the supervisor.

Text communications enable the supervisor/display to access all the data in the Plot record; the supervisor can select the data required for particular types of display. For Type 1 (Synchronised) Plot modules, timestamps and time offsets are generated when they are requested.

27.4.3 IQ4 Graph Webpages

The IQ4 Graph webpage takes values directly from the Plot records and displays them as a chart. It will only present the graph and values to two decimal places; the integer part can be as large as required by the value stored (up to the full analogue value). The graph will resolve daylight saving changes by adjusting the time points so that the times between the readings are correct.

27.4.4 Graphs

Graphs of the recorded values are available from the IQ4s web pages. They give a graph of the values in the Plot module, and a spot value which can be moved along the graph trace. As the point is moved along the graph its point value is displayed. A section of graph can be enlarged by the zoom feature. A list of the values used to create the graph can also be seen, and in some supervisors can be exported as a text file (e.g. for use by spreadsheet software).

In web pages the graph can be seen by clicking  in either the Plot module or the source module summary pages. If a module's output is plotted by more than one Plot module, a line is added to the module overview page for each Plot module, so each plot has a separate icon (see below).

S9	Sensor 9	603.00	DegC	Value - 1 Minute (P8) Value - 15 Minutes (P9) Value - 24 Hours (P10) Value - 30 Minutes (P11) Value - 20 Minutes (P12) Value - 1 Second (P13) Value - 6 Hours (P14) Value - 10 Minutes (P15)	No Alarm

27.5 Plot Memory

Plot modules use memory in two ways, each Plot module uses 12 brIQs and the stored records use plot memory (measured in log points). The IQ4's plot memory size (number of log points) is dependent on controller type - see the IQ4s data sheet for details.

Single Precision Values

For Type 1 (Synchronised) Plot modules each record requires 5 log points. For Type 2 (Triggered), Type 3 (Periodic) and Type 4 (COV) Plot modules each record requires 10 log points.

Double Precision Values

For Type 1 (Synchronised) Plot modules each record requires 10 log points. For Type 2 (Triggered), Type 3 (Periodic) and Type 4 (COV) Plot modules each record requires 20 log points.

27.6 Plot Module Limits

The maximum number of Plot modules is limited at 500.

The maximum number of recordings that can occur per second is 100 and this must be taken into account when designing the strategy. For example there could be 100 one second plots occurring at the same time, but no others.

Most strategies will contain Plot modules with different logging intervals, in which case the average number of recordings per second must be less then 100. This is calculated on the average plots serviced in 1 s, so a 1 minute plot would contribute a 1/60.

For example 90 off 1s, +360 off 1 min = 90+(360(1/60)) = 90 + 6 = 96 recordings per second.

Note: Type 2 (Triggered), Type 3 (Periodic) and Type 4 (COV) Plot modules must also be counted and it's up to the strategy designer to make an estimate.

27.7 Effect of Time Synchronisation

If the IQ4's Time module is re-synchronised and this results in a change to the IQ4's time this will effect the recorded data. The effect of time synchronisation is dealt with differently for Type 1 (Synchronised), Type 2 (Triggered), Type 3 (Periodic) and Type 4 (COV) Plot modules.

27.7.1 Effect of Time Synchronisation on Type 1 Plot Modules

With Type 1 (Synchronised) Plot modules using the traditional or the new compact protocols the record times are calculated by the supervisor or display when the plot is received. This results in a continuous plot where the earlier values may have been moved with respect to their original recording times.

E.g. A 1 minute plot has been running for several days and the time is now 14:00. The time is to be re-synchronised to 14:30.

If the original graph ran from 12:30 to 14:00, the new graph will run from 13:00 to 14:30. The graph's first sample time has been moved by 30 minutes to correct the graph to current time.

27.7.2 Effect of Time Synchronisation on Type 2, Type 3 and Type 4 Plot Modules

For Type 2 (Triggered), Type 3 (Periodic) and Type 4 (COV) Plot modules, the time re-synchronisation is stored as an event in the plot record. This includes the amount by which the time has changed. This may result in two records having overlapping timestamps (e.g. if the time change sets the time back).

If using the new text communications protocol, then the plot can be requested with its timestamp, and the supervisor/display can then resolve the individual record times with respect to the time change as it sees fit

27.8 Effect of Daylight Saving

If daylight saving is operating it will change the time twice a year. A Plot record will indicate in its status bits that daylight saving was in operation when the sample was taken.

The effects on the various plot types and protocols is similar to the effects of time synchronisation - see **"Effect of Time Synchronisation" on page 445**.

However the daylight savings status is saved for all types of Plot modules, so if using the new text communications protocol for Type 1 (Synchronised) Plot modules, the timestamps will be generated by the IQ4 taking into account the daylight saving information. This may result in two records having the same timestamp (e.g. if the time change sets the time back by an hour).

The supervisor or display has access to the daylight saving status and the Time module's 'Daylight Hours Change' parameter and will be able to use these (if required) to resolve the record times.

27.9 Plot Records

A Plot record is normally the value recorded at a specific time, but it may also be a record of a time change or some other event. The individual plot record contains the value of the sample taken along with associated data.

Plot records are not visible on a webpage. The data is presented in graphical and list format by supervisors and displays, it can also be accessed by text communications or BACnet communications.

Note that event type records will reduce the number of records available for taking samples.

27.9.1 Plot Record Parameters

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Record Number	Value of 'Total Record Count' parameter when reading was taken. This provides a fixed index for the record where the times may not run in sequence due to time changes or daylight savings.	0 to 2^{32}-1	0	O	B
Status	The status of the reading - see **"Status" on page 447**.	00000000 to 11111111	00000000	O	S
Time Offset	The time (seconds) since the last record was taken. It is generated when requested.	0 to 31536000 (1 year)	0	O	O
Timestamp	Not Type 1 (Synchronised). Time and date when reading was taken. It is stored for Type 2 (Triggered) Type 3 (Periodic) Plot modules, and Type 4 (COV) but not for Type 1 (Synchronised) Plot modules. If the timestamp of a synchronised plot is requested it is generated using the most recent time and back-calculating using the fixed period, daylight savings status and Time module's 'Daylight Hours Change' parameter.	date&time;yyyy-mm-dd&Thh:mm:ss	0	O	T
Type	The type of record. Always 2 for Type 1 (Synchronised) Plot modules.	0, 1, 2, 9 0 = Log status 1 = Digital value 2 = Analogue value 9 = Time change (in secs)	-	O	Y
Value	The value of the record. The type of value is defined by the 'Type' parameter.	**Type parameter = 0** 0, 1, 2 0 = Plot being disabled 1 = Plot being enabled 2 = Plot count being cleared **Type parameter = 1** 0 or 1 **Type parameter = 2** Full analogue **Type parameter = 9** Time change as an integral number of seconds (positive or negative) -3153 to max	0	O	V

Status

This is a digital byte which shows the status associated with the recorded value according to the table below:

Module Type	*Bit 7*	*Bit 6*	*Bit 5*	*Bit 4*	*Bit 3*	*Bit 2*	*Bit 1*	*Bit 0*
Digital input	Disabled	Overridden	Invalid Data	Daylight saving	-	-	-	Digin
Sensor	Disabled	Overridden	Invalid Data	Daylight saving	Read	Out of Limits	Low	High
Loop	Disabled	-	Invalid Data	Daylight saving	-	Default O/P	PV Fail	SP Deviation
Driver	Disabled	Overridden	Invalid Data	Daylight saving	HOA Override	-	Maintenance	Readback
Others	Disabled	-	Invalid Data	Daylight saving	Read	Out of Limits	Low	High

Bit 7 (Disabled): Set whenever the module sourcing the value being recorded is disabled (i.e. by the module's disable input).

Bit 6 (Overridden): Set when the module whose value is being recorded is being overridden.

Bit 5 (Invalid Data): Set to indicate that this value was backfilled to cover an interruption in power supply. Only applies to Type 1 (Synchronised) Plot modules.

Bit 4 (Daylight Saving): Set to indicate that the time module 'Daylight Savings Applied' parameter was set to '1' at the time this value was taken.

Bits 3 to 0 (Module Alarm Bits): Bits 3 to 0 reflect the source module's alarm status. The meaning of these bits varies with module type.

The BACnet Log_Buffer property includes Status_Flags which has the four flags:

IN_ALARM	True if any of the source module alarm bits are set (bits 0 to 3)
FAULT	True if Read, Out of Limits, or Maintenance status bits are set
OVERRIDDEN	True if Overridden (Bit 6) or HOA Override bit is set
OUT_OF_SERVICE	True if Disabled (Bit 7) is set.

27.10 Plot Module Connectives

For Type 2 (Triggered) Plot modules the following parameters are available as connectives:

Log Full
Trigger Output

For Type 3 (Periodic) and Type 4 (COV) Plot modules the following parameters are available as connectives:

Log Full

There are no connectives available for Type 1 (Synchronised) plot modules.

28 PROGRAM MODULE

Module Identifier	s1
Default overview parameters	-
Size in brIQs	0

28.1 Functionality

The Program module indicates the state of the strategy running in the IQ4. It is not sequenced and does not appear on the strategy page.

28.1.1 User Module Merging (v3.60 or greater)

User module merging controls what happens to users in the controller when a strategy is downloaded, this makes it easier to update and maintain users. The 'User Merge Option' parameter specifies how the User modules are merged during a strategy download.

Option	*Description*
Maintain existing users	Ignores any User module defined in the IQSET strategy leaving those in the IQ4 unchanged.
Merge and Maintain	Merges the User modules being downloaded by IQSET with those already present in the IQ4, leaving the User modules in the IQ4 unchanged if there are any duplicates (e.g. same user name and type).
Merge and Overwrite	Merges all User modules downloaded by IQSET with those in the IQ4, overwriting duplicate modules (same user name and type) in the IQ and leaving other User modules in the IQ4 unchanged.
Overwrite	Deletes all User modules in the IQ4 and replaces them with the User modules in the downloaded IQSET strategy.

28.2 Program Module Parameters

Parameter	*Comments*	*Range*	*Default*	*Type*	*Text comms*
Archive State	Indicates current status of the last strategy archive to Flash memory. The way in which the IQ4 archives strategy changes is described in - see **"Writing Changes to Flash" on page 34**.	0, 1, 2, 3, 4 0 = Never Archived (no strategy) 1 = Archive Primed 2 = Archive Active 3 = Last Archive Succeeded 4 = Last Archive was Bad	0	O	S
Filename	The filename of the current strategy file. Set up when a strategy file is transferred to or from the IQ4. It has standard filename format, Lyyynxxx. IQ4, where yyy is LAN number, xxx is the device address.	Lyyynxxx.IQ4	0	O	F
Fingerprint	Not currently used.	32 bits	Blank	O	f
Last Action Timestamp	The time and date the last action occurred.	date&time;yyyymm-dd&Thh:mm:ss	0	O	T
Last Load Attempt	The state of the last load attempt using the same states as used for the 'Program State' parameter.	0 to 12	0	O	R
Last Strategy Error	The text communications identifier of a module which caused a strategy download failure. It is set up if there is a strategy download failure and describes which module caused the strategy to fail.	Valid module code	Blank	O	E

Parameter	*Comments*	*Range*	*Default*	*Type*	*Text comms*
User Merge Option	v3.60 or greater. Specifies how users are processed when a strategy is downloaded - see **"User Module Merging (v3.60 or greater)" on page 449**.	0, 1, 2, 3 0 – Overwrite 1 – Merge and Maintain 2 – Merge and Overwrite 3 – Maintain existing users	1	N	u
Program State	The current state of the strategy file.	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17 0 = No Strategy 1 = Loading 2 = Running (normal state) 3 = Failed to Load 4 = Primed for Command 5 = Stopped 6 = No File 7 = File not Found 8 = Bad File 9 = File Version (invalid header) 10 = brIQs Exceeded 11 = External IO Exceeded 12 = Onboard IO Exceeded 13 = Feature Error 14 = Bad Library Licence 15 = Required Library Missing 16 = Bad Strategy Licence 17 = Cleared	0	O	P
Progress Percent	Progress of current download to the IQ4. 100% means file is fully loaded which should be quickly followed by a program state of 'Running'.	0 to 100	0	O	r

29 SCHEDULE OFFSET MODULES (Schedule Modules)

Module Identifier	Hn (where n is module number)
Default overview parameters	($, D)
Size in brIQs	19 (OTP) 21 (OSS)

Note: In versions of firmware earlier than 3.1 the Schedule Offset module was known as the Schedule module, and other products may still use this terminology. In these situations 'Schedule Offset' and 'Schedule' should be used interchangeably.

29.1 Functionality

Schedule Offset modules generate an analogue variable which changes with occupation state. It can be offset from the specified Time Schedule module's occupation times. There are two types of Schedule Offset module each performing a different calculation - see **"Schedule Offset Module Types" on page 451**.

Schedule Offset modules appear on the strategy page and can be sequence or event driven.

29.1.1 Manual Override

Schedule Offset modules can be manually overridden to so that the output uses the value set by the 'Manual Level' parameter by setting the 'Manual Select' parameter to '1'.

29.1.2 Alarm Bit Handling

The alarm bits associated with the selected input are transferred to be associated with the output.

29.1.3 Service Timing

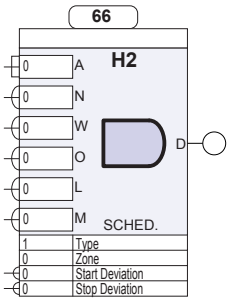
The selection of the relevant input according to the OSS or Time Schedule module's times and any change in the 'Manual Select' parameter is only acted upon once per minute. The transfer of data from the selected input to the output is serviced once per sequence table entry (or on a change in one of the inputs if event driven) i.e. if the selected input changes then the output will change within the sequence table cycle time.

29.2 Schedule Offset Module Types

There are two Schedule Offset module types specified using the 'Type' parameter:

Module Type	*Description*
Type 1 (OSS)	Type 1 (OSS) Schedule Offset modules are controlled by the optimum start and stop times calculated by the associated OSS module plus the Schedule Offset module's own start and stop deviation.
Type 2 (OTP)	Type 2 (OTP) Schedule Offset modules are controlled by the occupation times set in the associated Time Schedule module plus the Schedule Offset module's own start and stop deviation.

29.2.1 Type 1 (OSS)



Type 1 (OSS) Schedule Offset modules set output (D) to one of four values (Warmup Value, Occupied Value, Cooldown Value, and Non Occupied Value) depending on the occupation status of the connected OSS module and the Schedule Offset module's start and stop deviation. It calculates the following start and stop times:

Scheduled optimum start = OSS start time + Start Deviation
Scheduled OTP start = OTP start time + Start Deviation
Scheduled optimum stop = OSS stop time + Stop Deviation
Scheduled OTP stop = OTP stop time + Stop Deviation

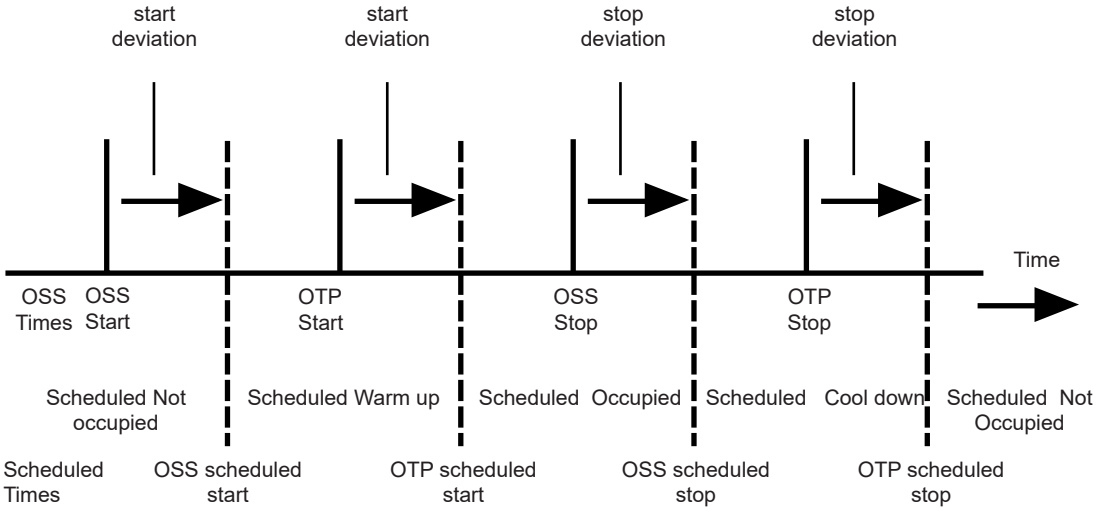
When time is between Scheduled OTP stop and Scheduled OSS start times the module's output will use the non occupied value and the 'State' parameter is set to 4 (Non-occupied).

When time is between Scheduled OSS start and Scheduled OTP start times the module's output will use the warmup value and the 'State' parameter is set to '2 (Warmup)'.

When time is between Scheduled OTP start and Scheduled OSS stop times the module's output will use the occupied value and the 'State' parameter is set to '1 (Occupied)'.

When time is between Scheduled OSS stop and Scheduled OTP stop times the module's output will use the cooldown value and the 'State' parameter is set to '3 (Cooldown)'.

This is illustrated in the diagram.



29.2.1.1 Manual Override

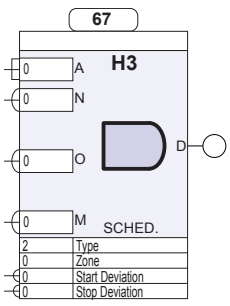
Supported - see **"Manual Override" on page 451**.

29.2.1.2 Type 1 (OSS) Schedule Offset Parameters

Type 1 (OSS) Schedule Offset modules include the parameters common to all Schedule Offset modules - see **"Schedule Offset Module Common Parameters" on page 453**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Cooldown Value	The output level used during the cooldown phase.	Full analogue	0	I	L
OSS Connection	The OSS module used by this schedule	1 to 100	0	N	Z
Warmup Value	The output level used during the warmup phase.	Full analogue	0	I	W

29.2.2 Type 2 (OTP)



Type 2 (OTP) Schedule Offset modules set output (D) to one of two values (Occupied Value, and Non Occupied Value) depending on the occupation status of the connected Time Schedule module and the Schedule Offset module's start and stop deviation. It calculates the start and stop time of the occupation period (i.e when the output uses the occupied value) as below:

Schedule start time = OTP start time + Start Deviation
Schedule stop time = OTP stop time + Stop Deviation

During this occupied period the 'State' parameter will be set to '1 (Occupied)'. Outside this period the modules's output is set to the non occupied value and the 'State' parameter will be set to 4 (Non-occupied).

This means that the output uses the occupied value during the occupation period of the connected Time Schedule module, and the non occupied value during the unoccupied period of the connected Time Schedule module subject to the application of the start and stop deviations.

29.2.2.1 Manual Override

Supported - see **"Manual Override" on page 451**.

29.2.2.2 Type 2 (OTP) Schedule Offset Module Parameters

Type 2 (OTP) Schedule Offset modules include the parameters common to all Schedule Offset modules - see **"Schedule Offset Module Common Parameters" on page 453**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Timezone Connection	The Time Schedule used by this Schedule Offset module.	1 to 100	0	N	Z

29.3 Schedule Offset Module Common Parameters

Schedule Offset modules include the following parameters that are common to most modules: 'Label', 'Page', 'Sequence Step', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Manual Level	The output level used when the 'Manual Select' parameter is set to '1'.	Full analogue	0	I	M
Manual Select	Specifies whether the output will be driven to the manual level. If set to '1' the Schedule Offset module will be forced into manual and the output level will be set to that specified by the 'Manual Level' parameter. If set to '0' Schedule Offset module will run in automatic mode.	0 or 1 0 = Automatic 1 = Manual	0	I	A
Non-Occupied Value	The output level used during the non-occupied phase.	Full analogue	0	I	N
Occupied Value	The output level used during occupation.	Full analogue	0	I	O
Output	The module's analogue output.	Full analogue	0	O	D
Start Deviation	The difference (in minutes) between the start time from the OSS or Time Schedule module, and the start time used by the Schedule Offset module. If the start deviation is positive, the schedule start is delayed.	-1440 to +1440	0	I	T
State	The current state of the module.	1, 2, 3, 4 1 = Occupied 2 = Warmup 3 = Cooldown 4 = Non-occupied	0	O	S
Stop Deviation	The difference (in minutes) between the stop time from the OSS or Time Schedule module, and the stop time used by the Schedule Offset module. If the stop deviation is positive, the schedule stop is delayed.	-1440 to +1440	0	I	P
Type	Schedule Offset module type	1 or 2 1 = OSS 2 = OTP	2	N	Y

29.4 Schedule Offset Module Connectives

The following parameters are available as connectives:

Destination

30 SENSOR MODULES

Module Identifier	Sn (where n is module number)
Default overview parameters	Analogue $,V,% Digital $,V
Size in brIQs	76

30.1 Functionality

Sensor modules allow analogue inputs to be scaled into engineering units and the values checked against alarm levels. There are also sensor modules which are used to monitor digital states. There are four types of sensor module - **"Sensor Module Types" on page 458**.

Sensor modules appear on the strategy page and can be either event driven or sequence table driven.

30.1.1 Alarms

Type 0 (Analogue) and Type 1 (Internal Analog) Sensor modules can generate four alarms.

Read
Out of Limits
Low
High

Each alarm has an associated clear alarm.

Note: Type 2 (Digital) and Type 3 (Internal Digital) Sensor modules do not generate alarms; digital status alarms can be generated by External or Internal Digital Input modules.

30.1.1.1 Read Alarm

The Read alarm (READ) is generated by Type 0 (Analogue) Sensor modules when communications between the processor and the input channel fail to operate, and its associated clear alarm is generated when the alarm clears. Type 1 (Internal Analog) Sensor modules pass the source's associated alarm status through from its source module to its destination. When the alarm condition clears, a Clear READ alarm (O/K) is generated.

The alarm generation is performed by an alarm sub-module which sets its 'Alarm State' parameter to '1' when the alarm condition exists for longer than the alarm delay. For details of alarm generation - see **"Item Alarm Generation" on page 77**.

30.1.1.2 Out of Limits Alarm

The Out of Limits alarm (OUTL) is generated by Type 0 (Analogue) Sensor modules when the Sensor module's output is outside the range defined by the upper and lower limits set up in the associated Sensor Type module. Type 1 (Internal Analog) Sensor modules pass the source's associated alarm status through from its source module to the destination. When the alarm condition clears, a clear Out of Limits alarm (COUT) is generated.

Note: The limits should be the calibration limits of the sensor and for a 4 to 20 mA output range sensor they would be the values of the parameter at 20 mA (upper) and 4 mA (lower). In this case, short-circuited sensor wires can be detected by a current of greater than 20 mA and an open circuit sensor by a current of less than 4 mA.

The Out of Limits alarm means that the sensor reading is invalid and downstream strategy modules may need to take fail action - see **"Analogue Alarm Status Bits" on page 78**.

The alarm generation is performed by an alarm sub-module which sets its 'Alarm State' parameter to '1' when the alarm condition exists for longer than the alarm delay. For details of alarm generation - see **"Item Alarm Generation" on page 77.**

30.1.1.3 Low Alarm

The Low alarm (LOW) is generated when the Sensor module's output is below the sensor low alarm level. This should be set to the limit required for the process in question (e.g. occupied space comfort level). When the alarm condition clears a Clear Low alarm (CLOW) is generated.

It can have both alarm hysteresis and the alarm delay applied to it. Once the Sensor module's value has risen above the low alarm level plus hysteresis, it must remain above that level for the low alarm delay before the clear alarm can be declared - see **"Sensor Alarm Hysteresis" on page 456**.

The alarm generation is performed by an alarm sub-module which sets its 'Alarm State' parameter to '1' when the alarm condition exists for longer than the alarm delay. For details of alarm generation - see **"Item Alarm Generation" on page 77**.

30.1.1.4 High Alarm

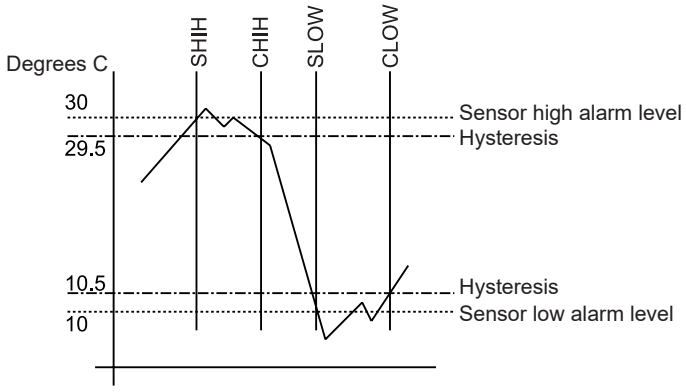
The High alarm (HIGH) is generated when the Sensor module's output is above the sensor high alarm level. This should be set to the limit required for the process in question (e.g. occupied space comfort level). When the alarm condition clears a clear High alarm (CHIH) is generated.

It can have both alarm hysteresis and the alarm delay applied to it. Once the Sensor module's value has fallen below the high alarm level alarm minus hysteresis, it must remain below that level for the high alarm delay before the clear alarm can be declared - see **"Sensor Alarm Hysteresis" on page 456**.

The alarm generation is performed by an alarm sub-module which sets its 'Alarm State' parameter to '1' when the alarm condition exists for longer than the alarm delay. For details of alarm generation - see **"Item Alarm Generation" on page 77**.

30.1.1.5 Sensor Alarm Hysteresis

Sensor alarm hysteresis is automatically applied to each Sensor module's high and low alarms. The purpose of the hysteresis is to avoid repeated alarms and clear alarms when the Sensor module's output is fluctuating about the alarm limit. Hysteresis prevents the clear alarm from being sent as soon as the parameter returns to within the allowed range, the value must return within the allowed range by the hysteresis amount before the clear alarm is sent.



The size of the hysteresis is calculated as a fixed percentage of the difference between the high and low alarm levels:

Hysteresis = 2.5% x (High alarm level - Low alarm level)

Alarm hysteresis and alarm delays can both be applied to an alarm, so it is possible to have for example, a hysteresis applied to the High alarm and a hysteresis and alarm delay applied to the Low alarm.

E.g:

if High alarm level = 30 deg C
if Low alarm level = 10 deg C
if Sensor High alarm delay = 0
if Sensor Low alarm delay = 10 seconds

Then hysteresis = 2.5/100 x (30 - 10) = 0.5 deg C

Consequently:

the High alarm will be set at 30 deg C
the Clear High alarm will occur when level <29.5 deg C
the Low alarm will be set at or below 10 deg C after a 10 second delay
the Clear Low alarm will occur when level has been >10.5 deg C for 10 seconds.

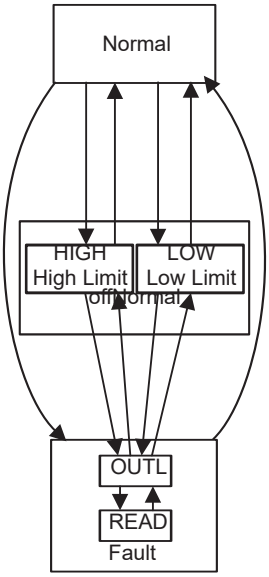
30.1.1.6 Sensor BACnet Alarm State Transitions

High and Low alarms: The High and Low alarms are both 'OffNormal' states and are treated separately so there are transitions To/From High Limit, and To/From Low Limit.

Read and Out of Limits alarms: The Read and Out of Limits alarms are both fault states and are treated the same, so the transitions To/From Fault apply to both Out of Limits and Read alarms. There are transitions within the Fault state, so a change from Out of Limits to Read will generate a From Fault To Fault transition.

It is also possible to go directly from 'Normal' to 'Out of Limits' which will create a 'From Normal to 'Fault' transition (in which case the return path should be 'From Fault To Normal').

The diagram shows possible sensor alarm state transitions with the downward paths being alarms, and the upward paths being clear alarms. The upward path should follow should follow the reverse of the downward path.



The BACnet alarm state transitions are explained in **"Alarm State Transitions" on page 65**.

Each Sensor module has a BACnet Event Time Stamps array that contains three values which are the date and time the last transition to into each of the BACnet alarm states (OffNormal, Fault, and Normal) occurred:

- n1 in the array indicates the time and date the module last transitioned into an 'offnormal' state.
- n2 in the array indicates the time and date the module last transitioned into a 'Fault' state.
- n3 in the array indicates the time and date the module last transitioned into the 'Normal' state.

30.1.2 COV Operation

COV mode enables the Sensor module to send its value to a list of subscribers using BACnet protocol when the value is changed by more than a specified amount.

30.1.3 Module Disable

Sensor modules can be disabled - see **"Disabling Modules" on page 29**. The 'Disable module' parameter will stop the sensor being serviced. This means that its internal states and output will remain at their last calculated value or status before the module was disabled.

30.1.4 Module Override

The Sensor module's input value can be overridden to a specified value defined by the 'Override Status' or 'Override Value' parameters by setting the 'Override Output' parameter to '1'.

30.1.5 Logging Sensor Modules

All Sensor modules can be logged by connecting the module's output to a Plot module. If a Sensor module is being logged will appear against it on the Sensor module summary page. If it is plotted at various intervals, each plot will be on a new line. Selecting will produce the graph from which the table of values may be selected if required. The plot module summary page will also have which operates in the same way. Trend supervisors, tools and displays are able to view the graphs and tables of values.

Note: The exact functionality of the graph display depends on the supervisor, tool or display being used.

30.1.6 Double Precision (v3.30 or greater)

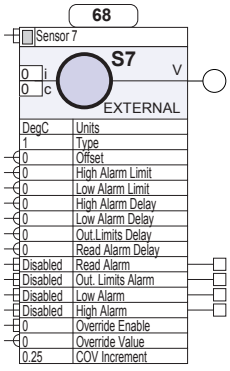
Type 1 (Internal Analog) Sensor modules measure the value they are connected to as a double precision value and output the value as a double precision value - see "2.1.2 Module Parameters" on page 23.

30.2 Sensor Module Types

There are four types of Sensor modules specified by the 'Input Type' parameter.

Input Type	*Description*
Type 0 (Analogue)	Provides an analogue sensor value from an analogue input channel.
Type 1 (Internal Analog)	Provides an analogue sensor value from a module's analogue output.
Type 2 (Digital)	Provides a digital sensor value from a digital input channel.
Type 3 (Internal Digital)	Provides a digital sensor value from a module's digital output.

30.2.1 Type 0 (Analogue)



Type 0 (Analogue) Sensor modules take the reading from a real sensor connected to the analogue input of the IQ4 or I/O module, scale it into engineering units, and check the value against alarm levels.

The IQ4's input channels can accept several different types of inputs (e.g. current 0 to 20 mA, voltage 0 to 10 V) - see the data sheet for inputs available for each IQ4 variant.

The way the input channel is connected and the strategy configured defines the signal conditioning circuit used. The conditioned signal is then converted to a digital value by an analogue to digital convertor this value is then scaled using parameters specified by the Sensor Type module associated with the Sensor module.

30.2.1.1 Sensor Offset

The 'Offset' parameter enables the engineer to compensate for sensor reading errors (e.g due to thermal conduction) by changing the sensor's value by a specified amount. The offset can be either positive or negative to compensate for a low or high value as required

30.2.1.2 Alarms

Supports High, Low, Read and Out of Limits alarms - see "Alarms" on page 455.

30.2.1.3 COV Operation

Supported - see "COV Operation" on page 457.

30.2.1.4 Module Disable

Supported - see "Module Disable" on page 457.

30.2.1.5 Module Override

Supported - see **"Module Override" on page 458**.

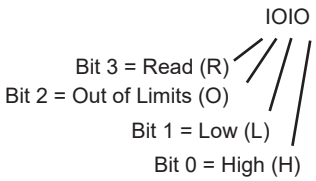
30.2.1.6 Double Precision (v3.30 or greater)

Not supported

30.2.1.7 Type 0 (Analogue) Sensor Module Parameters

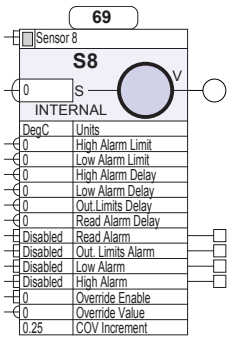
Type 0 (Analogue) Sensor modules include the parameters common to all Sensor modules - see **"Sensor Module Common Parameters" on page 466**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
ACK Transactions	v3.50 or greater. The status of the modules user alarm acknowledgement - see **"BACnet User Alarm Acknowledgments" on page 67**.	000 to 111	111	O	U
Alarm Delay (secs) (High)	The period (in seconds) for which the High alarm must be present before the high alarm is generated.	0 to 178200	0	I	h1(D)
Alarm Delay (secs) (Low)	The period (in seconds) for which the Low alarm must be present before the low alarm is generated.	0 to 178200	0	I	l1(D)
Alarm Delay (secs) (Out of Limits)	The period (in seconds) for which the Out of Limits alarm must be present before the out of limits alarm is generated.	0 to 178200	0	I	o1(D)
Alarm Delay (secs) (Read)	The period (in seconds) for which the Read alarm must be present before the read alarm is generated.	0 to 178200	0	I	r1(D)
Alarm Group (High)	The Alarm Group module to which the High alarm will be sent.	0 to 500	1	N	h1(G)
Alarm Group (Low)	The Alarm Group module to which the Low alarm will be sent.	0 to 500	1	N	l1(G)
Alarm Group (Out of Limits)	The Alarm Group module to which the Out of Limits alarm will be sent.	0 to 500	1	N	o1(G)
Alarm Group (Read)	The Alarm Group module to which the Read alarm will be sent.	0 to 500	1	N	r1(G)
Alarm Sent (High)	Indicates if the High alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	h1(K)
Alarm Sent (Low)	Indicates if the Low alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	l1(K)
Alarm Sent (Out of Limits)	Indicates if the Out of Limits alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	o1(K)
Alarm Sent (Read)	Indicates if the Read alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	r1(K)
Alarm State (High)	The state of the High alarm condition. It will be '1' if the High alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	h1(S)
Alarm State (Low)	The state of the Low alarm condition. It will be '1' if the Low alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	l1(S)

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm State (Out of Limits)	The state of the Out of Limits alarm condition. It will be '1' if the Out of Limits alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	o1(S)
Alarm State (Read)	The state of the Read alarm condition. It will be '1' if the Read alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	r1(S)
Alarm State	The alarm status of the four alarms as four digital bits. E.g.  The appropriate bit is set to '1' when the corresponding alarm is in an alarm state.	0000 to 1111	0000	O	M
Clear Alarm (High)	Enables/disables reporting of the High clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	h1(C)
Clear Alarm (Low)	Enables/disables reporting of the Low clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	l1(C)
Clear Alarm (Out of Limits)	Enables/disables reporting of the Out of Limits clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	o1(C)
Clear Alarm (Read)	Enables/disables reporting of the Read clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	r1(C)
Clear Alarm Delay (secs) (High)	v3.50 or greater. The period (in seconds) for which the High alarm must have cleared before the clear alarm is generated.	0 to 178200	0	I	h1(d)
Clear Alarm Delay (secs) (Low)	v3.50 or greater. The period (in seconds) for which the Low alarm must have cleared before the clear alarm is generated.	0 to 178200	0	I	l1(d)
Clear Alarm Delay (secs) (Out of Limits)	v3.50 or greater. The period (in seconds) for which the Out of Limits alarm must have cleared before the clear alarm is generated.	0 to 178200	0	I	o1(d)
Clear Alarm Delay (secs) (Read)	v3.50 or greater. The period (in seconds) for which the Read alarm must be present before the read alarm is generated, and for the alarm to have cleared before the clear alarm is generated.	0 to 178200	0	I	r1(d)
Enable (High)	Enables/disables reporting of the High alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	h1(E)
Enable (Low)	Enables/disables reporting of the Low alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	l1(E)
Enable (Out of Limits)	Enables/disables reporting of the Out of Limits alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	o1(E)
Enable (Read)	Enables/disables reporting of the Read alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	r1(E)

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Hardware Channel	The hardware analogue input channel which the Sensor module is monitoring. The range of channel numbers available depends on the IQ4 variant. '0' makes the source input (the 'Source(mV)' parameter) a constant; it is then possible to set the value of the 'Source(mV)' parameter from the IQ4's sensor web page.	0 to 32	0	N	c
Hardware Module	The address of the physical I/O module that contains the input channel providing the module's input - non zero values only applies to IQ4s that support external I/O modules, set to '0' for all other types of IQ4.	0 to 30 0 = IQ4 1 to 30 = I/O modules	0	N	i
High Alarm Level	The level above which a sensor High alarm is generated subject to alarm hysteresis and high alarm delay - see **"Sensor Alarm Hysteresis" on page 456**.	Full analogue	0	I	H
High	Indicates the High alarm state. Clicking on **High** on the web page will display the High alarm sub-module - see **"Alarms" on page 455**.	-	-	-	-
In Alarm	Indicates an alarm has been present for longer than its delay time, and the alarm is enabled.	0, 2, 3, 4, 5 0 = No alarm 2 = High alarm 3 = Low alarm 4 = Out of Limits alarm 5 = Read alarm	0	O	!
Low Alarm Level	The level below which the sensor Low alarm is generated subject to alarm hysteresis and low alarm delay - see **"Sensor Alarm Hysteresis" on page 456**.	Full analogue	0	I	L
Low	Indicates the low alarm state. Clicking on **Low** on the web page will display the low alarm sub-module - see **"Alarms" on page 455**.	-	-	-	-
Offset	The amount added to the Sensor module's value to account for sensor reading errors. It is in the same units as the Sensor module's value, and can be positive or negative.	Full analogue	0	I	O
Out of Limits	Indicates the Out of Limits alarm state. Clicking on **Out of Limits** on the web page will display the Out of Limits alarm sub-module - see **"Alarms" on page 455**.				
Read	Indicates the Read alarm state. Clicking on **Read** on the web page will display the Read alarm sub-module - see **"Alarms" on page 455**.				
Time in Alarm (secs) (High)	The elapsed time (seconds) since the High alarm state commenced.	Full analogue	0	O	h1(T)
Time in Alarm (secs) (Low)	The elapsed time (seconds) since the Low alarm state commenced.	Full analogue	0	O	l1(T)
Time in Alarm (secs) (Out of Limits)	The elapsed time (seconds) since the Out of limits alarm state commenced.	Full analogue	0	O	o1(T)
Time in Alarm (secs) (Read)	The elapsed time (seconds) since the Read alarm state commenced.	Full analogue	0	O	r1(T)

30.2.2 Type 1 (Internal Analog)



Type 1 (Internal Analog) Sensor modules allow digital values to be monitored. They take the value of a specified digital node, and put it into an analogue node.

Note: Type 1 (Internal Analog) Sensor modules do not require the scaling to be set up.

Note: Connecting a non-presentation module (e.g. Loop, or Schedule Offset module) output to an Type 1 (Internal Analog) Sensor module enables it to be labelled and seen on most supervisors whereas the source modules' outputs themselves are more difficult to see.

30.2.2.1 Sensor Offset

Not supported

30.2.2.2 Alarms

High, Low, Read and Out of Limits alarms - see **"Alarms" on page 455**.

30.2.2.3 COV Operation

Supported - see **"COV Operation" on page 457**.

30.2.2.4 Module Disable

Supported - see **"Module Disable" on page 457**.

30.2.2.5 Module Override

Supported - see **"Module Override" on page 458**.

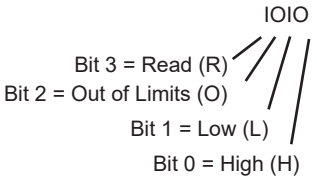
30.2.2.6 Double Precision (v3.30 or greater)

Supported - see **"30.1.6 Double Precision (v3.30 or greater)" on page 458**.

30.2.2.7 Type 1 (Internal Analog) Sensor Module Parameters

Type 1 (Internal Analog) Sensor modules include the parameters common to all Sensor modules - see **"Sensor Module Common Parameters" on page 466**, plus the following parameters:

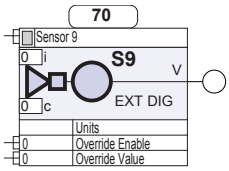
Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
ACK Transactions	v3.50 or greater. The status of the modules user alarm acknowledgement - see **"BACnet User Alarm Acknowledgments" on page 67**.	000 to 111	111	O	U
Alarm Delay (secs) (High)	The period (in seconds) for which the High alarm must be present before the High alarm is generated.	0 to 178200	0	I	h1(D)
Alarm Delay (secs) (Low)	The period (in seconds) for which the Low alarm must be present before the Low alarm is generated.	0 to 178200	0	I	l1(D)
Alarm Delay (secs) (Out of Limits)	The period (in seconds) for which the Out of Limits alarm must be present before the Out of Limits alarm is generated.	0 to 178200	0	I	o1(D)
Alarm Delay (secs) (Read)	The period (in seconds) for which the Read alarm must be present before the Read alarm is generated.	0 to 178200	0	I	r1(D)
Alarm Group (High)	The Alarm Group module to which the High alarm will be sent.	0 to 500	1	N	h1(G)

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Alarm Group (Low)	The Alarm Group module to which the Low alarm will be sent.	0 to 500	1	N	l1(G)
Alarm Group (Out of Limits)	The Alarm Group module to which the Out of Limits alarm will be sent.	0 to 500	1	N	o1(G)
Alarm Group (Read)	The Alarm Group module to which the Read alarm will be sent.	0 to 500	1	N	r1(G)
Alarm Sent (High)	Indicates if the High alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	h1(K)
Alarm Sent (Low)	Indicates if the Low alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	l1(K)
Alarm Sent (Out of Limits)	Indicates if the Out of Limits alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	o1(K)
Alarm Sent (Read)	Indicates if the Read alarm has been sent to the Alarm Group module.	0 or 1 1 = Alarm sent 0 = Alarm not sent	0	O	r1(K)
Alarm State (High)	The state of the High alarm condition. It will be '1' if the High alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	h1(S)
Alarm State (Low)	The state of the Low alarm condition. It will be '1' if the Low alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	l1(S)
Alarm State (Out of Limits)	The state of the Out of Limits alarm condition. It will be '1' if the Out of Limits alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	o1(S)
Alarm State (Read)	The state of the Read alarm condition. It will be '1' if the Read alarm condition has been present for longer than its alarm delay time. It takes no account of its alarm enable.	0 or 1 1 = Alarm 0 = No alarm	0	O	r1(S)
Alarm State	The alarm status of the four alarms as four digital bits. e.g.  The appropriate bit is set to '1' when the corresponding alarm is in an alarm state.	0000 to 1111	0000	O	M
Clear Alarm (High)	Enables/disables reporting of the High clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	h1(C)
Clear Alarm (Low)	Enables/disables reporting of the Low clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	l1(C)
Clear Alarm (Out of Limits)	Enables/disables reporting of the Out of Limits clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	o1(C)

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Clear Alarm (Read)	Enables/disables reporting of the Read clear alarm.	0 or 1 1 = Enabled 0 = Not enabled	1	N	r1(C)
Clear Alarm Delay (secs) (High)	The period (in seconds) for which the High alarm must have cleared before the clear alarm is generated.	0 to 178200	0	I	h1(D)
Clear Alarm Delay (secs) (Low)	The period (in seconds) for which the Low alarm must have cleared before the clear alarm is generated.	0 to 178200	0	I	l1(D)
Clear Alarm Delay (secs) (Out of Limits)	The period (in seconds) for which the Out of Limits alarm must have cleared before the clear alarm is generated.	0 to 178200	0	I	o1(D)
Clear Alarm Delay (secs) (Read)	The period (in seconds) for which the Read alarm must be present before the read alarm is generated, and for the alarm to have cleared before the clear alarm is generated.	0 to 178200	0	I	r1(D)
Enable (High)	Enables/disables reporting of the High alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	h1(E)
Enable (Low)	Enables/disables reporting of the Low alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	l1(E)
Enable (Out of Limits)	Enables/disables reporting of the Out of Limits alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	o1(E)
Enable (Read)	Enables/disables reporting of the Read alarm.	0 or 1 1 = Enabled 0 = Not enabled	0	I	r1(E)
High Alarm Level	The level above which a sensor High alarm is generated subject to alarm hysteresis and High Alarm Delay - see **"Sensor Alarm Hysteresis" on page 456**.	Full analogue	0	I	H
High	Indicates the read alarm state. Clicking on **High** on the web page will display the High alarm sub-module - see **"Alarms" on page 455**.				
In Alarm	Indicates an alarm has been present for longer than its delay time, and the alarm is enabled.	0, 2, 3, 4, 5 0 = No alarm 2 = High alarm 3 = Low alarm 4 = Out of Limits alarm 5 = Read alarm	0	O	!
Low Alarm Level	The level below which the sensor Low alarm is generated subject to alarm hysteresis and Low Alarm Delay - see **"Sensor Alarm Hysteresis" on page 456**.	Full analogue	0	I	L
Low	Indicates the Low alarm state. Clicking on **Low** on the web page will display the Low alarm sub-module - see **"Alarms" on page 455**.				
Out of Limits	Indicates the Out of Limits alarm state. Clicking on **Out of Limits** on the web page will display the out of limits alarm sub-module - see **"Alarms" on page 455**.				
Read	Indicates the read alarm state. Clicking on **Read** on the web page will display the Read alarm sub-module - see **"Alarms" on page 455**.				
Time in Alarm (secs) (High)	The elapsed time (seconds) since the High alarm state commenced.	Full analogue	0	O	h1(T)

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Time in Alarm (secs) (Low)	The elapsed time (seconds) since the Low alarm state commenced.	Full analogue	0	O	l1(T)
Time in Alarm (secs) (Out of Limits)	The elapsed time (seconds) since the Out of Limits alarm state commenced.	Full analogue	0	O	o1(T)
Time in Alarm (secs) (Read)	The elapsed time (seconds) since the Read alarm state commenced.	Full analogue	0	O	r1(T)

30.2.3 Type 2 (Digital)



Type 2 (Digital) Sensor modules convert the digital status of the input into an analogue value (of 0 or 1). They are intended for use with digital input channels or universal inputs set to read digital values that require logging.

30.2.3.1 Sensor Offset

Not supported

30.2.3.2 Alarms

Not supported

30.2.3.3 COV Operation

Supported - see **"COV Operation" on page 457**.

30.2.3.4 Module Disable

Supported - see **"Module Disable" on page 457**.

30.2.3.5 Module Override

Supported - see **"Module Override" on page 458**.

30.2.3.6 Double Precision (v3.30 or greater)

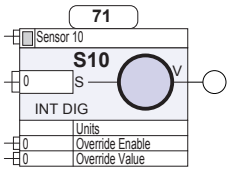
Not supported

30.2.3.7 Type 2 (Digital) Sensor Module Parameters

Type 2 (Digital) Sensor modules include the parameters common to all Sensor modules - see **"Sensor Module Common Parameters" on page 466** plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Hardware Channel	The hardware digital input channel which the sensor is monitoring. The range of channel numbers available depends on the type of IQ4. '0' makes the 'Source' parameter a constant; it is then possible to set the value of the 'Source' parameter from the IQ4's sensor web page.	0 to 32	0	N	c
Hardware Module	The address of the physical I/O module that contains the input channel providing the module's input - non zero value only applies to IQ4s that support external I/O modules, set to '0' for all other types of IQ4.	0 to 30 0 = IQ4 1 to 30 = I/O modules	0	N	i

30.2.4 Type 3 (Internal Digital)



Type 3 (Internal Digital) Sensor modules convert the digital status of the module input into an analogue value (of 0 or 1). They are intended for use with non-presentation module outputs that need to be logged and seen on a supervisor, display or tool. Connecting a non-presentation module's output to an Type 3 (Internal Digital) Sensor module enables it to be labelled and seen on most supervisors whereas the source modules' outputs themselves are more difficult to see.

30.2.4.1 Sensor Offset

Not supported

30.2.4.2 Alarms

Not supported

30.2.4.3 COV Operation

Supported - see **"COV Operation" on page 457**.

30.2.4.4 Module Disable

Supported - see **"Module Disable" on page 457**.

30.2.4.5 Module Override

Supported - see **"Module Override" on page 458**.

30.2.4.6 Double Precision (v3.30 or greater)

Not supported

30.2.4.7 Type 3 (Internal Digital) Sensor Module Parameters

Type 3 (Internal Digital) Sensor modules include the parameters common to all sensor modules - see **"Sensor Module Common Parameters" on page 466**. They do not include any additional parameters.

30.3 Sensor Module Common Parameters

Sensor modules include the following parameters that are common to most modules: 'Disable Module', 'Label', 'Page', 'Sequence Step', 'Sequence Table', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Note: The Label parameter can be used as an item attribute for text communications.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
BACnet Event Time Stamps	BACnet only parameters. An array of three parameters; the times of the last transitions into alarm states: n1: To OffNormal n2: To Fault n3: To Normal	date&time; yyyy-mm-dd&Thh:mm:ss	0000-00-00T00:00:00	O	n1 to n3

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
COV Increment	The amount by which the sensor value must change before a BACnet COV notification message is sent to those subscribers in the list of subscribers who have subscribed to this parameter. The 'COV Increment' parameter must not be set to '0'.	+Full analogue	0	N	C
Input Type	The type of module.	0, 1, 2, 3 0 = Type 0 (Analogue) 1 = Type 1 (Internal Analog) 2 = Type 2 (Digital) 3 = Type 3 (Internal Digital)	-	N	I
Override Output	Switches the module into override operation where input to the sensor's algorithm is set to the 'Override Value' or 'Override Status' parameter.	0 or 1 1 = Override 0 = No override	0	I	^
Override Value/ Status	The value to which the module is overridden when the 'Override Output' parameter is set to '1'.	Double precision Or 0, 1	0	I	v
Source (mV)/ Source	The value of the input channel. *Note: For Type 0 (Analogue) Sensor modules the value is NOT in mV even though the parameter is called 'Source (mV)'.*	Type 0 (Analogue) = 0 to 10000 Type 1 (Internal Analog) = Double precision Type 2 (Digital) and Type 3 (Internal Digital) = 0 or 1	0	I	S
Type	Sensor Type module is used to scale the sensor value and provide limits for Out of Limits alarms. (Type 0 (Analogue) and Type 1 (Internal Analog) only).	0 to 99	1	N	Y
Units	The units of the sensor. For compatibility with BACnet the units should be chosen from the list of BACnet units in **"BACnet Object/Property Codes and Units" on page 519**. (Type 0 (Analogue) only).	0 to 4 character string.	-	N	%
Value	The module's output. For analogue Sensor modules it is an analogue value, for digital Sensor modules it is an analogue value of '0' or '1'.	Type 0 (Analogue) = Full analogue Type 1 (Internal Analog) = Double precision Type 2 (Digital) = 0 or 1 Type 3 (Internal Digital) = 0 or 1	0	O	V

30.4 Sensor Module Connectives

The following parameter is available as a connective for all sensor types:

Value

For Type 0 (Analogue) and Type 1 (Internal Analog) following parameters are available as connectives:

Alarm State (High)
Alarm State (Low)
Alarm State (Out of Limits)
Alarm State (Read)

31 SENSOR TYPE MODULES

Module Identifier	Yn (where n is module number)
Default overview parameters	$
Size in brIQs	12

31.1 Functionality

Sensor Type modules are used to convert the signal received from an external sensor to the required engineering units for use in the strategy.

The electrical signal received from the sensor is read from the physical input and converted from its signal level to a digital number using an analogue to digital convertor, and then scaled according to the scaling specified in the Sensor Type module associated with the Sensor module reading the value to give the input value in engineering units (e.g. DegC). Each external sensor module uses a Sensor Type module to define how the signal is scaled.

The validity of the sensor reading is checked by comparing it with the upper and lower limits outside of which the sensor's operation is suspect; these limits are specified by the Sensor Type module's 'Upper Limit' and 'Lower Limit' parameters.

Sensor Type modules are configured for a particular type of sensor (e.g. Trend Thermistor Sensor). This means that if the same type of sensor is used by more than one sensor module the scaling information and alarm limits only need to be defined once (in the Sensor Type module) and referenced by each Sensor module monitoring a sensor of that type.

Sensor Type modules do not appear on the strategy page, and are accessed when the Type 0 (Analogue) Sensor module, or Type 19 (Lookup) Function module is serviced.

31.1.1 Sensor Type Library

IQSET is supplied with standard scaling tables (Sensor Library) for the sensors supplied by Trend which facilitate the configuring of sensor type modules for the Trend Sensors. The IQSET Strategy Library also has pre-configured sensor types for particular applications.

31.1.2 Input Type

Sensor Type modules have an 'Input Type' parameter which specifies the type of input being received from the actual sensor which can be set as follows:

Volts (V) - if sensor input channel linked for voltage (V)
Thermistor Volts (V) - if sensor input channel linked for thermistor (T)
Current (mA) - if sensor input channel linked for current (I)
Thermistor Ohms(kΩ) - if sensor input channel linked for thermistor (T)
Analogue - if used with lookup function module

Note: A Sensor module whose input channel is linked for thermistor may use either thermistor volts (V), or thermistor (kΩ). Normally the sensor characteristic will be given in kilohms, so thermistor (kΩ) can be used directly. However, thermistor volts input type is normally used in IQSET standard scaling tables because the voltage characteristic is more linear which gives greater accuracy in the linear interpolation described below.

The Sensor Type module uses this information in conjunction with a table of values that specify the value in engineering units of the sensor at a particular value of the sensors electrical signal to scale the sensor's signal.

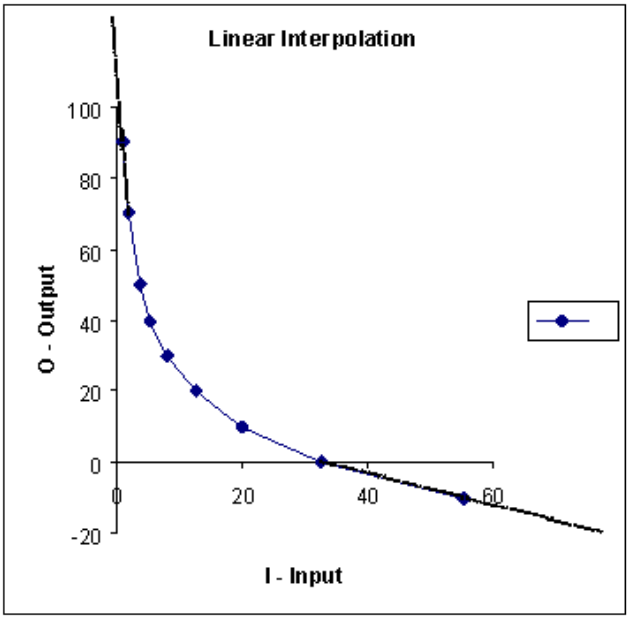
31.1.3 Scaling

The output value of the module is calculated by linearly interpolating between the relevant 2 points. If the input value falls above or below the table range then the output value is calculated from the extrapolated straight line between the first two points or the last two points respectively.

The example shows a 9 point table with the input ascending from 0.916 kΩ to 55.34 kΩ.

Entry	*Input (kohm)*	*Output (°C)*
1	0.916	90
2	1.751	70
3	3.602	50
4	5.326	40
5	8.058	30
6	12.49	20
7	19.9	10
8	32.66	0
9	55.34	-10

The graph shows the straight lines between the points, with the first and last straight lines extrapolated.



Note: The points in the table must be entered in ascending order of the input, and the 'Number of Points' parameter specified.

Note: The input resistance of the IQ4s' analogue voltage input configuration is about 9.4 kohm as opposed to 208 kohm in the equivalent IQ3 input so that the scaling for a potential divider input will be different; the selection is facilitated by IQSET - see **"Voltage Input Scaling (IQ4 vs IQ3)" on page 474**.

There are two types of scaling

Linear Sensor Scaling
Non-Linear Sensor Scaling

31.1.3.1 Linear Sensor Scaling

If the characteristics of the sensor providing the input is a straight line the scaling can be configured using only two points as other values can be extrapolated from the straight line graph. Such sensors are normally defined by their measurement range and their signal output.

Example 1: For a T/PI platinum resistance temperature sensor that has measurement range -10 to +40 DegC with an output signal of 4 to 20 mA the sensor type module should be set up as shown:

Parameter	*Value*
Input Type	2:Current

Parameter	*Value*
Number of Points	2
Input 1	4
Input 2	20
Output 1:	-10
Output 2:	+40
Upper Limit	+40
Lower Limit	-10

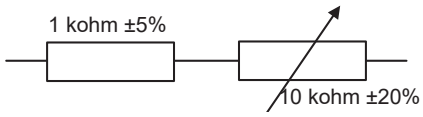
Example 2: For an AV/D air velocity sensor set to its defaults of measurement range: 0 to 10 m/s, and output signal: 0 to 10 V.

The Sensor Type module should be set up as shown:

Parameter	*Value*
Input Type	0:Volts
Number of Points	2
Input 1	0
Input 2	10
Output 1:	0
Output 2:	10
Upper Limit	10.1
Lower Limit	0

Note: The upper limit is chosen as 10.1 10 V is a valid measurement; this shows that the upper and lower limits can be outside the defined range, as the values outside the range are found by extrapolation.

Example 3: For a TB/TS/K ±20% a Knob module is often used to produce a -3 to +3 deg trim on the local space temperature setpoint.



Note: It is common practice for the lower limit signal to be 1 kohm for a potentiometer as then an open circuit condition can be detected by a value of 0 kohm which will give a value outside the working limits and hence an Out of Limits alarm condition.

Since great accuracy is not required the characteristic can be entered as temperature versus resistance and assume a straight line and the sensor type module should be set up as shown:

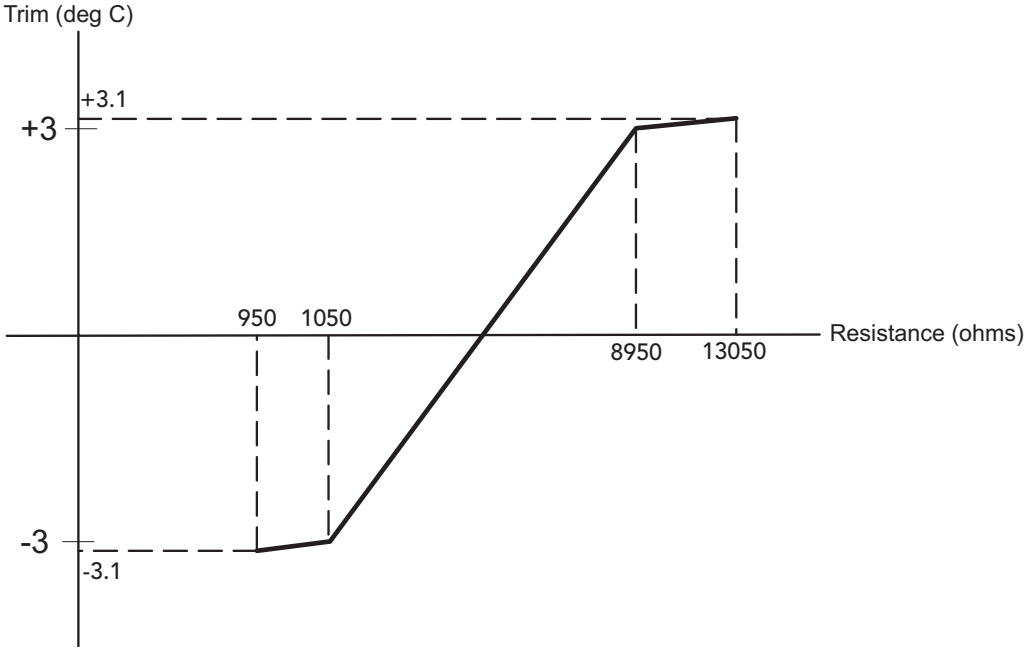
Parameter	*Value*
Input Type	3:Thermistor Ohms
Number of Points	2
Input 1	1
Input 2	11
Output 1:	-3
Output 2:	+3
Upper Limit	+3.1
Lower Limit	-3.1

However to guarantee a -3 to +3 trim, the resistance tolerances should be taken into account. This is done by taking the highest minimum value and the lowest maximum value and allocating the trim range across these values. Outside this range variations in resistance can be dealt with by tail-offs where the change in resistance produces little change in trim.

The tolerances give ranges of values as follows:

1 kohm ±5 % = 950 - 1050
10 kohm ±20 % = 8000 - 12000

This gives the highest minimum resistance (potentiometer value = 0) as 1050 ohm, and the lowest maximum resistance (950 + 8000) as 8950. These values correspond to the trim range -3 to +3. Outside this range we can allow the trim range to increase to -3.1 to + 3.1 as shown graphically below.



This gives a non-linear response specified by four points and the sensor type module should be set up as shown:

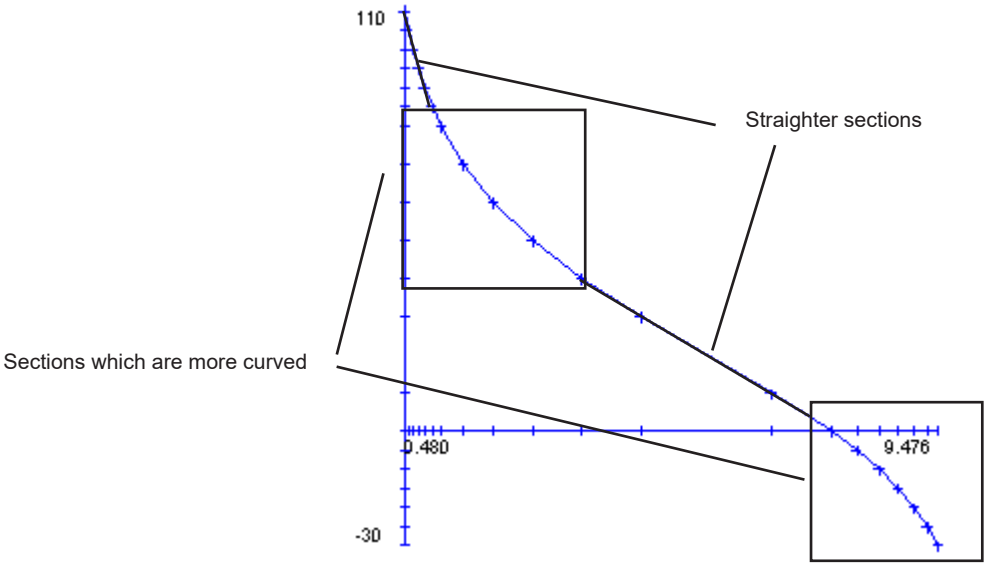
Parameter	***Value***
Input Type	3:Thermistor Ohms
Number of Points	4
Input 1	950
Input 2	1050
Input 3	8950
Input 4	13050
Output 1	-3.1
Output 2	-3
Output 3	+3
Output 4	+3.1
Upper Limit	+3.2 (corresponds to 18100 ohms)
Lower Limit	-3.2 (corresponds to 750 ohms)

31.1.3.2 Non-Linear Sensor Scaling

If the characteristics of the sensor providing the input is not a straight line, the scaling must be configured using more than two points to define the sensor characteristic graph. In practice it is normally sufficient to divide the range up into 19 equal sections (i.e. 20 points) and to enter these values. For characteristics which are just slightly non-linear, fewer points may be required.

This is because the input channel accuracy swamps the sensor scaling accuracy so that the niceties described below are not required. Hence the standard Trend thermistor scaling tables use the same input/output points but with different upper and lower limits.

Theoretically for maximum linearisation accuracy the sensor scaling points should be chosen so that there are more points covering the parts of the characteristic which are curved and fewer points over the straighter parts. This is because a straight characteristic can be defined by two points as described above.

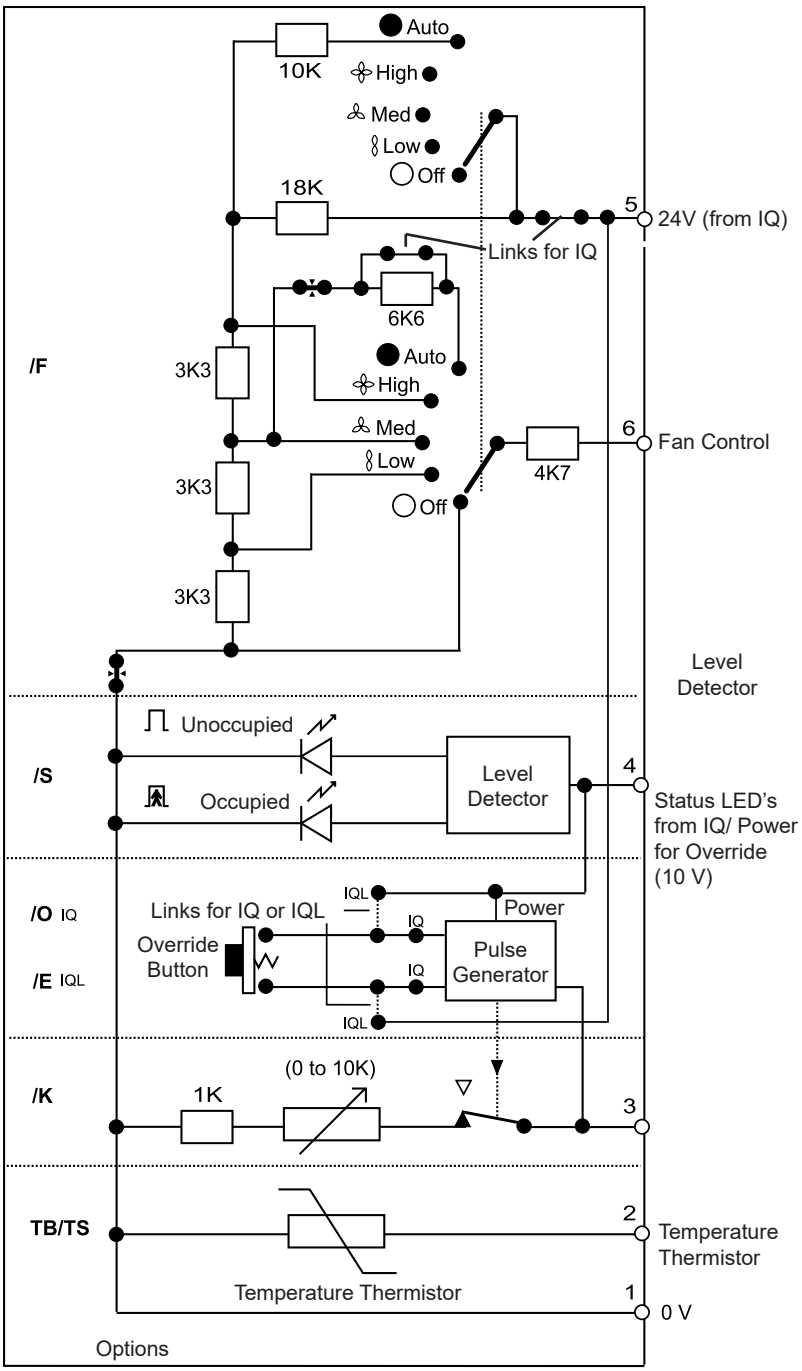


The above characteristic is for the standard Trend thermistor sensor over its full range (-30 to +110 deg C). It is shown for a Thermistor Volts input type; it would be possible to directly enter the temperature versus resistance values using the Thermistor Ohms(kΩ) input type, but this would give greater inaccuracies due to the linear interpolation used by the module.

The diagram shows the straight and curved sections and more points should be used on the curved sections, and the number of points should be spread over the range to be used, so for the Trend thermistor sensor if the sensor is only used indoors (e.g. TB/TS) the range can be reduced to 0 to 40 deg C, and the 20 points spread over that reduced range.

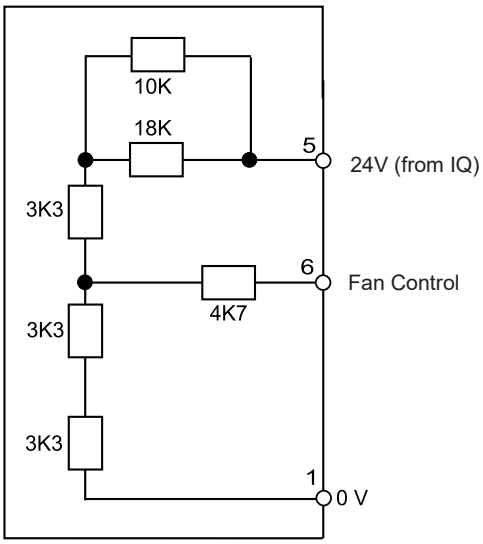
31.1.3.3 Voltage Input Scaling (IQ4 vs IQ3)

The input resistance of the IQ4 analogue voltage input is approximately 9.4 kohm as opposed to 208 kohm for the equivalent IQ3 input. This has an effect on the scaling for a voltage input which is produced by a relatively high impedance source (e.g. potential divider). We can use the TB/TS as an example.

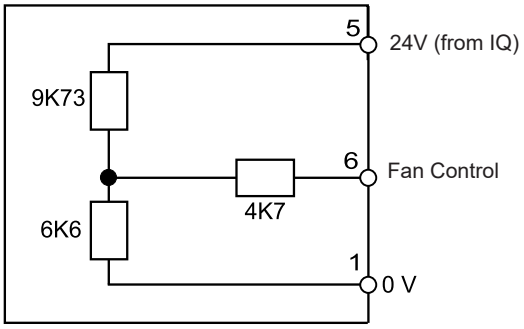


The top part of the diagram refers to the fan speed switch which uses a potential divider on the 24 Vdc IQ auxiliary supply to produce a voltage input to the controller.

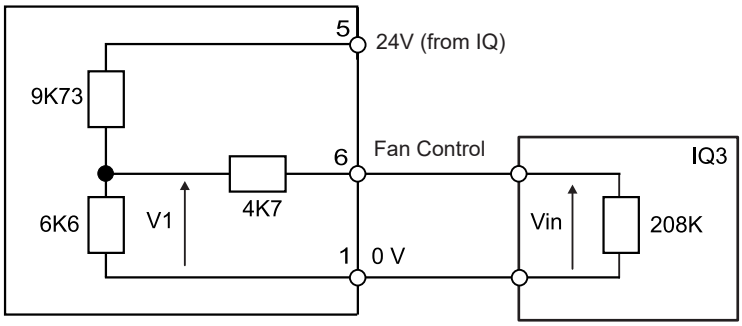
If we consider the AUTO setting, the circuit of the fan speed switch becomes:



The 10k//18k in series with 3k3 is equivalent to 9k73, and the two 3k3 resistors are equivalent to 6k6 so the circuit becomes:

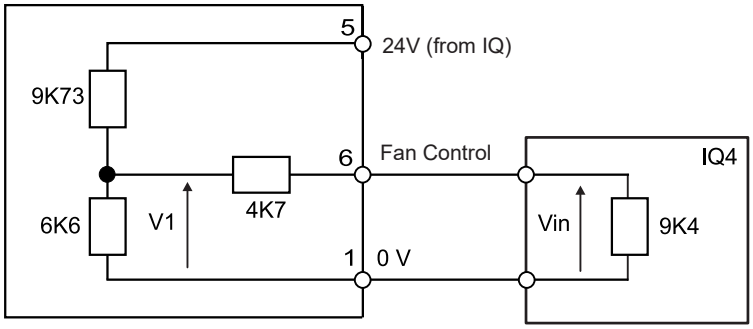


For IQ3 the input impedance is 208k so to a reasonable approximation the loading of the controller input and the effect of the series 4k7 can be ignored:



So Vin = V1 approximately, i.e. Vin = 24x(6k6/(9k73+6k6)) = 9.69 V . (If the loading of the controller and the 4k7 are taken into account Vin = 9.29 V).

For IQ4 the input impedance is 9k4 and this is relatively small compared to the impedance of the potential divider so its loading must be taken into account.

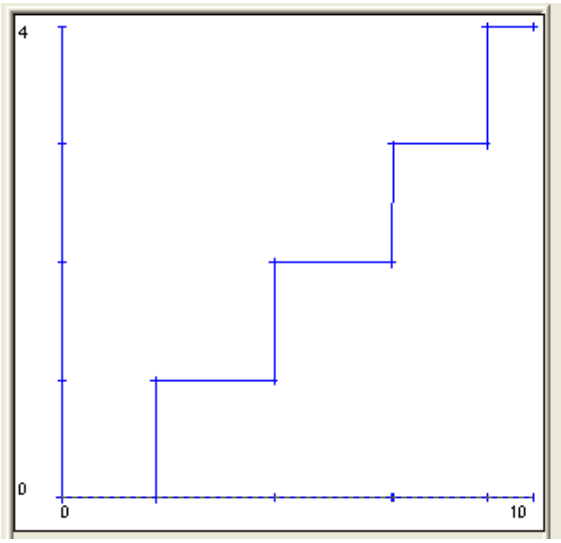


The combination of the 6k6 in parallel with the 4k7 and 9k4 in series gives an equivalent resistance of 4k49 which using 21 Vdc (typical) produces a V1 of 6.64 V. This is then divided by the 4k7 and 9k4 in series to produce a Vin of 4.43 V.

It can this be seen that a potential divider chain that gives 9.69 V approximately in an IQ3 gives 4.43 V in an IQ4.

IQSET will automatically convert the scaling for the TB/TS fan speed switch when converting an IQ3 strategy to IQ4.

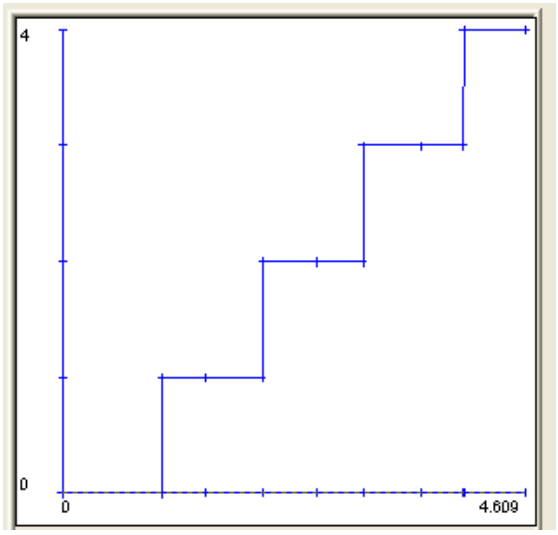
The Unique Sensor Reference ‘Fan KO enum’ is used to produce an enumeration of the voltage from the fan speed switch.



1	0	0
2	2.0	0
3	2.01	1
4	4.5	1
5	4.51	2
6	7.0	2
7	7.01	3
8	9.0	3
9	9.01	4
10	10	4

The above diagrams show the Fan KO enum sensor scaling for IQ3. The AUTO setting corresponds to the enumeration 4 which is between 9.01 V and 10 V to cater for the resistance tolerance.

For IQ4 the Fan KO enum has a different scaling; the AUTO setting (value) now lies between 4 and 4.608 V. The calculated value of 5.05 V will be higher than 4, so will produce an enumeration of 4.



1	0	0
2	0.99	0
3	1	1
4	1.42	1
5	1.99	1
6	2	2
7	2.539	2
8	2.99	2
9	3	3
10	3.562	3
11	3.99	3
12	4	4
13	4.609	4

31.1.3.4 Thermistor Volts Scaling

The ‘thermistor volts’ input type should be used when the input channel is set for thermistor (T). It has an advantage over the ‘thermistor ohms’ input type as the voltage characteristic is more linear which gives greater accuracy in the linear interpolation. However, whereas the resistance values required for thermistor ohms are normally given for a particular sensor, the voltage values have to be found. If they are not already available they can be calculated:

Although the voltage used to develop a voltage across the sensor is 3.3 V; the sensor type module for thermistor volts assumes the supply is 10 V. Similarly although the series resistor is about 12 kohms, the sensor type module assumes it to be 10 kohms. So the voltage values to be entered into the sensor type module should be calculated from the equation:

$$V = 10\left(\frac{R}{R + 10000}\right)$$

The IQ firmware will compensate for the actual input circuit values of 3.3 V and 12 kohms so that the output of the sensor module is correct.

Note: The early IQ1 series controllers had a 10 V supply with a 10 kohm series resistor and in order that, for a particular sensor, the same scaling table could be used in all IQ controllers, the firmware has been designed to compensate for the different voltage and resistance levels ever since.

31.1.4 Use with Type 19 (Lookup) Function Modules

Sensor Type modules are also used to provide a characteristic table for use with the Type 19 (Lookup) Function modules. To be used in this way, the ‘Input Type’ parameter must be set to ‘Analogue Value’. The Type 19 (Lookup) Function module has a ‘Scaling Type’ parameter which specifies the Sensor Type module to be used. If the Type 19 (Lookup) Function module specifies a Sensor Type module which does not have an input type of ‘Analogue Value’ the output is set to ‘0’.

31.2 Sensor Type Module Parameters

Sensor Type modules also include the following parameter common to most modules: 'Label' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Input Type	The type of scaling.	0, 1, 2, 3, 4 0 = Volts (V) 1 = Thermistor Volts (V) 2 = Current (mA) 3 = Thermistor Ohms (kohm) 4 = Analogue	0	N	I
Number of Points	Number of points used in the conversion table.	2 to 20	0	N	P
Upper Limit	The upper limit at which the Out of Limits alarm is detected by a sensor module. This will normally be the upper working limit of the sensor. Not used if used for Type 19 (Lookup) Function modules.	Full analogue	0	N	U
Lower Limit	The lower limit at which the Out of Limits alarm is detected by a sensor module. This will normally be the lower working limit of the sensor. Not used if used for Type 19 (Lookup) Function modules.	Full analogue	0	N	L
Inputs 1 to 20	The input value of point 'n' in the conversion table. (n = 1 to 20). Input 1 corresponds to output 1 etc. The input units correspond to the 'Input Type' parameter (e.g. if set to Input Type 0, Volts, the input units will be in volts). If used for reference by a Type 19 (Lookup) Function module these will be the analogue input values.	Full analogue	0	N	In
Outputs 1 to 20	The output value of point 'n' in the conversion table. (n = 1 to 20). Output 1 corresponds to input 1 etc. The output will be in the engineering units of the variable being measured by the sensor (e.g. DegC). If used for reference by a Type 19 (Lookup) Function module these will be the analogue output values.	Full analogue	0	N	On

32 SEQUENCE TABLE MODULE

Module Identifier	Q1
Default overview parameters	-
Size in brIQs	106

32.1 Functionality

The Sequence Table module defines the order the modules in the strategy are processed (serviced). The sequence table has an individual step for each module that is to be serviced. The number of steps is dependent on the type of IQ4. All modules in the table are processed in 1s.

Alternatively modules may be serviced on an event driven basis - see **"Event Driven Servicing" on page 28**.

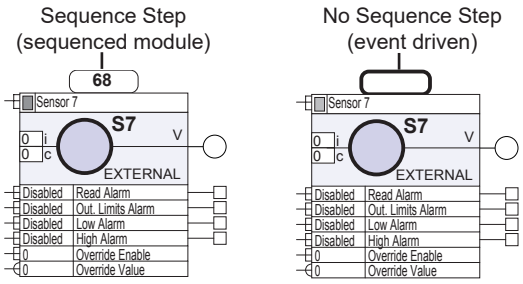
The following types of modules may be entered into the sequence table:

Sensor
Loop
Function
Driver
Schedule Offset
Logic
Internal Digital Input
Knob
Switch

Note: The Knob and Switch modules should be entered into the sequence table if their inputs are connected to other modules' outputs; if their inputs are set to a specific value (default), the Knob and Switch modules should be event driven (blank sequence step).

Each module is entered into the sequence table in the order of signal flow, so that the modules entered into the table with the lowest sequence step number are serviced first. This order can be critical - see **"Sequence Table Order" on page 27**.

In IQSET when a module is added to the strategy page the sequence step is added automatically. On the strategy diagram, the module has a sequence table entry number shown above it in an oval shape, If it is to be event driven the sequence step should be deleted which results in a blank oval above the strategy symbol.



32.1.1 Multiple Entries

A module may be serviced more frequently than once per cycle time by putting it in the sequence table more than once.

Note: Some modules should not be given multiple entries - see **"Multi-sequencing" on page 28**.

Multiple entries should be distributed through the sequence table and separated by an equal number of steps to divide the sequence table service time equally.

It is possible to put a whole sub strategy into the sequence table more than once to save on modules at the expense of sequence table steps - see **"Multiplexing" on page 30**.

Note: If the need is for a module to respond more rapidly to an input change it may be better to make it event driven - see **"Event Driven Servicing" on page 28**.

32.1.2 Disabled Modules

If a module is disabled - see **"Disabling Modules" on page 29** it is no longer serviced and its output(s) and internal states are maintained at their values prior to disabling.

32.1.3 Servicing Non-Sequence Table Modules

Some modules do not appear in the sequence table but are serviced under specific conditions:

Module	*Description*
Plots	Plot modules are checked once per second at the beginning of the sequence table cycle to see if they are due for an update.
Digital Input	The number of starts is passed to the module from the hardware 3 times per second (maximum).
Driver	The output is sent to the hardware on the sequence step or event and the hardware will drive the output for the required time e.g. for time proportional.
Time Schedule	Serviced once per minute.
OSS	Serviced once per minute.
Schedule Offset	The state is updated once per minute.

Note: This functionality is different to event driven strategy described in **"Event Driven Servicing" on page 28**.

32.2 Sequence Table Module Parameters

The Sequence Table module includes the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Label	The label for the sequence table.	Sequence Table 1		N	$
Last Sequence Step	Sequence table size.	IQ41x: 0 to 300 IQ422: 0 to 600 IQ4NC/../230: 0 to 600 IQ4NC/../24VAC: 0 to 600 IQ4NC/16/XNC/...: 0 to 600 IQ4NC/32/XNC/...: 0 to 640 IQ4E: 0 to 600 IQ4E/32: 0 to 600 IQ4E/64: 0 to 640 IQ4E/96: 0 to 960 IQ4E/96/XNC: 1280 IQ4E/128: 0 to 1280 IQ4E/160: 0 to 1600 IQ4E/192: 0 to 1920 IQ422/00/XNC: 0 to 1280 IQ422/12/XNC: 0 to 600		N	L
Sequence Step 1	The module serviced at sequence step 1.	Valid module code e.g. D1	-	N	S1
Timing	The sequence table has a fixed connection to the time module's 1 second pulse.	T(J)		N	G

33 STATES CATEGORY MODULES

Module Identifier	yn (where n is module number)
Default overview parameters	-
Size in brIQs	-

33.1 Functionality

v3.50 or greater firmware only. States Category modules enable a number of different states and associated text to be defined. They are used to specify the available states for modules that use multistate values, e.g. Type 1 (Multistate) time Schedule modules.

Each State Category module can have up to 20 different states each represented by State sub-module which defined the state, and the associated text.

They are not sequenced and do not appear on the strategy page.

33.2 States Category Module Parameters

The parameters of a States Category module are made up of those from the main States Category module (see **"Main States Category Module Parameters" on page 481**), and the parameters of the sub-modules that define the states (see **"States Sub-module Parameters" on page 481**).

33.2.1 Main States Category Module Parameters

The main States Category Modules include the following parameter common to most modules: 'Label' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameter:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Number of States	The number of states the module is to define.	1 to 20	1	N	N

33.2.2 States Sub-module Parameters

Each of the States Category module contains up to 20 State sub-modules (state 1 is State sub-module 1, state 2 is State sub-module 2 etc.) that define the text associated with each state which are identified in text comms as:

y<n>(T<n>)

Where y<n> specifies the State Category module and T<n> specifies the State sub-module. E.g. y2(T1) would read the 'State Text' parameter from the first State sub-module from State Category module 2.

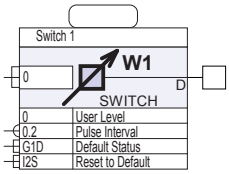
State sub-modules have the following parameters:

Parameter	*Description*	*Range (m)*	*Default*	*Type*	*Text Comms*
State Text	The text associated with the state.	String	-	N	T<n>

34 SWITCH MODULES

Module Identifier	Wn (where n is module number)
Default overview parameters	$, S
Size in brIQs	10

34.1 Functionality



Switch modules enable digital values to be changed by system users they can be viewed and adjusted on all Trend user interfaces. The function performed by any switch is determined by its connections within the strategy.

They appear on the strategy page and do not need to be sequenced unless the input is connected.

The Switch module's normal mode of operation is to set the 'Destination' parameter to the value of the 'Status' parameter. From version 3.40 the Switch module has five other modes of operation:

Reset to '0'
Reset to '1'
Self reset to '0'
Self reset to '1
Toggle Switch

The table below provides a summary of the operation:

Mode	***Parameter***			
	Default Status	***Reset to Default***	***Pulse Interval***	***Destination***
Normal (i.e stays at value it is set to)	0	0	0	'Source' Parameter
Reset to '0'	0	1	0	0
Reset to '1'	1	1	0	1
Self reset to '0'	0	1	>0	0 after pulse interval
Self reset to '1'	1	1	>0	1 after pulse interval
Toggle Switch	Inverse of output	0 to 1 to 0	>0	Default Status

34.1.1 Reset Operation

Reset operation (Reset to '0' and Reset to '1' modes) allows the Switch module's 'Destination' parameter to be set to a default value when required by the strategy. The 'Destination' parameter is set to the default status ('Default Status' parameter) when the 'Reset to Default' parameter is set to '1'.

34.1.2 Self Reset Operation

Self reset operation (Self reset to '0' and Self reset to '1' modes) allows the Switch module's 'Destination' parameter to automatically be set back to the default status a predetermined time after it is set to the non default state. This is done using the 'Pulse Interval' parameter. When the 'Pulse Interval' parameter is set to a value greater than '0' the module will automatically set its 'Destination' parameter back to the default status after it has been in the non-default state for the pulse period - see **"Pulse interval" on page 485**.

34.1.3 Toggle Switch Operation

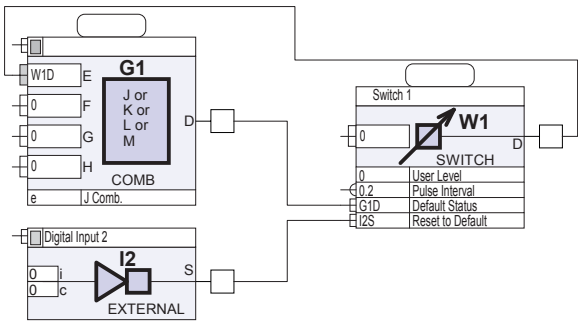
Toggle switch mode enables a Switch module to toggle its 'Destination' parameter when the 'Reset to Default' parameter changes from OFF to ON and back to OFF (0 to 1 to 0). This is useful if the switch is to be used to turn lights ON/OFF.

The 'Status' parameter should be left unconnected. The 'Default Status' parameter should be the inverse of the 'Destination' parameter, which may be provided by a **Type 1 (Combination)** Logic configured to invert its input.

When the ‘Reset to Default’ parameter changes from OFF to ON the value of the ‘Destination’ parameter is set to the value of the ‘Default Status’ parameter (which is the inverse of the ‘Destination’ parameter). This has the affect of turning the Switch module from OFF to ON or from ON to OFF. The ‘Default Status’ parameter will also be changed to the inverse of the ‘Destination’ parameter ready for the next change to the ‘Reset to Default’ parameter. No action is taken when the ‘Reset to Default’ parameter changes from OFF to ON.

Note: If the ‘Reset to Default’ parameter is changed again during the time specified by the ‘Pulse Period’ parameter the module’s output will not be changed. This prevent unwanted actions if the button is pressed more than once in quick succession, it also allows for a minimum ON time to be specified

An example of the strategy required for Toggle switch operation is shown below.



Note: This strategy is best configured as event driven. It does require feedback which is not generally recommended for event driven strategy.

Note: For Digital Input modules the number of transactions is limited to three per second so repeated operation that exceeds this rate will not have guaranteed results.

34.1.4 COV Operation

COV operation enables Switch modules to send a value to a list of subscribers using BACnet protocol when the value is changed.

34.1.5 Security

Switch modules have a configurable user level to protect the value from being changed by an unauthorised person.

If the level is left set to ‘0’ (default) any user except the ‘Guest’ user can adjust the status. If the level is non-zero the there must be a Type 0 User Module set up in the IQ4 with a user level of equal to or greater than the user level set in the Switch module; the user can then log on and make the change.

This user level is specific to a particular Switch module. Different Switch modules can have different levels. Changes to the other switch parameters are protected at a fixed level - see **“Fixed User Levels” on page 511**.

34.1.6 Switch Module Parameters

Switch modules include the following parameters that are common to most modules: ‘Label’, ‘Page’, ‘Sequence Step’, ‘Sequence Table’, ‘X co-ordinate’ and ‘Y co-ordinate’ - see **“Common IQ4 Module Parameters” on page 165**, plus the following parameters:

Note: The ‘Label’ parameter can be used as an item attribute for text communications - see **“Label Matching Item Selection” on page 286**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Destination	The output state.	0 or 1	0	O	D
Default Status	v3.40 or greater. The value to which the ‘Destination’ and ‘Status’ parameters are set when the ‘Reset to Default’ parameter is set to ‘1’.	0 or 1	0	I	F

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Pulse interval	v3.40 or greater. The minimum time (in seconds) after which the 'Destination' parameter is set back to the default status. When set to '0' the Destination' parameter will not be changed - see **"Pulse interval" on page 485**.	+Analogue	0	I	I
Reset to Default	v3.40 or greater. When set to '1' the 'Destination' and 'Status' parameters are set to the default status.	0 or 1 1 = reset	0	I	R
Status	The current state of the switch.	0 or 1	0	I	S
User Level	The PIN level of the user with which the user must be logged in, to change the switches value.	0 to 99	0	N	P

Pulse interval

If the Switch module is serviced during the pulse interval and the input status is different to the default status the pulse timer is restarted. When the input is set to the same as the default status the timer will not be restarted and will continue to count down unless the input changes again.

34.2 Switch Module Connectives

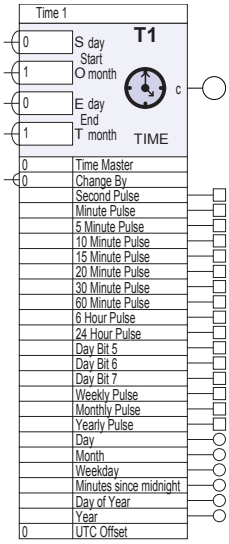
The following parameters are available as connectives:

Destination

35 TIME MODULE

Module Identifier	T1
Default overview parameters	$
Size in brIQs	38

35.1 Functionality



The Time module stores information about the current date and time which is supplied by the IQ4's Real Time Clock (RTC). It keeps track of the current time (as a 24 hour clock), the day of the week, and the date, and provides time pulses for use in the strategy.

It reads its value regularly from the IQ4's RTC. If the time is changed in the Time module e.g. by a timemaster, or the user the RTC's time is updated accordingly.

The Time module appears on the strategy page, is not sequenced, but is regularly serviced.

35.1.1 Power Failure

The IQ4's RTC is protected against power failure by a supercap which will support the RTC for approximately 8 days. When power is restored the Time module will update its time from the RTC. If if the supercap is discharged (i.e. IQ4 has lost its time), the IQ4 will request time synchronisation from an IQ3 or IQ4 set as a timemaster.

35.1.2 Timing Pulses

The timing pulse outputs all last for 1 second, and occur at the beginning of the period, (e.g. the 1 minute pulse will last for 1s at the beginning of every minute).

Note: The 1s pulse can only be used on event driven modules. It will not operate on sequenced servicing using the sequence table.

35.1.3 Time Synchronisation

An accurate time and date is required by the IQ4 so that it can be used for:

Logging of alarm messages.
Time and date stamping of alarm messages.
Use in the strategy.
Start time of plots.
BACnet time stamps

To ensure the accuracy of the date and time, the Time module can be synchronised in several ways:

By Timemaster
By supervisor or display panel
By web client using Time module web page
By an NTP Server (Timemaster only) - see **"Network Time Protocol Operation" on page 488**
Daylight saving time change (e.g. BST/GMT) - see **"Daylight Saving" on page 489**.

Note: The time is not set up by the strategy download.

35.1.3.1 Timemaster Operation

IQ4s can provide the timemaster function on the system. The timemaster function is enabled by setting the 'Time Master' parameter to '1'. It enables time and date synchronisation and daylight saving time changes of all IQ controllers from a single source. Controllers that are updated by the timemaster are known as time followers.

The timemaster will control the time and date synchronisation of all IQ controllers for an entire system (with the exception of auto-dialled sites). It can synchronise remote LANs across the internetwork. The timemaster may be placed at any address on any LAN. However, address 11 is recommended as IQLs request the time from address 11.

The timemaster will send a time and date update message to all devices on all LANs (except autodialled) at the following times:

00:05 hours each day
when it receives a request for time and date synchronisation from any LAN (not autodialled).
after any edit of its own Time module time and date
after any NTP changes
after effecting a daylight saving time change (e.g. BST/GMT)

In addition to the time and date, a timemaster will transmit UTC offset, daylight savings applied status, and daylight savings offset. These will enable the individual time follower controllers to respond to BACnet requests for the controller's UTC time.

On a combined system with IQ3, IQ4s, IQecos, IQLs, and earlier IQs (IQ1s, IQ2s) the timemaster must be an IQ3 or IQ4 (as an IQ2 Timemaster would not respond to an IQ3 or IQ4's request for synchronisation).

Note: The time synchronisation message will be rejected by a controller which is has security enabled unless there is a valid PIN in the synchronisation message. Therefore, if the controller to be synchronised has security enabled, the User 1 module in the timemaster must be set up with a PIN that matches one of level 80 or greater in the controller to be synchronised.

Note: A timemaster is necessary for each section of the system separated from other sections by auto-dialled links. However, there should only be one Timemaster on the system (excluding autodialled sections)

Note: When using an IQ212/BOX/ATM as the timemaster the IQ4's 'Time Master' parameter must be set to '0' as the ATM is unable change this itself (it is able to for IQ1 and 2 controllers). The ATM cannot respond to a request for synchronisation by an IQ4, but it sends out an hourly time synchronisation message, and hence in the worst case an IQ4 will not become synchronised until one hour after power up.

Note: Pre 1989 IQ1s cannot be synchronised by an IQ3 or IQ4.

35.1.3.1.1 Network Time Protocol Operation

IQ4s with v3.30 firmware or greater have Network Time Protocol (NTP) capability. This enables the IQ4 to obtain its time from an NTP server.

NTP operation is only available if the IQ4 is the timemaster ('Time Master' parameter set to '1'). In order to use the NTP functionality the 'Time Source' parameter must be set to '1' (NTP) and the 'Time Server Address Source' and 'Time Server Address' parameters set to specify the address of the NTP server. The addresses can be obtained from the DCHP server, or specified manually in the 'Time Server Address' parameter. Trend provide an NTP server at '0.trendcontrols.pool.ntp.org' which can be used. If the NTP functionality is to be used a time zone must be selected using the 'Time zone' parameter.

The NTP server will only sync the IQ4s time if the IQ4's time and the server's time are within 49 days of each other. When downloading the strategy you should ensure that the option to set the controller's time is selected.

Note: There must be an IP connection between the IQ4 and the NTP Server.

Note: The IQ4's NTP functionality does not support authenticated NTP servers.

35.1.3.2 Timekeeper Operation

The IQ4 cannot fulfil the role of timekeeper. This role was required for IQ90s; they should be supported by an IQ2 or IQ1 timekeeper at address 11 on their own LAN. Timekeeper controllers can be synchronised by the timemaster so there is still the one central timemaster.

The IQLs also have a timekeeper which is normally at address 11 on their own LAN; they will request time synchronisation from an IQ at address 11 on a specified LAN.

35.1.4 Daylight Saving

The IQ4's daylight saving feature allows the time to be adjusted to account for daylight saving time changes. The IQ4's time can be changed by a specified amount on a specified date and then changed back again at a later date.

While the time has been changed due to the daylight saving function, the 'Daylight Savings Applied' parameter is set to '1'.

The time changes at 01:00 when entering DST and at the time which means the new time becomes 01:00 when exiting DST. E.g.: If Hours Change = 1:

DST Entry 01:00 -> 02:00
DST Exit 02:00 -> 01:00

The dates between which the daylight saving occurs, and the amount of the change is determined by the 'Daylight Start Day', 'Daylight Start Month', 'Daylight Stop Day', 'Daylight Stop Month' and 'Daylight Hours Change' parameters. These parameters can be set up in two ways:

Automatically
Manually

Note: If a timemaster is being used to synchronise the time on a system, daylight saving should only be set up in the timemaster. The timemaster will synchronise a time follower controller's time and date, and set up its UTC offset, daylight savings applied status, and daylight savings offset appropriately. This enables the time follower to respond correctly to BACnet communications.

35.1.4.1 Automatic Daylight Saving Configuration

Automatic daylight saving configuration is only available on IQ4s with v3.30 firmware or greater. When the 'DST Source' parameter is set to '1' (Auto) the IQ4 will use the time zone specified in the 'Time zone' parameter to automatically calculate 'Daylight Start Day', 'Daylight Start Month', 'Daylight Stop Day', 'Daylight Stop Month' and 'Daylight Hours Change' parameters.

Note: If the strategy is writing directly to the 'Daylight Start Day', 'Daylight Start Month', 'Daylight Stop Day', and 'Daylight Stop Month' parameters i.e. strategy is calculating when daylight saving is to occur; manual daylight saving configuration must be used.

35.1.4.2 Manual Daylight Saving Configuration

When the 'DST Source' parameter is set to '0' (Manual Entry) the 'Daylight Start Day', 'Daylight Start Month', 'Daylight Stop Day', 'Daylight Stop Month' and 'Daylight Hours Change' parameters must be specified manually.

35.1.4.3 Disabling Daylight Saving

If daylight saving is not required, or the IQ4's time is being controlled by a timemaster daylight saving should be disabled by setting the 'Time zone' parameter to 'Not Used', the 'DST Source' parameter to 'Manual Entry' and the 'Daylight Hours Change' parameter set to '0'.

35.1.5 UTC Time

UTC time (Coordinated Universal Time) is the agreed international atomic time (with minor corrections for earth's rotation etc.). It is approximately the same as Greenwich Mean Time (GMT). BACnet protocol includes time synchronisation (date and time) using UTC time, so that it can synchronise internationally across the time zones.

The IQ4's 'UTC Offset' parameter can be used in conjunction with the 'Current Time' parameter to calculate the controller's UTC time:

UTC time = Current Time + UTC Offset

If Daylight Saving is in operation

UTC time = Current Time + UTC Offset - Daylight Savings Offset

Note: UTC time is not stored as a parameter in the IQ4; if it is required to provide UTC time it is necessary to use strategy to calculate it.

35.1.6 Differences between IQ2 and IQ4 Time Modules

- The IQ4 timemaster function is switched on by the 'Time Master' parameter; the IQ2 timemaster function is switched on by the Change By (Daylight Hours Change) value being non-zero.
- The IQ4 cannot fulfil the timekeeper function.
- The IQ2 'Change By' parameter can only be set to 0, 1, 2, whereas in IQ4 it can be set to 0, 1, 2, 3.
- IQ2s cannot respond to an IQ4's request for synchronisation, IQ4s can.
- The IQ4 has additional timing pulses: 1 s, 5 m, 10 m, 20 m, 6 h.
- The IQ4 Time module can appear on the strategy page, and connections can be made to the Start day, Start Month, End Day, End Month, and Change By.
- IQ4 can operate with UTC Time (BACnet), IQ2 can't.
- IQ4 can obtain its time from and NTP server IQ2 can't
- IQ4 can automatically calculated daylight saving changes based on the specified timezone

35.2 Time Module Parameters

The Time module includes the following parameters that are common to most modules: 'Label', 'Page', 'X co-ordinate' and 'Y co-ordinate '- see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
10 Minute Pulse	The pulse generated every 10 minutes.	0 or 1	0	O	n
15 Minute Pulse	The pulse generated every 15 minutes.	0 or 1	0	O	o
20 Minute Pulse	The pulse generated every 20 minutes.	0 or 1	0	O	a
24 Hour Pulse	The pulse generated every 24 hours.	0 or 1	0	O	t
30 Minute Pulse	The pulse generated every 30 minutes.	0 or 1	0	O	q
5 Minute Pulse	The pulse generated every 5 minutes.	0 or 1	0	O	l
6 Hour Pulse	The pulse generated every 6 hours.	0 or 1	0	O	s
60 Minute Pulse	The pulse generated every 60 minutes.	0 or 1	0	O	r
Current Time	The current time in hours and minutes.	00.00 to 23.59 hh.mm hh = 00 to 23 mm = 00 to 60	00.00	O	c
Day Bit 5	The least significant bit of the 3 bit expression of the weekday - see **"Day bits" on page 492**.	0 or 1	0	O	u
Day Bit 6	The middle bit of the three bit expression of the weekday - see **"Day bits" on page 492**.	0 or 1	0	O	v
Day Bit 7	The most significant bit of the three bit expression of the weekday - see **"Day bits" on page 492**.	0 or 1	0	O	w
Day of Year	The day of the year number.	1 to 366	0	O	e

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Daylight End Day	The day of the month at which daylight stops. When the 'DST Source' parameter is set to '1' (Auto) read only and calculated using the 'Time zone' parameter.	0 to 31 0 = No daylight saving	0	I	E
Daylight End Month	The month when daylight saving stops. When the 'DST Source' parameter is set to '1' (Auto) read only and calculated using the 'Time zone' parameter.	1 to 12	1	I	T
Daylight Hours Change	The length of time (hours) by which the time is changed for daylight saving. When the 'DST Source' parameter is set to '1' (Auto) read only and calculated using the 'Time zone' parameter.	0 to 3 0 = no change 1 = 1 hour 2 = 2 hours 3 = 3 hours	0	I	G
Daylight Savings Applied	Indicates whether the IQ4's time has been adjusted for daylight saving. In a system with a timemaster this parameter will be set by the timemaster.	0 or 1 1 = Time adjusted	0	O	F
Daylight Savings Offset (Mins)	The length of time (minutes) by which time is changed for daylight saving. In a system with a timemaster, the time follower controller's 'its value Offset (mins)' parameter will be set by the timemaster.	0 to 240	0	O	V
Daylight Start Day	The day of the month when daylight saving starts. When the 'DST Source' parameter is set to '1' (Auto) read only and calculated using the 'Time zone' parameter.	0 to 31 0 = No daylight saving	0	I	S
Daylight Start Month	The month when daylight starts. When the 'DST Source' parameter is set to '1' (Auto) read only and calculated using the 'Time zone' parameter.	1 to 12	1	I	O
Day	The current day of the month.	1 to 31	1	O	D
DST Source	Determines whether the IQ4 calculates daylight saving based on the value of the 'Time zone' parameter (Auto) or uses 'Daylight Start Day', 'Daylight Start Month', 'Daylight Stop Day', and 'Daylight Stop Month' parameters to determine when daylight saving starts and stops, and the 'Daylight Hours Change' parameter to determine the amount the time changes (Manual).	0 or 1 0 = Manual Entry 1= Auto	0	N	h
Hours	The hours part of the current time	0 to 23	0	O	H
Minute Pulse	The pulse generated every minute.	0 or 1	0	O	k
Minutes Past Midnight	The number of minutes elapsed since midnight.	0 to 1439	0	O	m
Minutes	The minutes part of the current time.	0 to 59	0	O	N
Monthly Pulse	The pulse generated every month.	0 or 1	0	O	d
Month	The current month.	1 to 12	1	O	M
Second Pulse	The pulse generated every second.	0 or 1	1	O	j
Seconds	The seconds part of the current time.	0 to 59	0	O	C
Time Master	Enables/disables the timemaster function - see page 488.	0 or 1 1 = Enabled 0 = Disabled	0	N	A
Time Server Address Source	IQ4 v3.30 or greater. Only available if the 'Time source' parameter is set to '1' (NTP). Specifies whether the address of the NTP server is to be obtained from the DHCP server, or entered manually in the 'Time Server Address' parameter.	0 or 1 1 = DHCP 0 = Manual	0	N	f
Time Server Address	IQ4 v3.30 or greater. The IP address or hostname of the NTP server. Read only when the 'Time Server Address Source' parameter is '1'.	Valid IP address	0.0.0.0	N	g

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Time Source	IQ4 v3.30 or greater. The source of the IQ4s time. Can be set to use the IQ4's RTC (internal) or an NTP server (NTP)	0, 1 0 = Internal 1 = NTP	0	N	I
Time zone	IQ4 v3.30 or greater. The time zone in which the IQ4 is located. Used to determine the 'Daylight Start Day', 'Daylight Start Month', 'Daylight Stop Day', 'Daylight Stop Month' and 'Daylight Hours Change' parameters when the 'DST Source' parameter is set to '1' (Auto), and the 'UTC Offset (Mins)' parameter. Must be specified if an NTP server is to be used.	List of time zones	Not used	N	Z
UTC Offset (Mins)	The length of time (minutes) the IQ4's local Time zone time differs from the UTC time - see **"UTC Time" on page 490**.	-780 to +780	0	N	U
Weekday	The current day of the week.	1, 2, 3, 4, 5, 6, 7 1 = Monday 2 = Tuesday 3 = Wednesday 4 = Thursday 5 = Friday 6 = Saturday 7 = Sunday	1	O	W
Weekly Pulse	The pulse generated every week.	0 or 1	1	O	u
Yearly Pulse	The pulse generated every year.	0 or 1	1	O	z
Year	The last two digits of the current year.	00 to 99	00	O	Y

Day bits

The day bits are three output bits that are used to indicate the day of the week. They are the Weekday parameter converted to a binary number. i.e.

Bit 7	Bit 6	Bit 5	
0	0	1	= Day 1 = Monday
0	1	0	= Day 2 = Tuesday
0	1	1	= Day 3 = Wednesday
1	0	0	= Day 4 = Thursday
1	0	1	= Day 5 = Friday
1	1	0	= Day 6 = Saturday
1	1	1	= Day 7 = Sunday

35.3 Time Module Connectives

The following parameters are available as connectives:

Day
Month
Weekday
Minutes Past Midnight
Day Bit 5
Day Bit 6
Day Bit 7
Second Pulse
Minute Pulse
5 Minute Pulse
10 Minute Pulse
15 Minute Pulse
20 Minute Pulse
30 Minute Pulse
60 Minute Pulse
6 Hour Pulse
24 Hour Pulse
Weekly Pulse
Monthly Pulse
Yearly Pulse
Current Time
Day of Year

36 TIME SCHEDULE MODULES (Time Zone Modules)

Module Identifier	Zn (where n is module number)
Default overview parameters	$, V
Size in brIQs	566 (includes 20 exceptions; further exceptions require 13 brIQs each)

Note: In versions of firmware earlier than 3.10 the Time Schedule module was known as the Time Zone module, and other products may still use this terminology. In these situations 'Time Schedule' and 'Time Zone' should be used interchangeably.

Important
Some features of Time Schedule modules in IQ4v3.50 firmware and greater are not compatible with Trend supervisor products.

Some features of Time Schedule modules in IQ4v3.50 firmware and greater are not compatible with Trend supervisor products (963, IQ™VISION, IQ™VIEW8, and IQ™VIEW4) - see **"Legacy Configuration" on page 503**. To maintain compatibility with these products, ensure that the B-BC Sched option is NOT selected when creating the controller in IQSET.

36.1 Functionality

Time Schedule modules are designed to set the weekly changes of occupation or value for the controller. The output ('Value' parameter) is set to the value specified by the schedule in the module.

v3.50 introduced two new types of Time Schedule modules, **Type 1 (Multistate)** and **Type 2 (Analogue)**. **Type 1 (Multistate)** allow scheduling of more complex program states than boolean Occupied/Non-occupied allows, such as Occupied, standby, non-occupied. **Type 2 (Analogue)** allow scheduling of analogue values - e.g. to set up light level profiles or setpoint profiles for periods in the day.

They can be allocated to an area of a building which has a profile which is different from other areas (e.g. factory, office, service bureau, and canteen). It may also be beneficial to allocate separate Time Schedule modules to areas of the building which normally have the same daily profile, but have different exceptions.

Time Schedule modules appear on the strategy page but are not sequenced as they are serviced once a minute.

36.1.1 When the Time Schedule Applies (Effective Period)

From v3.50 Time Schedule modules have an effective period that defines when the output of the module is controlled by the specified profile. In firmware versions prior to 3.50 the Time Schedule module applied all the time. Outside the effective period the output is set to the value of the 'Default Value' parameter. This is useful, if for example a Time Schedule module is to only apply during the school term time.

Note: If a Time Schedule module has been disabled ('Disable Module' parameter set to '1') the module will remain disabled even during the effective period.

Note: If a Time Schedule module is to be connected to a Calendar module (see **"Link to Time Schedule Modules" on page 191***) the Time Schedule module's effective period must be set to be ON during any period the Calendar module might be ON to ensure that the effective period is correctly applied.*

The effective period is specified by defining start date and end date of the period.

Start Date

The start date is defined using a combination of the 'Start Day', 'Start Month' and 'Start Year' parameters e.g:

Start Day = 1
Start Month = 1
Start Year = 2017

would set the effective period's start date to the 1st of January 2017.

End Date

The end date is defined using a combination of the ‘End Day’, ‘End Month’ and ‘End Year’ parameters e.g:

End Day = 31
End Month = 1
End Year = 2017

would set the effective period’s end date to the 31st of January 2017.

Wildcards

The Start Day’, ‘Start Month’, ‘Start Year’, ‘End Day’, ‘End Month’ and ‘ End Year’ parameters can be left as ‘0’ (Unspecified) which makes the Time Schedule always effective.

36.1.2 Calculating the Output

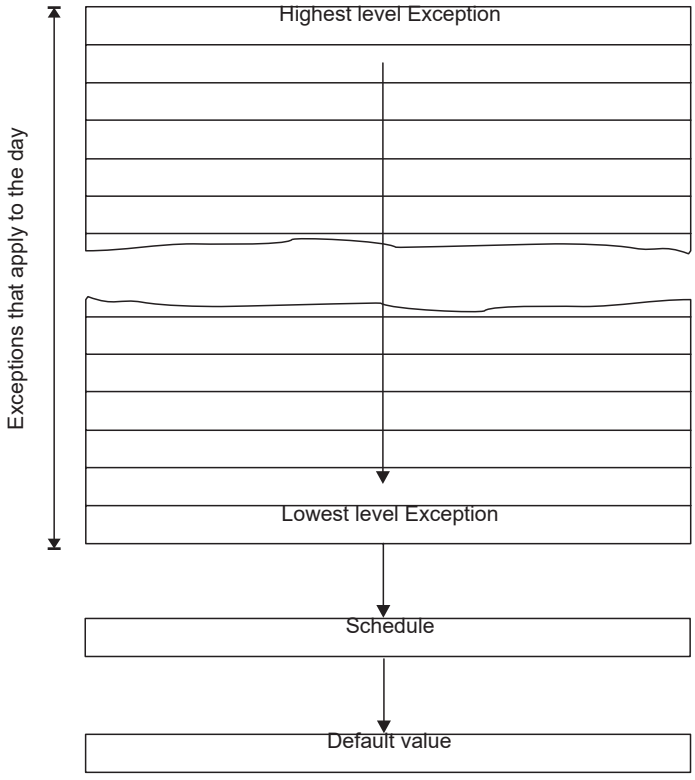
During the effective period Time Schedule modules set their output (‘Value’ parameter) depending on the result of combining the following:

Default Value: The default value defines the output of the module when a value is not specified by the normal weekly pattern or an exception - see **“Default Value” on page 498**.

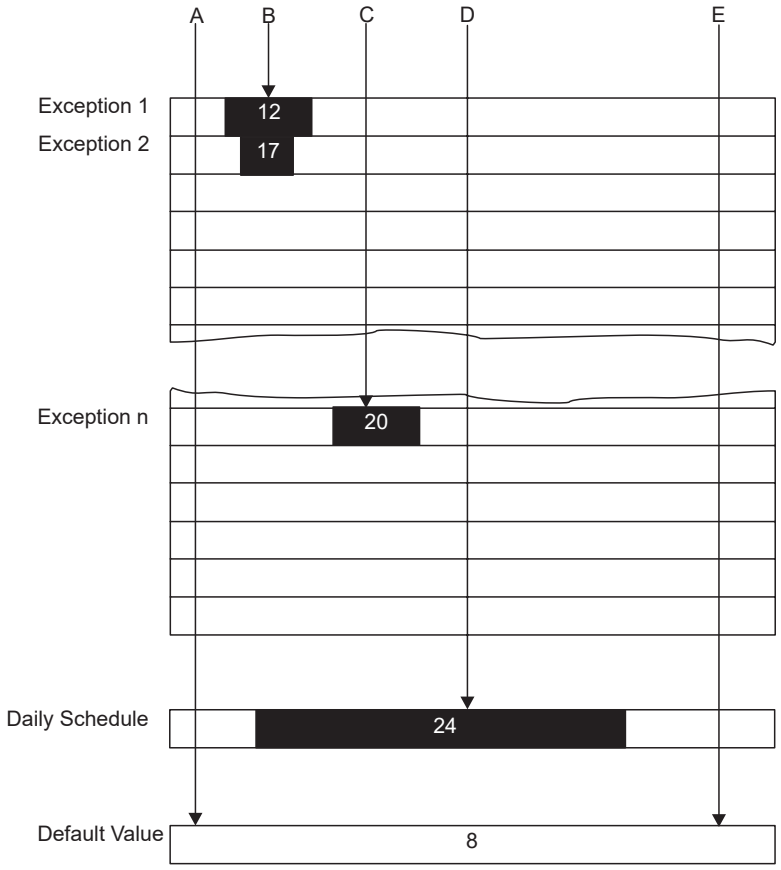
Schedule: The schedule defines the normal daily profile for each day of the week - see **“Schedule” on page 498**.

Exceptions: Exceptions specify a daily profile that can be applied to one or more days that is different to the profile defined by the schedule. This enables a different daily profile to be used on a particular day e.g. for a bank holiday - see **“Exceptions” on page 498**. If more than one exception applies to a day the exception with the highest priority is used - see **“Priority” on page 503**.

These three items are combined to determine the value. They can be considered as individual layers, with the highest priority exception at the top followed by other exceptions, then the schedule for that day and finally the default value. The module will take the value defined by the topmost layer. If a layer is null, at a particular time, then the value specified by the next layer at that time will be used, if that is null the next layer is used, then the next until if all layers are null the default value is used.



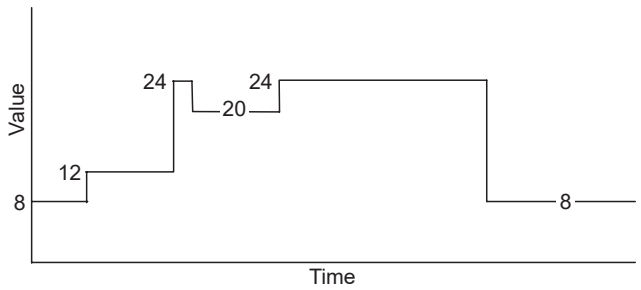
Example



In the example above the output value is calculated as:

Time	*Value Used*	*Reason*
A	Default value	No layers define a value (all are null)
B	Value specified by Exception 1	Exception 1 is the first layer to define a value during this period. *Note: The value of Exception 2 is not used as Exception 1 has a higher priority.*
C	Value specified by Exception n	Exception n is the first layer to define a value during this period. All higher layers are null.
D	Value specified by Schedule	The schedule is the first layer to define a value during this period. All higher layers are null.
E	Default value	No layers define a value (all are null)

The graph below shows the how the module's output (V) changes over time.



36.1.2.1 Default Value

v3.50 or greater. When the schedule and any applied exceptions do not specify a specific value (i.e. null) the module's output ('Value' parameter) is set to the value of the 'Default Value' parameter.

For example, if the default value is '0' and the schedule and applied exceptions do not specify a specific value the output is set to '0', and changing the default value will have the effect of changing the module's output.

Note: Ensure that the 'Default Value' parameter is configured to ensure the required operation when the schedule and any applied exceptions do not specify a specific value.

Note: Prior to v3.50 the Time Schedule's output was set to '0' outside when a specific value was not defined.

Legacy Configuration

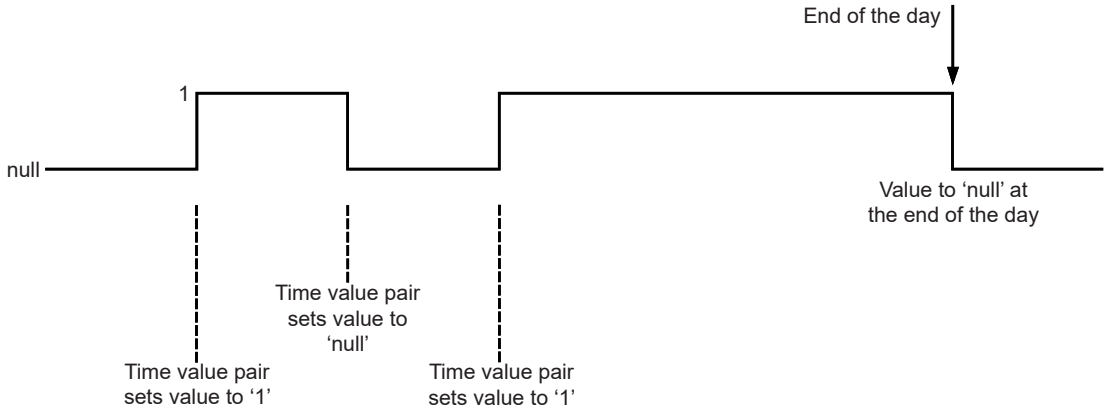
From v3.50 to maintain compatibility with Trend Supervisors the 'Default Value' parameter must be set to '0' - see **"Legacy Configuration" on page 503**.

36.1.2.2 Schedule

The Time Schedule module's normal weekly pattern is stored in a seven day record called the 'Schedule' which contains the daily profile for each day of the week (Monday to Sunday). Each day of the week can have a different profile e.g. 9 to 5 Monday to Friday and 9 to 12 on Saturday and Sunday.

When nothing is configured the entire day is set to 'null'. The daily profile for each day of the week is configured by adding 'Time value pairs' (max 20 per day) which define a time, and the value at that time which are used to define the daily profile. The value can either be a specific value or 'null'.

Once a value is set it will remain set until another time value pair is encountered, or the end of the day (last second of the last minute of the last hour) whichever occurs first. At the end of the day the value will be set to 'null' or the next day's profile will apply.



Note: When a value is not specified the profile will default to 'null', an empty daily profile defaults to 'null'.

36.1.2.3 Exceptions

Exceptions (Exception sub-modules) allow the Time Schedule module to use a different daily profile on specific day or a over a defined set of days or range of dates. Each Time Schedule module can have up to 50 exceptions.

An exception defines the profile for a single day and the days over which those times are used i.e. each day to which the exception apples will use the same profile.

Each exception can have a maximum of 20 time value pairs (20 change times in a day) which for Type 0 (Digital) Time Schedule modules allows up to 10 periods (10 ON, 10 OFF) and for Type 1 (Multistate) and Type 2 (Analogue) Time Schedule modules allows up to 20 value changes a day.

36.1.2.3.1 Defining the Days to Which an Exception Applies

There are six ways the days to which an exception applies can be specified:

Calendar Module
Date or Date Pattern
Range of Dates
Week and Day
Next
Every

Selection is made using a combination of the 'Date Option', 'Use' and 'Calendar Reference' parameters.

	Parameter		
	Date Option	***Use***	***Calendar Reference***
Calendar Module	Any	Any	Calendar Module
Next	0 = Legacy Configuration	Next	Blank
Every		Every	
Free - Not used		Free	
Date or Date Pattern	1 = Date or date pattern	Parameter ignored	
Range of Dates	2 = Range of dates		
Week and Day	3 = Week and Day		

Note: Calendar Module, Date or Date Pattern, Range of Dates and Week and Day require v3.50 firmware of greater.

If a Calendar module is specified then both the 'Use' and 'Date Option' parameters are ignored and the Calendar module specifies the days the exception applies to. When the 'Date Option' parameter is set to '0' (Legacy Configuration) the exception's use is specified by the 'Use' parameter. When the 'Date Option' parameter is set not set to '0' (Legacy Configuration) the 'Use' parameter is ignored and the exception's use specified by the 'Date Option' parameter.

Note: To maintain compatibility with Trend Supervisors the exception's use must be specified by the 'Use' parameter i.e. Next or Every.

36.1.2.3.1.1 Calendar Module

v3.50 or greater: Calendar modules can be used to determine the day(s) an Exception applies to, by specifying the Calendar module in the Exception's 'Calendar Reference' parameter.

In this configuration the Calendar module determines the date(s) the Exception applies. The times used are those specified by the exception and it is not possible to specify any dates. This enables easy control of the days different exceptions apply. E.g. a Calendar module, connected to exceptions in different Time Schedule modules, would contain the complete list of banks holidays. This would apply the connected exceptions, which could specify a different daily profile on those days.

Legacy Configuration

This method of defining when an exception applies should not be used if compatibility with Trend Supervisors is required - see **"Legacy Configuration" on page 503**.

36.1.2.3.1.2 Date or Date Pattern

v3.50 or greater: Date or Date Pattern enables an exception to be applied to a specific date e.g. 25th December 2017 or a repeating pattern e.g. 25th December every year, or the last day of every month.

The day(s) an Exception applies to are determined by the combination of the Exception sub-module's 'Start Day', 'Start Month' and 'Start Year' parameters e.g:

Start Day = 25
Start Month = 12
Start Year = 2017

would apply the Exception on 25th of December 2017.

Wildcards

Wildcards can be used for one or more of the 'Start Day', 'Start Month' and 'Start Year' parameters to enable a date pattern to be specified, e.g. to repeat a date every year. To specify the 25th of December every year the 'Start Day', 'Start Month' and 'Start Year' parameters would be set as shown:

Start Day = 25
Start Month = 12
Start Year = 0 (Unspecified)

Start Day wildcards: It is possible to specify any day, odd days, even days or the last day of the specified month by setting the 'Start Day' parameter appropriately:

0 = Unspecified (any day of specified month)
32 = Last day of specified month
33 = Every odd day of specified month
34 = Every even day of specified month

Start Month wildcards: It is possible to specify any month, odd months (January, March, May, July, September, November), or even months (February, April, June, August, October, December) of the specified year by setting the 'Start Month' parameter appropriately:

0 = Unspecified (any month of specified year)
13 = Every odd month
14 = Every even month

Start Year wildcards: It is possible to specify any year by setting the 'Start Year' parameter appropriately:

0 = Unspecified (any year)

Legacy Configuration

This method of defining when an exception applies should not be used if compatibility with Trend Supervisors is required - see **"Legacy Configuration" on page 503**.

36.1.2.3.1.3 Range of Dates

v3.50 or greater: Range of Dates enables an exception to be applied between two specified dates determined by the period(s) between the start date (determined by the combination of the Exception sub-module's 'Start Day', 'Start Month' and 'Start Year' parameters) and the end date (determined by the combination of the Exception Entry sub-module's 'End Day', 'End Month' and 'End Year' parameters) e.g:

Start Day = 1
Start Month = 12
Start Year = 2017
End Day = 31
End Month = 12
End Year = 2017

would apply the Exception for all of December 2017.

Wildcards

The 'Start Day', 'Start Month', 'Start Year', 'End Day', 'End Month' and 'End Year' parameters can be left as '0' (Unspecified) e.g. to enable the range to start on any date. To specify the range to start on any day and end on 31st of December 2017 the 'Start Day', 'Start Month', 'Start Year', 'End Day', 'End Month' and 'End Year' parameters would be set as shown:

Start Day = 0 (Unspecified)
Start Month = 0 (Unspecified)
Start Year = 0 (Unspecified)
End Day = 31
End Month = 12
End Year = 2017

Legacy Configuration

This method of defining when an exception applies use should not be used if compatibility with Trend Supervisors is required - see **"Legacy Configuration" on page 503**.

36.1.2.3.1.4 Week and Day

v3.50 or greater: Week and Day enables an exception to apply to a specific day of the specified week, specific week of the specified month, or a specific month using a combination of the 'Day of Week', 'Week of Month' and 'Start Month' parameter, e.g. every Tuesday in March could be specified by setting the parameters as shown:

Day of Week = 2 (Tuesday)
Week of Month = 0 (Unspecified)
Start Month = 3 (March)

Wildcards

Wildcards can be used for one or more of the 'Day of Week', 'Week of Month' and 'Start Month' parameters.

Day of Week wildcards: It is possible to specify any day of the week:

0 = Unspecified (any day of the week)

Week of Month wildcards: It is possible to specify any week or the last week of the setting the 'Week of Month' parameter appropriately:

0 = Unspecified (any week of the month)
6 = Last week of the month

Start Month wildcards: It is possible to specify any month or odd /even months by setting the 'Month of Year' parameter appropriately:

0 = Unspecified (any month)
13 = Every odd month
14 = Every even month

Legacy Configuration

This method of defining when an exception applies use should not be used if compatibility with Trend Supervisors is required - see **"Legacy Configuration" on page 503**.

36.1.2.3.1.5 Next

Next enables an exception to apply to the next occurrence of a specified day or range of days(s) specified as the period(s) between the start date (determined by the combination of the Exception sub-module's 'Start Day', and 'Start Month' parameters) and the end date (determined by the combination of the Exception sub-module's 'End Day', and 'End Month' parameters). The Exception will only apply the next time the date(s) occur only once and will be set to use Free once the exception has occurred. e.g:

Start Day = 1
Start Month = 12
End Day = 31
End Month = 12

would apply the Exception once for all of the next December and then set the exception to use Free.

Legacy Configuration

This method of defining when an exception applies can be used if compatibility with Trend Supervisors is required, the 'Date Option' parameter must be set to 0 (Legacy) - see **"Legacy Configuration" on page 503**.

36.1.2.3.1.6 Every

Every enables an exception to apply to every occurrence of a specified day or range of days(s) specified as the period(s) between the start date (determined by the combination of the Exception sub-module's 'Start Day', and 'Start Month' parameters) and the end date (determined by the combination of the Exception sub-module's 'End Day', and 'End Month' parameters). The Exception occurs every year. e.g:

Start Day = 1
Start Month = 12
End Day = 31
End Month = 12

would apply the Exception for all of the December every year.

Legacy Configuration

This method of defining when an exception applies can be used if compatibility with Trend Supervisors is required, the 'Date Option' parameter must be set to 0 (Legacy) - see **"Legacy Configuration" on page 503**.

36.1.2.3.1.7 Free

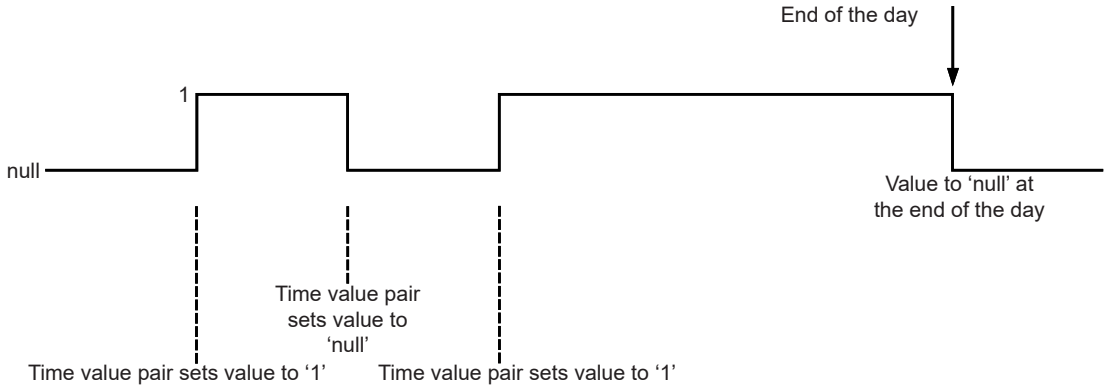
Free is not be used. Exceptions that use Next will be set to use Free once the data has passed.

36.1.2.3.2 Defining an Exception's Daily Profile

The Exception's daily profile specifies the output of the Time Schedule module on the day(s) to which it applies. The daily profile of an Exception is similar to the daily profile of an individual day in the schedule. All days to which the exception applies will use the same profile

When nothing is configured the entire day is set to 'null'. An Exception is configured by adding 'Time value pairs' (max 20 per day) which define a time, and the value at that time which are used to define the daily profile. The value can either be a specific value or 'null'.

When a time value pair is added the value is used from the specified time until the end of the day when the Exception will return to a 'null' state unless another time value pair changes the value. In the diagram below the value starts at null at the time of the first time value pair the value is set to '1'. At the time of the second time value pair it is set back to 'null'. At the time of the third time value pair it is set to '1' where it will stay until the end of the day when the value is set to 'null' until the next time value pair.



Legacy Configuration

From v3.50 to maintain compatibility with Trend Supervisors if the 'Date Option' parameter must be set to '0' (Legacy Configuration) - see **"Legacy Configuration" on page 503**. The IQ4 will check to see if there is a time value pair at the end of the day. If there isn't one it will automatically add one at the end of the day that set the value to '0'.

Note: If a time value pair exists at the end of the day, but it is set to value to something other than '0' it will be necessary to change it to '0' to maintain compatibility with Trend Supervisors.

36.1.2.3.3 Priority

If multiple exceptions have been specified that apply to the same day(s), i.e. the Exceptions overlap, the Exception with the highest priority (defined by the 'Priority' parameter) will be used. If overlapping exceptions have the same priority the Exception with the lowest number (instance) will be used.

36.1.3 Use with OSS and Schedule Offset Modules

Type 0 (Digital) Time Schedule modules can be used with an associated OSS module, and can be attached to Schedule Offset modules to provide inputs to control strategies.

36.1.4 Security

Changes to Time Schedule modules parameters (Normal week's Time Schedule Periods - Exceptions: Start and Stop dates, Priority, Use and Time Schedule Periods) are protected by the module's User Level. To make a change from a web page the user must be logged in with a user level equal to or greater than the Time Schedule module's User level. For a supervisor, tool or display to make a change it must supply a PIN for a Type 0 User Module with a user level equal to or greater than the Time Schedule module's User level If the level is left set to '0' (default) any user except the 'Guest' user can adjust the value.

If the level is non-zero there must be a Type 0 User Module set up in the IQ4 with a user level of equal to or greater than the user level set in the Time Schedule module; the user can then log on and make the change.

This user level is specific to a particular Time Schedule module. Different Time Schedule modules can have different levels. Changes to the other Time Schedule parameters are protected at a fixed level as defined in the Type 0 User Module - see **"Fixed User Levels" on page 511**.

36.1.5 Module Disable

If required Time Schedule modules can be disabled - see **"Disabling Modules" on page 29**. From v3.50 firmware a Calendar module can be connected to the Time Schedule module's 'Disable Module' parameter allowing it to be enabled/ disabled based on the schedule defined in the Calendar module - see **"Link to Time Schedule Modules" on page 191**.

36.1.6 Legacy Configuration

Some features of Time Schedule modules in IQ4v3.50 firmware or greater are not compatible with Trend supervisor products (963, IQVISION, IQVIEW8, and IQVIEW4). IQSET will maintain compatibility with these products if the B-BC Sched option is **NOT** selected when creating the controller.

To enable compatibility with these products it is necessary to configure any Time Schedule module as described below.

Time Schedule Modules

- Only use Type 0 (Digital) time Schedule Modules. Trend supervisors are not compatible with Type 1 (Multistate) and Type 2 (Analogue) Time Schedule modules.
- Set the 'Default Value' parameter to '0'.
- Configure the effective period so that it applies all the time.
- Do not link the 'Disable Module' parameter to a Calendar module

Exceptions

- Set the 'Date Option' parameter of any exceptions to '0' (Legacy Configuration). This will ensure that the 'Use' Parameter is used to specify the exception's use. It will also cause the IQ4 to check to see if there is a time value pair at the end of the day. If there isn't one it will automatically add one at the end of the day that sets the value to '0'.
- Do not specify a Calendar module in the Exception's 'Calendar Reference' parameter.
- Only set the exception's use to Next or Every.
- Ensure that any time value pair at the end of the day sets the value to '0'

Setting up Daily Profile

When creating the daily profile for day in the schedule, or an exception the following rules should be followed.

- The periods must be in sequence
- The periods must not overlap.
- Start and stop times must not be made equal.
- The value of a time value pair should not be set to ‘null’.
- The end of daily profile for the first day should be set to 24.00, and the start of daily profile for the next day should be set to 0.00 in order for the Time Schedule module to remain in uninterrupted occupation through the end of the day.

Note: When using a supervisor, tool, or display to make the changes, the product making the change will ensure that these rules are followed.

36.1.6.1 Upgrading a Pre v3.50 Strategy

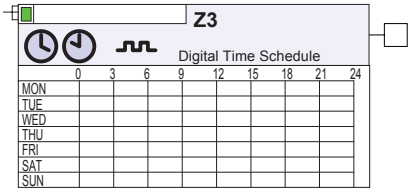
When upgrading a pre v3.50 strategy all the Time Schedule modules will be configured to enable legacy operation.

36.1.7 Time Schedule Module Types

There are several Time Schedule module types specified using the 'Data Type' parameter:

Type	*Description*
Type 0 (Digital)	Provides a digital output that can be used to switch plant and control loops ON or OFF at times entered by the user.
Type 1 (Multistate)	Provides Multistate output that can be used to control plant at times entered by the user.
Type 2 (Analogue)	Provides an analogue output that can be used to control plant at times entered by the user.

36.1.8 Type 0 (Digital)



Type 0 (Digital) Time Schedule modules work in the way described in "Functionality" on page 495. They provide a digital output that can be used to control plant at times entered by the user.

36.1.8.1 Connection to Calendar Modules

Supported - see "Link to Time Schedule Modules" on page 191 and "Calendar Module" on page 499.

36.1.8.2 Use with OSS and Schedule Offset Modules

Supported - see "Use with OSS and Schedule Offset Modules" on page 503.

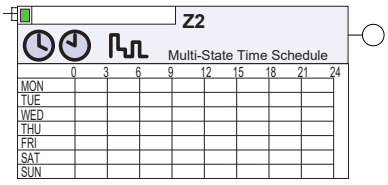
36.1.8.3 Module Disable

Supported - see "Link to Time Schedule Modules" on page 191 and "Calendar Module" on page 499.

36.1.8.4 Legacy Operation

Supported - see "Legacy Configuration" on page 503.

36.1.9 Type 1 (Multistate)



v3.50 or greater. Type 1 (Multistate) Time Schedule modules work in the way described in "Functionality" on page 495. They provide a multistate output that can be used to control plant at times entered by the user.

36.1.9.1 Connection to Calendar Modules

Supported - see "Link to Time Schedule Modules" on page 191 and "Calendar Module" on page 499.

36.1.9.2 Use with OSS and Schedule Offset Modules

Not supported

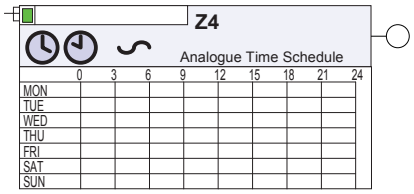
36.1.9.3 Module Disable

Supported - see "Module Disable" on page 503.

36.1.9.4 Legacy Operation

Not supported

36.1.10 Type 2 (Analogue)



v3.50 or greater. Type 2 (Analogue) Time Schedule modules work in the way described in **"Functionality" on page 495**. They provide an analogue output that can be used to control plant at times entered by the user.

36.1.10.1 Connection to Calendar Modules

Supported - see **"Link to Time Schedule Modules" on page 191** and **"Calendar Module" on page 499**.

36.1.10.2 Use with OSS and Schedule Offset Modules

Not supported

36.1.10.3 Module Disable

Supported - see **"Module Disable" on page 503**.

36.1.10.4 Legacy Operation

Not supported

36.2 Time Schedule Module Parameters

The parameters of a Time Schedule module are made up of those from the main Time Schedule module (see **"Main Time Schedule Modules Parameters" on page 506**), and the parameters of the sub-modules that define the schedule (see **"Day of Week Sub-module Parameters" on page 508** and **"Day of Week Time value Sub-module Parameters" on page 508**) and Exceptions (see **"Exception Sub-module Parameters" on page 508** and **"Exception Time value Sub-module Parameters" on page 510**).

36.2.1 Main Time Schedule Modules Parameters

Time Schedule modules also include the following parameters that are common to most modules: 'Disable Module', 'Label', 'Page', 'X co-ordinate' and 'Y co-ordinate' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Advanced Scheduling	Indicates if advanced scheduling is being used by the time Schedule - see **on page 507**.	0 or 1 0 = Legacy operation 1 = Advanced scheduling	0	N	A
BACnet only references	v3.50 or greater. String representing a list of object property references.	-	-	N	R
Data Type	Specifies the type of Time Schedule module.	0 to 2 0 = Digital 1 = Multistate 2 = Analogue	0	N	Y
Default Value	v3.50 or greater. The value output when the module is in the Null' state - see **"Default Value" on page 498**. For Type 1 (Multistate) Time Schedule modules set to '0' if the 'States Category module' parameter is not defined.	Type 0 0 or 1 0 = OFF 1 = ON Type 1 Multistate (Constrained by the 'States Category module' parameter). Type 2 Full analogue	Type 0 = 0 Type 1 = 1 Type 2 = 0.0	N	D

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Edit Timeout	v3.50 or greater. Internal use only	-	-	N	d
End Day	v3.50 or greater. The day of the month used for the end of the effective period - see **"When the Time Schedule Applies (Effective Period)" on page 495**.	0 to 31 0 = Unspecified	0	N	e
End Month	v3.50 or greater. The month of the year used for the end of the effective period - see **"When the Time Schedule Applies (Effective Period)" on page 495**.	0 to 12 0 = Unspecified	0	N	n
End Year	v3.50 or greater. The year used for the end of the effective period - see **"When the Time Schedule Applies (Effective Period)" on page 495**.	0, to 9999 0 = Unspecified	0	N	t
Exceptions 1	Not Used.	0	0	N	-
Exceptions	List of Exceptions	Index 1 to no max	-	N	C
references	v3.50 or greater. List of object property references.	String	" "	N	R
Start Day	v3.50 or greater. The day of the month used for the beginning of the effective period - see **"When the Time Schedule Applies (Effective Period)" on page 495**.	0 to 31 0 = Unspecified	0	N	a
Start Month	v3.50 or greater. The month of the year used for the beginning of the effective period - see **"When the Time Schedule Applies (Effective Period)" on page 495**.	0 to 12. 0 = Unspecified	0	N	m
Start Year	v3.50 or greater. The year used for the beginning of the effective period - see **"When the Time Schedule Applies (Effective Period)" on page 495**.	0 to 9999 0 = Unspecified	0	N	s
States Category module	v3.50 or greater. The States Category module used to provide the available states for multistate values.	Valid States Category module instance number.	First state	N	g
Number of states	v3.50 or greater. The number of states	1 to 20	1	N	N
User Level	The PIN level required to change the Time Schedule parameters.	0 to 100	0	N	P
Value	The current value of the currently active profile. For Type 1 (Multistate) Time Schedule modules the value is constrained by the States Category module. It is set to '0' if the 'States Category module' parameter is not defined.	Type 0 0, 1, null 0 = OFF 1 = ON Type 1 1 to 20 max. Type 2 Full analogue, null	0	O	V
Week Day Profiles	The schedule, an array of 7 day profiles.	Index 1 to 7	0	N	W

Advanced Scheduling Parameter

The 'Advanced Scheduling' parameter intended for use by tools, supervisors and displays to enable them to determine whether the Time Schedule module has been configured for legacy operation - see **"Legacy Configuration" on page 503**.

36.2.2 Day of Week Sub-module Parameters

The normal week consists of seven individual Day of week sub-modules which are identified in text comms as:

Z<n>(W<n>)

Where Z<n> specifies the Time Schedule module and W<n> specifies the day of the week sub-module (1, 2, 3, 4, 5, 6, 7 - Monday = 1). E.g. Z2(W1(*)) would read all the parameters from Time Schedule module 2's Monday's day of the week sub-module.

36.2.3 Day of Week Time value Sub-module Parameters

Each of the Day of week sub-module contains up to 20 Day of Week Time value sub-modules that define a time and the Time Schedule's value at that time which are identified in text comms as:

Z<n>(W<n>(P<n>))

Where Z<n>W<n> specifies the Time Schedule module, and Day of week sub-module - see **"Day of Week Sub-module Parameters" on page 508**, and P<n> specifies the Day of Week Time value sub-module. E.g. Z2(W1(P1(*)) would read all the parameters from the first Day of Week Time value sub-module from Time Schedule module 2's Monday's day of the week sub-module.

Day of Week Time value sub-module have the following parameters:

Parameter	*Description*	*Range (m)*	*Default*	*Type*	*Text Comms*
Time	The change time based on 24 hour clock.	00.00 to 24:00	00.00	N	T
Type	The data type of the value	0 to 2 0 = Digital 1 = Multistate 2 = Analogue	0	N	Y
Value	The value for the change time.	Type 0 0, 1, null 0 = OFF 1 = ON Type 1 Multistate (Constrained by the 'States Category module' parameter), null Type 2 Full analogue, null	0	N	V

36.2.4 Exception Sub-module Parameters

Each Time Schedule module can contain up to 50 Exception sub-modules that define when the IQ4 will operate a different daily profile to that defined by the normal week which are identified in text comms as:

Z<n>(C<n>)

Where Z<n> specifies the Time Schedule module and C<n> specifies the Exception sub-module. E.g. Z2(C1(*)) would read all the parameters from Time Schedule module 2 first Exception sub-module.

Exception sub-modules also include the following parameter common to most modules: 'Label' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Calendar	v3.50 or greater. The Calendar module that specifies the days the exception applies to - see Link to Time Schedule Modules" on page <?>).	Valid Calendar module instance number.	0	N	C

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Date Option	v3.50 or greater. The use of the exception - see **"Defining the Days to Which an Exception Applies" on page 499**. If set to '0' Legacy Configuration the 'Use' parameter is used to specify the exception's use.	0 to 3 0 = Legacy Configuration 1 = Date or date pattern 2 = Range of dates 3 = Week and Day	0	N	u
Day of Week	v3.50 or greater. The day of the week when the exception will apply. When set to 'Unspecified' any day is used. Only available when 'Date Option' is set to 'Week and Day'.	0 to 7 0 = Unspecified 1 = Monday	-	N	A
End Day	The day of the month when the exception stops - see **"Defining the Days to Which an Exception Applies" on page 499**. v3.50 or greater only available when 'Date Option' is set to 'Range of Dates'.	Pre v3.50 1 to 31 v3.50 or greater 0 to 31 0 = Unspecified	1	N	E
End Month	The month in which the exception stops - see **"Defining the Days to Which an Exception Applies" on page 499**. v3.50 or greater only available when 'Date Option' is set to 'Range of Dates'.	Pre v3.50 1 to 12 v3.50 or greater 0 to 12 0 = Unspecified 1 = January)	1	N	T
End Year	v3.50 or greater. The year in which the exception stops - see **"Defining the Days to Which an Exception Applies" on page 499**. When set to 'Unspecified' any year is used. Only available when 'Date Option' is set to 'Range of Dates'.	0 to 9999 0 = Unspecified	0	N	R
Priority	The priority of the exception. In case of exception overlap, the exception with highest priority level will be used.	Pre IQ4 v2.0 1 to 6 6 = Highest priority v2.0 or greater 1 to 16 16 = Highest priority	1	N	P
Start Day	The day of the month from which the exception is to apply - see **"Defining the Days to Which an Exception Applies" on page 499**. v3.50 or greater not used when 'Date Option' is set to 'Week and Day'.	Pre v3.50 1 to 31 v3.50 or greater 0 to 34. 0 = Unspecified 1 to 31 = Day of month 32 = Last day of month 33 = Every odd day 34 = Every even day	1	N	S
Start Month	The month from which the exception is to apply - see **"Defining the Days to Which an Exception Applies" on page 499**.	Pre v3.50 1 to 12. v3.50 or greater 0 to 14. 0 = Unspecified 1 to 12 = Month of year (1 = January) 13 = Every odd month 14 = Every even month	1	N	O
Start Year	v3.50 or greater. The year when the exception starts. When set to 'Unspecified' any year is used. Not used when 'Date Option' is set to 'Week and Day'.	0 to 9999 0 = Unspecified	-	N	Y

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Use	The use of the exception - see **"Defining the Days to Which an Exception Applies" on page 499**. From v3.50 this parameter is only used when the 'Date Option' parameter is set to '0' (Legacy Configuration).	0 to 2 0 = Next (i.e. once only) 1 = Every (yearly) 2 = Free (not used)	0	N	U
Week of Month	v3.50 or greater. The week of the month when the exception will apply. When set to 'Unspecified' any week is used. Only available when 'Date Option' is set to 'Week and Day'.	0 to 6 0 = Unspecified 1 = First (days 1 to 7) 2 = Second (days 8 to 14) 3 = Third (days 15 to 21) 4 = Forth (days 22 to 28) 5 = Fifth (days 29 to 31) 6 = Last	-	N	W

36.2.5 Exception Time value Sub-module Parameters

Each of the Exception sub-modules contains up to 20 Exception Time value sub-modules that define a time and the Time Schedule module's value at that time which are identified in text comms as:

Z<n>(C<n>(D1(P<n>)))

Where Z<n>W<n> specifies the Time Schedule module, and Exception sub-module - see **"Exception Sub-module Parameters" on page 508**, and D1(P<n> specifies the Exception Time value sub-module. E.g. Z2(C2(D1(P1(*)))) would read all the parameters from the first Time value sub-module of the second Exception sub-module.

Exception Time value sub-modules have the following parameters:

Parameter	*Description*	*Range (m)*	*Default*	*Type*	*Text Comms*
Time	The change time based on 24 hour clock.	00.00 to 24:00	00.00	N	T
Type	The data type of the value.	0 to 2 0 = Digital 1 = Multistate 2 = Analogue	0	N	Y
Value	The value for the change time.	Type 0 0, 1, null 0 = OFF 1 = ON Type 1 Multistate (Constrained by the 'States Category module' parameter). Type 2 Full analogue, null	0	N	V

36.3 Time Schedule Module Connectives

The following parameters are available as connectives:

Value

37 USER MODULES

Module Identifier	Un (where n is module number)
Default overview parameters	$, O
Size in brIQs	12

37.1 Functionality

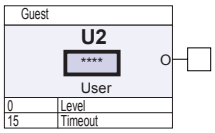
User modules provide security for the IQ4. There are two types of User module - see **"User Module Types" on page 511**.

37.2 User Module Types

There are two User module types specified using the 'Type' parameter:

Module Type	*Description*
Type 0 User Modules	Contains credentials and permissions for the IQ4's security for web pages and file transfer.
Type 1 CNC User Modules (v3.60 or greater)	Contains the credentials necessary for connection to a vCNC.

37.2.1 Type 0 User Modules



Type 0 User modules (also known as User modules) contain credentials and permissions for the IQ4's security. Security is enabled as soon as a Type 0 User Module is set up . If no Type 0 User Modules are set up, the IQ4's security is disabled. For details of how IQ4 security works see **"Security" on page 99**.

Each Type 0 User Module has a PIN, password, username and user level. The PIN is used to protect Trend network text communications with the IQ4 (e.g. supervisor changes, display panel changes, IC communications, Timemaster synchronisation, and IQSET Live Values/ Live Edit). The user name and password are used to protect access to web pages, strategy upload and strategy download and XML web services and file transfer.

If the network communications do not require to be protected the Type 0 User Modules can be have network access disabled. This will mean that the Type 0 User Module only protects the Web, XML and File upload Download, and NOT the Network communications. It will also not be possible to setup or change the PIN.

The user level determines what parameters a user can adjust, and what information they can see on the IQ4's web pages once a valid PIN or user name and password has been supplied.

Some changes require the user level to be at or above a certain fixed level before they can be made - see **"Fixed User Levels" on page 511**. Other changes have levels that are configurable (e.g. Knob value, Switch status, Time Schedule occupation parameters) - see **"Configurable User Levels" on page 512**. Because of this there may be more than one Type 0 User Module in an strategy enabling different users to have different sets of parameters that they may change.

Note: A user cannot change any Type 0 User Module parameters (including their own password) if they have a user level of less than '99'. There must also be at least one Type 0 User Module with a user level greater than or equal to 99.

Type 0 User Modules appear on the strategy page, but are not sequenced.

37.2.1.1 User Level

37.2.1.1.1 Fixed User Levels

The following lists the fixed user levels required for various communications and changes:

User Level	*Task*
<80	Restrict the user to only viewing Sensor, Digital Input, Knob, Switch, Driver, Time Schedule, Time, Plot, User modules, and Alarm enables, Alarm levels, Label, Value, and Units on module detail pages.

> = 81	View all module types and module parameters.
> = 90	Module changes (not user module, and value of Knob, Switch, and Time Schedule modules)
> = 90	Module changes (not user module, and value of Knob, Switch, and Time Schedule modules)
	File download
	File upload
> = 95	Text communications directed to one controller (i.e. directed text comms)
> = 99	Text communications directed to more than one controller on Local LAN or all LANs (i.e. local global or global global text comms), installing security certificates and changes to User module parameters.

37.2.1.1.2 Configurable User Levels

Knob, Switch, and Time Schedule modules have individually configurable user levels which can be set to any level between 0 and 99. These levels protect against changes to Knob value, Switch status, and Time Schedule occupation parameters (Normal week: Time Schedule Periods - Exceptions: Start and Stop dates, Priority, Use and Time Schedule Periods); changes to other parameters of these modules will require a fixed level of 95.

Note: If the user level is set to '0' (default), the values are unprotected and can be changed without logging on unless the user is logged on with a level of '-1' in which case the values cannot be changed.

Note: If a configurable user level is set up there must be a Type 0 User Module set up in the IQ4 with a user level equal to or greater than the configured user level; if this is not done the parameter will not be able to be changed (unless there are no Type 0 User Modules set up).

37.2.1.2 Changes Using Text Communications

See **"Changes Using Text Communications" on page 102**.

37.2.1.3 Strategy Upload and Download

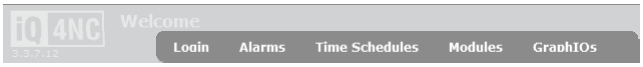
See **"Strategy Upload and Download" on page 102**.

37.2.1.4 Access to Web Pages

See **"Access to Web Pages" on page 102**

37.2.1.5 View Defaults

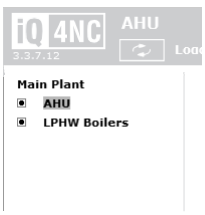
The links available in the banner (Alarms, Time Schedules, Modules, and GraphIQs) across the top of web pages, can be restricted by setting the View Alarms. View GraphIQs, View Modules, and View Time Schedules parameters.



Note: The Login/Logout link shown in the banner cannot be turned off.

Access to the navigator on GraphIQs can be restricted using the 'View Navigator' parameter.

The last View Default is View Navigator which can be used to turn off the Navigator displayed on the GraphIQs pages.



The removal of these links can be used to stop the user having access to other pages in the IQ4.

37.2.1.6 Home Page

When a user logs in, a defined web page can be displayed. If this is a customised GraphIQs page it enables a defined set of pages to be accessed which can be bespoke to that user.

The Home Page is defined by a URL for the page that is required. IQSET facilitates this configuration with a drop down list of codes, and the ability to select a specific GraphIQ page.

Note If the Guest User module is set up with a Home Page, the Home Page will be displayed to all users before they login, not the 'Welcome' page.

If linking to a GraphIQ page the URL should be in the format below:

GraphIQs/<Directory Module Structure>/index.htm

Where <Directory Module Structure> is the directory module structure of the Display module that defines the page. E.g.

GraphIQs/Root/Albery House/First Floor/Index.htm

This will select the GraphIQs page shown for the first floor of Albery House

If linking to a module overview page the URL should be in the format below:

<Module Type>.htm

Where <Module Type> specifies the type of module using its text comms identifier. For details of the module identifiers see the appropriate module section of this manual. E.g.

S.htm

This will select the sensors module summary page

In linking to a module details page the URL should be in the format below:

<Module Type><Module Number>.htm

Where <Module Type> specifies the type of module using its text comms identifier. For details of the module identifiers see the appropriate module section of this manual, and <Module Number> specifies the number of the module. E.g.

S1.htm

This will select the sensor details for Sensor module 1.

Note: Module page URLs are case-sensitive.

If linking to a web site enter the website's URL. E.g.

http://www.trendcontrols.com

37.2.1.7 Creating Type 0 User Modules

A Type 0 User Module can be created from the Type 0 User Module overview web page by clicking **Create New** and entering the User Name, Password, Pin, and other parameters. However the Type 0 User Module cannot be deleted by using the web pages.

Type 0 User Modules can be both created and deleted in IQSET by changing and downloading the strategy file.

When creating Type 0 User Modules the first Type 0 User Module added must have a user level of '>=99' otherwise it will not be possible to add other user, and make changes that require a user level greater than the one specified. The IQ4 will enforce the fact that there MUST be at least 1 Type 0 User Module with a user level greater than or equal to '99'.

Note: From v3.30 onwards all Type 0 User Modules created via the web will not enforce Trend network security and this cannot be changed. This means that this user does not affect the IQ4's ability to communicate via the Trend Network, and the ability to add or change the PIN is disabled.

It is recommended to always create an Administrator user with a user level '100', before creating any other users.

37.2.1.8 BACnet Security

For /BAC variants and IQ4NC even if Type 0 User Modules are set up correctly, the IQ4 is not protected from BACnet writes. An IQ4 is only protected from BACnet writes if BACnet is disabled.

37.2.1.9 Password Recovery

37.2.1.9.1 Forgotten User Name and Password

If the user name or password has been forgotten a temporary user name and password can be obtained from Trend.

The temporary user name and password will remain valid until midnight that day unless:

- The user tries to log in 3 times with an invalid user name and password combination or,
- The user uses the temporary user name and password to log in (i.e. the temporary user name/password can only be used once).

The temporary user name and password should be used immediately to log in and set up the user name and password on one of the existing Type 0 User Modules. The updated user name and password should then be used as the temporary user name and password will no longer be valid.

Note: The temporary user name and password can only be used to edit the Type 0 User Modules.

37.2.1.9.2 Forgotten PIN

If a user forgets their PIN, a default PIN which has a level 99 can be obtained from Trend for use on that day. This is useful if there is no Ethernet access to the IQ4.

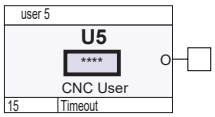
37.2.1.10 Type 0 User Module Parameters

Type 0 User modules include the parameters common to all User modules - see **"User Module Common Parameters" on page 516**.

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Application Key	A number for the user which should be included in an XML write to the IQ4. It is only displayed when the user is logged on. It will change if the user name or password are changed.	-	-	N	-

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Disable Network Use	Specifies whether the Type 0 User Module network has access rights. When '0' the Type 0 User Module will protect network access (text communications) and its PIN can be used to authorise those changes. When '1' the Type 0 User Module will not protect network access (text communications). However it can be used to access the IQ4's web pages or use XML or authorise strategy upload download. The default is '0' however if the user is created from the IQ4's web pages it will always be '1'. It is not possible to change this parameter from the web pages	0 or 1 1 = Is not part of network access rights management	0	N	w
Home Page	A URL of the web page that is displayed after user has logged on.	Valid URL	Blank	N	H
Level	The user level associated with the Type 0 User Module; defines what communications and changes can be made when logged on with this module.	-1 to 100	100	N	L
PIN	A 4 digit number used for security on text communications, e.g. entered into supervisors to authorise parameter change.	0000 to 9999	Blank	N	P
View Alarms	Displays/hides the alarm log link from banner.	0 or 1 0 = hidden 1 = Visible	1	N	V
View GraphIQs	Displays/hides GraphIQs link from banner	0 or 1 0 = hidden 1 = Visible	1	N	G
View Modules	Displays/hides module link from banner	0 or 1 0 = hidden 1 = Visible	1	N	M
View Navigator	Displays/hides Navigator from GraphIQs pages	0 or 1 0 = hidden 1 = Visible	1	N	V
View Time Schedules	Displays/hides Time Schedule link from banner	0 or 1 0 = hidden 1 = Visible	1	N	Z

37.2.2 Type 1 CNC User Modules (v3.60 or greater)



Type 1 CNC User modules contain the usernames and passwords that enable a client that supports secure communications to a vCNC operating in 'Secure' mode to connect to the vCNC. Each module has a password, and username which must be provided by a client to authorise the secure connection to the vCNC.

User Modules appear on the strategy page, but are not sequenced.

37.2.2.1 Type 1 CNC User Module Parameters

Type 1 CNC User modules include the parameters common to all User modules except the 'Password Timeout (Minutes)' parameter - see **"User Module Common Parameters" on page 516**.

37.3 User Module Common Parameters

User modules also include the following parameter common to most modules: ‘Page’, ‘X co-ordinate’ and ‘Y co-ordinate’ - see **“Common IQ4 Module Parameters” on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Last failed log in	The time of the last failed login attempt	timestamp	0	N	s
Last successful log in	The time of the last successful login	timestamp	0	N	s
Logged On	The module’s logged on state. Indicates whether a user is logged on from the web pages using this module.	0 or 1 1 = Logged On 0 = Logged Off	0	O	O
Password	A hashed representation of the 30-character password that must be entered by the user. DO NOT write to this parameter. Changes to the password should ONLY be made using IQSET.	0 to 30 characters.	Blank	N	W
Timeout (Minutes)	Not Type 1 CNC User - The time (minutes) when using web pages after the last key press after which the user will be automatically logged off.	1 to 15	15	N	T
Type	The type of User module.	0, 1 0 = User 1 = CNC	-	N	Y
User Name	The user name of the user. Changes to the User Name should ONLY be made using IQSET.	0 to 30 character string.	Blank	N	$

Note: Editing user modules using text comms could invalidate the module and make it impossible to log into the controller.

38 VIRTUAL CNC MODULES

Module Identifier	Cn (where n is module number)
Default overview parameters	-
Size in brIQs	9

38.1 Functionality

Virtual CNC modules enable the connection between a Trend supervisory or tool and the Trend network to be made over Ethernet using a Virtual Communications Node Controller (vCNC). Once a device is connected to the vCNC, it can communicate over the Trend network using normal Trend communications (i.e. LAN number and device address). The vCNC appears to be located on the IQ4's Local LAN and will respond to LAN mapping in this way.

A Virtual CNC module does not exist until its 'Address' and 'Port Number' parameters are set up, once created it will be shown on the network map. Virtual CNC modules are not included in the strategy file, do not appear on the strategy page, and do not require a sequence table entry.

The number of Virtual CNC modules available is dependent on the type of IQ4 - see the IQ4's data sheet for details.

38.1.1 Secure Communications (v3.60 or greater)

From v3.60 firmware onwards Virtual CNC modules can operate in two security modes, 'Secure' - see **"Secure Mode (v3.60 or greater)" on page 517** and 'None' - see **"Security Mode 'None'" on page 517**. The security mode is specified by the 'Security Mode' parameter.

38.1.1.1 Secure Mode (v3.60 or greater)

When a Virtual CNC module is added to the strategy it defaults to 'Secure', communications between it and any clients are encrypted and authenticated, and must be authorised with a user name and password matching a CNC User module in the IQ4.

Note: Not all clients support connection to a Virtual CNC module operating in 'Secure' mode. In these situations, it will be necessary to configure the Virtual CNC module to operate in Security mode 'None'. However, it is recommended that, where possible, the supervisor product be upgraded to a version that does support secure connections. If this is not possible, then consideration should be given to using another Trend product that does support secure connections. Use of an unsecured connection must be restricted to situations where there is no alternative.

38.1.1.1.1 Connecting to a vCNC operating in Secure Mode

When operating in this mode the vCNC will tell any client attempting to make a connection that it supports secure communications and that it is operating in 'Secure' mode and whether there are CNC User Modules in the strategy. The connection made will be encrypted and authenticated (using a user name and password)

If the client supports secure connection to a vCNC and there are CNC User Modules in the strategy the IQ4 will require the client to provide a user name and password that matches a CNC user module in the strategy. Once the credentials are validated the connection will be made.

If the client supports secure connection to a vCNC, and there are not any CNC User Modules in the strategy, it will be necessary to connect for the first time using IQSET to configure the CNC user module within the IQ4. If already connecting using IQSET the IQ4 will allow IQSET to create a new CNC User Module with a user name and password specified by the user. It will then be necessary to prove that you have physical access to the IQ4 by pressing the service button, IQSET will guide you through this process - see **"Connecting to an Unconfigured IQ4's vCNC" on page 44**.

38.1.1.2 Security Mode 'None'

When operating in Security Mode 'None' communication between the vCNC and a client that supports secure connections will still be encrypted, but will not require authentication.. Clients that do not support secure connection to are able to connect to the vCNC in the same way as for earlier IQ4 versions but the connection will not be encrypted or authenticated.

'Unsecured' mode should only be used where connection to a Trend device that does not support secure connections is required, and there is no alternative.

38.1.2 Alarms

The IQ4 vCNCs only operate in supervisor mode; they cannot operate in alarm mode. This means that they cannot initiate a connection to a supervisor when the IQ4 needs to send an alarm. Therefore, if the IQ4 must send alarms to a supervisor which is only connected temporarily, it is best done using IP Address type alarms - see **"IP Address Alarm Destinations" on page 81**.

Note: The IQ4NC's virtual CNC modules have the INC alarm forwarding feature which can be used instead of the alarm mode available in vCNCs in XTENDs - see **"INC Alarm Forwarding (IQ4NC only)" on page 91**.

38.2 Virtual CNC Module Parameters

Virtual CNC modules include the following parameter common to most modules: 'Label' - see **"Common IQ4 Module Parameters" on page 165**, plus the following parameters:

Parameter	*Description*	*Range*	*Default*	*Type*	*Text Comms*
Address	The network address of the vCNC. It must be unique on the Local LAN.	0, 1, 4 to 119 excluding 10 0 = Disabled	0	N	A
Connected IP Address	IP address of connected device.	Valid IP address 0.0.0.0 = Device not connected.	0.0.0.0	O	I
DeviceDead Alarm Enable	Enables/disables device dead alarms form the vCNC	0 or 1 0 = Disabled 1 = Enabled	1	N	D
Port Number	Port number of the vCNC.	1 to 65535	10000 + Address	N	P
Security Mode	v3.60 or greater. The security mode of the vCNC. *Set to 'Secure' by default when a new vCNC is created. Set to 'None' by default if firmware is updated and vCNCs exist in the strategy.	0 or 1 0 = None 1 = Secure	*Secure	N	M
Socket Timeout	The timeout of the CNC socket.	0 to 65535	0	N	T
Status	Status of vCNC.	0, 1, 2 0 = Not used 1= Not connected 2 = Connected	0	O	S

APPENDICES

A1 BACNET OBJECT/PROPERTY CODES AND UNITS

This section lists the codes that can be used to select BACnet objects and BACnet properties for use in IC communications. It also lists the units that can be coded and sent over BACnet (used in sensors and knobs).

A1.1 BACnet Object Codes

Code	*Object*
AI	Analogue Input
AO	Analogue Output
AV	Analogue Value
BI	Binary Input
BO	Binary Output
BV	Binary Value
MI	Multi-state input
MO	Multi-state Output
MV	Multi-state Value
NC	Notification Class
DV	Device
AG	Averaging
CL	Calendar
CD	Command
EE	Event Enrollment
FL	File
GR	Group
LP	Life Safety Point
LZ	Life Safety Zone
LO	Loop
PG	Program
SC	Schedule
TL	Trend Log

A1.2 BACnet Property Codes

Code	*Property*
AK	ACKED_TRANSITIONS
AQ	ACK_REQUIRED
AN	PROP_ACTION
AX	ACTION_TEXT
AB	ACTIVE_COV_SUBSCRIPTIONS
AT	ACTIVE_TEXT
AC	ACTIVE_VT_SESSION
AV	ALARM_VALUE
AS	ALARM_VALUES
AL	ALL
AW	ALL_WRITES_SUCCESSFUL
AM	APDU_SEGMENT_TIMEOUT
AO	APDU_TIMEOUT
AF	APP_SOFTWARE_VERSION
AE	ARCHIVE
AP	ATTEMPTED_SAMPLES
AG	AVERAGE_VALUE
BF	BACKUP_FAILURE_TIMEOUT
BI	BIAS
BS	BUFFER_SIZE
CC	CHANGE_OF_STATE_COUNT

Code	*Property*
CH	CHANGE_OF_STATE_TIME
CL	CLIENT_COV_INCREMENT
CF	CONFIGURATION_FILES
CR	CONTROLLED_VAR_REFERENCE
CU	CONTROLLED_VAR_UNITS
CV	CONTROLLED_VAR_VALUE
CI	COV_INCREMENT
CB	COV_RESUBSCRIBE_INTERVAL
CT	CURRENT_NOTIFY_TIME
DR	DATABASE_REVISION
DT	DATELIST
DS	DAYLIGHT_SAVINGS_STATUS
DB	DEADBAND
DC	DERIVATIVE_CONSTANT
DU	DERIVATIVE_CONSTANT_UNITS
DN	DESCRIPTION
DH	DESCRIPTION_OF_HALT
DV	DEVICE_ADDRESS_BINDING
DY	DEVICE_TYPE
DG	DIRECT_READING
ED	EFFECTIVE_PERIOD
ET	ELAPSED_ACTIVE_TIME
EL	ERROR_LIMIT
EE	EVENT_ENABLE
EP	EVENT_PARAMETERS
ES	EVENT_STATE
EM	EVENT_TIME_STAMPS
EY	EVENT_TYPE
EH	EXCEPTION_SCHEDULE
FV	FAULT_VALUES
FB	FEEDBACK_VALUE
FM	ACCESS_METHOD
FS	FILE_SIZE
FY	FILE_TYPE
FR	FIRMWARE_REVISION
GL	GROUP_MEMBERS_LIST
HL	HIGH_LIMIT
IT	INACTIVE_TEXT
IO	INSTANCE_OF
IC	INTEGRAL_CONSTANT
IU	INTEGRAL_CONSTANT_UNITS
IP	IN_PROCESS
IN	ISSUE_CONF_NOTIFICATIONS
LR	LAST_RESTORE_TIME
LV	LIFE_SAFETY_ALARM_VALUES
LM	LIMIT_ENABLE
LD	LOCAL_DATE
LT	LOCAL_TIME
LN	LOCATION
LB	LOG_BUFFER
LP	LOG_DEVICE_OBJ_PROP
LE	LOG_ENABLE
LI	LOG_INTERVAL
LL	LOW_LIMIT
MQ	MAINTENANCE_REQUIRED
MF	MANIPULATED_VAR_REFERENCE
MS	MAX_SEGMENTS_ACCEPTED

Code	*Property*
MO	MAXIMUM_OUTPUT
MV	MAXIMUM_VALUE
MT	MAXIMUM_VALUE_TIMESTAMP
MA	MAX_APDU_LENGTH_ACCEPTED
MI	MAX_INFO_FRAMES
MM	MAX_MASTER
MP	MAX_PRES_VALUE
MB	MEMBER_OF
mF	MINIMUM_OFF_TIME
mN	MINIMUM_ON_TIME
mO	MINIMUM_OUTPUT
mV	MINIMUM_VALUE
mT	MINIMUM_VALUE_TIMESTAMP
mP	MIN_PRES_VALUE
ME	MODE
ML	MODEL_NAME
MD	MODIFICATION_DATE
NC	NOTIFICATION_CLASS
NT	NOTIFICATION_THRESHOLD
NY	NOTIFY_TYPE
NR	NUMBER_OF_APDU_RETRIES
NS	NUMBER_OF_STATES
OI	OBJECT_IDENTIFIER
OL	OBJECT_LIST
ON	OBJECT_NAME
OR	OBJECT_PROPERTY_REFERENCE
OY	OBJECT_TYPE
OP	OBJ_PROP_REFERENCES_LIST
OE	OPERATION_EXPECTED
OT	OPTIONAL
OU	OUTPUT_UNITS
OS	OUT_OF_SERVICE
PO	POLARITY
PV	PRESENT_VALUE
PY	PREVIOUS_NOTIFY_TIME
PR	PRIORITY
PA	PRIORITY_ARRAY
PW	PRIORITY_FOR_WRITING
PI	PROCESS_IDENTIFIER
PG	PROGRAM_CHANGE
PL	PROGRAM_LOCATION
PS	PROGRAM_STATE
PM	PROFILE_NAME
PP	PROPORTIONAL_CONST
PU	PROPORTIONAL_CONST_UNITS
PN	PROTOCOL_REVISION
PC	PROT_CONFORMANCE_CLASS
PJ	PROT_OBJ_TYPES_SUPPORTED
PD	PROT_SERVICES_SUPPORTED
PT	PROT_VERSION
RO	READ_ONLY
RH	REASON_FOR_HALT
RT	RECIPIENT
RL	RECIPIENT_LIST
RN	RECORDS_SINCE_NOTIFICATION
RC	RECORD_COUNT
RB	RELIABILITY

Code	*Property*
RD	RELINQUISH_DEFAULT
RQ	REQUIRED
RV	RESERVED_18
RS	RESOLUTION
SM	SEGMENTATION_SUPPORTED
SL	SESSION_KEYS_LIST
SP	SETPOINT
SR	SETPOINT_REFERENCE
SG	SETTING
SD	SILENCED
SA	START_TIME
ST	STATE_TEXT
SF	STATUS_FLAGS
SO	STOP_TIME
SW	STOP_WHEN_FULL
SS	SYSTEM_STATUS
TD	TIME_DELAY
TA	TIME_OF_ACTIVE_TIME_RESET
TS	TIME_OF_STATE_COUNT_RESET
TR	TIME_SYNC_RECIPIENTS
TC	TOTAL_RECORD_COUNT”
TV	TRACKING_VALUE
UN	UNITS
UI	UPDATE_INTERVAL
UO	UTC_OFFSET
VS	VALID_SAMPLES
VV	VARIANCE_VALUE
VI	VENDOR_IDENTIFIER
VN	VENDOR_NAME
VC	VT_CLASSES_SUPPORTED
WK	WEEKLY_SCHEDULE
WI	WINDOW_INTERVAL
WS	WINDOW_SAMPLES
ZM	ZONE_MEMBERS

A1.3 BACnet Units

Type	*Units text*	*4 Characters*
Acceleration	meters-per-second-per-second	m/s2
Area	square-meters	m2
Area	square-centimeters	cm2
Area	square-feet	ft2
Area	square-inches	in2
Currency	currency1	$USD
Electrical	milliamperes	mA
Electrical	amperes	A
Electrical	amperes-per-meter	A/m
Electrical	amperes-per-square-meter	A/m2
Electrical	ampere-square-meters	Am2
Electrical	farads	F
Electrical	henrys	H
Electrical	ohms	Ohm
Electrical	ohm-meters	Ohmm
Electrical	milliohms	mOhm
Electrical	kilohms	kOhm
Electrical	megohms	MOhm
Electrical	siemens	S
Electrical	siemens-per-meter	S/m
Electrical	teslas	T
Electrical	volts	V
Electrical	millivolts	mV
Electrical	kilovolts	kV
Electrical	megavolts	MV
Electrical	volt-amperes	VA
Electrical	kilovolt-amperes	kVA
Electrical	megavolt-amperes	MVA
Electrical	volt-amperes-reactive	var
Electrical	kilovolt-amperes-reactive	kvar
Electrical	megavolt-amperes-reactive	Mvar
Electrical	volts-per-degree-Kelvin	V/K
Electrical	volts-per-meter	V/m
Electrical	degrees-phase	degP
Electrical	power-factor	pf
Electrical	webers	Wb
Energy	joules	J
Energy	kilojoules	kJ
Energy	kilojoules-per-kilogram	kJkg
Energy	megajoules	MJ
Energy	watt-hours	Wh
Energy	kilowatt-hours	kWh
Energy	megawatt-hours	MWh
Energy	btus	btu
Energy	kilo-btus	kbtu
Energy	mega-btus	Mbtu
Energy	therms	thrm
Energy	ton-hours	th
Enthalpy	joules-per-kilogram-dry-air	J/kg
Enthalpy	kilojoules-per-kilogram-dry-air	kJkd
Enthalpy	megajoules-per-kilogram-dry-air	MJkg
Enthalpy	btus-per-pound-dry-air	b/ld
Enthalpy	btus-per-pound	b/lb
Entropy	joules-per-degree-Kelvin	J/K
Entropy	kilojoules-per-degree-Kelvin	kJ/K

Type	*Units text*	*4 Characters*
Entropy	megajoules-per-degree-Kelvin	MJ/K
Entropy	joules-per-kilogram-degree-Kelvin	JkgK
Force	newton	N
Frequency	cycles-per-hour	c/h
Frequency	cycles-per-minute	c/m
Frequency	hertz	Hz
Frequency	kilohertz	kHz
Frequency	megahertz	MHz
Frequency	per-hour	h-1
Humidity	grams-of-water-per-kilogram-dry-air	g/kg
Humidity	percent-relative-humidity	%rh
Length	millimeters	mm
Length	centimeters	cm
Length	meters	m
Length	inches	inch
Length	feet	feet
Light	candelas	cd
Light	candelas-per-square-meter	cdm2
Light	watts-per-square-foot	W/f2
Light	watts-per-square-meter	W/m2
Light	lumens	lum
Light	luxes	lux
Light	foot-candles	ftcd
Mass	kilograms	kg
Mass	pounds-mass	lb
Mass	tons	t
Mass Flow	grams-per-second	g/s
Mass Flow	grams-per-minute	g/m
Mass Flow	kilograms-per-second	kg/s
Mass Flow	kilograms-per-minute	kg/m
Mass Flow	kilograms-per-hour	kg/h
Mass Flow	pounds-mass-per-second	lb/s
Mass Flow	pounds-mass-per-minute	lb/m
Mass Flow	pounds-mass-per-hour	lb/h
Mass Flow	tons-per-hour	t/h
Power	milliwatts	mW
Power	watts	W
Power	kilowatts	kW
Power	megawatts	MW
Power	btus-per-hour	bt/h
Power	kilo-btus-per-hour	kb/h
Power	horsepower	hp
Power	tons-refrigeration	tr
Pressure	pascals	Pa
Pressure	hectopascals	hPa
Pressure	kilopascals	kPa
Pressure	millibars	mBar
Pressure	bars	Bar
Pressure	pounds-force-per-square-inch	psi
Pressure	centimeters-of-water	cmw
Pressure	inches-of-water	inwc
Pressure	millimeters-of-mercury	mmHg
Pressure	centimeters-of-mercury	cmHg
Pressure	inches-of-mercury	inHg
Temperature	degrees-Celsius	DegC
Temperature	degrees-Kelvin	K
Temperature	degrees-Kelvin-per-hour	K/h

Type	*Units text*	*4 Characters*
Temperature	degrees-Kelvin-per-minute	K/m
Temperature	degrees-Fahrenheit	DegF
Temperature	degree-days-Celsius	DDyC
Temperature	degree-days-Fahrenheit	DDyF
Temperature	delta-degrees-Fahrenheit	dDgF
Temperature	delta-degrees-Kelvin	dK
Time	years	year
Time	months	mnth
Time	weeks	week
Time	days	day
Time	hours	hour
Time	minutes	min
Time	seconds	sec
Time	hundredths-seconds	.01s
Time	milliseconds	mS
Torque	newton-meters	Nm
Velocity	millimeters-per-second	mm/s
Velocity	millimeters-per-minute	mm/m
Velocity	meters-per-second	m/s
Velocity	meters-per-minute	m/m
Velocity	meters-per-hour	m/h
Velocity	kilometers-per-hour	km/h
Velocity	feet-per-second	ft/s
Velocity	feet-per-minute	fpm
Velocity	miles-per-hour	mph
Volume	cubic-feet	cuft
Volume	cubic-meters	m3
Volume	imperial-gallons	gall
Volume	liters	l
Volume	us-gallons	USg
Volumetric Flow	cubic-feet-per-second	cfs
Volumetric Flow	cubic-feet-per-minute	cfm
Volumetric Flow	cubic-meters-per-second	m3/s
Volumetric Flow	cubic-meters-per-minute	m3/m
Volumetric Flow	cubic-meters-per-hour	m3/h
Volumetric Flow	imperial-gallons-per-minute	igpm
Volumetric Flow	liters-per-second	l/s
Volumetric Flow	liters-per-minute	l/m
Volumetric Flow	liters-per-hour	l/h
Volumetric Flow	us-gallons-per-minute	gpm
Other	degrees-angular	deg
Other	degrees-Celsius-per-hour	DC/h
Other	degrees-Celsius-per-minute	DC/m
Other	degrees-Fahrenheit-per-hour	DF/h
Other	degrees-Fahrenheit-per-minute	DF/m
Other	joule-seconds	Js
Other	kilograms-per-cubic-meter	kgm3
Other	kilowatt-hours-per-square-meter	kWhm
Other	kilowatt-hours-per-square-foot	kWhf
Other	megajoules-per-square-meter	MJm2
Other	megajoules-per-square-foot	MJsf
Other	no-units	
Other	newton-seconds	Ns
Other	newtons-per-meter	N/m
Other	parts-per-million	ppm
Other	parts-per-billion	ppb
Other	percent	%

Type	*Units text*	*4 Characters*
Other	percent-obscuration-per-foot	%/ft
Other	percent-obscuration-per-meter	%/m
Other	percent-per-second	%/s
Other	per-minute	pm
Other	per-second	ps
Other	psi-per-degree-Fahrenheit	psiF
Other	radians	rad
Other	radians-per-second	rd/s
Other	revolutions-per-minute	rpm
Other	square-meters-per-Newton	m2/N
Other	watts-per-meter-per-degree-Kelvin	WmK
Other	watts-per-square-meter-degree-kelvin	Wm2K

A2 XML SYNTAX

The Web service call is achieved by sending an HTTP request to an IQ4, the response from the IQ4 being an XML document. Both Reads and Writes can be achieved through the web services interface.

Writes to the IQ4 require an Application Key to be included with the write if a Type 0 User Module has been set up (i.e. security has been enabled). This key is provided by a Type 0 User Module (not the system guest).

Some Trend JavaScript libraries are available from the Trend PartnerNet website (Downloads/Tools). They provide examples of using the IQ4 XML Web Services - see **"Using IQ4 Web Services - Trend JavaScript Libraries" on page 535**.

A2.1 Request Structure

Web service requests can be submitted to the IQ4 using a HTTP GET service. This means that requests may be submitted as a URL.

The request format is:

> http://***Hostname or IP address***/ws/tsite.xml?Key = ***Application Key***&Type = ***Message Type***&Request = ***Text Comms Expression***

***Hostname or IP address*:** This argument is mandatory. Specifies the IP address or hostname of the IQ4 to which the request is being sent.

All responses are sent as UTF-8 encoded XML documents and as such have the HTTP header "Content-Type; text/xml; charset = utf-8".

***Application Key*:** This argument is optional. It is required to authorise the request when the IQ4 to which the request is being sent has security enabled. It is the value of the 'Application Key' parameter in the Type 0 User Module in the IQ4 to which the request is being sent that is to authorise the change and can be found on the Type 0 User Module's web page.

If the 'XML Web Services' parameter in the Ethernet IP network module is set to '1', the Application Key must be included in XML write messages. If the 'XML Web Services' parameter is set to '2' or '3' then encrypted authentication will be used - see **"Authentication Security" on page 540**.

With Basic IQ authentication if a Type 0 User Module has been set up (i.e. the IQ4's security has been enabled) and if either an Application Key is not included in the web service write, or the Key does not match one of those generated by any of the Type 0 User Modules in the IQ4, then web service writes will fail.

Note: The Application Key is generated by the IQ4's Type 0 User Module from the User Name and Password. If the User Name or Password are changed then the it will change. It is valid for all Type 0 User Modules in all IQ4s which have the same User Name and Password.

***Message Type*:** This argument is optional. It indicates whether a read or a write is to be carried out and should be set to 'Read' or 'Write' as required. If it is omitted, a 'Read' is executed.

Text Comms Request: This argument is mandatory. It specifies the actual request (i.e. what data is to be read or written). It can contain any structured comms data field as defined **"Text Communications" on page 543** and the relevant module sections of this manual - see **"Part 3" on page 163**.

Any HTTP special characters as defined by RFC1738 must be encoded using the %xx syntax. For example e.g. a tilde '~' is represented by '%7E' where 7E is the hexadecimal equivalent of 127, the decimal code for a tilde.

A2.2 Response Structure

A2.2.1 Invalid Request Response

The request is validated to ensure that the server is capable of producing a full response to all the requested parameters. If this validation fails in any respect, then an error response is generated.

A client using this web service should as a result be able to implement a parsing process as follows.

```
IF a validation error element exists in the document THEN
        Request failed, parse error information and inform user
ELSE
        IF Request was a Read THEN
           Parse response file down to expected parameters
           FOR each parameter of each device
              IF Parameter has no CommsState element THEN
                 Value can be used
              ELSE
                 Parameter communication failed – report CommsState
              ENDIF
           NEXT
              ELSE (Request is a Write)
                 FOR each device THEN
                    Deal with Write Success or Failure
           NEXT
        ENDIF
ENDIF
```

The general form of an invalid request response is as follows:

<?xml version = ”1.0" encoding = ”UTF-8" standalone = ”yes” ?>
- <!— written by IQ4 Iss3.0 Mar 14 2014 Saved 2014-03-14T22:55:52TZD —>
- <InvalidRequest>
- < Type = ”Some brief text” Code = ”X” Details = ”Some optional longer text” />
- < Some error code specific detail />
- <Type = ”Some more brief text” Code = ”Y” Details = ”Some other optional longer text” />
</InvalidRequest>

As can be seen in the above, the specification allows for many errors to be reported if required. It is not necessary to attempt to discover all possible errors in a request, processing may stop at the first discovered error and a document be returned. However, if multiple errors are discovered, they may be all be reported back in the one error response.

The following describes the specific errors that can be reported as invalid requests. The column titles in the following table correspond to the attributes shown in the XML fragment above.

Type	*Code*	*HTTP Code*	*Meaning*
FAILED - Web services are disabled	0	503	The XML server is disabled in the network module.
FAILED - Illegal web service message type	1	412	The message type requested (e.g. Type = Read) is not supported.
FAILED - Request is zero length	2	412	The request parameter was zero length or was not supported.
FAILED - illegal parameter - <name> is not supported	3	412	A parameter was included in the requests that the web service does not implement.
FAILED - Argument for Key not supplied	6	401	A write service did not contain a Application Key.
FAILED - Authentication failure	7	401	The value of the Application Key was incorrect, write is refused.
FAILED - No Trend parameters	8	412	The request parameter contained no valid Trend text comms parameters

Type	*Code*	*HTTP Code*	*Meaning*
FAILED - Parsing error on request parameter		412	An error occurred parsing the request parameter indicating an illegal text comms expression was used.
FAILED - One or more parameters not found	10		Comma separated list of parameters as: OxxxLyyy[Mz(p,q), My(r,s], e.g. O16l0[S23(V),S24(k)],
FAILED - web service is not licenced	14	412	The product is required to be licenced. A licence needs to be entered using the IQ4's web pages.
FAILED - Maximum response exceeded	15	412	The server maximum response size would have been exceeded.

Note: The HTTP Code is the code returned in the HTTP response header defined in the HTTP standard.

A2.2.2 Response to a Read

The response to a read is an XML document, the format of which matches the definition of the Trend .tsite file - see **"Tsite File Schema" on page 531**.

e.g.

```
<?xml version = "1.0" encoding = "UTF-8" standalone = "yes" ?>
- <!–
          written by IQ4 Iss3.0 Mar 14 2014 Saved 2014-03-14T22:55:52TZD
–>
- <TrendProject>
          - <Site CncAddress = "30" TuaString = "T/Albery RnD 6/TCP,,,10.10.10
          .70,10030///20/20/">
             - <Lan IsLocal = "True" LanNumber = "20">
                - <Device DeviceNumber = "20" VersionString = "IQ4 Iss3.0 Mar 14
                2014">
                   <DeviceOverview DeviceType = "q" MajorVersion = "1" MinorVersion
                   = "3" HexEncoded = "711B1B1B41313330" Revision = "0" />
                      <Module Name = "K1" SubTypeNumber = "-1">
                         <Param Name = "V" Type = "1" Value = "11" />
                      </Module
                   </Device>
                </Lan>
          </Site>
</TrendProject>
```

A2.2.3 Response to a Write

A successful write will result in a response containing the details in ‘tsite’ format for the targeted device, with a CommsResponse tag (ServiceResponse) in the body of the response.

A positive response will be as follows:

```
<?xml version = ”1.0" encoding = ”UTF-8" standalone = ”yes” ?>
- <!–
          written by IQ4 Iss3.0 Mar 14 2014 Saved 2014-03-14T22:55:52TZD
–>
- <TrendProject>
          - <Site CncAddress = ”30" TuaString = ”T/Albery RnD 6/TCP,,,10.10.10
          .70,10030///20/20/”>
            - <Lan IsLocal = ”True” LanNumber = ”20">
              - <Device DeviceNumber = ”20" VersionString = ”IQ4 Iss3.0 Mar 14
              2014">
                <DeviceOverview DeviceType = ”q” MajorVersion = ”1" MinorVersion
                = ”3" HexEncoded = ”711B1B1B41313330" Revision = ”0" />
                <ServiceResponse Type = ”Acknowledge - Write was OK” Code =
                ”6" />
              </Device>
            </Lan>
          </Site>
</TrendProject>
```

A negative response for Bad PIN when writing to a remote controller will be as follows:

```
<?xml version = ”1.0" encoding = ”UTF-8" standalone = ”yes” ?>
- <!–
          written by IQ4 Iss3.0 Mar 14 2014 Saved 201-03-14T22:55:52TZD
–>
- <TrendProject>
          - <Site CncAddress = ”30" TuaString = ”T/Albery RnD 6/TCP,,,10.10.10
          .70,10030///20/20/”>
            - <Lan IsLocal = ”True” LanNumber = ”20">
              - <Device DeviceNumber = ”20" VersionString = ”IQ4 Iss 3.0 Mar
              14 2014">
                <DeviceOverview DeviceType = ”q” MajorVersion = ”1" MinorVersion
                = ”3"HexEncoded = ”711B1B1B41313330" Revision = ”0" />
                  <ServiceResponse Type = ”FAILED – Authentication failure”
                  Code = ”21" />
                </Device>
            </Lan>
          </Site>
</TrendProject>
```

A list of the service response types given by the IQ4 is given by the table below:

Code	*ServiceResponse Type*
6	Acknowledge - Write was OK
21	FAILED - Authentication failure
22	FAILED - Device communication failed

A2.3 Tsite File Schema

This section defines the public structure of the tsite format XML file. Client applications are required to tolerate the presence of other elements and attributes in the file in addition to those requested, these may be present to support non published system features.

The specification for the configuration of a device (modules it contains, parameters contained in those modules) can be found in **“Text Communications” on page 543** and the relevant module sections of this manual - see **“Part 3” on page 163**).

The following covers the public schema for a tsite format XML file.

```
<TrendProject>
          <Site />
            <Internetwork>
              <Lan IsLocal = ”False” LanNumber = ”1">
                <Device DeviceNumber = ”4">
                  <Module Name = ”R” >
                    <Param Name = ”A” Type = ”X” Value = ”YYYYYY” />
                  </Module>
                </Device>
              </Lan>
            </Internetwork>
              <Lan IsLocal = ”True” LanNumber = ”123">
                <Device DeviceNumber = ”4">
                  <Module Name = ”R” >
                    <Param Name = ”A” Type = ”X” Value = ”YYYYYY” />
                  </Module>
                </Device>
              <Internetwork>
            </Lan>
          </Site>
</TrendProject
```

A2.3.1 Site Element

Current web services will only return files with a single site present in them.

A2.3.2 Internetwork Element

There will be ‘0’ or ‘1’ Internetwork element in each site. The Internetwork is a container element for all the LANs in the system except the Local LAN. The Local LAN is returned in a separate element within a site. The IQ4 is always Local LAN, so there is no internetwork element.

A2.3.3 LAN Element

The LAN element may be present as the child element of an Internetwork object, or the child element of the Site element. The LAN element contains the device elements.

A LocalLan element will be returned if the request made is for a controller which is connected to the same Trend LAN as the web service server itself (made as a request to LAN 0), otherwise the requested controller’s LAN will be a child of the Internetwork object.

In the case of a request to an IQ4 through its own server, only a LocalLAN element is returned.

A2.3.3.1 IsLocal Attribute

The IsLocal attribute of the LAN element indicates whether the LAN object element represents a Local LAN. In the case where it does represent a Local LAN, requests to access controllers must always be targeted to LAN number 0.

A value of 'True' indicates Local LAN, 'False” indicates the LAN is accessed via the Internetwork.

A2.3.3.2 LanNumber Attribute

The LAN number is the number which matches the number assigned to the L parameter of the request. An exception to this is when a device on the Local LAN is addressed, in this case the L parameter must be assigned the value 0, but the LanNumber attribute will return the LAN Number of the Local LAN in the system (which maybe 0).

Possible values are 0 to 125 but not 2,3 or 10.

A2.3.4 Device Element

The Device element represents a controller or other networking device. A device contains 0 to many parameter elements.

A2.3.4.1 DeviceNumber Attribute

The Device number is the number specified for the O parameter of a request. It can be used to match a device element in a response to the request.

Possible values are 1 to 126 but not 2,3 or 10.

A2.3.5 Module Element

The Module element represents a function block within a controller. There are 0 to many module elements within a device element. A module contains 0 to many parameters.

A2.3.5.1 Name Attribute

The module's Name attribute is the text comms reference used to address the module. What modules are present in a controller can be found in the module sections of this manual.

A2.3.6 Parameter Element

The Parameter element represents an individual parameter value of a function block within a IQ4. There are 0 to many parameter elements within a module element. A parameter element may itself contain 0 to many parameter elements.

A2.3.6.1 Name Attribute

The parameter's Name attribute is the Text comms reference used to address the parameter.

A2.3.6.2 Type Attribute

The Parameter's Type attribute defines the data type of the Value parameter as follows:

Type	*Description*	*Example*
1	A floating point number as a Double	23.45
2	A string value	Hello World
3	A string value	Hello World
4	Date and time as Microsoft Date datatype. The example given is for 10:30 on 1-Jul-06. Time of day would be a number < 1.	38899.438
5	This value indicates that the parameter did not exist	invalid
6	Any 32 bit integer value	2345
101	A single bit as a boolean, the value 1 indicates bit set, 0 indicates bit clear	1
102	A byte which should be interpreted as two bits	3
103	A byte which should be interpreted as three bits	7
104	A byte which should be interpreted as four bits	15
105	A byte which should be interpreted as five bits	31
106	A byte which should be interpreted as six bits	63
107	A byte which should be interpreted as seven bits	127
108	A byte which should be interpreted as eight bits	255
200	Reserved (indicates a custom object is contained as a child element)	

Type	*Description*	*Example*
300	Reserved (The parameter is itself made up of a number of child parameter elements)	
302	Reserved	
304	Reserved (The parameter is itself made up of a number of child parameter elements of the same data type structured as an array)	

A2.3.6.3 Value Attribute

The value of the parameter with data type as defined above

A2.3.6.4 CommsState Attribute

A CommsState attribute may be included in the tsite format file to indicate the result of the last communication operation carried out on the parameter.

If the attribute is missing from the Parameter element then the last communication can be assumed to have succeeded (normally indicating the last read was successful). Otherwise this attribute shall mean:

Value	*Description*
0	Comms OK (Attribute not normally included in this case)
1	Parameter is currently being read
2	Parameter is currently being written
3	Reserved
4	The last communication failed - the target device, module, or parameter did not reply
5	The last write attempt was refused due to a bad PIN
6	Reserved

A2.3.6.5 Enum Attribute

The Enum attribute provides a human readable value for an enumerated parameter.

E.g. <Module Name = "P1" >
<Param Name = "P1" Type = "1" Enum = "15 minutes" />

A2.4 System Compatibility

XML access to an IQ4 is via its in-built server.

The authentication scheme defined here is intended to allow a single user name and password to give access to all controllers which have that user name and password in one of their Type 0 User Modules.

A2.4.1 Logging

IQ4 logging is supported.

E.g.

http://albery.iq4.ftwr.co.uk/ws/tsite.xml?Request = P2(B1-2(V,S,E))

yields the following response:

```
<TrendProject>
          "<Site CncAddress = "101" TuaString = "T/IQ4 On the Internet/TCP,,,1
          94.242.149.206,10101///20/20/">
            "<Lan IsLocal = "True" LanNumber = "20">
              "<Device DeviceNumber = "20" VersionString = "IQ4 Iss 3.0 Mar 14
              2014">
                <DeviceOverview DeviceType = "q" MajorVersion = "2" MinorVersion
                = "1" HexEncoded = "711B1B1B41323130" Revision = "0"/>
                  <Module Name = "P2" SubTypeNumber = "-1">
                    <Param Name = "B1" Type = "300">
                    <Param Name = "V" Type = "1" Value = "41.450001"/>
                    <Param Name = "S" Type = "108" Value = "0"/>
                    <Param Name = "E" Type = "6" Value = "80685"/>
                    </Param>

                    <Param Name = "B2" Type = "300">
                    <Param Name = "V" Type = "1" Value = "41.009998"/>
                    <Param Name = "S" Type = "108" Value = "0"/>
                    <Param Name = "E" Type = "6" Value = "80684"/>
                    </Param>
                  </Module>
              </Device>
            </Lan>
          </Site>
</TrendProject>
```

A2.5 Performance

A2.5.1 Size of Response

The size of the XML file produced is governed by the size of the request issued. The IQ4 server has a limit on the size of response which if exceeded will result in an error code being returned (Invalid Request Response 15).

A2.6 Using IQ4 Web Services - Trend JavaScript Libraries

The Trend JavaScript libraries make use of the IQ4 Web Services to display values and plots from the IQ4 on a web page, and to enable values to be changed in the controller from the web page. They are available from the Trend PartnerNet website in the Downloads/Tools section.

A2.6.1 Overview

These JavaScript libraries provide utility functions to facilitate the use of the IQ4 web services. The libraries are split up based on their functionality, so only the ones relevant to the functions required need to be used.

The following libraries are provided:

- TrendControls (trendcontrols-base.js) - The basic TrendControls namespace. This creates a space for the TrendControls name so must be used on all pages using the Trend JavaScript Libraries.
- TrendControls.XMLWebservices (trendcontrols-xmlwebservices.js) - Functionality for making based requests on the XML Web Services in an IQ4, so again must be used on all pages using the Trend JavaScript Libraries.
- TrendControls.ActiveData (trendcontrols-activedata.js) - This gives enhanced functionality to allow for the easy development of live data displays, Trend Active Data Points; this need only be used when Trend Active Data Points are required on the web page.
- TrendControls.Flot (trendcontrols-flot.js) - This provides simple graphing of the values recorded by a plot module; this need only be used when graphs of plot module values are required on the web page.

In addition to the JavaScript libraries, the .jQuery library (v1.3.2). must be used on all pages using the Trend JavaScript Libraries. It can be downloaded from '**<http://docs.jquery.com/Downloading_jQuery>**'.

The TrendControls.Flot library makes use of the flot library in order to create graphs of values so it must be used on all pages using the TrendControls.Flot library. It can be downloaded from '**<http://code.google.com/p/flot/>**'.

Details of the browsers supported by Trend JavaScript are given in section **“Browser Support” on page 540**, and descriptions of response codes (including error codes) are given in section **“Response Codes” on page 540**.

A2.6.2 Basic XMLWebservices requests using TrendControls.XMLWebservices

These requests require the ‘jQuery library’, plus ‘TrendControls’, and ‘TrendControls.XMLWebservices’ libraries to be used on the page.

A2.6.2.1 Read a Single Parameter

This function fetches a particular parameter value from a controller and passes it back to a callback function you supply.

TrendControls.XMLWebservices().getParameterValue(“***Hostname or IP address***”,“***Module***”,“***Parameter***”, function***CallbackFunction***

***Hostname or IP address*:** Specifies the IP address or hostname of the controller.

***Module*:** The module to request on.

Parameter: The parameter requested.

CallbackFunction: The function to be called with the parameter value - function prototype is (aResultCode, aValue)

Example:

```
TrendControls.XMLWebservices().getParameterValue("TREND_00_00_00","R1","D",function(aResultCode, aValue)
{
    alert(aValue);
});
```

A2.6.2.2 Write a Single Parameter

This function writes a particular parameter value to a controller and calls a callback function you supply with the result.

TrendControls.XMLWebservices().setParameterValue("***Hostname or IP address***","***ApplicationKey***","***Module***","***Parameter***", "***Value***",function***CallbackFunction***

***Hostname or IP address*:** Specifies the IP address or hostname of the controller.

***ApplicationKey*:** The Application Key for a Type 0 User Module in the controller if security is enabled in the controller.

***Module*:** The module to request on.

Parameter: The parameter requested.

***Value*:** The value to write to the parameter.

CallbackFunction: The function to be called with the result of the request - function prototype is (aResultCode, aModule, aParameter)

Example:

```
TrendControls.XMLWebservices().setParameterValue("TREND_00_00_00", "ABCD1234",
"R1", "D", "Identifier", function (aResultCode, aModule, aParameter)
     {
          alert("Write Complete");
     });
```

A2.6.2.3 Read Several Parameters

TrendControls.XMLWebservices().get("***Hostname or IP address***","***Request***",function***CallbackFunction***")

***Hostname or IP address*:** Specifies the IP address or hostname of the controller.

***Request*:** The text comms request to be made.

***CallbackFunction*:** Function to be called when the request completes. Receives the xml response object - function prototype is (aResultCode, anXmlResponse).

Make a generic text comms read request on the server passing the response back to you. You can then use the TrendControls.XMLWebservices().extractParameterValue() function to get parameter values out of the response.

Example:

```
TrendControls.XMLWebservices().get("TREND_00_00_00","R1(D,C)",
function(aResult, aXml)
{
          if (TrendControls.XMLWebservices().responseCode.SUCCESS = = aResult)
          {
            alert(  TrendControls.XMLWebservices().extractParameterValue(aXml,
            "R1", "D")));
            alert(  TrendControls.XMLWebservices().extractParameterValue(aXml,
            "R1", "C")));
          }
});
```

A2.6.2.4 Write Several Parameters

TrendControls.XMLWebservices().set("***Hostname or IP address***","***ApplicationKey***","***Request***", function***CallbackFunction***)

***Hostname or IP address*:** Specifies the IP address or hostname of the controller.

***Request*:** The text comms request to be made.

ApplicationKey: The Application Key for a Type 0 User Module in the controller if security is enabled.

Request: The text comms request to be made.

CallbackFunction: Function to be called when the request completes. Receives the xml response object - function prototype is (aResultCode, anXmlResponse)

Make a generic text comms read request on the server passing the response back to you.

You can then use the TrendControls.XMLWebservices().extractParameterValue() function to get parameter values out of the response.

Example:

```
TrendControls.XMLWebservices().set("TREND_00_00_00","R1(D = "Example",G = "String")", function(aResult, aXml)
{
          if (TrendControls.XMLWebservices().responseCode.SUCCESS = = aResult)
          {
               alert("Multi parameter write done");
          }
});
```

A2.6.2.5 Extract Parameter from a Number of Parameters Returned

TrendControls.XMLWebservices().extractParameterValue(***XmlResponse***, ***Module***, ***Parameter***)

XmlResponse: The dom xml response object.

Module: The Module you are interested in.

Parameter: The Parameter you are interested in (if structured then specify with brackets e.g B1(V) for plot buffer value).

This function will extract a parameter value from the response xml from the controller and return it to you.

Example: See the example above in section **"Read Several Parameters" on page 536** for TrendControls. XMLWebservices().get

This example has requested 2 parameters, D, and C, from the address module. They need to be extracted from the reply.

The exractParameterValue function must be used for each parameter is shown in the example.

A2.6.2.6 Extract Item Identifier from a Number of Different Module Parameters Returned

TrendControls.XMLWebservices().extractModuleIdentifier(***XmlResponse***, ***Index***)

***XmlResponse*:** The dom xml response object.

***Index*:** The index of the module you are interested in (starts from 1).

This function goes through the response xml and returns the name of the <anIndex> of the module that is present.

Example:

This function is used after a 'get' request for parameters from a number of different modules. For example the text comms request 'S(V)' will normally return a number of sensor values; the configured sensors may start at any number and are not necessarily continuous (e.g S4, S7, S8, S11), so the user will need to discover which sensors are replying with the values:

```
TrendControls.XMLWebservices().get(myController   ,"S(V)",   function(aResult,
aXml)
          {
            if (TrendControls.XMLWebservices().responseCode.SUCCESS = = aResult)
          {
            //Get the first module in the response
            alert( TrendControls.XMLWebservices().extractModuleIdentifier(aXml,
            1) );
          }
});
```

This could then be followed by a 'extractParameter' function to extract the value of 'Sn'.

e.g. myIdentifier = TrendControls.XMLWebservices().extractModuleIdentifier(aXml, 1)
alert(TrendControls.XMLWebservices().extractParameterValue(aXml, myidentifier, "V")));

A2.6.3 Creating Active Data Points

These functions require the ‘jQuery library’, plus ‘TrendControls’, ‘TrendControls.XMLWebservices’, and ‘TrendControls.ActiveData’ libraries to be used on the page.

The ActiveData library allows you to enable a number of active data points in a page using simple HTML markup and a couple of lines of JavaScript.

Within the page you need to add a span tag for each active point you want displayed. This markup describes the module and parameters you want displayed and the update rate required.

The point is configured by adding special class names to the span, the available class names are:

Class Name	***Description***
trendcontrols-active-value	Marks the span as an active value point (required)
trendcontrols-active-module-XX	Specifies the module from which the parameters are requested (required)
trendcontrols-active-parameter-X-Y-Z	Specifies the parameters which are requested (required)
trendcontrols-active-rate-N	Specifies the update rate for the point (optional)
trendcontrols-active-controller-TREND_00_00_00	Specifies the controller the data point is from (optional)

Note that the default controller is defined in the enable Active Data function described below, so the trendcontrols-active-controller class name is only required if the data is from a different controller to the the default controller.

An example of this to show the Label, Value, and Units from sensor 2 every 2 seconds is:

```
<span class = "trendcontrols-active-value trendcontrols-active-module-S2
trendcontrols-active-parameter-$-V-% trendcontrols-active-rate-2"></span>
```

The data points are enabled by a call to the TrendControls.ActiveData().enableActiveData() function which takes a default controller name to use.

This call must be made after the page has completely loaded so it is recommended that you use the jQuery.ready() function to enable the functionality as follows:

```
jQuery(document).ready( function()
     {
          TrendControls.ActiveData().enableActiveData("TREND_00_00_00");
     });
```

A2.6.4 Displaying Graphs Using TrendControls.Flot.

These functions require the 'jQuery library', the 'flot library', plus 'TrendControls', 'TrendControls.XMLWebservices', and 'TrendControls.Flot' libraries to be used on the page.

The TrendControls.Flot library will display graphs using the 3rd party flot library.

It uses the TrendControls.XMLWebservices library to fetch the plot data from the controller and then displays it as a graph.

It fetches the data using multiple requests so you will see the graph drawn live on the screen.

Within the page you need to add a div (division) tag for each active point you want displayed. This markup describes what you want displayed.

The point is configured by adding special class names to the div, the available class names are:

Class Name	***Description***
trendcontrols-flot	Marks the span as an active value point (required)
trendcontrols-flot-module-XX	Specifies the plot module from which the graph is requested (required)
trendcontrols-flot-controller-TREND_00_00_00	Specifies the controller the data point is from (optional)

Note that the default controller is defined in the enable Flot function described below, so the trendcontrols-flot-controller class name is only required if the data is from a different controller to the default controller.

An Example of this to show the graph of the first plot module, P1:

```
<div class = "trendcontrols-flot trendcontrols-flot-module-P1"></div>
```

The graph will be sized based on the size of the div element, this must be specified in the normal way using Cascading Style Sheets (CSS).

The graphs are enabled by a call to the TrendControls.Flot().enableFlot() function which takes a default controller name to use.

This call must be made after the page has completely loaded so it is recommended that you use the jQuery.ready() function to enable the functionality as follows:

```
jQuery(document).ready( function()
     {
          TrendControls.Flot().enableFlot("TREND_00_00_00");
     });
```

A2.6.5 Browser Support

The JavaScript libraries have been developed and tested with Internet Explorer 8 and Firefox 3.5 and should also work in Internet Explorer 6, Internet Explorer 7, and Firefox 2.0.

Because of the same origin policy implemented in browsers for JavaScript, some of the more advanced features of the libraries can only be used with Internet Explorer 8 and Firefox 3.5 as these are the only versions of the browsers to support the Cross-Origin Resource Sharing specification which allows JavaScript code to access resources across domains.

This means that if the JavaScript accesses more than one IQ4, Internet Explorer 8 and Firefox 3 must be used.

Earlier versions of these browsers and other browsers are not recommended as the JavaScript libraries will only be able to request data back from the device that serves up the JavaScript.

A2.6.6 Response Codes

TrendControls.XMLWebservices().responseCode

This enumeration contains all the possible response code results returned from calls to the XMLWebservices functions.

The possible values and their meanings are as follows:

TrendControls.XMLWebservices().responseCode.**NOT_ENABLED** - XML Webservices are not enabled in the target controller.
TrendControls.XMLWebservices().responseCode.**NOT_LICENCED** - XML Webservices are not licenced in the target controller.
TrendControls.XMLWebservices().responseCode.**SUCCESS** - The request was successful.
TrendControls.XMLWebservices().responseCode.**FAILURE** - The request failed - usually a more detailed response code is provided.
TrendControls.XMLWebservices().responseCode.**RESPONSE_LIMIT_EXCEEDED** - The request generated a response larger than the controller can produce so a more specific request needs to be used.
TrendControls.XMLWebservices().responseCode.**AUTH_FAILURE** - The application key provided was not valid for the request.

A2.7 Authentication Security

A2.7.1 Authentication Types

The XML Web Services parameter in the Ethernet IP network module defines the level of security required on the XML messages. It can be set as follows:

0 = XML interface OFF
1 = Basic IQ authentication: The Application key (obtained from an engineered Type 0 User Module) must be included in XML write messages.
2 = Write authentication: As for Basic IQ authentication, but the Application key is sent in encrypted form.
3 = Read/Write authentication: As for Basic IQ authentication, but the Application key is sent in hashed form and is required for both read and write messages.

With a type 1 setting, the basic IO communication application key is sent with in the message.

With types 2 and 3 the application keys are included in a hashed form, and thus when sent with the message cannot be read and copied; the hash will change with the date and the item selected.

A2.7.2 Hashing

The hashing method involves using a function to hide the authentication key and some current data; this will verify both the data integrity and authenticity of the message. The function used is: HMAC_MD5, key hashing for message authentication.

The main hash is generated from the following data:

Key: The current Application Key (all uppercase) from the Type 0 User Module with User Name used below
Domain: The URL used to make the request to the controller, e.g. trend_00_00_e9 (a Host Name) or 165.195.23.45 (an IP Address) - however the controller is accessed, but without the protocol.
Request: The Request part of the message (e.g. K1(V = 10) in plain text form i.e. NOT K1%28V = 10%29)
Type: The type of the request as text (Read or Write)
User name: The User Name from the Type 0 User Module with Application Key used above .
Date: The current date as dd/mm/yyyy

The user name will also be an HMAC_MD5 hashed before sending in the URL. The client can hash the user name once and keep the hash for use in subsequent requests as it will not change.

The controller can also keep a list of users, keyed on this HMAC key for easy lookup.

The HMAC_MD5 has 2 data fields:

HMAC_MD5 (HMAC_key, HMAC_data)

The message with encrypted authentication will have two hashes, H1, H2 with the two data fields defined as below:

H1 = HMAC_MD5 (“<key>”, “<username>”)
H2 = HMAC_MD5 (“<key>”, “<domain>:<request>:<Type>:<username>:<date as dd/mm/yyyy>”)

For example:

H1 = HMAC_MD5 (“30CC89B2”, “Sam”) = 97b69a2bd191c8e972312f11f6e85e71
H2 = HMAC_MD5 (“30CC89B2”,“trend_00_00_e9:K1(V=10):Write:Sam:13/06/2012”)=073d9202dd31b81eaaa8c1beb703066d

The request URL will be:

URL = http://<domain>/ws/tsite.xml?Type = <Read|Write>&User = <H1>&HMACKey = <H2>Request = <request>

Using the example H1 and H2 hashes defined above, the XML client will then send the following command to the controller:

http://trend_00_00_e9/ws/tsite.xml?Type = Write&User = 97b69a2bd191c8e972312f11f6e85e71&HMACKey = 073d9202dd31b81eaaa8c1beb703066d&Request = K1%28V = 10%29

Below is a snippet of HTML/JavaScript showing the same value created in JavaScript, this process should be used to test any implementation.

```
<html>
<head>
          <script type = "text/javascript" src = "http://crypto-js.googlecode.
          com/files/2.5.3-crypto-md5-hmac.js"></script>
</head>
<body>
          <div id = "finalResult" style = "font-family: Courier New; font-size:
          0.8em; padding: 2em;"></div>
          <script type = "text/javascript">

             function pad(number, length) {
```

```
            var str = '' + number;
            while (str.length < length) {
              str = '0' + str;
            }
            return str;
        }
        var myUsername    = "Sam"; // This bit is kept secret
        var myAppkey        = "30CC89B2"; // This bit is also kept secret
        var myType           = "Write";
        var myRequest        = "K1(V = 10)";
        var myDomain        = "trend_00_00_e9";
        var myResult;
        var myDate = new Date();
        var myDateString = pad(myDate.getDate(), 2) + "/" + pad(myDate.
        getMonth() + 1, 2) + "/" + myDate.getFullYear();
        var myHMACData = myDomain + ":" + myRequest + ":" + myType + ":" +
        myUsername + ":" + myDateString;
        var H1 = Crypto.HMAC(Crypto.MD5, myUsername, myAppkey, { asBytes: true
        });
        var H2 = Crypto.HMAC(Crypto.MD5, myHMACData, myAppkey, { asBytes: true
        });
        myResult = "http://" + myDomain + "/ws/tsite.xml?Type = " + myType +
        "&User = " + Crypto.util.bytesToHex(H1) + "&HMACKey = " + Crypto.util.
        bytesToHex(H2) + "&Request = " + escape(myRequest);
        document.getElementById("finalResult").innerHTML = myResult;
</script>
</body>
</html>
```

A3 TEXT COMMUNICATIONS

Text Communications facilitate the control of large commercial buildings and unitary controllers by allowing the building control system to be treated as a database parameter values, and be read or written. Text Communications allow:

- Flexible grouping of controllers, without any hardware or inter-wiring imposed limitations.
- Controller groupings to overlap.
- Immediate access to plain language data in response to plain language requests.

Devices on the system can be referenced using codes (Absolute Addressing), or by using labels (Attribute Addressing). module on the system can be referenced using codes (Absolute Item Selection), or by using labels (Label Matching Item Selection) This is an advantage in the engineering of large systems, and in the user-friendliness of the system to the HVAC engineer. Trend supervisor and tools use text communications in their communications with other Trend devices. Some Trend tools also allow you to create and send your text communications.

A3.1 Addressing and Item Selection

Addressing in text communications refers to the method used to specify the controller(s) to which a text communications message applies and can be done using either absolute addressing or attribute addressing.

Item selection in text communications refers to the method used to specify the module(s) to which a text communications message applies and can be done using either absolute item selection or label matching item selection.

A3.1.1 Absolute Addressing/Item Selection

Absolute addressing is the selection of the module and parameter in the controller, using a code.

A3.1.2 Attribute Addressing

Attribute addressing allows controller(s) to which a text communications message will apply to be specified using a user friendly text string (attribute). A text communications message will apply to controller if the value of one or more of its attribute(s) match the string specified in the text communications message's device attribute argument.

For a controller the following parameters are used as attributes:

'Identifier'
'Attribute 2'
'Attribute 3'
'Attribute 4'
'Attribute 5'
'Attribute 6'
'Attribute 7'

Note: When matching the attribute's value with the string in the text communications message an exact match is required, case is taken into account i.e. 'Outside Air' will match 'Outside Air' but not 'Outside air'.

A3.1.3 Label Matching Item Selection

Label matching enables the module(s) to which a text communications message will apply to be specified using the module's 'Label' parameter. A text communications message will apply to module if the value of its 'Label' parameter matches the string specified in the appropriate argument in the text communications message.

Note: When matching the attribute's value with the string in the text communications message an exact match is required, case is taken into account i.e. 'Outside Air' will match 'Outside Air' but not 'Outside air'.

Note: To avoid selection problems, a label which consists of a single capital letter followed by a number may not be used as a module label in controllers, as this will be treated as an absolute reference and could result in the selection of the incorrect parameter.

A3.2 Text Communications Message Types

There are three types of text communications messages:

Targeted Messages
Global Messages
Visitor Messages

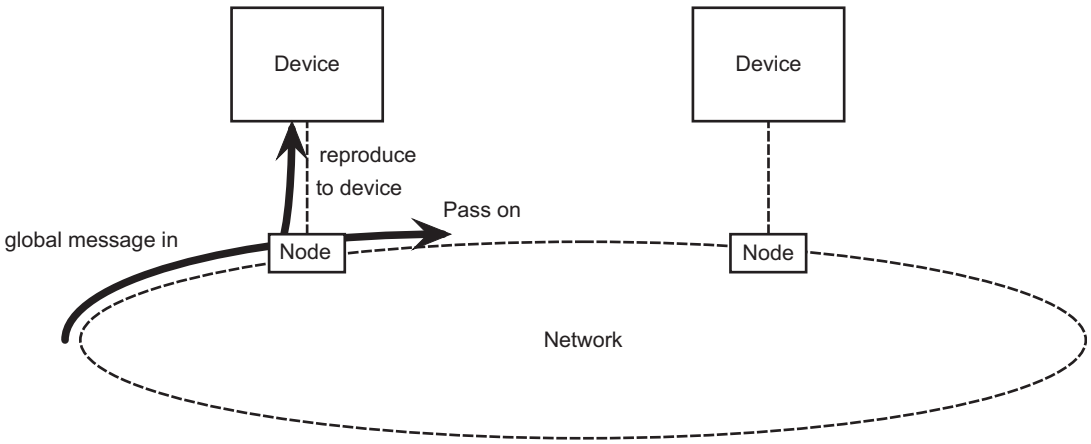
The type of message is determined by the way in which the message is addressed

A3.2.1 Targeted Messages

Targeted text communications messages are sent to a specific device.

A3.2.2 Global Messages

Global text communications messages are sent simultaneously to all devices on a LAN (global) or all devices on the network i.e all LANs (global global). Attribute addressing is then used to determine if the message is to apply to that device.



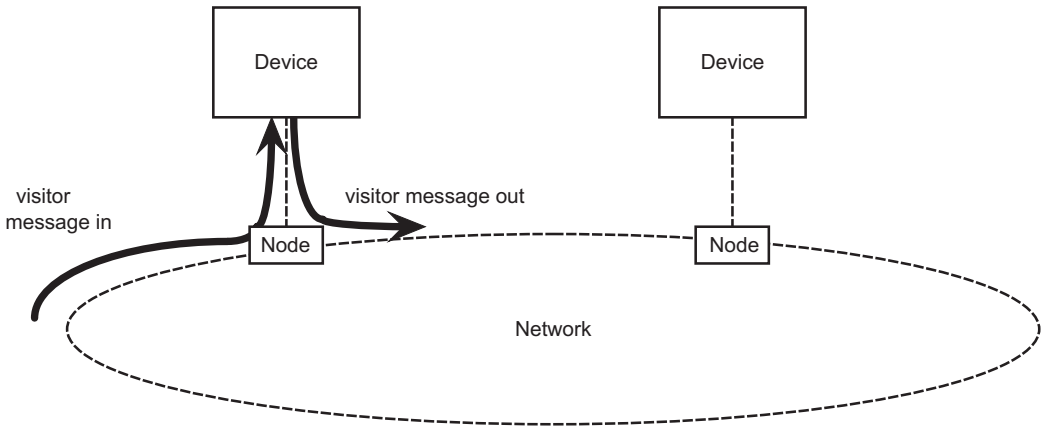
If this were a global request for values, each controller which had the relevant attributes would reply with a value, so a number of replies will normally be received.

Note: A global global branches at the INC type node so that it is received by each LAN on the internetwork at the same time.

A3.2.3 Visitor Messages

Visitor text communications messages are sent sequentially to all devices on a LAN. Attribute addressing is then used to determine if the message is to apply to that device. If the message applies to the controller it will process it an add its response to the message before passing kit to the next device.

The visitor message must go to each controller to have its device attributes checked, but the value it carries must be updated in sequence, so the message cannot be regenerated in parallel like a global message, instead it has to travel through each controller and back onto the network so that it visits each controller serially.



In the case of a visitor message only one reply is received.

This allows for the text communications function feature to be used to determine maximum, minimum and average values

For example if the function required is the minimum function. The requirement is satisfied by passing a large value (999) through each selected controller and if that controller has a value less than the current value it will overwrite it. Thus when the message is received back it will have the minimum value.

E.g.	Outstation		20	21	22	23	24	
	Value		30	29	32	28	28.5	
	Minimum	999	30	29	29	28	28	28
		Start						Finish

In the above example, as the message value passes through a controller, it only gets changed if the controller value is less than the message value (i.e. at controllers 20, 21, and 23). In this way the message returns a minimum value.

There are three types of function: maximum, minimum, and sum. In the case of sum each selected controller adds its value to the total carried by the message.

Note: Visitor text communications messages can only be sent to a single LAN.

A3.3 Addressing a Text Communications Message

The structure of the text communication message does not specify the devices to which the message will be sent, only the characteristics the device must for the message to apply. Therefore in addition to the text communications message the devices to which the message will be sent must me specified. This addressing a carried out by the supervisor/tool/device sending the message. The following information is included only to aid understanding of text communications.

A text communications message can be sent to depending on how it is addressed:

Specific device
All devices on a LAN
All devices on all LANs

A3.3.1 Specific Device

The text communications message is sent as a targeted message to a specific device determined by is LAN number and device address.

Note: 0 can be used to specify the local LAN.

A3.3.2 All Devices on a LAN

The text communications message is sent as either a global message or visitor message to all devices on a specific LAN.

To send the message as a global message it must be addressed to device address ‘0’ on the required LAN i.e LAN number must be specified and device address must be ‘0’.

To send the message as a visitor message it must be addressed to device address ‘127’ on the required LAN i.e LAN number must be specified and device address must be ‘127’.

Note: ‘0’ can be used to specify the local LAN.

A3.3.3 All Devices on all LANs

The text communications message is sent as a global global message. The message must be addressed to device address ‘0’ on LAN ‘128’ i.e LAN number must be ‘128’ and device address must be ‘0’.

Note: 0 can be used to specify the local LAN.

A3.4 Message Structure

This section describes the structure of the text communications message to allow creation of a message and understanding messages when fault finding.

There are two text communications protocols:

Traditional Text Communications (v comms)
Structured Text Communications (t comms)

A3.4.1 Traditional Text Communications (v comms)

Traditional text communications messages are in the following format:

PIN \ Device Attribute \ item selection {conditional} (data part) ; message data

Note: The tool used to enter the text communications message will add information to the message to define the message start and end, and the message header (which defines the message type). Without these additions the message will be unacceptable to the communications network.

A3.4.1.1 PIN Argument (optional)

The PIN argument is used to authorise any changes the message may attempt to make. It must match with the PIN of a Type 0 User Module in the target controller that has a high PIN level high enough to allow the change. A text communications message that changes Knob or and Switch module values must include a PIN of level equal to or greater than the level set up in the relevant module. It is only required if security is enabled in the controller(s). Controllers without Type 0 User Modules set up can be changed without specifying a PIN.

Note: If the message is targeted to multiple controllers they must all contain a Type 0 User Module with the specified PIN.

Note: When the message is initiated by a is entered into the tool or supervisor the tool or supervisor automatically includes a PIN. This PIN the one that it has been configured to use for that controller.

A3.4.1.2 Device Attribute Argument (optional)

The device attribute argument allows attribute addressing (see **“Attribute Addressing” on page 543**) to be used to limit the controller(s) to which the text communications message applies. The device attribute argument should be in the format below:

<Attribute String>

<Attribute String>: Specifies the string that must match with the value of the any ‘Identifier’, ‘Attribute 2’, ‘Attribute 3’, ‘Attribute 4’, ‘Attribute 5’, ‘Attribute 6’ and ‘Attribute 7’ parameter in the devices the text communications message is to apply, e.g.

NORTH

Selects all devices with one of their device attributes set to ‘NORTH’. Any controller without a match will not reply.

The * ? & | wildcards may be used - see **“Wildcards” on page 553**.

Note: The device attribute must be at least 2 characters long.

A3.4.1.3 Item Selection Argument (mandatory):

The item selection specifies the module(s) to which the text communications message applies. It allows the module(s) to be specified using either: Label Matching Item Selection (see **“Label Matching Item Selection” on page 543**), or Absolute Item Selection (see **“Absolute Addressing/Item Selection” on page 543**).

Label Matching Item Selection

If label matching item selection is used to specify the module(s) the string specified in the item selection argument is used for the match. The item selection argument is in the format:

<Label String>

<Label String>: Specifies the string that must match with the value of the label of any module in the devices the text communications messages is addressed to, e.g.

Outside Air Temp

Selects all modules with a label of ‘Outside Air Temp’.

The way the argument is matched is described in **“Addressing a Text Communications Message” on page 545**.

The * ? & | wildcards may be used - see **“Wildcards” on page 553**.

Absolute Item Selection

If absolute item selection is used to specify the module(s) the item selection argument must be in the format:

<Module>

<Module>: The type of module using its module identifier and module number. E.g.

S1

Specifies sensor 1.

All modules of one type may be selected by using the module identifier. E.g.

S

Specifies all sensors.

A range of modules of the same type can be specified by separating the first and last module in the range by a hyphen. E.g.

S1-3

Specifies sensors 1 to 3.

The module identifier for each module is available in **“IQ4 Modules” on page 165**.

Conditional Argument (optional)

A conditional statement may be added after the item selection to limit the number of parameters affected by the text communications message. It is of the format:

<Module Parameter><Condition><Value>

<Module Parameter>: Specifies the parameter that is to be used to limit the number of parameters affected by the text communications message using its text comms ID.

<Condition>: Specifies the condition that to be used to limit the number of parameters affected by the text communications message. It can be set to one of the following:

Condition	*Description*
<	The parameter affected if the specified parameter's value is less than the value specified by <Value>.
>	The parameter affected if the specified parameter's value is greater than the value specified by <Value>.
=	The parameter affected if the specified parameter's value is equal to the value specified by <Value>.

E.g.

S{V>19}(V)

Requests the values of all sensors with values greater than 19.

Note: Only = can be used with text or bit strings.

Logical operators may be used to combine the result of more than one condition.

Operator	*Description*
&	The conditions are AND'ed together to the result true if all conditions are true.
\| | The conditions are OR'ed together to the result true if at least one of the conditions are true. |

E.g.

S{V>25.0|V<19.0}(V)

Selects only those sensors whose values are outside the range 19 to 25.

S{V<25.0&V>19.0}(V)

Selects only those sensors whose values are within the range 19 to 25

The * ? wildcards may be used - see **"Wildcards" on page 553**.

E.g.

S{$ = "S* Fred"}($)

Only matches if the characters 'Fred' are at the end of the label.

The wildcard ? may be used in bit strings.

E.g.

S{M = OO??}(V)

Selects only those sensors without Read or Outside Limit alarms.

Data Part Argument (mandatory):
The data part specifies the parameter(s) to be read or written in the format:

<Module Parameter> = <Value>

Or

<Function>

<Module Parameter>: Specifies the parameter using its text comms ID.

Therefore:

V

Specifies the value of the specified module e.g S1(V) request the value of sensor 1.

Multiple parameters can be specified by separating the text comms ID for each parameter with a comma:

<Module Parameter>,<Module Parameter>,<Module Parameter>

E.g.

S1($,V,%)

Specifies the label, value, and units of sensor 1.

The wildcard * can be used to request all the parameters within an item, e.g:

S1(*)

Specifies all parameters in sensor 1.

Additional rules for data selection are as follows:

- D1(V) will select input level of a driver (input status if digital driver)
- Subsidiary properties of connectable inputs ('s' and 'v') in IQ1, IQ2 controllers can be selected using . (Sv) to select the source value, and (Ss) to select the source status. In IQ3 and IQ4 (S) selects the source value by default, but (Sv) will also work (e.g. S1(S) gives sensor 1 input value). (Ss) is not supported in IQ4.
- B1(S) selects whole bit pattern of Byte 1 (e.g. reply is B1(S = OOIIIIOI))
- B1(S0) selects the status of Byte 1, bit 0 (e.g. reply is B1(S0 = O) i.e. an array parameter)
- R(C) selects text string of controller type, prom issue, and date

= <Value>: The value of the specified parameter. It is included in the reply to a request for data but not the actual request e.g.

S1(V)

Will return a message similar to the one below:

S1(V = 33.2)

Where 33.12 is the value of sensor 1.

It must be included when the text communications message is to change the parameter's value to specify the new value e.g.

S1(V = 33.2)

Would change the value of sensor 1 to '33.2'.

Status bits are set by using letters O or I, e.g.

B1(S0 = I)

Sets Byte 1 bit 0 to '1'.

A byte is set by 8 letters O or I, e.g.

B1(S = OOIIOIII)

Sets Byte 1 to OOIIOIII

Labels can be set up, but the text must be included in quotes, e.g.

S1($ = "Space Sensor")

Sets the label of Sensor 1 to Space Sensor.

Note: Using text communications to change values should be undertaken with CAUTION. The user should ensure that they understand the message and its effects, especially when sending global messages.

Note: The new value must be within the configuration range.

Note: When sending text communications messages to IQ3s and IQ4s it is not possible to read data and write date with the same message (e.g. S1(V, V = 24), V). This is allowed when sending the message to an IQ2.

If the parameter has been changed successfully there will be a response of similar to the one shown below:

O18 L5 - Ack Received

If unsuccessful there will be a response of similar to the one shown below

O18 L5 - Nak Received

Note It is not possible to add modules using text communications.

If errors are found in the request the following error characters may be included in the = Value argument of the reply:

Error Character	***Description***	***Example***
?	Specified parameter does not exist	S999(?) - Sensor 999 does not exist.
		Z1(C3(?)) - array parameter, Time Schedule 1, exception 3, does not exist.
()	Specified item has no elements	Z1(W1()) - array parameter, Time Schedule 1 normal week 1, has no elements (change times).
X = inv	Specified parameter does not exist	S1(X = inv) - Sensor 1 does not have parameter X.

<Function>: Defines a function that is to be performed on the addressed data. The available functions allow the necessary data to be retrieved to allow supervisor or tool software to calculate the maximum or minimum values, or the sum total value of all addressed items. The format of <Function> is defined as follows:

<Module Parameter><Function><Value>

<Module Parameter>: Specifies the parameter the function is to be applied to parameter using its text comms ID.

<Function>: Specifies the function that is to be applied. The following functions are available:

Function	***Description***
<	Requests the replacement of the value in the 'message data' argument if the addressed value is smaller e.g. to obtain minimum.

Function	*Description*
>	Requests the replacement of the value in the 'message data' argument if the addressed value is greater e.g. to obtain maximum.
+<starting count>	Sums the values of all the parameter selected by the text communication message. e.g. to obtain average. <starting count> specifies the initial value of the sum.

E.g.

S1(V+0.0)

Sum of all values of sensor 1 starting with sum of 0.0

K1(V<25)

Minimum of all values of knob 1 if less than 25 (if no value below 25 the minimum remains at 25). The result of the function is returned in the data part of the message when it reaches the originating device.

When using a function in a text communications message there is an option for the reply to contain two extra fields; address and count in the message data argument - see **"Message Data Argument (optional)" on page 551**.

Note: Only one function can be included in a single message. Values with an associated read or sensor out of limits alarm will be inhibited from responding, only if {M = OO??} is sent within the conditional argument.

Note: Functions can only be applied to real or integer values.

Note: Text communications messages that use functions must be sent as a visitor message - see **"Visitor Messages" on page 544**.

The text comms ID for each parameter is available in **"IQ4 Modules" on page 165**.

A3.4.1.4 Message Data Argument (optional)

The message data argument can contain the address of the last device to contribute to the message, which in the case of maximum and minimum information is the address of the device having the maximum or minimum, and the number of items that contributed to the message depending, which in the case of a sum function can be used to calculate the average. The message data argument must be specified in the outgoing message for it to be included in the reply. It must be specified as below:

AD = 0,CT = 0

AD = 0 - Causes the address of the last device to contribute to the message to be included.
CT = 0 - Causes the number of items that contributed to the message to be included.

E.g.

WEST/S/SPACE TEMPERATURE/(V+0.0);CT = 0

Selects the sum of all sensors with the label 'SPACE TEMPERATURE' in controllers with the attribute 'WEST'. It also add the count of the number of contributions to the sum to the message data argument so that the average can be calculated. The reply will be of the form:

WEST/S/SPACE TEMPERATURE/(V+2313.33);CT = 100

Thus the sum is 2313.33, the number of contributions is 100, and the average is 23.13.

Multiple Items

More than one item parameter may be specified in a single text communications message by including the item selection, conditional and data part arguments the required number of time separated by a comma, e.g.

S1($),S2($),S3($)

Requests labels of sensors 1, 2, 3

A3.4.2 Structured Text Communications (t comms)

Structured text communications are required where a parameter contains a number of values. In this case they are considered to be in an array where each value (record) is numbered (1 to n where n is the number of records). The message structure of a structured text communications message is the same as that for a traditional text comms message with the following change to the data part. For structured text communications access array parameters the data part is in the following format:

<Record>(<Record Parameter>)

<Record> Specifies the record containing the required parameter using the record's text comms ID number. E.g.

B1

Specifies the record 1 from a Plot module.

<Record Parameter> Specifies the parameter within the record using the parameter's text comms ID

E.g.

V

Specifies the value of the specified record.

Therefore:

P1(B1(V))

Specifies the value of record 1 for Plot module 1 which has a number of records each an element in an array.

An item's array parameter may itself have array parameter. In this situation the data part is in the following format:

<Level 1 Record>(<Level 2 Record>(<Record Parameter>))

<Level 1 Record> Specifies the record containing the required parameter using the record's text comms ID and record number. E.g.

W1

Specifies standard week 1 in a Time Schedule module.

<Level 2 Record> Specifies the record inside the record specified by Level 1 Record containing the required parameter using the record's text comms ID and record number. E.g.

P1

Specifies change time 1 in the standard week of a Time Schedule module.

<Record Parameter> Specifies the parameter within the record using the parameter's text comms ID

E.g.

T

Specifies the actual time of a time change in the standard week of a Time Schedule module.

Therefore:

Z1(W1(P1(T)))

Specifies the time of change time 1 in standard week 1 of Time Schedule module 1.

A3.4.3 Wildcards

In addition to the simple matching of two text strings, wildcards can be used to enable the user to indicate that a pure match is not required, but some substitute characters are acceptable. They can also be used to combine the attributes required. The wildcards used are:

* ? .& |

***** Wildcard
The * is a wildcard character that can be included in an attribute to allow any character, no character, or group of characters to match that position in the initiator's attribute e.g.

controller	receiving controller
attribute **CANTEEN***	attribute **CANTEEN SOUTH**

In early versions of IQ4 firmware the controller would ignore any characters after the wildcard,characters. From v3.21 firmware the onwards characters after the wildcard are not ignored making it the same as that of IQ2 and IQ3 v3.06 or greater.

E.g. In early versions of firmware an attribute of 'AHU*Flow' would match an attribute of 'AHU 1 Damper' as the characters '1 Damper' after the wildcard are ignored. In version v3.21 firmware or greater it would not match as the characters '1 Damper' after the wildcard are not ignored.

An attribute of 'AHU*Damper' could be used to achieve a match in the receiving controller with an attribute of 'AHU 1 Damper'.

? Wildcard
The **?** is a wildcard character that can be included in an attribute to allow any single character to match that position in the initiator's attribute (there must be a character) e.g.

originating controller	IC comms receiving controller
attribute **CANTEEN?**	attribute **CANTEEN1**

& Wildcard
& is a special character performs a string AND function. When placed in an attribute, the message may only be delivered if the destination has two separate attributes that match the characters either side of the **&** e.g.

originating controller	receiving controller
attribute 2 **NORTH&OUT**	attribute 2 **NORTH**
	attribute 3 **OUT**

| Wildcard
| is a special character performs a string OR function. When placed in an attribute, the message may only be delivered if the destination has either or both of the two separate attributes that match the characters either side of the | e.g.

originating controller	receiving controller
attribute 3 **NORTH|OUT**	attribute 2 **NORTH**

A3.5 Examples of Text Communications

Example 1: To obtain the value from all Sensor 1 modules. It is sent as global global message. As the device attribute argument is not set up it will be acted on by all controllers.

Message arguments

PIN	
Device attribute	
Item attribute	S1
Conditional	
Data part	V

Message
R(D),S1(V)

Note: R(D) requests address module, identifier so replies can be associated with their controllers.

Replies
R(D = ”1FL3 A1 IQ91”),S1(V = 23.45)
R(D = ”1FL3 B14 IQ91”),S1(V = 23.45)
R(D = ”1FL3 C3 IQ92”),S1(V = 23.45) etc.

Example 2: To obtain the value of the ‘Label’, ‘Value, and ‘Units’ parameters for Sensor 5 module label from a single controller. It is sent as a targeted message to the required controller.

Message arguments

PIN	
Device attribute	
Item attribute	S5
Conditional	
Data part	$,V,% [label, value, units]

Message
R(D),S5($,V,%)

Note: R(D) requests address module, identifier so replies can be associated with their controllers.

Reply
R(D = ”1FL3 A1 IQ91”),S5($ = Space Temperature,V = 23.45,% = degC)

Example 3: To obtain the input level of Driver module 3 input level from a single controller. It is sent as a targeted message to the required controller.

Message arguments

PIN	
Device attribute	
Item attribute	D3
Conditional	
Data part	$,V,% [label, value, units]

Message
R(D),D3($,V,%)

Note: R(D) requests address module, identifier so replies can be associated with their controllers.

Reply

R(D = ”1FL3 A1 IQ91”),D3($ = Damper position,V = 45,% = %)

Example 4: To obtain input and output values of a Loop module from a single controller. It is sent as a targeted message to the required controller.

Message arguments

PIN	
Device attribute	
Item attribute	L3
Conditional	
Data part	P,D [input analog value, output analog value]

Message
R(D),L3(P,D)

Note: R(D) requests address module, identifier so replies can be associated with their controllers.

Reply

R(D = ”1FL3 6 IQ91”),L3(P = 19.7,D = 59.1)

Example 5: To change a digital bit, in all controllers in Office areas controlled by XYZ Co Ltd. It is sent as global global message using the device attribute argument to select controllers with attributes of both XYZ Co Ltd and Office. The PIN argument is included to authorise the change.

Message arguments

PIN	1234
Device attribute	XYZ Co Ltd&Office
Item attribute	
Conditional	
Data part	B13(S0 = I) [Byte xxx, make bit 0 equal to following value]

Message

1234\XYZ Co Ltd&Office\B13(S0 = I)

Reply

30 L6 - Ack Received

Note: The reply is only received if write with ACK is set (default)

Example 6: To search Sensors for Space Temperature label and obtain label and value. It is sent as global global message. As the device attribute argument is not set up it will be acted on by all controllers.

Message arguments

PIN	
Device attribute	
Item attribute	Space Temperature
Conditional	
Data part	$,V [label, value]

Message

R(D),Space Temperature ($,V)

Note: R(D) requests address module, identifier so replies can be associated with their controllers.

Replies

R(D = ”1FL3 A99IQ91”),S2($ = ”Space Temperature”,V = 21.4),S3($ = ”Space Temperature”,V = 21.5),
R(D = ”1FL3 A100IQ91”),S2($ = ”Space Temperature”,V = 21.3)

Example 7: To change the value of all Knob 2’s to 23.0. It is sent as global global message. As the device attribute argument is not set up it will be acted on by all controllers. The PIN argument is included to authorise the change

Message arguments

PIN	1234
Device attribute	
Item attribute	K2
Conditional	
Data part	V = 23.0

Message

K2(V = 23.0)

Replies

030 L6 - Ack Received
031 L6 - Ack Received
032 L6 - Ack Received etc.

Note: The reply is only received if write with ACK is set (default)

Example 8: To change the value of all modules with label of ‘Occ Set Point’. It is sent as global global message. As the device attribute argument is not set up it will be acted on by all controllers. The PIN argument is included to authorise the change.

Message arguments

PIN	1234
Device attribute	
Item attribute	Occ Set Point
Conditional	
Data part	V = 23.0

Message

R(D),Occ Set Point(V = 23.0)

Note: R(D) requests address module, identifier so replies can be associated with their controllers.

Reply

O21 L6 - Ack Received

Note: The reply is only received if write with ACK is set (default)

Example 9: Use of greater than condition to obtain the value of the ‘Label’, ‘Value, and ‘Units’ parameters of all modules with label of ‘Space Temperature’ and a value greater than ‘21.4’. It is sent as global global message. As the device attribute argument is not set up it will be acted on by all controllers.

Message arguments

Device attribute	
Item attribute	Space Temperature
Conditional	V>21.4
Data part	$,V,%

Message

R(D), Space Temperature{V>21.4}($,V,%)

Note: R(D) requests address module, identifier so replies can be associated with their controllers.

Reply

R(D = “1FL3 A1 IQ91”),S3($ = “Space Temperature”,V = 21.5, % = “DegC”)

A3.6 Differences Between IQ2 and IQ4 Text Communications

- IQ2 is able to select drivers by label matching. This is not supported on IQ4.
- IQ4’s Cannot mix reads and writes in the same message. E.g. In IQ2 it was possible to send a message:

  S1($,$ = ”new label”,$)

  In IQ4 this would have to be 3 separate messages:

  S1($)
  S1($ = ”new label”)
  S1($)

A4 BACNET PROPERTIES SUPPORTED

The following sections list the BACnet properties supported by IQ4.

BACnet Analogue Input Object Type A
BACnet Analogue Input Object Type B
BACnet Binary Input Object
BACnet Analogue Output Object
BACnet Binary Output Object
BACnet Analogue Value Object
BACnet Binary Value Object
BACnet Device Object
BACnet Notification Class Object
BACnet Schedule Object
BACnet Calendar Object
BACnet Trend Log Object

Note: The BP_OBJECT_NAME property in each of the objects is detailed as mapping to the module's 'Label' parameter. However, if the Type 13 (BACnet Application) Network module's 'Object Name Prefixes' parameter is set to '0' it will map to the module type and number followed by the label.

A4.1 BACnet Analogue Input Object Type A

The BACnet analogue input object maps to the IQ4's Type 0 (Analogue) and Type 1 (Internal Analog) Sensor modules.

<table>
<tr><th>BACnet Property</th><th>Data Type</th><th>Read/Write</th><th>Trend Parameter</th></tr>
<tr><td>Acked_Transitions</td><td>BACnetEventTransitionBits</td><td>R</td><td>No equivalent, 3 bit bit-string, always returns '1' for each bit.</td></tr>
<tr><td>COV_Increment</td><td>REAL</td><td>R/W</td><td>'COV Increment' parameter (C).</td></tr>
<tr><td>Deadband</td><td>REAL</td><td>R</td><td>2.5% of difference between the 'High Alarm Level' and 'Low Alarm Level' parameters (H) and (L).</td></tr>
<tr><td>Device_Type</td><td>CharacterString</td><td>R</td><td>'Input Type' parameter (I) as type number and text.
0: Analogue
1: Internal Analogue'</td></tr>
<tr><td>Event_Detection_Enable</td><td>BOOLEAN</td><td>R</td><td>No equivalent, always returns '1'</td></tr>
<tr><td>Event_Enable</td><td>BACnetEventTransitionBits</td><td>R/W</td><td>'Enabled' parameter of the Sensor module's alarm sub-modules as a digital bit string (left to right):
<table><tr><td>Bit 2</td><td>To normal enabled - Clear alarm enabled (any alarm enabled and associated clear alarm enabled for this sensor, i.e. (r1c) (o1c) or (h1c) and (l1c).</td></tr><tr><td>Bit 1</td><td>To fault enabled - Read or Out of Limits enabled i.e. (r1E) or (o1E)</td></tr><tr><td>Bit 0</td><td>To offnormal enabled - High or Low alarm enabled i.e. (h1E) or(l1E)</td></tr></table>A write writes to the corresponding alarm bits as described in the table above.</td></tr>
<tr><td>Event_State</td><td>BACnetEventState</td><td>R</td><td>'In Alarm' parameter with a different enumeration.
<table><tr><td>0</td><td>Normal</td><td>No alarm (normal)</td></tr><tr><td>1</td><td>Fault</td><td>Read or Out of Limits alarm</td></tr><tr><td>3</td><td>High</td><td>High Alarm</td></tr><tr><td>4</td><td>Low</td><td>Low Alarm</td></tr></table></td></tr>
<tr><td>Event_Time_Stamps</td><td>BACnetArray[3] of BACnetTimeStamp</td><td>R</td><td>'BACnet Event Time Stamps' parameter (n1), (n2, and (n3).
<table><tr><td>0</td><td>To offnormal (n1)</td></tr><tr><td>1</td><td>To fault (n2)</td></tr><tr><td>2</td><td>To normal (n3)</td></tr></table></td></tr>
<tr><td>High_Limit</td><td>REAL</td><td>R/W</td><td>'High Alarm Level' parameter (H).</td></tr>
<tr><td>Limit_Enable</td><td>BACnetLimitEnable</td><td>R/W</td><td>'Enabled' parameter (h1E) or (l1E) of the Sensor module's Low and High sub-modules as a digital bit string (left to right):
<table><tr><td>Bit 1</td><td>High alarm enabled - (h1E)</td></tr><tr><td>Bit 0</td><td>Low alarm enabled - (l1E)</td></tr></table></td></tr>
</table>

<table>
<tr><th>BACnet Property</th><th>Data Type</th><th>Read/Write</th><th>Trend Parameter</th></tr>
<tr><td>Low_Limit</td><td>REAL</td><td>R/W</td><td>‘Low Alarm Limit’ parameter (L).</td></tr>
<tr><td>Notification_Class</td><td>Unsigned</td><td>R/W</td><td>‘Alarm Group’ parameter of Sensor module’s alarm sub-modules (h1G), (l1G), (o1G), and (r1G). A write writes to all alarm sub-modules.</td></tr>
<tr><td>Notify_Type</td><td>BACnetNotifyType</td><td>R</td><td>No equivalent, always returns ‘0’.</td></tr>
<tr><td>Object_Identifier</td><td>BACntObjectIdentifier</td><td>R</td><td>Analogue input n where ‘n’ is module instance e.g. analogue input 1 for Sensor module 1.</td></tr>
<tr><td>Object_Name</td><td>CharacterString</td><td>R</td><td>‘Label’ parameter ($) e.g ‘Sensor 1’.</td></tr>
<tr><td>Object_Type</td><td>BACnetObjectType</td><td>R</td><td>Always ‘0’ (Analogue Input).</td></tr>
<tr><td>Out_Of_Service</td><td>BOOLEAN</td><td>R</td><td>No equivalent, always returns ‘0’.</td></tr>
<tr><td>Present_Value</td><td>REAL</td><td>R</td><td>‘Value’ parameter (V). A write always responds with ‘Write Access Denied’.</td></tr>
<tr><td>Reliability</td><td>BACnetReliability</td><td>R</td><td>‘Alarm State’ parameter (r1S) and (o1S) of the Sensor module’s Read and Out of Limits alarm sub-modules:
<table><tr><td>0</td><td>No fault detected</td><td>(r1S) and (o1S) both OFF</td></tr><tr><td>1</td><td>No sensor</td><td>(r1S) ON</td></tr><tr><td>7</td><td>Unreliable-other</td><td>(01S) ON</td></tr></table></td></tr>
<tr><td>Status_Flags</td><td>BACnetStatusFlags</td><td>R</td><td>Indicates the overall module status as a digital bit string.
<table><tr><td>Bit 3</td><td>Out of service</td><td>Always returns ‘0’</td></tr><tr><td>Bit 2</td><td>Overridden</td><td>‘Override Output’ parameter (^).</td></tr><tr><td>Bit 1</td><td>Fault</td><td>‘Alarm State’ parameter of the Sensor module’s Read and Out of Limits alarm sub-modules (r1S) or (o1S).</td></tr><tr><td>Bit 0</td><td>In alarm</td><td>‘Alarm State’ parameter of the Sensor module’s High and Low alarm sub-modules (h1S) or (l1S).</td></tr></table></td></tr>
<tr><td>Time_Delay</td><td>Unsigned</td><td>R/W</td><td>The parameter with the highest value of the ‘Alarm Delay (secs)’ parameter (h1D) and (l1D) of the Sensor module’s High, and Low alarm sub-modules. A write writes to both (h1D) and (l1D).</td></tr>
<tr><td>Time_Delay_Normal</td><td>Unsigned</td><td>R</td><td>The parameter with the highest value of the ‘Clear Alarm Delay (secs)’ parameter (h1d) and (l1d) of the Sensor module’s High, and Low alarm sub-modules.</td></tr>
<tr><td>Units</td><td>BACnetEngineeringUnits</td><td>R</td><td>‘Units’ parameter (%). The units that can be used with BACnet communications are limited - see “BACnet Units” on page 523.</td></tr>
</table>

The BACnet analogue input object type A supports COV and intrinsic reporting.

A4.2 BACnet Analogue Input Object Type B

The BACnet analogue input object maps to the IQ4's Type 2 (Digital) and Type 3 (Internal Digital) Sensor modules.

<table>
<tr><th>BACnet Property</th><th>Data Type</th><th>Read/Write</th><th>Trend Parameter</th></tr>
<tr><td>COV_Increment</td><td>REAL</td><td>R</td><td>'COV Increment' parameter (C).</td></tr>
<tr><td>Device_Type</td><td>CharacterString</td><td>R</td><td>'Input Type' parameter (I) as type number and text.
2: Digital
3: Internal Digital</td></tr>
<tr><td>Event_Detection_Enable</td><td>BOOLEAN</td><td>R/W</td><td>No equivalent, always returns '1'</td></tr>
<tr><td>Event_State</td><td>BACnetEventState</td><td>R</td><td>'In Alarm' parameter with a different enumeration.
<table><tr><td>0</td><td>Normal</td><td>No alarm (normal)</td></tr><tr><td>1</td><td>Fault</td><td>Read or Out of Limits alarm</td></tr><tr><td>3</td><td>High</td><td>High Alarm</td></tr><tr><td>4</td><td>Low</td><td>Low Alarm</td></tr></table></td></tr>
<tr><td>Object_Identifier</td><td>BACnetObjectIdentifier</td><td>R</td><td>Analogue input n where 'n' is module instance e.g. Analogue input 1 (for sensor 1)</td></tr>
<tr><td>Object_Name</td><td>CharacterString</td><td>R</td><td>'Label' parameter ($)</td></tr>
<tr><td>Object_Type</td><td>BACnetObjectType</td><td>R</td><td>Always '0' (Analogue Input)</td></tr>
<tr><td>Out_Of_Service</td><td>BOOEAN</td><td>R</td><td>No equivalent, always returns '0'.</td></tr>
<tr><td>Present_Value</td><td>REAL</td><td>R</td><td>'Value' parameter (V). A write always responds with 'Write Access Denied'.</td></tr>
<tr><td>Reliability</td><td>BACnetReliability</td><td>R</td><td>'Alarm State' parameter (r1S) and (o1S) of the Sensor module's Read and Out of Limits alarm sub-modules.
<table><tr><td>0</td><td>No fault detected</td><td>(r1S) and (o1S) both OFF</td></tr><tr><td>1</td><td>No sensor</td><td>(r1S) ON</td></tr><tr><td>7</td><td>Unreliable-other</td><td>(01S) ON</td></tr></table></td></tr>
<tr><td>Status_Flags</td><td>BACnetStatusFlags</td><td>R</td><td>Indicates the overall module status as a digital bit string.
<table><tr><td>Bit 3</td><td>Out of service</td><td>Always returns '0'</td></tr><tr><td>Bit 2</td><td>Overridden</td><td>'Override Output' parameter Sn(^)</td></tr><tr><td>Bit 1</td><td>Fault</td><td>'Alarm State' parameter of the Sensor module's Read and Out of Limits alarm sub-modules (r1S) or (o1S).</td></tr><tr><td>Bit 0</td><td>In alarm</td><td>'Alarm State' parameter of the Sensor module's High and Low alarm sub-modules (h1S) or (l1S).</td></tr></table></td></tr>
<tr><td>Units</td><td>BACnetEngineeringUnits</td><td>R</td><td>'Units' parameter (%). The units that can be used with BACnet communications are limited - see "BACnet Units" on page 523.</td></tr>
</table>

The BACnet analogue input object type B supports COV reporting but not intrinsic reporting

A4.3 BACnet Binary Input Object

The BACnet binary input object maps to the IQ4's Digital Input module.

<table>
<tr><th>BACnet Property</th><th>Data Type</th><th>Read/Write</th><th>Trend Parameter</th></tr>
<tr><td>Acked_Transitions</td><td>BACnetEventTransitionBits</td><td>R</td><td>No equivalent, 3 bit string, always returns '1' for each bit.</td></tr>
<tr><td>Alarm_Value</td><td>BACnetBinaryPV</td><td>R/W</td><td>Inverse of the 'Required State' parameter (R).</td></tr>
<tr><td>Device_Type</td><td>CharacterString</td><td>R</td><td>'Type' parameter (I) as type number and text.
0: External
1: Internal'</td></tr>
<tr><td>ElapsedActiveTime</td><td>Unsigned32</td><td>R/W</td><td>'Hours Run' parameter (H) converted to seconds. A write converts seconds to hours and writes to the 'Hours Run' parameter.</td></tr>
<tr><td>Event_Detection_ Enable</td><td>BOOLEAN</td><td>R/W</td><td>No equivalent, always returns '1'</td></tr>
<tr><td>Event_Enable</td><td>BACnetEventTransitionBits</td><td>R/W</td><td>'Enabled' parameter of the Sensor module's alarm sub-modules as a digital bit string (left to right):

<table><tr><td>Bit 2</td><td>To normal enabled</td><td>Clear alarm 'Enabled' parameter (d1c)</td></tr><tr><td>Bit 1</td><td>To fault enabled</td><td>Always '0'</td></tr><tr><td>Bit 0</td><td>To offnormal enabled</td><td>Alarm 'Enable' parameter (d1E).</td></tr></table>E.g. '0 1 0'

A write writes to the corresponding alarm bits as described in the table above.</td></tr>
<tr><td>Event_State</td><td>BACnetEventState</td><td>R</td><td>'In Alarm' parameter (!) with a different enumeration:

<table><tr><td>'0'</td><td>Normal - no alarm</td><td>! = 0</td></tr><tr><td>'2'</td><td>Offnormal - in alarm</td><td>! = 1</td></tr></table></td></tr>
<tr><td>Event_Time_Stamps</td><td>BACnetARRAY[3] of BACnetTimeStamp</td><td>R</td><td>'BACnet Event Time Stamps' parameter (n1), (n2), and (n3)

<table><tr><td>0</td><td>To offnormal (n1)</td></tr><tr><td>1</td><td>To fault (n2)</td></tr><tr><td>2</td><td>To normal (n3)</td></tr></table></td></tr>
<tr><td>Notification_Class</td><td>Unsigned</td><td>R/W</td><td>'Alarm Group' parameter (d1G) of the module's alarm sub-module.</td></tr>
<tr><td>Notify_Type</td><td>BACnetNotifyType</td><td>R</td><td>No equivalent, always returns '0'.</td></tr>
<tr><td>Object_Identifier</td><td>BACnetObjectIdentifier</td><td>R</td><td>Digital input n where 'n' is module instance e.g. binary input 1.</td></tr>
<tr><td>Object_Name</td><td>CharacterString</td><td>R</td><td>'Label' parameter ($).</td></tr>
<tr><td>Object_Type</td><td>BACnetObjectType</td><td>R</td><td>Always '3' (Binary Input)</td></tr>
<tr><td>Out_Of_Service</td><td>BOOLEAN</td><td>R</td><td>No equivalent, always returns '0'.</td></tr>
</table>

<table>
<tr><th>BACnet Property</th><th>Data Type</th><th>Read/Write</th><th>Trend Parameter</th></tr>
<tr><td>Polarity</td><td>BACnetPolarity</td><td>R</td><td>Always returns ‘0’ = normal.</td></tr>
<tr><td>Present_Value</td><td>BACnetBinaryPV</td><td>R</td><td>‘State’ parameter (S).</td></tr>
<tr><td>Reliability</td><td>BACnetReliability</td><td>R</td><td>‘Type’ (Y) and ‘Hardware Channel’ (c) parameters.
<table><tr><td>0</td><td>No fault detected - If (Y) = 1 internal, or (Y) = 0 external and ‘c’ = non zero (channel set up).</td></tr><tr><td>7</td><td>Unreliable-other - If (Y) = 0 external and (c) = 0 no channel.</td></tr></table></td></tr>
<tr><td>Status_Flags</td><td>BACnetStatusFlags</td><td>R</td><td>Indicates the overall module status as a digital bit string.
<table><tr><td>Bit 3</td><td>Out of service - always returns ‘0’</td></tr><tr><td>Bit 2</td><td>Overridden - ‘Override Output’ parameter (^)</td></tr><tr><td>Bit 1</td><td>Fault - ‘Alarm State’ parameter of the module’s alarm sub-module (d1S).</td></tr><tr><td>Bit 0</td><td>In alarm - ‘Alarm State’ parameter of the module’s alarm sub-module (d1S).</td></tr></table></td></tr>
<tr><td>Time_Delay</td><td>Unsigned</td><td>R/W</td><td>‘Alarm Delay (secs)’ parameter (d1D) of the module’s alarm sub-module. A write writes to this parameter</td></tr>
<tr><td>Time_Delay_Normal</td><td>Unsigned</td><td>R</td><td>‘Clear Alarm Delay (secs)’ parameter (d1d) of the alarm sub-modules.</td></tr>
<tr><td>BACnetDateTime</td><td>Time_Of_Active_Time_Reset</td><td>R</td><td>‘Time of Hours Run Reset’ parameter (t)</td></tr>
</table>

The BACnet binary input object supports COV and intrinsic reporting.

A4.4 BACnet Analogue Output Object

The BACnet analogue output object maps to the IQ4's Driver modules (except Type 1 (Digital)).

<table>
<tr><th>BACnet Property</th><th>Data Type</th><th>Read/Write</th><th>Trend Parameter</th></tr>
<tr><td>COV_Increment</td><td>REAL</td><td>R/W</td><td>'COV Increment' parameter (C).</td></tr>
<tr><td>Device_Type</td><td>CharacterString</td><td>R</td><td>'Type' parameter (Y) as type number and text.
2: Analogue
3: Time Proportional
4: Raise Lower
5: Binary Hysteresis
6: Time Proportional + Override
7: Raise Lower
8: Multistage</td></tr>
<tr><td>Event_State</td><td>BACnetEventState</td><td>R</td><td>Always '0: normal' - no alarm.</td></tr>
<tr><td>Object_Identifier</td><td>BACnetObjectIdentifier</td><td>R</td><td>Analogue output n where 'n' is module instance e.g. analogue output 1 (for driver 1).</td></tr>
<tr><td>Object_Name</td><td>CharacterString</td><td>R</td><td>'Label' parameter ($).</td></tr>
<tr><td>Object_Type</td><td>BACnetObjectType</td><td>R</td><td>Always '1' (Analogue Output).</td></tr>
<tr><td>Out_Of_Service</td><td>BOOL</td><td>R</td><td>No equivalent, always returns '0'</td></tr>
<tr><td>Present_Value (Commandable)</td><td>REAL</td><td>R/W</td><td>'Value' parameter (V). A write writes into the driver priority array at the priority level specified in the BACnet write message. If priority level is not specified it will write at level 16.</td></tr>
<tr><td>Priority_Array</td><td>BACnetPriorityArray</td><td>R</td><td>'BACnet Priority Array' parameter (o1 to o16).</td></tr>
<tr><td>Reliability</td><td>BACnetReliability</td><td>R</td><td>'In Alarm' parameter (!).
<table><tr><td>0</td><td>No fault detected</td><td>'In Alarm' parameter = 0</td></tr><tr><td>7</td><td>Unreliable - other</td><td>'In Alarm' parameter = 8 or 9</td></tr></table></td></tr>
<tr><td>Relinquish_Default</td><td>REAL</td><td>R</td><td>'Source' parameter (S).</td></tr>
<tr><td>Status_Flags</td><td>BACnetStatusFlags</td><td>R</td><td>Indicates the overall driver status as a digital bit string.
<table><tr><td>Bit 3</td><td>Out of service.</td><td>Always returns '0'.</td></tr><tr><td>Bit 2</td><td>Overridden</td><td>Driver overridden, 'Override Output' parameter (^) = '1' or 'HOA State' parameter (h) = '1' or '3' (in Hand or Auto).</td></tr><tr><td>Bit 1</td><td>Fault</td><td rowspan="2">Always returns '0'</td></tr><tr><td>Bit 0</td><td>In alarm</td></tr></table></td></tr>
<tr><td>Units</td><td>BACnetEngineeringUnits</td><td>R</td><td>No equivalent, always returns the enumeration for '%' which is '98'</td></tr>
</table>

The BACnet analogue output object supports COV reporting but not intrinsic reporting.

A4.5 BACnet Binary Output Object

The BACnet binary output object maps to the IQ4's Type 1 (Digital) Driver module.

BACnet Property	*Data Type*	*Read/Write*	*Trend Parameter*
Change_Of_State_Count	Unsigned	R/W	Number of Starts' parameter (N) x 2 plus 'Value' parameter (V).
Change_Of_State_Time	BACnetDateTime	R	'Time of Last Value Change' parameter (c)
Time_Of_State_Count_Reset	BACnetDateTime	R	'Time of Number of Starts Reset' (e)
Device_Type	CharacterString	R	'Type' parameter (Y) as type number and text. 1: Digital
Elapsed_Active_Time	Unsigned32	R/W	'Hours Run' parameter (H)
Event_State	BACnetEventState	R	Always 0, normal - no alarm
Object_Identifier	BACnetObjectIdentifier	R	Binary output n where 'n' is module instance e.g. binary output 1 (for driver 1)
Object_Name	CharacterString	R	'Label' parameter ($).
Object_Type	BACnetObjectType	R	Always '4' (Binary Output).
Out_Of_Service	BOOLEAN	R	No equivalent, always returns '0'.
Polarity	BACnetPolarity	R	'Invert' parameter (I).
Present_Value (Commandable)	BACnetBinaryPV	R/W	'Value' parameter (V). A write writes into the driver priority array at the priority level specified in the BACnet write message. If priority level is not specified it will write at level 16.
Priority_Array	BACnetPriorityArray	R	'BACnet Priority Array' parameter (o1 to o16).
Reliability	BACnetReliability	R	'In Alarm' parameter (!). 0 – No fault detected – 'In Alarm' parameter = 0 7 – Unreliable - other – 'In Alarm' parameter = 8 or 9
Relinquish_Default	BACnetBinaryPV	R/W	'Source' parameter (S).
Status_Flags	BACnetStatusFlags	R	This parameter declares overall driver status as a digital bit string. Bit 3 – Out of service. – Always returns '0'. Bit 2 – Overridden – Driver overridden, 'Override Output' parameter (^) = '1' or 'HOA State' parameter (h) = '1' or '3' (in Hand or Auto). Bit 1 – Fault – Always returns '0'. Bit 0 – In alarm
Time_Of_Active_Time_Reset	BACnetDateTime	R	'Time of Hours Run Reset' (t)

The BACnet binary output object supports COV but not intrinsic reporting.

A4.6 BACnet Analogue Value Object

The BACnet analogue value object maps to the IQ4's Knob module.

BACnet Property	*Data Type*	*Read/Write*	*Trend Parameter*
COV_Increment	REAL	R/W	'COV Increment' parameter (C).
Event_State	BACnetEventState	R	No equivalent, always returns enumeration '0' = normal.
Object_Identifier	BACnetObjectIdentifier	R	Analogue value n where 'n' is module instance e.g. analogue value 1 (for knob 1)
Object_Name	CharacterString	R	'Label' parameter ($).
Object_Type	BACnetObjectType	R	Always '2' (Analogue Value)
Out_Of_Service	BOOLEAN	R	No equivalent, always returns '0' = (module in service).
Present_Value	REAL	R/W	'Value' parameter (V). A write is only allowed when OUT_OF_SERVICE is true
Reliability	BACnetReliability	R	No equivalent, always returns enumeration '0' = no fault detected.
Status_Flags	BACnetStatusFlags	R	No equivalent, always returns a digital bit string of 4 bits, all set to '0' indicating all normal.
Units	BACnetEngineeringUnits	R/W	'Units' parameter (%). Limits the units that can be used with BACnet communications to those on the enumerated list - see **"BACnet Units" on page 523**.

The BACnet analogue value object supports COV reporting but not intrinsic reporting.

A4.7 BACnet Binary Value Object

The BACnet binary value object maps to the IQ4's Switch module.

BACnet Property	*Data Type*	*Read/Write*	*Trend Parameter*
Event_State	BACnetEventState	R	No equivalent, always returns enumeration '0' = normal.
Object_Identifier	BACnetObjectIdentifier	R	Binary value n where 'n' is module instance e.g. binary value 1 (for switch 1)
Object_Name	CharacterString	R	'Label' parameter ($).
Object_Type	BACnetObjectType	R	Enumeration always '5' = Binary Value'.
Out_Of_Service	BOOLEAN	R	No equivalent, always returns '0' = module in service.
Present_Value	BACnetBinaryPV	R/W	'Status' parameter (S). A write is only allowed when OUT_OF_SERVICE is true.
Reliability	BACnetReliability	R	No equivalent, always returns enumeration '0' = no fault detected'.
Status_Flags	BACnetStatusFlags	R	No equivalent, always returns digital bit string of 4 bits, all set to '0' indicating all normal.

The BACnet binary value object supports COV reporting but not intrinsic reporting.

A4.8 BACnet Device Object

The BACnet device object maps to the IQ4's Address, Program, Type 5 (BACnet IP) Network and Time modules.

BACnet Property	*Data Type*	*Read/Write*	*Trend Parameter*
Object_Identifier	BACnetObjectIdentifier	R	(I) parameter of Device n where n = Network 6, Device Instance - n6(i); the controller's address on the BACnet system e.g. device 6021 (for LAN 6, device 21 if calculated automatically.
Object_Name	CharacterString	R/W	If the Address module's 'Identifier' parameter (D), is set up it will map to it. If not set up it maps to the IQ4's Ethernet MAC address. Writes will write to the Address module's 'Identifier' parameter.
Object_Type	BACnetObjectType	R	Always '8' (Device).
Vendor_Identifier	Unsigned16	R	Always '91' = Trend.
Model_Name	CharacterString	R	Address module's 'Type Name' parameter (y).
Firmware_Version	CharacterString	R	Address module's 'Version' parameter - (C).
Application_Software_Version	CharacterString	R	
Location	CharacterString	R	Address module's 'Site Name' parameter - ($).
Database_Revision	Unsigned	R	Address module's 'Strategy Revision' parameter (d).
Protocol_Version	Unsigned	R	Not available as Trend Parameters
Protocol_Revision	Unsigned	R	
Protocol_Services_Supported	BACnetServicesSupported	R	
Protocol_Object_Types_Supported	BACnetObjectTypesSupported	R	
Object_List	BACnetArray[N] of BACnetObjectIdentifier	R	
Max_APDU_Length_Accepted	Unsigned	R	
Segmentation_Supported	BACnetSegmentation	R	
Max_Segments_Accecpted	Unsigned	R	
Local_Time	Time	R/W	Time module's 'Hours', 'Minutes' and 'Seconds' parameters (H), (N) and (C).
Local_Date	Date	R/W	Time module's 'Weekday', 'Day', 'Month' and 'Year' parameters (W), (D), (M) and (Y).
UTC_Offset	INTEGER	R/W	Time module's 'UTC Offset (Mins)' parameter (U).
Daylight_Savings_Status	BOOLEAN	R	Time module's 'Daylight Savings Applied' parameter (F).
APDU_Segment_Timeout	Unsigned	R	Not available as Trend Parameters
APDU_Timeout	Unsigned	R	
Number_Of_APDU_Retries	Unsigned	R	
Device_Address_Binding	BACnetList of BACnetAddressBinding	R	

<table>
<tr><th>BACnet Property</th><th>Data Type</th><th>Read/Write</th><th>Trend Parameter</th></tr>
<tr><td>System_Status</td><td>BACnetDeviceStatus</td><td>R</td><td>Program module’s ‘State’ parameter (P):
<table><tr><td>0</td><td>Download required</td><td>No strategy</td></tr><tr><td>1</td><td>Download in progress</td><td>Loading</td></tr><tr><td>2</td><td>Operational</td><td>Running</td></tr></table></td></tr>
<tr><td>Vendor_Name</td><td>CharacterString</td><td>R</td><td>Always ‘Trend Control Systems Ltd’.</td></tr>
</table>

A4.9 BACnet Notification Class Object

The BACnet notification class object maps to IQSET's BACnet Alarm Block - see **"BACnet Alarm Destinations (/BACnet variants and IQ4NC only)" on page 81**. Most properties in the BACnet Notification Class object map to parameters in the IQ4's Alarm Group module. The BACnet Notification Class object's BP_RECIPIENT_LIST property within the Notification Class object is a list of BACnet destinations. Each element (recipient) in this list maps to a Type 4 (BACnet) Alarm destination module routed to from the Alarm Group module. Each recipient has sub-properties associated with the time scheduling of the alarm; these are catered for by the Type 4 (BACnet) Alarm destination module.

Note: Each recipient in the BP_RECIPIENT_LIST property is equivalent to a Alarm Route or Alarm Destination module combination. If the BACnet Notification Class Object has several recipients this is equivalent to the Alarm Group module being connected to several Alarm Route/Alarm Destination module combinations. Thus having created a BACnet Alarm Block on the strategy page, additional Type 4 (BACnet) Alarm destination modules can be added by adding more alarm blocks and editing them so that the Type 4 (BACnet) Alarm destination modules are linked to the same Alarm Group module.

BACnet Property	***Data Type***	***Read/Write***	***Trend Parameter***
Ack_Required	BACnetEventTransitionBits	R	No equivalent, always returns '0'. Trend does not have the facilities for user acknowledgements.
Notification_Class	Unsigned	R	The number of the Alarm Group module.
Object_Identifier	BACnetObjectIdentifier	R	Notification class n where 'n' is the number of the Alarm Group module e.g. notification class 1 for Alarm Group module 1.
Object_Name	CharacterString	R	Alarm Group module's 'Label' parameter ($).
Object_Type	BACnetObjectType	R	Always '15' (Notification Class).
Priority	BACnetArray[3] of Unsigned	R	The alarm priorities for the three types of alarm transitions all are set to 255.
Recipient_List	BACnetList of BACnetDestination	R/W	The list of BACnet devices to which the alarms will be sent. Each element in this list maps to an Alarm Destination module routed to from the Alarm Group module (direct mapping from the notification class). The order of devices in the list is the same as the order of Alarm Route modules attached to the Alarm Group.

Each element in the Recipient List has the following BACnet properties:

BACnet Property	***Data Type***	***Trend Parameter***
from Time	Time	The first occupied ON time for the day. Mapped to the BACnet Alarm Destination module's 'Start time' parameter. *Note: A time of 24:00 is sent on BACnet as 23:59:59.*
IssConfirmedNotifications	BOOLEAN	The inverse of the Alarm Destination module's 'Service Type' parameter (T).
process Identifier	Unsigned32	The Alarm Destination module's 'Process ID' parameter. This has to match the number in the receiving device which identifies the process for which the alarm is intended.
recipient	BACnetRecipient	The address on the BACnet system of the remote BACnet device. By preference this will be the Alarm Destination module's 'Device Instance' parameter (D), but if this is not set up it will be a combination of the 'Network Number' (N) and 'MAC Address' (M) parameters.
to Time	Time	The 'to Time' is the first OFF time for the day. Mapped to the BACnet Alarm Destination module's 'End time' parameter. *Note: A time of 24:00 is sent on BACnet as 23:59:59.*
transitions	BACnetEventTransitionBits	The Alarm Destination module's 'Transitions' parameter.
validDays	BACnetDaysOfWeek	Mapped to the BACnet Alarm Destination module's 'Valid days' parameter.

A4.10 BACnet Schedule Object

The BACnet schedule object maps to the IQ4's Time Schedule module.

BACnet Property	*Data Type*	*Read/Write*	*Trend Parameter*
Effective_Period	BACnetDateRange	R	Time Schedule Module's 'Start Date' and 'End Date' parameters. Both the start and end date of the effective period maybe either a specific date or an unspecified date. Nether start nor end date may be partially specified - e.g. 31/01/* is not allowed, but 31/01/2018 is allowed.
Exception_Schedule	BACnetArray [N] of BACnetSpecialEvent	R/W	Maps onto the Time Schedule, Exceptions All Trend 24:00 period values are read as BACnet 23:59:59
List_Of_Object_ Property_References	BACnetList of BACnetDeviceObjectProeprtyReference	R/W	If the schedule object (Time Schedule module output) is connected to no modules or internal Trend modules only, the controller returns a list containing no elements. If the schedule is connected to Trend modules which are BACnet objects (digital input (BACnet binary input), switch (BACnet binary value), or digital driver (BACnet binary output)), the controller returns a list containing the connected objects writing.
Object_Identifier	BACnetObjectIdentifier	R	Schedule n where 'n' is module instance e.g. schedule 1 (Time Schedule 1)
Object_Name	CharacterString	R	'Label' parameter ($)
Object_Type	BACnetObjectType	R	Always '17' (Schedule).
Out_Of_Service	BOOLEAN	R	No equivalent; always returns '0' = module in service)'.
Present_Value	Any Restricted to NULL BinaryPV, REAL or Integer	R	'Value' parameter (V).
Priority_For_Writing	Unsigned	R	'Priority For Writing' parameter.
Reliability	BACnetReliability	R	No equivalent, always returns '0' = no fault detected.
Schedule_Default	Any Restricted to NULL or one of BinaryPV, REAL or Integer - depending upon the 'TimeSchedule SpecificType' parameter.	R	'Default Value' parameter
Status_Flags	BACnetStatusFlags	R	No equivalent, always returns digital bit string of 4 bits, all set to '0' (indicating all normal)
Weekly_Schedule	BACnetArray[7] of BACnetDailySchedule	R/W	Maps to Time Schedule module's , Normal Week. All Trend 24:00 period values are read as BACnet 23:59:59 Write: Time values MUST be written to as BinaryPV. All 7 days MUST be written to. All BACnet 23:59 time values are written as Trend 24:00 period values.

A4.11 BACnet Calendar Object

BACnet Property	*Data Type*	*Read/Write*	*Trend Parameter*
Date_List	BACnetLIST of BACnetPropertyIdentifier	R	'Date List' parameter (D)
Object_Identifier	BACnetObjectIdentifier	R	Calendar n where 'n' is module instance.
Object_Name	CharacterString	R	'Label' parameter ($).
Object_Type	BACnetObjectType	R	Always '6' (Calendar).
Present_Value	BOOLEAN	R	'Module Status' parameter (S)

A4.12 BACnet Trend Log Object

The BACnet 'trend log' object maps to the IQ4's Plot module. Only Type 3 (Periodic) Plot modules are compliant with the requirements of the BACnet standard,

Note: The 'Enable' parameter must not be connected.

BACnet Property	***Data Type***	***Read/Write***	***Trend Parameter***
Object_Identifier	BACnetObjectIdentifier	R	Trend log n where 'n' is module instance e.g. trend log 1 (for plot 1)
Object_Name	CharacterString	R	'Label' parameter ($).
Object_Type	BAcnetObjectType	R	Always '20' (trend log)
Buffer_Size	Unsigned32	R	'Number Of Records' parameter (N).
Record_Count	Unsigned32	R/W	'Record Count' parameter (R).
Stop_When_Full	BOOLEAN	R	'Stop When Full' parameter(W)
Total_Record_Count	Unsigned32	R	'Total Record Count' parameter (C)
Event_Enable	BACnetEventTransitionBits	R	No equivalent, always returns '111'.
Event_State	BACnetEventState	R	Always returns '0' = 'Normal'
Log_Enable	BOOLEAN	R/W	'Enable' parameter (E). (Only available on Type 2 (Triggered) and Type 3 (Periodic) Plot modules). For Type 1 (Synchronised) Plot modules a read returns true, and writes are ignored.
Log_Interval	Unsigned	R/W	'Interval' parameter (I) for Type 3 (Periodic) Plot modules. Always '0' for Type 1 (Synchronised) and Type 2 (Triggered) Plot modules. *Note: The 'Interval' parameter is in seconds, BACnet Plot_interval is in hundredths of seconds.* Write is OK for Type 3 (Periodic) Plot modules returns 'Write Access Denied' for Type 1 (Synchronised) and Type 2 (Triggered) Plot modules.
Notification_Threshold	Unsigned32	R/W	'Notification Threshold' parameter (n)
Records_Since_Notification	Unsigned32	R	'Records Since Last Notification' parameter (r).
Last_Notify_Record	Unsigned32	R	'Last Notify Record' parameter (l).
Notification_Class	Unsigned	R	'Alarm Group' parameter (g).
Acked_Transtions	BACnetEventTransitionBits	R	No equivalent, returns '111'
Notify_Type	BACnetNotifyType	R	Notifications are always events; always returns '1'
Event_Time_Stamps	BACnetArray[3] of BACnetTimeStamp	R	Returns 255/255/yy-mm-ddThh:mm:ss The To-Normal timestamp maps to the Plot module's 'Time Of Last Notification' parameter (b).

<table>
<tr><th>BACnet Property</th><th>Data Type</th><th>Read/Write</th><th>Trend Parameter</th></tr>
<tr><td>Log_Buffer</td><td>BACnetList of BACnetLogRecord</td><td>R</td><td>Always returns ‘READ_ACCESS_DENIED’. BACnet ReadRange Service must be used to access the log buffer.</td></tr>
<tr><td colspan="4">BACnetLogRecord</td></tr>
<tr><td>Timestamp</td><td>BACnetDateTime</td><td></td><td>Plot Record sub-module’s ‘Timestamp’ parameter (T). Returns yy-mm-ddThh:mm:ss</td></tr>
<tr><td>LogDatum</td><td>CHOICE</td><td></td><td>Plot Record sub-module’s ‘Type ‘ parameter (Y).

<table><tr><td>0</td><td>BACnet log-status</td></tr><tr><td>1</td><td>BACnet boolean value</td></tr><tr><td>2</td><td>BACnet real value</td></tr><tr><td>9</td><td>BACnet time-change, real</td></tr></table></td></tr>
<tr><td></td><td>Value</td><td></td><td>If status (BACnet log-status):

<table><tr><td>0: log-disabled</td><td>Corresponds to Trend value 0</td></tr><tr><td>1: buffer-purged</td><td>Corresponds to Trend value 2</td></tr></table>If digital, Value = 0 or 1

If analogue, Value, is analogue

If time-change, Value is time change in secs</td></tr>
<tr><td>StatusFlags</td><td>BACnetStatusFlags</td><td></td><td>Plot Record sub-module’s ‘Status’ parameter (S).

<table><tr><td>Bit 3</td><td>Out of service - Log record status bit 7 Pn(Bm(S(S7)))</td></tr><tr><td>Bit 2</td><td>Overridden - Log record status bit 6 S(S6)</td></tr><tr><td>Bit 1</td><td>Fault - Log record status Pn(Bm(S(bits as below)))

if sensor/other, read or out of limits (S3 or S2)

if driver, maintenance or readback (S0 or S1)</td></tr><tr><td>Bit 0</td><td>In alarm - Log record status S(S0, S1, S2, or S3)</td></tr></table></td></tr>
</table>

A4.13 BACnet over MS/TP

IQ4NCs in Ethernet Internetwork to MS/TP LAN or Current Loop Internetwork to MS/TP LAN modes are capable of both BACnet over IP, and BACnet over MS/TP.

IQ4NCs have the BACnet capability of the /BAC variants over the Ethernet network, but have additional capability to enable it to act as an interface to an MS/TP network.

IQ4NCs are able to communicate with other BACnet devices over the MS/TP trunk. This allows them to access/adjust information in BACnet devices on MS/TP, and for BACnet devices to access/change data in the IQ4NC. They can be viewed both as a Trend device, and as a BACnet device because they perform both Trend and BACnet communications.

When operating in Ethernet Internetwork to MS/TP LAN or Current Loop Internetwork to MS/TP LAN mode IQ4NCs have the following additional modules:

Network module 5 - Type 12 (BACnet MSTP)
Option (MSTP LAN Card)
Option (INC)

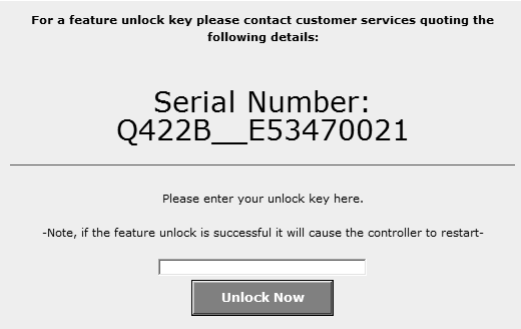
For full details of the objects, properties, and BIBBS supported by the IQ4NC see the IQ4NC/.../ BACnet Protocol Implementation Conformance Statement (TP201286).

A5 UPGRADE THE LICENCE

The IQ4's licence can be upgraded to add additional functionality e.g an IQ4E can be upgraded to have more I/O channels available.

To upgrade the licence:

1. Obtain an unlock key from Trend. This will normally be chargeable.
2. Login to the IQ4 using a web client.
3. Select **Modules**>**Options**.
4. Click **Add New Feature**.



5. Enter the unlock key.
6. Click **Unlock Now**.

Appendix 5 Upgrade the Licence

INDEX

Symbols

+Full analogue 24
5 Minute Pulse Parameter 490
6 Hour Pulse Parameter 490
10 Minute Pulse Parameter 490
15 Minute Pulse Parameter 490
20 Minute Pulse Parameter 490
24 Hour Pulse Parameter 490
30 Minute Pulse Parameter 490
60 Minute Pulse Parameter 490

A

About This Manual 17
Absolute Controller Selection 284
Absolute Item Selection 287
Access
- Plot Data 443
- Web Pages 102
- XML Web Services 104

ACK Transactions Parameters 199, 459, 462
Active Priority Level Parameter 234
Actual Type Parameter 330
Adaption Inhibit Parameter 423
Add
- I/O Modules 121
- Modules 25, 121

Additional 24 Vdc Supply 129
Address Module 167–172
- Connectives 172
- Parameters 167
- Set up 118

Address Parameter 174, 315, 317, 330, 387, 518
Advanced Scheduling Parameter 506
Alarm Bit Handling 289
- Schedule Offset Modules 451

Alarm Delay (secs) (High) Parameter 459, 462
Alarm Delay (secs) (Low) Parameter 459, 462
Alarm Delay (secs) (maintenance) Parameter 234
Alarm Delay (secs) (Out of Limits) Parameter 459, 462
Alarm Delay (secs) Parameter 199
Alarm Delay (secs) (PV Fail) Parameter 383
Alarm Delay (secs) (readback) Parameter 234
Alarm Delay (secs) (Read) Parameter 459, 462
Alarm Delay (secs) (SP Deviation) Parameter 383
Alarm Delivery Failure 173
Alarm Destination Modules 173–178
- Common Parameters 177
- Connectives 178
- Types 173

Alarm Forwarding 91
Alarm Group (High) Parameter 459, 462
Alarm Group (Low) Parameter 459, 463
Alarm Group (maintenance) Parameter 234
Alarm Group Modules 179–180
- Connectives 179
- Parameters 179

Alarm Group (Out of Limits) Parameter 459, 463
Alarm Group Parameter 185, 199, 391, 400, 442
Alarm Group Parameters 393
Alarm Group (PV Fail) Parameter 383
Alarm Group (readback) Parameter 234
Alarm Group (Read) Parameter 459, 463
Alarm Group (SP Deviation) Parameter 383
Alarm Handling Strategy 121
Alarm Hysteresis 456
Alarm Label 1 Parameter 311
Alarm Label 2 Parameter 311
Alarm Label 3 Parameter 311
Alarm Label 4 Parameter 311
Alarm Label 5 Parameter 311
Alarm Label 6 Parameter 311
Alarm Log Module 181–184
- Add/Removing Alarms 181
- Format 182
- High Water Mark 181
- Parameters 182

Alarm Reporting Lan Parameter 311
Alarm Reporting OS Parameter 311
Alarm Retries 173
Alarm Route Modules 185–186
- Connectives 185
- Parameters 185

Alarms 71–94
- BACnet 65
- Digital Input Modules 195
- Driver Modules 207
- Function Modules 241
- Loop Modules 369
- Plot Modules 435
- Sensor Modules 455
- Type 1 (IQ LAN) Network Modules 391
- Type 1 (XNC) Interface Modules 310
- Type 4 (Ethernet IP) Network Modules 392
- Type 5 (BACnet IP) Network Module 397
- Type 12 (BACnet MSTP) Network Module 400
- Virtual CNC Modules 518

Alarm Sent (High) Parameter 459, 463
Alarm Sent (Low) Parameter 459, 463
Alarm Sent (maintenance) Parameter 234
Alarm Sent (Out of Limits) Parameter 459, 463
Alarm Sent Parameter 199
Alarm Sent (PV Fail) Parameter 383
Alarm Sent (readback) Parameter 234
Alarm Sent (Read) Parameter 459, 463
Alarm Sent (SP Deviation) Parameter 383
Alarm State (High) Parameter 459, 463
Alarm State (Low) Parameter 459, 463
Alarm state (maintenance) Parameter 235
Alarm State (Out of Limits) Parameter 460, 463
Alarm State Parameter 199, 460, 463
Alarm State (PV Fail) Parameter 383
Alarm State (readback) Parameter 235
Alarm State (Read) Parameter 460, 463
Alarm State (SP Deviation) Parameter 383
Alarm State Transitions 65, 457
- Digital Input Modules 196
- Plot Modules 435
- Sensor Modules 457

Alarm Status Parameter 182, 187
Alarm Type Parameter 182
Alarm Types 71
Algorithm Type Parameter 423
Align Parameter 205
Analogue and Binary BACnet Objects 66
Analogue Input Object 558, 560
Analogue Node Modules 187–188
- Connectives 188
- Parameters 187

Analogue Output Channel and Relay Module Operation 221
Analogue Output Object 563
Analogue Value Object 565
AND Function 340
Angle Type Parameter 265
Anti-phase Hardware Channel Parameter 235
Anti-phase Output 227
Anti-Phase Output (mV) Parameter 235
Anti-Phase Output Parameter 235
Application Key 105, 514, 527
Application Key Parameter 514
Archive Failure 76
Archive Failure Parameter 167
Archive State Parameter 449
Assign I/O Channels 122
Attribute 2 Parameter 167, 387
Attribute 3 Parameter 167, 387
Attribute 4 Parameter 167, 387
Attribute 5 Parameter 167, 387
Attribute 6 Parameter 167, 387
Attribute 7 Parameter 167, 387
Attribute Addressing 543

Attribute Parameter ... 305
Attribute Selection - Wildcards ... 285
Authentication Types ... 540
Autobaud ... 55
Autobaud Parameter ... 391
Automatic Daylight Saving Configuration ... 489
Automatic IP Addressing ... 50, 68
Automatic Strategy I/O Module Creation ... 329
Auxiliary Supply ... 114
B
Back Colour Parameter ... 201, 205
Backdrop Files ... 34
Backdrop Parameter ... 201
Backup and Restore ... 66
Backup_and_Restore_state Parameter ... 403
Backup Failure Timeout Parameter ... 402
Backup the Strategy ... 157
BACnet
  Backup and Restore ... 66
  User Alarm Acknowledgments ... 67
BACnet Address ... 68
BACnet Alarms ... 65
BACnet Alarm State Transitions
  Digital Inputs ... 196
  Plot Modules ... 435
  Sensor Modules ... 457
BACnet Analogue Input Object ... 558
BACnet Analogue Input Object Type A ... 558
BACnet Analogue Input Object Type B ... 560
BACnet Analogue Output Object ... 563
BACnet Analogue Value Object ... 565
BACnet Backup/Restore Parameter ... 403
BACnet Binary Input Object ... 561
BACnet Binary Output Object ... 564
BACnet Binary Value Object ... 565
BACnet Broadcast Management Device ... 398
BACnet Communications ... 63–70
BACnet COV Reporting and Receiving ... 64
BACnet Device Object ... 566
BACnet Event Time Stamps Parameter ... 199, 235, 466
BACnet GetEventInformation Service ... 85
BACnet IC Comms ... 284
BACnet IP Communications ... 63
BACnet IP Network Module ... 397
BACnet Notification Class Object ... 568
BACnet Object Codes ... 519
BACnet Object/Property Codes and Units ... 519–526
BACnet only references Parameter ... 506
BACnet over MS/TP ... 69, 573
BACnet Preparation Timeout Parameter ... 403
BACnet Priority Array Parameter ... 235
BACnet Properties ... 69
  Codes ... 519
  Supported ... 557–574
BACnet Schedule Object ... 66, 569
BACnet Security ... 514
BACnet Services Password ... 403
BACnet Trend Log Object ... 570, 571
BACnet with Automatic IP Addressing ... 68
BACnet Writable Properties Parameter ... 403
BACnet Write Priority Parameter ... 306
Baseboard version Parameter ... 167
Baud Rate ... 399, 406
Baud Rate Parameter ... 391, 400
BBMD ... 68, 398
BBMD Parameter ... 398
BBUF ... 77, 435, 436
BDT Remote IP Address Parameter ... 398
BDT Remote Subnet Parameter ... 398
BDT Remote UDP Parameter ... 398
BIBBS ... 69
Binary Input Object ... 561
Binary Output Object ... 564
Binary Value Object ... 565
Bit 0 Parameter ... 193, 260, 352
Bit 1 Parameter ... 193, 260, 352
Bit 2 Parameter ... 193, 260, 352
Bit 3 Parameter ... 193, 260, 352
Bit 4 Parameter ... 193, 260, 352
Bit 5 Parameter ... 193, 260, 352
Bit 6 Parameter ... 193, 260, 352
Bit 7 Parameter ... 193, 260, 352
Bootloader version Parameter ... 167
BO State Count And Active Time Properties Parameter ... 403
Bottom range Parameter ... 334
Boundless Operation ... 222, 228
brIQ capacity Parameter ... 167
brIQs used Parameter ... 167
BTL Compliance ... 63, 117, 401
  Plot Modules ... 435
BTL Compliance Parameter ... 403
BTL Compliant Strategy
  Creating ... 117
  Requirements ... 117
Buffer Ready ... 436
Building an Internetwork on a Trend Current Loop Network ... 57
Building an Internetwork on Ethernet ... 40
Building an Internetwork on Ethernet Across Routers ... 41, 135
Building LANs on a Trend Current Loop Network ... 56
Building LANs on Ethernet ... 40
Building LANs on MS/TP Trunk ... 58
Build Parameter ... 167
Byte Order ... 317
C
Calculate
  I/O Module Power Requirements ... 129
  Power the IQ4E can Supply to the I/O Bus ... 129
  Power Used by the Combined Supply ... 116
Calendar Reference Parameter ... 508
Card ID Parameter ... 429
Cascade Control ... 377
Cascade Operation (Type 12 (Digital Selector) Logic Modules ... 356
Casual Users ... 103
CDGT ... 77
CDI0 ... 77
CDI1 ... 77, 195
Changes Using Text Communications ... 102
Check
  Current Loop Network Indicators ... 135
  Input & Output Operation ... 136
  I/O Bus Indicator ... 159
  I/O Module Addressing ... 161, 162
  LAN OK Indicator ... 136
  MS/TP Trunk Indicators ... 135
  Power Indicator ... 132, 159
  Supply Voltage ... 159
  Watchdog Indicator ... 132, 159
Checksum OK Parameter ... 168
CHIH ... 77
Clear Alarm Delay (secs) (High) Parameter ... 460, 464
Clear Alarm Delay (secs) (Low) Parameter ... 460, 464
Clear Alarm Delay (secs)(maintenance) Parameter ... 236
Clear Alarm Delay (secs) (Out of Limits) Parameter ... 460, 464
Clear Alarm Delay (secs) Parameter ... 199
Clear Alarm Delay (secs) (PV Fail) Parameter ... 384
Clear Alarm Delay (secs) (readback) Parameter ... 236
Clear Alarm Delay (secs) (Read) Parameter ... 460, 464
Clear Alarm Delay (secs) (SP Deviation) Parameter ... 384
Clear Alarm (High) Parameter ... 460, 463
Clear Alarm (Low) Parameter ... 460, 463
Clear Alarm (maintenance) Parameter ... 236
Clear Alarm (Out of Limits) Parameter ... 460, 463
Clear Alarm Parameter ... 199
Clear Alarm (PV Fail) Parameter ... 383
Clear Alarm (readback) Parameter ... 236
Clear Alarm (Read) Parameter ... 460, 464
Clear Alarm (SP Deviation) Parameter ... 383
Clickable Parameter ... 204
Clip to Limits Parameter ... 334
CLOW ... 77
CMNT ... 77
Cold & Warm Restart ... 403

Combination J Parameter 340
Combination K Parameter 340
Combination M Parameter 340
Combined Supply 116
Common IQ4 Module Parameters 165
Comms Direction Parameter 306
Communication Channel 399
Communication Channel Mode Parameter 408
Communication Channel Parameter 405, 406, 407, 408
Communication Channel Status 399
Communication Channel Status Parameter 405, 406, 407, 408
Communication Modes 59
Communications 39–62
Compatibility with IQ3 TCL 108
Compliance with BACnet Testing Laboratory Requirements 63
Configurable User Levels 512
Configure
  I/O Modules 159–162
  IQ4 131–158
  Modules 26
  SSL Certificates 136
  Strategy 25, 118
  Web Servers 124
Configured Type Parameter 331
CONL 168
Connect
  3rd Party System 114
  Auxiliary Supply 114
  Current Loop Network 114
  Displays/Supervisors/Tools 52, 136
  Ethernet 113
  Inputs 113, 128, 129
  I/O Bus 127, 128
  I/O Channels 26
  Modules 25
  MS/TP LAN to Internetwork 59
  MS/TP Trunk 114
  Outputs 113, 128, 129
  PC/Laptop 132
  Power 113, 127, 128
  Wallbus Devices 55, 114
  Web Client 48
Connected IP Address Parameter 518
Connection Status 409
Connection Types 25
Connectives 172, 178, 179, 185, 188, 193, 200, 240, 278, 307, 328, 332, 335, 358, 386, 485, 493, 510
Controller on line 76
Controller Online Parameter 168
Controller Selection 284
Cooldown Limit Parameter 423
Cooldown State Parameter 423
Cooling and Heating 381
Cooling Gain Time Constant 418
Cooling Mode 411
Corrupt Strategy 76
Corrupt Strategy Parameter 168
Corrupt Time Parameter 168
Counter Logic Modules 345
COUT 77
COV Increment Parameter 236, 334, 442, 467
COV Mode Parameter 302, 305
COV Operation 197, 333
  Driver Modules 212
  Sensor Modules 457
  Switch Modules 484
COV Receiving 64
COV Reporting 64
COV Service 282
CPU Usage (%) Parameter 431
CPVF 77
Create
  SSL Certificates 137
  Strategy 117–126
  Strategy File 125
  User Modules 513
Create/Delete Service 68, 402
Create/Delete Service Parameter 403
Critical Data 34
Cross-router Master 41
Cross Routers 42
CSDV 77
Cumulative Count Parameter 346
Cumulative Output Parameter 346
Cumulative Output (Type 4 (Counter) Logic Modules) 345
Current Line Parameter 311
Current Loop Baud Rate 135
Current Loop Communications 55
Current Loop Internetwork to MS/TP Lan 134
Current Loop Internetwork to MS/TP LAN 69
Current Loop Port 39
Current Time Parameter 490
Cut Screen Earth Link 115, 159

D

Data Parameter 505, 506
Data Precision Parameter 442
Data Types 23
Date List Parameter 191
Date Option parameter 509
Day Bit 5 Parameter 490
Day Bit 6 Parameter 490
Day Bit 7 Parameter 490
Day Bit Parameters 492
Daylight Saving 445, 487, 489
  Automatic Daylight Saving Configuration 489
  Disable 489
  Manual Daylight Saving Configuration 489
Day of Week Parameter 509
Day of Year Parameter 490
day Parameter 182
Deadband 368
Deadband Parameter 384
Debug Active Parameter 311
Default Action Parameter 384
Default Input 275
Default Input Parameter 356
Default Status Parameter 484
Default Value Parameter 334, 355, 384, 506
Delay Logic Modules 347
Delivery Failure (Alarm) 173
Derivative Parameter 384
Derivative Time (mins) Parameter 384
Description Parameter 429
Destination D 275
Destination d Parameter 354
Destination D Parameter 270, 357
Destination Parameter 174, 185, 244, 246, 247, 249, 250, 251, 252, 253, 254, 255, 256, 257, 261, 262, 263, 264, 265, 267, 269, 334, 357, 484
Destination Port Parameter 175
Developing a Security Program 99
Device Dead 74
DeviceDead Alarm Enable parameter 518
Device Discovered Parameter 168
Device Firmware Version Parameter 315
Device Instance 68
Device instance Parameter 176, 387, 404
Device Object 566
Device Online 74
Device Profile Parameter 315, 318, 326
Device Type Parameter 315
Dew point Parameter 272
DHCP 50
  Alarms 75
  Fixing Address 51
DI=0 77
DI=1 ..... 77, 195
Differences Between IQ4 and IQ2 IC Comms 307
DIGIN alarms 195
Digin Parameter 199
Digin (reqd state=0) 77
Digin (reqd state=1) 77

Digital Byte Modules ... 193–194
Connectives ... 193
Parameters ... 193
Digital Data Types ... 24
Digital Input Modules ... 195–200
Alarms ... 195
BACnet Alarm State Transitions ... 196
Common Parameters ... 199
Connectives ... 200
COV Operation ... 197
Hours Run ... 195
Module Disable ... 197
Number of Starts ... 195
State ... 195
Types ... 197
Use with Type 6 (Hours Run) Logic Module ... 196
Directory Modules ... 201–202
Parameters ... 201
Disable
Daylight Saving ... 489
Modules ... 29
Web Servers ... 124
Disabled Modules ... 480
Disable Module Parameter 165, 315, 318, 327, 391, 398, 400, 405, 406
Disable Network Use Parameter ... 515
Disaster Recovery Planning ... 99
Display and Directory Modules ... 31
Set up ... 124
Supervisor/Tool Access ... 32
Web Client Access ... 31
Display Modules ... 203–206
Common Parameters ... 205
Types ... 203
Display Subtype Parameter ... 205
DNS Server ... 50
DNS Server 1 Parameter ... 393
DNS Server 2 Parameter ... 393
DNS Server 3 Parameter ... 393
DNS Server 4 Parameter ... 393
Double Precision
Data Types ... 24
Logs ... 444
Plot Modules ... 436
Sensor Modules ... 458
Download ... 49, 55, 57, 102, 512
Configuration Files ... 49, 57
Exclusions ... 136
Security ... 102
Strategy ... 136
Strategy Files ... 53, 54
Drive Position Parameter ... 223, 229
Driver Modules ... 207–240
Alarms ... 207
Common Parameters ... 234
Connectives ... 240
COV Operation ... 212
Hand Off Auto (HOA) Operation ... 211
Hours Run ... 210
Invert ... 210, 220, 228
Maintenance Interval Alarm ... 208, 209
Module Disable ... 209
Module Override ... 210
Offset and Range ... 214
Readback Alarm ... 207
Response to Changing Inputs ... 217
Reverse Operation ... 220
Types ... 212
Duplicate
Addresses ... 74, 161
LAN ... 74
DVDD ... 73
DVOK ... 73
Dynamic Text Display Modules ... 204
Parameters ... 204
E
Earth Link Track ... 115
Edit Module Parameters ... 26, 122
Edit Timeout Parameter ... 507
Effective Period ... 495
Effect of Daylight Saving ... 445
Effect of Time Synchronisation ... 445
Email Alarms ... 48
Destinations ... 81
Email Min TLS version parameter ... 394
Email Password Parameter ... 393
Email Port Parameter ... 393
Email Security Parameter ... 394
Email Server Address ... 50
Email Server Address Parameter ... 394
Email Username Parameter ... 394
Enabled (maintenance) Parameter ... 236
Enabled Parameter ... 199
Enabled (PV Fail) Parameter ... 384
Enabled (readback) Parameter ... 236
Enabled (SP Deviation) Parameter ... 384
Enable (High) Parameter ... 460, 464
Enable (Low) Parameter ... 460, 464
Enable (Out of Limits) Parameter ... 460, 464
Enable Parameter ... 185, 442
Enable (Read) Parameter ... 460, 464
End Day Parameter ... 507, 509
End Month Parameter ... 507, 509
End Year ... 509
End Year Parameter ... 507
Enforce Single Write Function Codes Parameter ... 318
Enthalpy Parameter ... 272
Enum Data Types ... 24
Ethernet
Access Across Routers ... 135
Building Internetworks ... 40
Building LANs ... 40
Communications ... 40
Connect ... 113
Network Planning and Security ... 100
Port ... 39
Prepare Network ... 131
Ethernet Internetwork to Current loop Internetwork ... 134
Ethernet Internetwork to Current Loop Lan ... 134
Ethernet Internetwork to MS/TP Lan ... 134
Ethernet Internetwork to MS/TP LAN ... 69
Ethernet MAC Parameter ... 394
Ethernet Port ... 40
Ethernet to MS/TP ... 389
Ethernet to MS/TP Mode ... 59
Ethernet to Trend Internetwork ... 389
Ethernet to Trend Internetwork Mode ... 61
Ethernet to Trend LAN ... 389
Ethernet to Trend LAN Mode ... 60
Event Driven Strategy ... 28
Feedback ... 29
Service Order ... 29
Set up ... 122
Type 2 (Triggered) Plot Modules ... 439
Event Time Stamp ... 196, 457
Exceptions ... 498
Exceptions 1 Parameter ... 507
Exceptions Parameter ... 507
External Analogue Sensor ... 458
External Digital Inputs and Drivers with Type 6 (Hours Run) Logic Modules ... 348
External IO Count Parameter ... 168
External IO Enabled Parameter ... 168
External IO Limit Parameter ... 168
Extra Text Parameter ... 175
F
Factory Defaults ... 35
Failed Bit ... 282
Failed Bit Parameter ... 306
Failed Parameter ... 177, 328, 331
Failed Peripheral Interface Adapter Parameter ... 168
Failed PROM Parameter ... 168
Failed RTC ... 76

Failed RTC Parameter ... 168
Failed Software Parameter ... 168
Falling Edge Enable Parameter ... 353
FARC ... 76
Fault Indications ... 159, 160
Feedback ... 29
Feedback Parameter ... 223, 229
Feed Forward Control ... 379
File Error Alarm ... 76
File Error Parameter ... 168
File List Parameter ... 168
Filename Parameter ... 315, 449
Filter Factor ... 243
Fingerprint Parameter ... 449
Firmware ... 23–24
  Upgrade ... 24
Firmware Version Parameter ... 331
Fixed User Levels ... 511
Fixing Address on DHCP System ... 51
FLER ... 76, 168
Font Name Parameter ... 205
Font Size Parameter ... 205
Fore Colour Parameter ... 205
From Email Address Parameter ... 175
FRTC ... 76, 168
Full Analogue Data Types ... 24
Full Scale Drive Parameter ... 223, 229
Function Modules ... 241–278
  Alarms ... 241
  Common Parameters ... 277
  Connectives ... 278
  Module Disable ... 241
  Types ... 242

G

Gain Parameter ... 384
General Alarm Group Parameter ... 169
Grace Time Parameter ... 344, 347
Graphical Display Pages ... 96
GraphIQs ... 31. See Graphical Display Pages
group Parameter ... 182
Guest user ... 103, 123

H

Hand Off Auto (HOA) Operation ... 211
Hardware ... 37
Hardware Channel Parameter ... 198, 461, 465
Hardware I/O Module Address
  Check ... 161
  Set ... 160
Hardware Module Parameter ... 198, 236, 461, 465
Heating and Cooling ... 381
Heating and Cooling Loops ... 419
Heating and Cooling Mode ... 411
Heating Gain Time Constant ... 417
Heating Mode ... 411
Height Parameter ... 201, 272
HELP ... 168
HIGH ... 77
High Alarm ... 456
High Alarm Level Parameter ... 461, 464
High Alarm Parameter ... 187
High Parameter ... 461, 464
High Water Mark ... 181
HOA State Parameter ... 237
Hold Priority ... 80, 179
Hold Priority Parameter ... 177, 179
Home Page ... 513
Home Page Parameter ... 515
Host name ... 50
Hostname Parameter ... 394
Hours Parameter ... 182
Hours Run ... 195, 210
Hours Run Output Parameter ... 349
Hours Run Parameter ... 199, 237, 349
HTTPS Certificate Expiry Parameter ... 393
HTTPS Certificate Name Parameter ... 393
HTTPS Redirect Parameter ... 395
Hyperlinks ... 203
Hysteresis ... 456

I

IC Comms
  BACnet ... 284
  Controller Selection ... 284
  COV Service ... 282
  Failed Bit ... 282
  From Other Controllers ... 279
  Item Selection ... 286
  Module Disable ... 280
  Retries ... 282
  Set up ... 123
  To Other Devices ... 280
  Writes ... 283
  Writes Security ... 280
IC Comms Modules ... 279–308
  Alarm Bit Handling ... 289
  Common Parameters ... 305
  Connectives ... 307
  Types ... 288
Identifier Parameter ... 169, 387
ID Parameter ... 311
Ignore Email Certificate Parameter ... 394
In Alarm Colour Parameter ... 204
In Alarm Parameter ... 199, 237, 384, 461, 464
INC Alarm Forwarding ... 91
Inhibit Clear Parameter ... 178
Initiating IC comms to Other Devices ... 280
In-Phase Hardware Channel Parameter ... 237
In-Phase Output (mV) Parameter ... 237
In-Phase Output Parameter ... 237
Input 1 Parameter ... 478
Input 2 Parameter ... 478
Input 3 Parameter ... 478
Input 4 Parameter ... 478
Input 5 Parameter ... 478
Input 6 Parameter ... 478
Input 7 Parameter ... 478
Input 8 Parameter ... 478
Input 9 Parameter ... 478
Input 10 Parameter ... 478
Input 11 Parameter ... 478
Input 12 Parameter ... 478
Input 13 Parameter ... 478
Input 14 Parameter ... 478
Input 15 Parameter ... 478
Input 16 Parameter ... 478
Input 17 Parameter ... 478
Input 18 Parameter ... 478
Input 19 Parameter ... 478
Input 20 Parameter ... 478
Input a ... 275
Input a Parameter ... 356
Input b ... 275
Input b Parameter ... 356
Input c ... 276
Input c Parameter ... 356
Input d ... 276
Input d Parameter ... 357
Input e ... 276
Input e Parameter ... 357
Input f ... 276
Input f Parameter ... 357
Input g Parameter ... 276, 357
Input h Parameter ... 276, 357
Input I Parameter ... 267
Input Label Parameter ... 311, 315
Input Monitoring (Type 4 (Counter) Logic Modules) ... 345
Input Parameters ... 24
Inputs
  Connect ... 113, 128, 129
  Type 1 (XNC) Interface Modules ... 309
  Type 2 (Wallbus) Interface Modules ... 312, 317
Inputs Base Register Address ... 319
Inputs (Data Format) ... 318

Inputs (PV Index) Parameter....315, 319
Inputs (Refresh Period)....318
Inputs (Refresh Period) Parameter....315
Inputs Register Bitmask....319
Inputs (Scaling Factor)....319
Inputs (Significant change)....319
Inputs (Significant change) Parameter....315
Input Type Parameter....467, 478
Input Value Parameter....311, 315
Inside Temp Parameter....423
Installation Procedure....113
Installing an IQ4....113–116
Install I/O Modules....114, 127–130
Integer Data Types....24
Integral Parameter....384
Integral Time Constant....366
Integral Time (mins) Parameter....384
Interface Modules....309–328
  Common Parameters....328
  Connectives....328
  Types....309
Interfacing....108
Interframe Pause....399, 406
Internal Analogue Sensor Modules....462
Internal Digital Sensor Modules....466
Internal LAN....39
Internal Parameters....24
Internetwork
  Building Across Routers....41, 135
  Building on a Trend Current Loop Network....57
  Building on Ethernet....40
Internetwork Broken....72
Internetwork Changed....72
Internetwork OK....72
Interval Limit Alarm Parameter....349
Interval Parameter....349, 441
Intrinsic Reporting....65
Invert Driver Modules....220
Invert inputs Parameter....354
Invert Output Parameter....270, 353
Invert Parameter....237, 350
I/O Bus
  Connect....127, 128
  Indicator....159
  Power Requirements....116, 129
  Terminate....127, 129
IO Bus Failure Parameter....169
I/O Channels (Assign)....122
I/O Module Address....159
I/O Module Matching....330
I/O Modules....329–332
  Add....121
  Configure....159
  Connectives....332
  Create....329
  Creation in IQSET....329
  Installing....114, 127
  Parameters....330
  Power Requirements....129
    Calculating....129
IP Address Alarm Destination....81
IP Addressing....50
  Automatic....50
  Static....51
IP Addressing Mode....393
IP Addressing Mode Parameter....394
IP Address or Hostname....409
IP Address Parameter....395
IP Alarms....49
IQ4 and IQ2 IC Comms....307
IQ4E....22
IQ4 File....34, 136
IQ4/IO and XCITE I/O Modules (Mixing)....129
IQ4/IO Modules....22, 127, 159
  Installation Procedure....127
  Mount....127
  Power Requirements....127
IQ4 Modules....165
IQ4NC....21
IQ4NC Communication Mode....59
  Ethernet to MS/TP....59, 389
  Ethernet to Trend Internetwork....61, 389
  Ethernet to Trend LAN....60, 389
  Manual....61, 389
  Manual Configuration....390
  Selection....389
  Set up....134
  Trend Internetwork to MS/TP....60, 389
IQ4 Security System....101
IQ4 Variants....21
IQ41x Series Controllers....21
IQ411....21
IQ412....21
IQ422....21
IQ Interfacing....109
IQSET....117
  Diagnostic Plots Feature....431
  Network Connection....54
Item Parameter....205
Item Selection....286, 287
  Wildcards....286

K

Knob Modules....333–336
  Connectives....335
  COV Operation....333
  Knob Adjustment Mode....333
  Parameters....334
  Range....333
  Resetting Operation....333
  Security....333
  Sequencing....334

L

Label Matching
  Controller Selection....284
  Item Selection....286
Label Parameter....165, 182, 201, 203, 328, 480
LAN Broken....71
LAN Changed....71
Language selection Parameter....169
LAN Label Parameter....391, 395, 400
Lan Number Parameter....391, 395, 400
LAN OK....71
LAN Parameter....174, 387
LANs
  Building on a Trend Current Loop Network....56
  Building on MS/TP Trunk....58
Last Action Timestamp Parameter....449
Last Cumulative Count Parameter....346
Last Error Parameter....311
Last failed log in parameter....516
Last Load Attempt Parameter....449
Last Notify Record Parameter....442
Last number of Starts Parameter....199
Last_Restore_Time....404
Last Sent Parameter....292, 295
Last Sequence Step Parameter....480
Last Strategy Error Parameter....449
Last successful log in parameter....516
Last Value Parameter....270
Lead Output Parameter....233
Left Position Parameter....206
Level Parameter....515
Licence Error Parameter....169
Licence Upgrade....575
Linear Sensor Scaling....470
Link/Local Operation....51
Link Modules....121
Local Address Parameter....169, 405
Local Lan Parameter....169
Local Module Parameter....306
Log Full....435
Log Full Parameter....442

Logged in Users 103
Logged On Parameter 516
Logging Sensor Modules 458
Logic Modules 337–358
  Common Parameters 357
  Connectives 358
  Module Disable 337
  Types 337
Loop Modules 359–386
  Alarms 369
  Connectives 386
  Deadband 368
  Integral Only Action 367
  Manual Override 368
  Module Disable 368
  Output 363
  Parameters 383
  Process Variable Fail Alarm 370
  Reschedule Time 385
  Setpoint Deviation Alarm 369
  Setpoint Selection 368
Loop Tuning 371
Loss Time Constant Parameter 423
LOW 77
Low Alarm 456
Low Alarm Level Parameter 461, 464
Low Alarm Parameter 187
Lower Limit Parameter 478
Low Flash Parameter 169
Low Memory Parameter 169
Low Parameter 461, 464

M

MAC address Parameter 387
MAC Address Parameter 176, 400
Maintenance input Parameter 237
Maintenance Interval Alarm 77, 208, 209
Maintenance Parameter 238
Manual Communication Mode 61
Manual Configuration
  IQ4NC Communication Mode 390
  Manual Daylight Saving 489
Manual device instance Parameter 404
Manual Level Parameter 384, 453
Manual MAC address Parameter 400
Manual Network Number Parameter 401
Manual Override
  Loop Modules 368
  Schedule Offset Modules 451
Manual Select Parameter 384, 453
Manufacturer Code 327
Maximum Number of Modules Serviced 29
Max Info Frames Parameter 401
Max Master Parameter 401
Memory Allocated (bytes) Parameter 431
Memory Capacity 23
Memory Usage (%) Parameter 431
Message Format Parameter 174, 175
Message Structure 546
Minimum ON 341
Minimum on-time 341
MINT 77
minutes Parameter 182
Mixing IQ4/IO and XCITE I/O Modules 129
Modbus Communication Mode 399
Mode parameter 260
Mode Parameter 423
Module Disable 197, 209, 226, 228, 241, 280
  Logic Modules 337, 503
  Loop Modules 368
  Sensor Modules 457
Module Number Parameter 182
Module Override 31, 210
  Digital Input Modules 196
  Sensor Modules 458
Module Parameters 23
  Edit 122
Modules 23
  Add 121
  Limits 23
  Link 121
  Servicing 26
Module Status Parameter 191, 238
module type Parameter 183
Module Types 23
Moisture content Parameter 272
month Parameter 183
Mount
  IQ4 113
  IQ4/IO Modules 127
  XCITE I/O Modules 128
MS/TP 69, 573
  Communications 58
  Connect 114
  LAN Connection to Internetwork 59
  Port 39
Multiple Entries in the Sequence Table 479
Multiplexing 30
Multi-sequencing 28

N

NAND Function 340
Network Alarms 71
Network Connection of IQSET 54
Network Modules 389–410
  Types 390
Network Number Parameter 176, 387, 398, 401
Network Parameter 328
Network Planning and Security 100
  BACnet IP Network 100
  I/O Bus 100
  MS/TP 100
  RS232 100
  Trend Current Loop Network 100
  USB 100
  Wallbus 100
Network Time Protocol 488
New Alarm Parameter 179
New Compact Protocols 443
New Text Comms Protocols 444
NKBK 71, 72
NKCH 71, 72
NKDA 74
NKOK 71, 72
Non-Linear Sensor Scaling 473
Non Occupied State Parameter 423
Non-Occupied Value Parameter 453
Non-Trend Device Modules 387–388
  Parameters 387
NOR Function 340
Normal Week 496
Notification Class Object 65, 69, 568
Notification Threshold Parameter 442
NTD Modules. See Non-Trend Device Modules
Number of Lines Parameter 311
Number of Points Parameter 478
Number of Records
  Type 1 (Synchronised) Plot Modules 438
  Type 2 (Triggered) Plot Modules 439
  Type 3 (Periodic) Plot Modules 440
  Type 4 (COV) Plot Modules 441
Number of Records Parameter 442
Number of Stages Parameter 233
Number of Starts 195
Number of Starts Parameter 199, 238
Number of states parameter 507
Number of States parameter 481

O

Object Name Prefixes 401
Object Name Prefixes Parameter 404
Occupation Times 122
Occupied Setpoint Parameter 384
Occupied State Parameter 423
Occupied Value Parameter 453

OFF Delay....341
OFF Level Parameter....225
Offset and Range....214
Offset Parameter....215, 276, 357, 461
O/K....77
Onboard IO Count Parameter....169
ON Delay....341
ON Level Parameter....225
On/Off Pump Control....374
Operation with Proportional Feedback....222
Optimum Off Time Parameter....423
Optimum On Time Parameter....423
Optimum Start Stop Modules....411–428
  Parameters....423
Options Module....429
  Parameters....429
OR Function....340
OSS Cooling Module - Average Medium Temp Parameter....423
OSS Cooling Module - Gain Time Constant Parameter....423
OSS Cooling Module - Medium Temp Parameter....423
OSS Cooling Module - OSS Setpoint Parameter....424
OSS Cooling Module - Start Elevation Parameter....424
OSS Cooling Module - Stop Elevation Parameter....424
OSS Heating Module - Average Medium Temp Parameter....424
OSS Heating Module - Gain Time Constant Parameter....424
OSS Heating Module - Medium Temp Parameter....424
OSS Heating Module - OSS Setpoint Parameter....424
OSS Heating Module - Start Elevation Parameter....424
OSS Heating Module - Stop Elevation Parameter....424
OSS State Parameter....424
OTP State Parameter....425
OUTL....77
Out of Limits....77
Out of Limits Alarm....455
Out of Limits Alarm Parameter....187
Out of Limits Parameter....461, 464
Output 1 Parameter....478
Output 2 Parameter....478
Output 3 Parameter....478
Output 4 Parameter....478
Output 5 Parameter....478
Output 6 Parameter....478
Output 7 Parameter....478
Output 8 Parameter....478
Output 10 Parameter....478
Output 11 Parameter....478
Output 12 Parameter....478
Output 13 Parameter....478
Output 14 Parameter....478
Output 15 Parameter....478
Output 16 Parameter....478
Output 17 Parameter....478
Output 18 Parameter....478
Output 19 Parameter....478
Output 20 Parameter....478
Output J Parameter....350
Output K Parameter....350
Output Label Parameter....311, 316
Output Parameter....179, 187, 384, 453
Output Parameters....24
Outputs
  Connect....113, 128, 129
  Type 1 (XNC) Interface Modules....309
  Type 2 (Wallbus) Interface Modules....313, 317, 326
Outputs Base Register Address....321
Outputs (Data Format)....320, 327
Outputs (Expiry time) Parameter....316
Outputs L Parameter....350
Outputs M Parameter....350
Outputs N Parameter....350
Outputs O Parameter....350
Outputs (PV Index) Parameter....316
Outputs (Refresh Period)....320, 327
Outputs (Register Address)....320
Outputs (Register Bitmask)....320
Outputs Register Type....321
Outputs Scaling Factor....321, 327
Output status Parameter....233
Output Value Parameter....311, 316
Outside Temp Parameter....425
Overflow parameter....261
Override Modules....31
Override Output Parameter....199, 238, 467
Override Parameter....227
Override Status Parameter....200
Override Timer (mins) Parameter....170
Override Value Parameter....238
Override Value/Status Parameter....467
Overview Parameters....165

P

Page Modules....433
  Access....433
  Parameters....433
Page Parameter....165
Parameter Download Exclusions....136
Parameter Types....23
Parent Parameter....201, 206, 433
Parity....399
Password....101
  Recovery....514
Password Parameter....516
Performance Module....431
  Parameters....431
Periodic Plot Modules....436
Period Parameter....219, 227, 438
Permanent vCNC Connections....43
Physical Algorithm....411
PIN....101
PIN Parameter....515
Plan the System....117
Plot Memory....444
Plot Memory Size Parameter....170
Plot Memory Used Parameter....170
Plot Modules....435–448
  Access Plot Data....443
  Alarms....435
  BACnet Alarm State Transitions....435
  BACnet Compliance....435
  Common Parameters....442
  Connectives....447
  Double Precision....436
  Effect of Daylight Saving....445
  Effect of Time Synchronisation....445
  Limits....445
  Plot Memory....444
  Plot Records....446
  Status Parameter....447
  Stop When Full....435
  Threshold Notification....436
  Types....436
Plot Records....446
  Parameters....446
Port Number parameter....518
Position Synchronisation Parameter....223
Power
  Auxiliary Supply....116
  Connect....113, 127, 128
  Failure....437
    Time Module....487
    Type 1 (Synchronised) Plot Modules....437
  I/O Bus....116, 129
  I/O Modules....129
  On Board Inputs and outputs....116
  RS232 Supply....116
  Wallbus....116
Power on Delay (secs) Parameter....238
Power Parameter....346
Power Reset Parameter....170
Power (Type 4 (Counter) Logic Modules)....345
Precision Logs....443
Prefix Object Names....557
Prepare Ethernet Network....131

Primary Address 327
priority Parameter 183
Priority Parameter 509
Process ID Parameter 176
Process Variable Fail 77, 370
Process Variable Fail Alarm 370
Process Variable Parameter 384
Process Variables (PV) 312, 313
Program Module 449
    Parameters 449
Program State Parameter 311, 450
Program Type Parameter 328
Progress Percent Parameter 450
Proportional + Integral + Derivative (PID) Control 367
Proportional and Integral Control 366
Proportional Control 362
Protocol Parameter 292, 295, 302, 305
Pulse Duration Parameter 270, 353
Pulse interval Parameter 485
PV Fail Parameter 385
PVFL 77

R

Raise Lower Driver with Two Digital Outputs 221
Range Parameter 215
Rate Parameter 346
Rate (Type 4 (Counter) Logic Modules) 345
READ 77
Read Alarm 455
Read Alarm Parameter 187
Readback 77
Readback Alarm 207
Readback input Parameter 238
Readback Parameter 238
Read Parameter 461, 464
Real Time Clock 487
Receiving IC comms from Other Controllers 279
Record Count Parameter 442
Record Number Parameter 446
Records Since Last Notification Parameter 442
references Parameter 507
Refresh Rate Parameter 201
Register Addressing Mode 321
Relative Humidity Parameter 272
Remote Controller Address Parameter 292, 302, 305
Remote Device 1 Parameter 395
Remote Device 2 Parameter 395
Remote Device 3 Parameter 395
Remote Device 4 Parameter 395
Remote Device 5 Parameter 395
Remote Device 6 Parameter 395
Remote Device 7 Parameter 395
Remote Device 8 Parameter 395
Remote Device 9 Parameter 395
Remote Device 10 Parameter 395
Remote Device 11 Parameter 395
Remote Device 12 Parameter 395
Remote Device 13 Parameter 395
Remote Device 14 Parameter 395
Remote Device 15 Parameter 395
Remote Device 16 Parameter 395
Remote Device 17 Parameter 395
Remote Device 18 Parameter 395
Remote Device 19 Parameter 395
Remote Device 20 Parameter 395
Remote Device Subnet 1 Parameter 395
Remote Device Subnet 2 Parameter 395
Remote Device Subnet 3 Parameter 395
Remote Device Subnet 4 Parameter 395
Remote Device Subnet 5 Parameter 395
Remote Device Subnet 6 Parameter 395
Remote Device Subnet 7 Parameter 395
Remote Device Subnet 8 Parameter 395
Remote Device Subnet 9 Parameter 395
Remote Device Subnet 10 Parameter 395
Remote Device Subnet 12 Parameter 395
Remote Device Subnet 14 Parameter 395
Remote Device Subnet 15 Parameter 395
Remote Device Subnet 16 Parameter 395
Remote Device Subnet 17 Parameter 395
Remote Device Subnet 18 Parameter 395
Remote Device Subnet 19 Parameter 395
Remote Device Subnet 20 Parameter 395
Remote Lan Parameter 306
Remote Module Parameter 306
Replacing an XCITE I/O Module with an IQ4/IO Module 130
reported state Parameter 183
Required State Parameter 200
Reschedule Time Parameter 347, 385
Reset Parameter 347, 349, 354, 355
Resetting Operation
    Knob Modules 333
Resetting (Type 4 (Counter) Logic Modules) 345
Reset to Default Parameter 334, 485
Resolve Hardware I/O Module Duplicate Addresses 161
    IQ4/IO Modules 161
    XCITE/IO Modules 161
Response to Changing Inputs 217
Restore to Factory Defaults 35
Retransmission Interval 281
Retransmission Interval Parameter 306
Re-tries 399
Retries 282, 406
Retry Interval Parameter 178
Reverse Operation 220, 228
Reverse Operation Parameter 223, 229
Rising Edge Enable Parameter 353
Router 1 Parameter 395
RS232 Addressing 135
RS232 Communications 52
RS232 Port 39, 52
    Connection of Displays/Supervisors/Tools 52

S

Saturated Vapour Pressure Parameter 272
SavedVars 309, 310
Saving Critical Data 34
Scale Factor Parameter 347
Scaling Type Parameter 261
Schedule Modules. **See Schedule Offset Modules**
Schedule Object 569
Schedule Offset Modules 451–454
    Alarm Bit Handling 451
    Common Parameters 453
    Connectives 453
    Manual Override 451
    Service Timing 451
    Types 451
SDEV 77
SDGT 77, 207
Secondary Address 327
seconds Parameter 183
Secure
    Access to the Operating System 101
    IQ4 99
    Wireless Devices 101
Secure Access Port 396
Secure Web Server Port Number Parameter 396
Security 99–104
    BACnet 514
    Checklist 99
    Download Strategy 102
    Features 101
    IC Comms Writes 280
    Knob Modules 333
    Physical and Environmental Consideration 99
    Set up 123
    Switch Modules 484
    System 101
    Time Schedule Modules 503
    Updates and Service Packs 99
    Upload Strategy 102
    Web Only Access 104
    Web Server 97, 102

XML Web Services ... 104
XML Webs Services ... 105
Security Mode parameter ... 518
Selector Parameter ... 276, 357
Select Universal Input Channel Type ... 159
Send I-Am at Startup Parameter ... 404
Send Service PIN Parameter ... 170
Sensor Fail Parameter ... 385
Sensor Modules ... 455–468
Alarm Hysteresis ... 456
Alarms ... 455
BACnet Alarm State Transitions ... 457
Common Parameters ... 466
Connectives ... 468
COV Operation ... 457
High Alarm ... 456
Logging ... 458
Low Alarm ... 456
Module Disable ... 457
Module Override ... 458
Out of Limits Alarm ... 455
Read Alarm ... 455
Sensor Offset ... 458
Types ... 458
Sensor Scaling ... 470
Sensor Type Library ... 469
Sensor Type Modules ... 121, 189–192, 469–478
Parameters ... 191, 478
Scaling ... 469
Sensor Type Library ... 469
Voltage Input Scaling ... 474
Sequenced Servicing ... 27
Sequence Step 1 Parameter ... 480
Sequence Step Parameter ... 165
Sequence Table ... 122
Multiple Entries ... 479
Order ... 27
Overrun ... 76
Set up ... 122
Sequence Table Execution (%) Parameter ... 431
Sequence Table Module ... 479–480
Parameters ... 480
Sequence Table Overrun Parameter ... 170
Sequence Table Parameter ... 165
Sequencing (Knob Modules) ... 334
Serial Comms Factor Parameter ... 170
Serial Number Parameter ... 170, 331
Service Class Parameter ... 401
Service Order ... 29
Service Timing
Schedule Offset Modules ... 451
Service Type Parameter ... 177
Servicing Modules ... 26
Event Driven Modules ... 28
Non-Sequence Table Modules ... 480
Sequenced Modules ... 27
Set Hardware I/O Module Addresses ... 159, 160
IQ4/IO Modules ... 160
XCITE/IO Modules ... 161
Set Parameter ... 354
Setpoint Deviation Alarm ... 77, 369
Setpoint Deviation Parameter ... 385
Setpoint Select Parameter ... 385
Set up
Address Module ... 118
Alarm Handling Strategy ... 121
Current Loop Baud Rate ... 135
Daily Profile ... 122, 504
Display and Directory Modules ... 124
Ethernet Access Across Routers ... 135
Event Driven Strategy ... 122
IC Comms ... 123
IP Addressing Details ... 132
IQ4NC Communication Mode ... 134
RS232 Addressing ... 135
Security ... 123
Sensor Type Modules ... 121
Sequence Table ... 122
Strategy for Use with Wallbus Devices ... 122
Trend Addressing ... 135
Type 1 (IQ LAN) Network Module ... 119
Type 1 (XNC) Interface Module ... 121
Type 4 (Ethernet IP) Network Module ... 119
Type 5 (BACnet IP) Network Module ... 120
Type 12 (BACnet MSTP) Network Module ... 120
Type 13 (BACnet Application) Network Module ... 120
Type 17 (Trend Serial) Network Module ... 121
USB Port Addressing ... 135
vCNCs ... 133
Significant Change ... 281
Significant Change Parameter ... 270, 292, 295
Single Precision Logs ... 443
Site GUID Parameter ... 170
Site Name Parameter ... 170
Size of Modules Parameter ... 328
Socket Timeout parameter ... 518
Solicited COV reception ... 282
Solutions ... 117
Source B Parameter ... 255
Source E Parameter ... 244, 246, 247, 249, 250, 251, 252, 253, 254, 255, 256, 257, 262, 263, 264, 267, 340, 344, 347
Source F Parameter ... 246, 247, 249, 251, 252, 253, 254, 255, 256, 257, 262, 263, 264, 267, 269, 340, 344
Source G Parameter ... 244, 246, 247, 249, 250, 251, 252, 253, 254, 257, 261, 262, 263, 264, 265, 266, 267, 269, 340, 344, 347
Source H Parameter ... 251, 253, 254, 262, 267, 269, 340, 344
Source (mV)/Source Parameter ... 467
Source Parameter ... 178, 200, 238, 270, 347, 349, 350, 353, 355, 442
SP Deviation Parameter ... 385
SSL Certificates
Configure ... 136
Create ... 137
Upload to the IQ4 ... 155
Stability Limit Method ... 371
Stand Alone IQ4 ... 42
Standalone Mode Parameter ... 396
Start Day Parameter ... 507, 509
Start Deviation Parameter ... 453
Start Month Parameter ... 507, 509
Start Time Error Parameter ... 425
Start Year Parameter ... 507, 509
State ... 195
State Parameter ... 200, 223, 229, 233, 332, 425, 453
States Category module parameter ... 507
States Category Module Parameters ... 481
Static IP Addressing ... 51
Static Text Display Modules ... 203
Hyperlinks ... 203
Parameters ... 203
Statistical Algorithm ... 413
Status Parameter ... 316, 446, 447, 485, 518
Step Method ... 372
Stop Deviation Parameter ... 453
Stop Time Error Parameter ... 425
Stop When Full ... 435
Stop When Full Parameter ... 443
STOR ... 76, 170
Store-Name Parameter ... 312
Stores ... 309, 310
Store-String Value Parameter ... 312
Strategy ... 25–36
Backup ... 157
Configure ... 118
Creating ... 117
Download ... 136
Set up for Use with Wallbus Devices ... 122
Test ... 124
Strategy Blocks ... 117, 122
Strategy File ... 34
Create ... 125
Strategy I/O Modules. See I/O Modules
Strategy Modules ... 23, 165

Strategy Revision Parameter ... 170
Strategy Status ... 168
Strings ... 24
Subnet Mask Parameter ... 396
Sub Type Parameter ... 266
Supervisor Mode ... 518
Supervisor Port Parameter ... 171
Supply Voltage ... 159
Supported BACnet Properties ... 69, 557–574
Switching Type Parameter ... 233
Switch Modules ... 483–486
  Connectives ... 485
  COV Operation ... 484
  Security ... 484
System Guest User ... 103
System Monitoring ... 101

T

TCL ... 108, 309
  Compatibility with IQ3 TCL ... 108
TCL Application ... 121, 309
  Configuration ... 108
t comms ... 552
TCP Port ... 409
Temperature Parameter ... 272
Temporary vCNC Connections ... 43
Terminate I/O Bus ... 127, 129
Test the Strategy ... 124
Text Communications ... 543–556
  Identifier ... 23
  Security ... 102
Threshold Notification ... 436
Time Data Types ... 24
Time Errors (OSS) ... 420
Time in Alarm (secs) (High) Parameter ... 461, 464
Time in Alarm (secs) (Low) Parameter ... 461, 465
Time in Alarm (secs) (maintenance) Parameter ... 239
Time in Alarm (secs) (Out of Limits) Parameter ... 461, 465
Time in Alarm (secs) Parameter ... 200
Time in Alarm (secs) (PV Fail) Parameter ... 385
Time in Alarm (secs) (readback) Parameter ... 239
Time in Alarm (secs) (Read) Parameter ... 461, 465
Time in Alarm (secs) (SP Deviation) Parameter ... 385
Time into Interval Parameter ... 306
Timekeeper ... 489
Timemaster ... 487
Time Module ... 487–494
  Connectives ... 493
  Parameters ... 490
  Power Failure ... 487
  Time Synchronisation ... 487
  Timing Pulses ... 487
  UTC Time ... 490
Time Offset Parameter ... 446
Time of Hours Run Reset Parameter ... 200, 213
Time of Last Notification Parameter ... 443
Time of Last Record Parameter ... 443
Time of Last Value Change Parameter ... 213
Time of Number of Starts Reset Parameter ... 213
Timeout ... 399, 406
Timeout (Minutes) Parameter ... 516
Time Parameter ... 508, 510
Time Schedule Modules ... 495–510
  Connectives ... 510
  Day of Week Sub-module Parameters ... 508
  Day of Week TimeValue Sub-module Parameters ... 508
  Exception Sub-module Parameters ... 508
  Exception TimeValue Sub-module Parameters ... 510
  Module Disable ... 503
  Parameters ... 506
  Security ... 503
  Setting up the Daily Profile ... 504
  Types ... 505
  Use with OSS and Schedule Offset Modules ... 503
Time Schedule Parameter ... 425
Time Server Address Parameter ... 491
Time Source Parameter ... 492
Timestamp Parameter ... 446
Time Synchronisation ... 445, 487
  Network Time Protocol Operation ... 488
  Timekeeper Operation ... 489
  Timemaster Operation ... 488
Timezone Connection Parameter ... 453
Timezone Modules. See Time Schedule Modules
Time zone Parameter ... 492
Timing Parameter ... 480
Timing Pulses ... 487
Timing Type Parameter ... 443
Title Parameter ... 429
To Email Address Alarm Destination Modules ... 175
Top Position Parameter ... 206
Top Range Parameter ... 334
Total Record Count Parameter ... 443
Transitions (BACnet Alarm) ... 196, 435, 457
Transitions Parameter ... 177
Trend Addressing (Set up) ... 135
Trend Communications ... 39–62
Trend Current Loop
  Communications ... 55
  Connect ... 114
  Port ... 39
Trend Custom Language. See TCL
Trend Internetwork to MS/TP ... 389
Trend Internetwork to MS/TP Mode ... 60
Trend Log Object ... 66, 571
Trend Mode ... 390, 392, 400
Trend Mode Parameter ... 392, 396, 401
Trigger Input Parameter ... 440
Trigger Output Parameter ... 440
Trigger Parameter ... 355
Tuning an integral only Loop ... 372
Type 0 (Analogue) Sensor Modules ... 458
  Parameters ... 459
  Sensor Offset ... 458
Type 0 User Modules ... 511
Type 1 CNC User Module Parameters ... 515
Type 1 CNC User Modules ... 515
Type 1 (Combination) Logic Modules ... 338
  Parameters ... 340
Type 1 (Digital) Driver Modules ... 213
  Parameters ... 213
Type 1 (Filter) Function Modules ... 243
  Parameters ... 244
Type 1 (Internal Analog) Sensor Modules ... 462
  Parameters ... 462
Type 1 (Internal) Digital Input Modules ... 197
  Parameters ... 197
Type 1 (IQ Lan) Alarm Destination Modules ... 174
  Parameters ... 174
Type 1 (IQ Lan) Network Modules ... 390
  Parameters ... 391
  Set up ... 119
Type 1 (OSS) Schedule Offset Modules ... 451
  Parameters ... 452
Type 1 (Synchronised) Plot Modules ... 436
  Number of Records ... 438
  Parameters ... 438
  Power Failure Operation ... 437
Type 1 (To) IC Comms Modules ... 288
  Parameters ... 292
Type 1 (XNC) Interface Module ... 107, 309
  Inputs ... 107, 309
  Outputs ... 107, 309
  Parameters ... 311
  SavedVars ... 107, 310
  Set up ... 121
  Stores ... 107, 310
Type 2 (Analogue) Driver Modules ... 214
  Parameters ... 215
Type 2 (Digital) Sensor Modules ... 465
  Parameters ... 465
Type 2 (External) Digital Input Modules ... 198
  Parameters ... 198

Type 2 (Global To) IC Comms Modules 292
Parameters 295
Type 2 (IP Address) Alarm Destination Modules 174
Parameters 174
Type 2 (OTP) Schedule Offset Modules 452
Parameters 453
Type 2 (Rescale From) Function Modules 244
Parameters 245
Type 2 (Timer) Logic Modules 341
Parameters 342
Type 2 (Triggered) Plot Modules 436, 438
Event Driven Strategy 439
Number of Records 439
Type 2 (Wallbus) Interface Module 122, 312
Inputs 312, 317
Outputs 313, 317, 326
Parameters 315, 317, 326, 328
Type 3 (Email) Alarm Destination Modules 175
Parameters 175
Type 3 (Internal Digital) Sensor Modules 466
Parameters 466
Type 3 (Min) IC Comms Modules 296
Parameters 296
Type 3 (Periodic) Plot Modules 440
Number of Records 440
Parameters 441
Type 3 (Readback) Logic Modules 342
Parameters 344
Type 3 (Rescale To) Function Modules 245
Parameters 246
Type 3 (Time Proportional) Driver Modules 216
Parameters 219
Type 4 (BACnet) Alarm Destination Modules 176
Parameters 176
Type 4 (Counter) Logic Modules 345
Parameters 346
Type 4 (COV) Plot Modules 441
Number of Records 441
Type 4 (Ethernet IP) Network Module 392
Parameters 393
Set up 119
Type 4 (Limit At) Function Modules 247
Parameters 247
Type 4 (Max) IC Comms Modules 296
Parameters 298
Type 4 (Raise Lower with End Stop) Driver Modules 220
Parameters 223
Type 5 (BACnet IP) Network Module 397
Parameters 398, 399, 409
Set up 120
Type 5 (Binary Hysteresis) Driver Modules 224
Parameters 225
Type 5 (Delay) Logic Modules 347
Parameters 347
Type 5 (Limit To) Function Modules 248
Parameters 249
Type 5 (Sum) IC Comms Modules 298
Parameters 299
Type 6 (Average) IC Comms Modules 299
Parameters 300
Type 6 (Hours Run) Logic Modules 348
Parameters 349
Type 6 (Log) Function Modules 249
Parameters 249
Type 6 (Time Proportional + OR) Driver Modules 226
Anti-phase Output 227
Parameters 227
Type 7 (Fan Out) Logic Modules 350
Parameters 350
Type 7 (From Analogue) IC Comms Modules 301
Parameters 302
Type 7 (Raise Lower Continuous) Driver Modules 228
Boundless Operation 228
Invert 228
Module Disable 228
Parameters 229
Reverse Operation 228
Type 7 (Square Root) Function Modules 249
Parameters 250
Type 8 (Adder/Scaler) Function Modules 250
Parameters 251
Type 8 (D to A) Logic Modules 350
Parameters 352
Type 8 (From Byte) IC Comms Modules 302
Parameters 304
Type 8 (Multistage Digital) Driver Modules 230
Parameters 233
Type 9 (Change of State) Logic Modules 352
Parameters 353
Type 9 (From Bit) IC Comms Modules 304
Parameters 305
Type 9 (Multiplier) Function Modules 251
Parameters 251
Type 10 (Divider) Function Modules 252
Parameters 252
Type 10 (Flip Flop) Logic Modules 354
Parameters 354
Type 11 (Latch) Logic Modules 355
Parameters 355
Type 11 (Minimum) Function Modules 252
Parameters 253
Type 12 (BACnet MSTP) Network Module 400
Parameters 400
Set up 120
Type 12 (Digital Selector) Logic Modules 356
Cascade Operation 356
Parameters 356
Type 12 (Maximum) Function Modules 253
Parameters 253
Type 13 (Average) Function Modules 254
Parameters 254
Type 13 (BACnet Application) Network Module 401
BTL Compliance 401
Object Name Prefixes 401
Parameters 402
Set up 120
Type 14 (Analogue Gate) Function Modules 254
Parameters 255
Type 14 (Wallbus) Network Module 122, 405
Parameters 405, 406
Type 15 (Comparator) Function Modules 255
Parameters 255
Type 16 (Enthalpy) Function Modules 255
Parameters 256
Type 16 (Trend Client) Network Module 407
Parameter 407
Type 17 (Hysteresis Band) Function Modules 256
Parameters 257
Type 17 (XNC Serial) Network Module 408
Parameters 408
Set up 121
Type 18 (Analogue to Digital Converter) Function Modules 257
Parameters 260
Type 19 (Lookup) Function Modules 261
Parameters 261
Type 20 (Adder) Function Modules 261
Parameters 262
Type 21 (Hysteresis Limits) Function Modules 262
Parameters 263
Type 22 (Proximity) Function Modules 263
Parameters 263
Type 23 (Power) Function Modules 263
Parameters 264
Type 24 (Absolute) Function Modules 264
Parameters 264
Type 25 (Truncate) Function Modules 265
Parameters 265
Type 26 (Trigonometric) Function Modules 265
Parameters 265
Type 27 (Heat Meter) Function Modules 266
Parameters 267
Type 28 (Integrator) Function Modules 267

Parameters ... 269
Type 29 (Change of Value) Function Modules ... 270
Parameters ... 270
Type 31 (Psychrometrics) Function Modules ... 270
Parameters ... 272
Type Name Parameter ... 171
Type Parameter 178, 200, 239, 277, 337, 357, 387, 409, 429, 446, 453, 467, 508, 510, 516
U
UDP ... 51
UDP Port Parameter ... 396, 398
Units Parameter ... 272, 335, 467
UNKN ... 74
Unoccupied Setpoint Parameter ... 385
Unsolicited COV reception ... 282
Upgrade
Firmware ... 24
Licence ... 575
Upload ... 102
Configuration Files ... 49, 57
Security ... 102
SSL Certificate ... 155
Strategy Files ... 53, 54
Upper Limit Parameter ... 478
URL Parameter ... 203
USB Communications ... 53
USB Local Engineering Port ... 39, 53
Addressing (Set up) ... 135
USB Supervisor Port Parameter ... 171
Use Parameter ... 510
User Alarm Acknowledgments ... 67
User Dependent Displays ... 33
User Level ... 101
User Level Parameter ... 201, 206, 335, 485, 507
User Levels ... 511
Configurable User Levels ... 512
Fixed User Levels ... 511
User Merge Option parameter ... 450
User Module Common Parameters ... 516
User Modules ... 101, 511, 511–516
Configurable User Levels ... 512
Fixed User Levels ... 511
Home Page ... 513
Parameters ... 514
Password Recovery ... 514
View Defaults ... 512
User Module Types ... 511
User name ... 101
User Name Parameter ... 516
Using the HTTPS Web Server ... 124
Using the HTTP Web Server ... 124
UTC Offset (Mins) Parameter ... 492
UTC Time ... 490
V
Value Parameter ... 183, 188, 193, 239, 335, 446, 467, 507, 508, 510
Vapour Pressure Parameter ... 272
Variable Type Parameter ... 307
v comms ... 546
Version parameter ... 327
Version Parameter ... 171, 429
View Alarms Parameter ... 515
View GraphIQs Parameter ... 515
View Modules Parameter ... 515
View Navigator Parameter ... 515
View Time Schedules Parameter ... 515
Virtual CNC Modules ... 517
Alarm Mode ... 518
Alarms ... 518
Parameters ... 518
Supervisor Mode ... 518
Virtual CNC (vCNC) ... 39
Connect ... 43
Set up ... 133
Virtual Environments ... 100
Virtual INC (vINC) ... 39
Virus Protection ... 99
Visitor Messages ... 544
Voltage Input Scaling ... 474
W
Wallbus
Communications ... 54
Connect ... 55, 114
Port ... 39, 54
Power Requirements ... 114, 116
Warmup Limit Parameter ... 425
Warmup State Parameter ... 425
Weather Compensated Control ... 375
Web Client ... 96
Connection ... 48
Web Only Access ... 104
Web Server ... 95–98
Configure ... 124
Disable ... 124
Security ... 97, 100, 102
Using HTTP ... 124
Using HTTPS ... 124
Web Server Port Number Parameter ... 397
Weekday Parameter ... 492
Week Day Profiles Parameter ... 507
Weekly Pulse Parameter ... 492
Week of Month Parameter ... 510
What is IQ4? ... 21–22
Width Parameter ... 201, 206
Wildcards ... 285, 286, 553
Windows Domains ... 101
Wink Parameter ... 332
WINS Server 1 Parameter ... 397
WINS Server 2 Parameter ... 397
WINS Server 4 Parameter ... 397
Write Changes to Flash ... 34
Writes over BACnet ... 402
Write TCL Application ... 121
X
XCITE I/O Modules ... 127, 128, 160
Installation Procedure ... 128
Mount ... 128
Power Requirements ... 128
Replacing with an IQ4/IO Module ... 130
Xco-ordinate Parameter ... 165
XML Syntax ... 527–542
XML Web Services ... 105
Security ... 104, 105
XML Web Services Parameter ... 397
XNC
Alarm Generation ... 310
Facility ... 107–108
File ... 34
Program File ... 108
Variants ... 22
XNC Enabled Parameter ... 171
XNC Interface Module. See Type 1 (XNC) Interface Module
XNC Variants ... 309
Y
Yco-ordinate Parameter ... 165
Yearly Pulse Parameter ... 492
year Parameter ... 183
Year Parameter ... 492
Z
Zone Modules. See Time Schedule Modules