Powered by Possibility

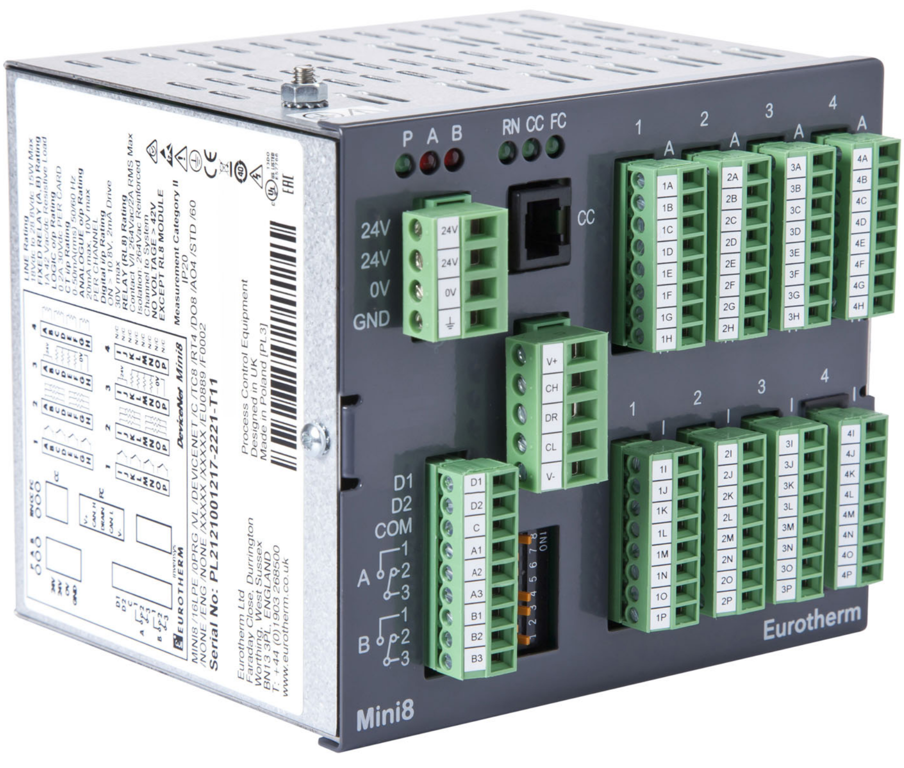
Mini8® Loop Controller

(Firmware version V5+)

User Guide

HA033635 Issue 2

July 2023



Eurotherm

Table of Contents

Safety Information

Important Information

Read these instructions carefully and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this manual or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of either symbol to a “Danger” or “Warning” safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a hazardous situation which, if not avoided, **could result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **could result in** minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

Notes:

1. Electrical equipment must be installed, operated, serviced and maintained only by qualified personnel. No responsibility is assumed by Eurotherm Limited for any consequences arising out of the use of this material.
2. A qualified person is one who has skills and knowledge related to the construction, installation, and operation of electrical equipment, and has received safety training to recognize and avoid the hazards involved.

Before you Start

Important Information

Reasonable Use and Responsibility

The safety of any system incorporating this product is the responsibility of the assembler/installer of the system.

The information contained in this manual is subject to change without notice. While every effort has been made to maintain the accuracy of the information, your supplier shall not be held liable for inaccuracies contained herein.

This programmable controller is intended for industrial temperature and process control applications, which meet the requirements of the European Directives on Safety and EMC.

Use in other applications, or failure to observe the installation instructions of this manual may compromise safety or EMC. The installer must ensure the safety and EMC of any particular installation.

To comply with the European EMC directive certain installation precautions are necessary:

- General guidance. Refer to EMC Installation Guide, Part no. HA025464.
- Relay outputs. It may be necessary to fit a suitable filter to suppress conducted emissions.
- Table top installation. If using a standard power socket, compliance with commercial and light industrial emissions standard is required. To comply with conducted emissions standard, a suitable mains filter must be installed.

Failure to use approved software/hardware with our hardware products may result in injury, harm, or improper operating results.

Please Note

Electrical equipment must be installed, operated, serviced, and maintained only by qualified personnel.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

No responsibility is assumed by Eurotherm Limited for any consequences arising out of the use of this material.

Qualification of Personnel

Only appropriately trained persons who are familiar with and understand the contents of this manual and all other pertinent product documentation are authorized to work on and with this product.

The qualified person must be able to detect possible hazards that may arise from parameterization, modifying parameter values and generally from mechanical, electrical, or electronic equipment.

The qualified person must be familiar with the standards, provisions, and regulations for the prevention of industrial accidents, which they must observe when designing and implementing the system.

Intended Use

The product described or affected by this document, together with software and options, is the Mini8 Controller - Firmware V5.0+ (referred to herein as “programmable controller”, or "controller" or "Mini8"), intended for industrial use according to the instructions, directions, examples, and safety information contained in the present document and other supporting documentation.

The product may only be used in compliance with all applicable safety regulations and directives, the specified requirements, and the technical data.

Prior to using the product, a risk assessment must be performed in respect of the planned application. Based on the results, the appropriate safety-related measures must be implemented.

Since the product is used as a component within a machine or process, you must ensure the safety of this overall system.

Operate the product only with the specified cables and accessories. Use only genuine accessories and spare parts.

Any use other than the use explicitly permitted is prohibited and can result in unanticipated hazards.

Dangers and Warnings

⚠ ⚠ DANGER

HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH

Turn off all power to product and all I/O circuitry (alarms, control I/O etc.) before starting the installation, removal, wiring, maintenance or inspection of the product.

Power line and output circuits must be wired and fused in compliance with local and national regulatory requirements for the rated current and voltage of the particular equipment, i.e. UK, the latest IEE wiring regulations, (BS7671), and USA, NEC class 1 wiring methods.

The unit must be installed in an enclosure or cabinet.

Do not exceed the device's ratings.

This product must be installed, connected and used in compliance with prevailing standards and/or installation regulations. If this product is used in a manner not specified by the manufacturer, the protection provided by the product WILL be impaired.

Do not insert anything through the case apertures.

All connections must be tightened in conformance with the specified torque specifications.

Apply appropriate personal protective equipment (PPE) and follow safe electrical work practices. See NFPA 70E, CSA Z462 BS 7671, NFC 18-510.

Ensure the mandatory protective ground connection is connected during installation. Connection of this protective ground connection must be made before turning on any power supplying this product.

Failure to follow these instructions will result in death or serious injury.

⚠ DANGER

FIRE HAZARD

Do not install if the unit or any part of the unit is damaged. Contact your supplier.

Do not allow anything to fall through the case apertures and ingress the controller.

Ensure the correct wire gauge size is used per circuit and it is rated for the current capacity of the circuit.

Ensure when using ferrules (cable ends) that the correct size is selected and each is securely fixed to the wire using a crimping tool.

Correct rated power supply unit or supply voltage must be used for the product.

Failure to follow these instructions will result in death or serious injury.

Symbols

Various symbols may be used on the controller. They have the following meaning:

Risk of electric shock.

Take precautions against static.

The RCM is a trademark owned by Australian and New Zealand Regulators with RCM mark.

Complies with the 40 year Environment Friendly Usage Period.

Hazardous Substances

This product conforms to European **R**estriction **o**f **H**azardous **S**ubstances (RoHS) (using exemptions) and **R**egistration, **E**valuation, **A**uthorisation and Restriction of **Ch**emicals (REACH) Legislation.

RoHS Exemptions used in this product involve the use of lead. China RoHS legislation does not include exemptions and so lead is declared as present in the China RoHS Declaration.

Californian law requires the following notice:

WARNING: This product can expose you to chemicals including lead and lead compounds which are known to the State of California to cause cancer and birth defects or other reproductive harm. For more information go to: http://www.P65Warnings.ca.gov

Cybersecurity

What is in this Chapter?

This chapter outlines some good practice approaches to cybersecurity as they relate to use of the Mini8 loop controller, and draws attention to several features that could assist in implementing robust cybersecurity.

⚠ CAUTION

EQUIPMENT OPERATION HAZARD

To minimize any potential loss of control or controller status when communicating across a network or being controlled via a third party master (i.e. another controller, PLC or HMI) ensure all system hardware, software, network design, configuration and cybersecurity robustness have been correctly configured, commissioned and approved for operation.

Failure to follow these instructions can result in injury or equipment damage.

Introduction

When utilizing a Eurotherm Mini8 loop controller in an industrial environment, it is important to take 'cybersecurity' into consideration: in other words, the installation's design should aim to prevent unauthorized and malicious access. This includes electronic access (via network connections and digital communications).

Cybersecurity Good Practices

Overall design of a site network is outside the scope of this manual. The Cybersecurity Good Practices Guide, Part Number HA032968 provides an overview of principles to consider. This is available from www.eurotherm.com.

Typically, an industrial controller such as the Mini8 loop controller together with any controlled devices should *not* be placed on a network with direct access to the public Internet. Rather, good practise involves locating these devices on a firewalled network segment, separated from the public Internet by a so-called 'demilitarized zone' (DMZ).

Security Features

The sections below draw attention to some of the cybersecurity features of the Mini8 loop controller.

Principle of Secure by Default

Some of the digital communication features on the Mini8 loop controller can provide greater convenience and ease-of-use (particularly in regards to initial configuration), but also can potentially make the controller more vulnerable. For this reason, the following feature is turned off by default:

Bonjour Auto-discovery Disabled by Default

When an Ethernet comms module is installed into the Mini8 loop controller, the Bonjour auto-discovery feature becomes available for use. Bonjour enables the controller to be automatically discovered by other devices on the network without the need for manual intervention. However, for cybersecurity reasons, it is disabled by default when using a fixed IP address as it could be exploited by a malicious user to gain information about the controller. This feature is enabled automatically when DHCP is used, as it is the only method of discovering the device when the IP address is unknown.

Port Use

The following ports are being used:

Port	Protocol
502 TCP	Modbus (Master (Client) and Slave (Server))
5353 UDP	Bonjour/auto-discovery/zeroconf

The following should be noted about the Ethernet ports:

- Modbus TCP port is always enabled as the primary method of communicating with the device.
- UDP Port 5353 (Auto-discovery/ZeroConf/Bonjour, open only when Comms.FC.Network.AutoDiscovery parameter is ON.

Access Control

The Mini8 loop controller has two levels of access - Operator mode and Configuration mode. Operator mode provides basic functionality required on a day-to-day basis, whereas Configuration mode provides full functionality for Initial set-up and process configuration. Passwords are supported by default to control access to configuration mode. Strong passwords should be used (see below). After five unsuccessful login attempts, password entry is blocked for 30 minutes (including over a power interruption). This helps protect against ‘brute force’ attempts to guess a password.

Strong Passwords

It is recommended that a strong password is used for the Configuration password and Config Lock password. By ‘strong’, we recommend a password that is:

- At least eight characters in length.
- Has a mixture of both upper and lower case characters.
- Has a minimum of one special punctuation character (#, %, or @ for example).
- Has at least one numeric digit.

NOTICE

POTENTIAL LOSS OF INTELLECTUAL PROPERTY OR CONFIGURATION

Ensure all passwords configured in the programmable controller are ‘strong’ to help prevent the loss of intellectual property or unauthorized configuration changes.

Failure to follow these instructions can result in equipment damage.

Config Lock Password

An optional Config Lock feature is provided to give Original Equipment Manufacturers (OEMs) a layer of protection against theft of their intellectual property, and is designed to help prevent unauthorized cloning of controller configurations. This protection includes application-specific internal (soft) wiring and limited access to certain parameters via comms (by iTools or a third party comms package).

Configuration Password

The password for Configuration Level access via iTools has the following features to help protect against unauthorized access:

- There is no default password for comms configuration level.
- User needs to set the comms configuration password on first connect from iTools.
- If password is not set, FC comms will be in Comms Lockdown mode (see below).
- Configuration password is encrypted before sending via comms.
- Passwords are salted and hashed before being stored.
- Number of password attempts is five. If more than five unsuccessful attempts are made, the Password Lock function is triggered.
- iTools will enforce a minimum password length of eight characters.

Comms Lockdown mode

In Comms Lockdown mode, FC comms will only have read/write access to a limited set of parameters that allow for iTools to connect and set a password. CPI and CC comms connections will not be affected.

Ethernet security features

Ethernet connectivity is available as an option in the Mini8 loop controller. The following security features are specific to Ethernet:

Ethernet rate protection

One form of cyberattack is to try to make a controller process so much Ethernet traffic that it drains systems resources and useful control is compromised. For this reason, the Mini8 loop controller includes an Ethernet rate protection algorithm, which will detect excessive network activity and help to ensure the controller's resources are prioritized on the control strategy rather than servicing the Ethernet traffic. If this algorithm is running, the RateProtection diagnostic parameter will be set to ON.

Broadcast Storm protection

A 'broadcast storm' is a condition which may be created by cyberattack whereby spurious network messages are sent to devices which cause them to respond with further network messages, creating a chain reaction that escalates until the network is unable to transport normal traffic. The Mini8 loop controller includes a broadcast storm protection algorithm, which will automatically detect this condition, stopping the controller from responding to the spurious traffic. If this algorithm is active, the BroadcastStorm diagnostic parameter will be set to ON.

Communications watchdog

The Mini8 loop controller includes a 'comms watchdog' feature. This can be configured to raise an alert if any of the supported digital communications are not received for a specified period of time. These provide a way to configure appropriate action if malicious action interrupts the controller's digital communications.

Note: This watchdog may not function as expected for multiple Ethernet connections, due to the shared timer and flag for this interface. If the device is configured to receive a setpoint from a remote master via Ethernet connection, it should be routed through the 'Remote Input' block. This block has an independent timeout (default to 1s), allowing the loss of comms to this parameter to be flagged independently of any other Ethernet connections.

Configuration backup and recovery

Using Eurotherm's iTools software, you can 'clone' a Mini8 loop controller, saving all its configuration and parameter settings to a file. This can then be copied onto another controller, or used to restore the original controller's settings—see "Cloning" on page 61.

For cybersecurity reasons, password-restricted parameters are not saved in the clone file.

Clone files include a cryptographic integrity hash, meaning that if the file contents is tampered with, it will not load back into a controller.

A clone file cannot be generated or loaded if the Config Lock feature option is configured and active.

User Sessions

Communication connections only have two permission levels - an 'Operator mode' and a 'Configuration mode'. Any connection via comms (Ethernet or serial) is separated into its own unique session. A user logged in via the TCP socket will not share permissions with a different user logged in, for example, via the serial port and vice versa.

In addition, only a single user can be logged into a Mini8 loop controller in Configuration mode at any one time. If another user attempts to connect and select Configuration mode, the request will be denied until the other user exits the Configuration mode.

User sessions are not persistent across power cycles.

Data Integrity

FLASH Integrity

When a Mini8 loop controller powers up, it automatically performs an integrity check on the entire contents of its internal flash memory. If the application is detected as being corrupted, the Mini8 loop controller will fail to boot, indicated with RUN LED off, advice must be sought from the manufacturer.

Non-volatile Data Integrity

When a Mini8 loop controller powers up, it automatically performs an integrity check on the contents of its internal non-volatile memory devices. Additional periodic integrity checks are performed during normal runtime and when non-volatile data is being written. If any integrity check detects a difference from what is expected, the controller enters Standby mode and sets bits 1 or 2 in the Instrument.Diagnostics function block, StandbyCondStatus (Standby Condition Status Word) (refer to "Instrument / Diagnostics" on page 87).

Cryptography Usage

Cryptography usage is employed in the following areas:

- ROM startup integrity checking.
- Clone files.
- Custom linearization tables.
- Config Lock Password.
- Configuration Password.

Achilles® Communications Certification

The Mini8 loop controller has been certified to Level 1 under the Achilles® Communications Robustness Test Certification scheme. This is an established industry benchmark for the deployment of robust industrial devices recognized by the major automation vendors and operators.

Decommissioning

When a Mini8 loop controller is at the end of its life and being decommissioned, Eurotherm advises reverting all parameters to their default settings. This can help to protect against subsequent data and intellectual property theft if the controller is then acquired by another party.

Legal Information

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Eurotherm Limited software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

Installation

DANGER

HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH

Only qualified personnel must install, operate and maintain this equipment.

Turn off all power to product and all I/O circuitry (alarms, control I/O etc.) before starting the installation, removal, wiring, maintenance or inspection of the product.

Power line and output circuits must be wired and fused in compliance with local and national regulatory requirements for the rated current and voltage of the particular equipment, i.e. UK, the latest IEE wiring regulations, (BS7671), and USA, NEC class 1 wiring methods.

Failure to follow these instructions will result in death or serious injury.

WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure all electrostatic discharge precautions are taken before handling the unit.

Ensure electrically conductive pollution is excluded from the cabinet in which the controller is mounted.

Avoid ingress of conductive materials during installation.

The unit must be installed in an enclosure or cabinet.

Ensure wires are routed to minimize the pick-up of EMI (Electromagnetic interference) keeping cable lengths to a minimum.

Ensure all cables and wiring harness are secured using a relevant strain relief mechanism.

Wiring, it is important to connect the product in accordance with the data in this User Guide and use copper cables (except the thermocouple wiring).

Only connect wires to identified terminals shown on the product warning label, the wiring section of the product User Guide or Installation sheet.

Safety and EMC protection can be seriously impaired if the unit is not used in the manner specified. The installer must ensure the safety and EMC of the installation.

Ensure only persons with expertise in the design and programming of control systems are allowed to program, install, alter and commission this product.

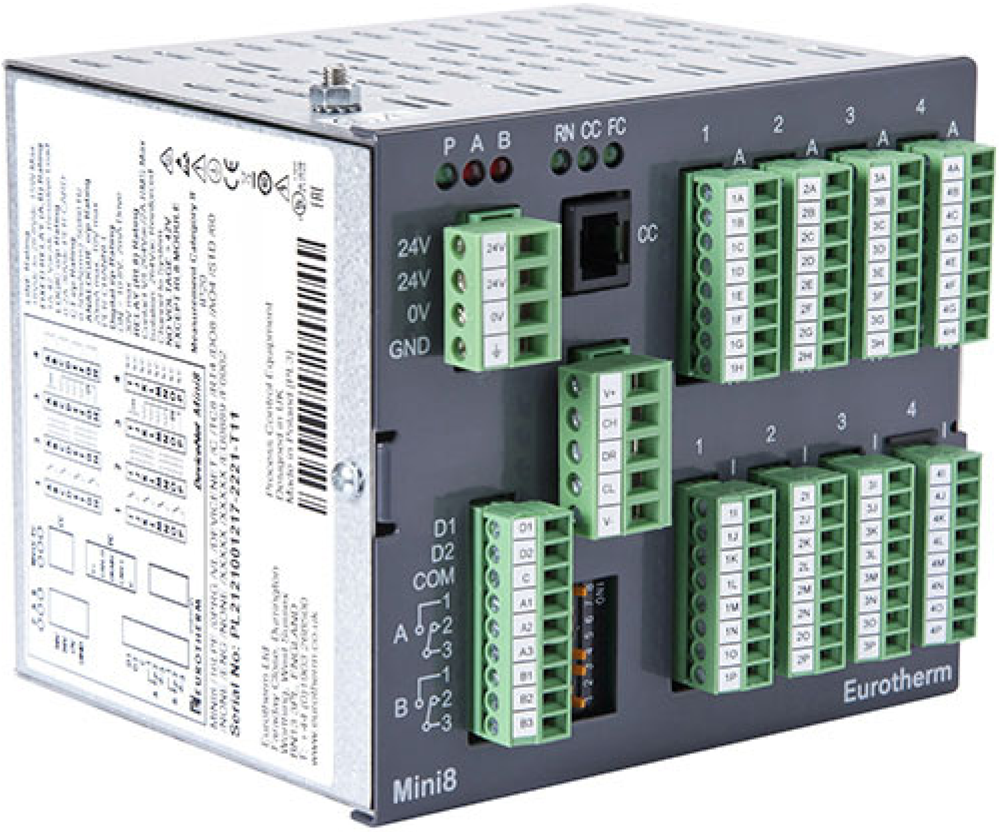
Do not use, or implement a controller configuration (control strategy) into service without ensuring the configuration has completed all operational tests, been commissioned and approved for service.

During commissioning ensure all operating states and potential fault conditions are carefully tested.

Failure to follow these instructions can result in death, serious injury or equipment damage.

What's in this Chapter

What Instrument Do I Have?



The Mini8 loop controller is a compact DIN rail mounted multi-loop precision PID controller and data acquisition unit. It offers a wide choice of I/O and a selection of Ethernet, DeviceNet and serial industrial communications protocols.

The controller mounts on 35mm Top Hat DIN Rail. It is designed for permanent installation, for indoor use only, and to be enclosed in an electrical panel or cabinet.

It is delivered pre-assembled with I/O and communications options as specified in the order code.

Eurotherm iTools PC based configuration software is used for commissioning and programming, this is available free of charge from the Eurotherm website. Mini8 loop controller with firmware V5.0+ is an updated model of the previous Mini8 loop controller allowing faster processing and more wiring. Applications can be converted from previous versions by means of an iTools migration tool. Some functions have been changed or removed, and details of these are given in "Comparison To Previous Versions" on page 25.

All Safety & EMC information is in the Chapter titled "Before you Start" on page 13.

See "Technical Specification" on page 360 for further details.

Note: Whenever the symbol ☺ appears in this User Guide it indicates a helpful hint.

Comparison To Previous Versions

What Has Changed?

The enhancements to the Mini8 loop controller with the introduction of the V5.0 firmware are as follows:

- A new higher performance microcontroller
- Integrated Ethernet communications with Achilles communication qualification
- The latest Eurotherm control algorithms
- SuperLoop with Cascade function

There has been no changes to the external dimensions and casing, or to the physical wiring of the unit. In most cases, the Mini8 loop controller V5.0+ can be used as a functional replacement for an pre-V5.0 Mini8 loop controller, without changes to technical drawings or external communications interfaces.

What is not supported?

The following functions are NOT supported by Mini8 loop controller V5.0+:

- **Real time clock**
  The real time clock requires a battery to maintain the time when the unit is unpowered. This is also used in pre-V5.0 Mini8 loop controllers to retain the serial number and configuration data – once the battery expires, it requires replacement which must be done in an approved Eurotherm service center on an approximately 7 year cycle, including units in stores used for spares. If this is not done, serial numbers and configuration data are lost. The lithium battery technology is problematic from the point of view of transport and environmental impact. We are therefore taking this opportunity to remove it from the product, replacing the serial number and configuration storage with non-volatile FRAM memory.

  The real time clock related functions (Alarm Log, Timed Events) are therefore not available in the Mini8 loop controller V5.0+.
- **Profibus DP and EtherNet/IP**
  These protocols are not supported in the Mini8 loop controller V5.0+.
- **Programmer**
  The programmer is not supported in the Mini8 loop controller V5.0+.

- **Non-isolated Serial Communications**
  There are two options for serial communication hardware in the pre-V5.0 Mini8 loop controller: isolated and non-isolated. The vast majority of existing applications use the non-isolated version, which is slightly lower cost. However, this creates some risks of electromagnetic interference (EMI), particularly with the new higher performance microprocessor, and so the non-isolated option has been discontinued.

  It is mandated that the point to point communications via the RJ11 configuration connector is not used in applications, for example for panel connections or I/O. This connection is non-isolated and creates potential for EMI risks.

  It is strongly recommended that users convert, where possible, to Ethernet-based protocols which support multiple connections on the same cable and are generally of a similar list price to the lower performance serial communications.

  The Clone Migration Tool (CMT) will automatically migrate applications using non-isolated serial comms, to use the isolated equivalent.

- **Changes to Function Blocks**
  There have been various changes to the function block library, reflecting the pre-V5.0 and V5.0+ Eurotherm control library. Generally, equivalent functions are available.

  Most significantly:

  - LIN16 block has been replaced with the new LIN32 block.
  - The Zirconia block has been removed, as there are no Zirconia high impedance inputs on the Mini8 loop controller V5.0+.
  - Removal of Alarm (Analog) and DigAlarm (Digital) alarm blocks, replaced with the “Generic” Alarm block as found in EPC3000. All previous alarm types and modes continue to be supported. Some of the type enums have changed, but are replaced with the equivalent.
  - Removal of the Load simulation function block.

- There are minor differences with the communications and instrument blocks, with changes to parameterization and the logical presentation of information.

- **Sub-assemblies**
  Communications boards, PSU, and microprocessor sub-assemblies are not available in the Mini8 loop controller V5.0+. I/O Modules are available.

- **EC8 and FC8 Options**
  These options provide an 8-loop extrusion and furnace controller respectively, and have been discontinued due to very low usage.

All other existing Mini8 loop controller functions remain available.

How Do I Migrate To The New Version?

A software feature (Clone Migration Tool) is provided to enable automatic migration of pre-V5.0 Mini8 loop controller applications.

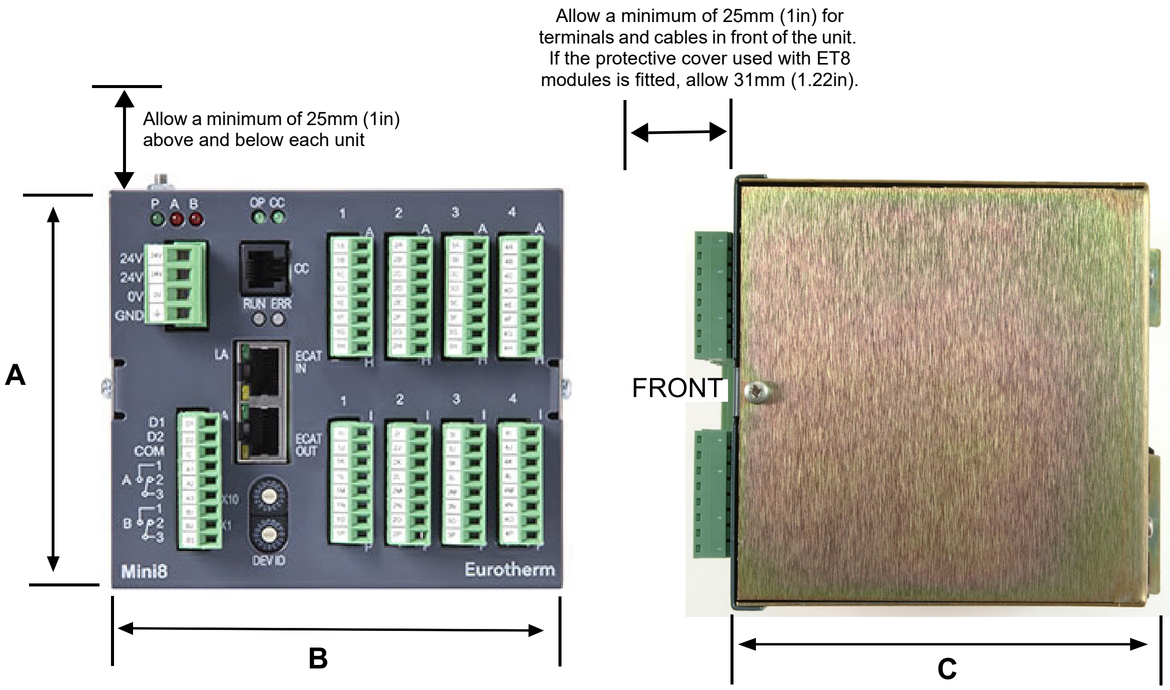
Refer to the iTools help for further details of this tool and its use, see "Using iTools" on page 154.

How to Install the Controller

This instrument is designed for permanent installation, for indoor use only, and to be enclosed in an electrical panel.

Select a location where minimum vibrations are present and the ambient temperature is between 0 and 55°C (32 and 131°F).

Dimensions



Dimension	mm	in
A	108	4.25
B	124	4.88
C	115	4.53

Figure 1 Mini8 Loop Controller Dimensions

To Install the Controller

Proceed as follows:

1. Use 35mm (1.38in) symmetrical steel DIN Rail to EN50022-35 x 7.5 or 35 x 15. The DIN Rail must be suitably bonded to protective earth ground.
2. Mount the DIN Rail horizontally as indicated in Figure 1. The Mini8 loop controller is NOT designed to be mounted in other orientations.
3. Hook the upper edge of the DIN rail clip on the instrument on the top of the DIN rail and push.
4. To remove use a screwdriver to lever down the lower DIN rail clip and lift forward when the clip has released.
5. A second unit on the same DIN rail may be mounted adjacent to the unit.
6. A second unit mounted above or below the unit requires a gap of at least 25mm (1in) between the top of the lower one and the bottom of the higher one.

7. Allow a minimum of 25mm (1in) for terminals and cables in front of the unit. If the protective cover used with ET8 modules is fitted, allow 31mm (1.22in).

Protective/Input Cover

If at least one ET8 module is fitted, the protective/input cover should be fitted. This provides thermal stability so that the high specification of the ET8 card is met. Figure 2 shows the Mini8 loop controller with this cover fitted. The image shows the protective cover mounted with the slot at the bottom, to accommodate alternative cabling requirements, this cover can be mounted with the slot at the top.

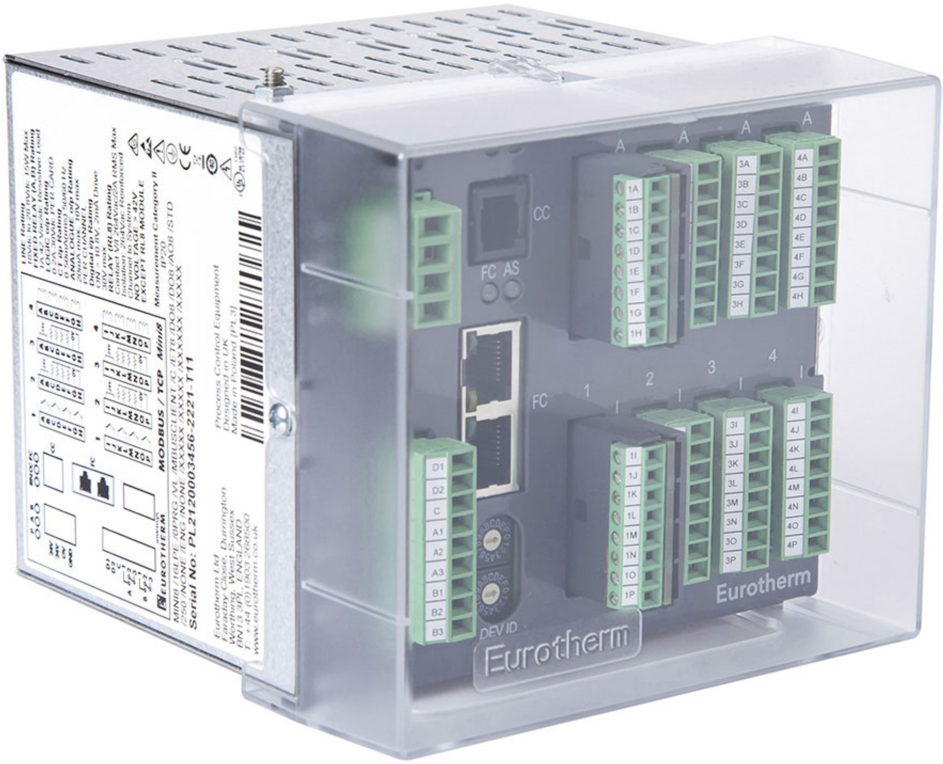


Figure 2 View of Mini8 Loop Controller with Protective Cover Fitted

Environmental Requirements

Mini8 loop controller	Minimum	Maximum
Temperature	0°C (32°F)	55°C (131°F)
Humidity (non condensing)	5% RH	95% RH
Altitude		2000m (6562ft)

Electrical Connections – Common to All Instruments

DANGER

HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH

Ensure the mandatory protective ground connection is connected during installation. Connection of this protective ground connection must be made before turning on any power supplying this product.

The Mini8 loop controller is intended for operation at safe low voltage levels, except the RL8 relay module. Voltages in excess of 42V must not be applied to any terminals other than the RL8 relay module.

Failure to follow these instructions will result in death or serious injury.

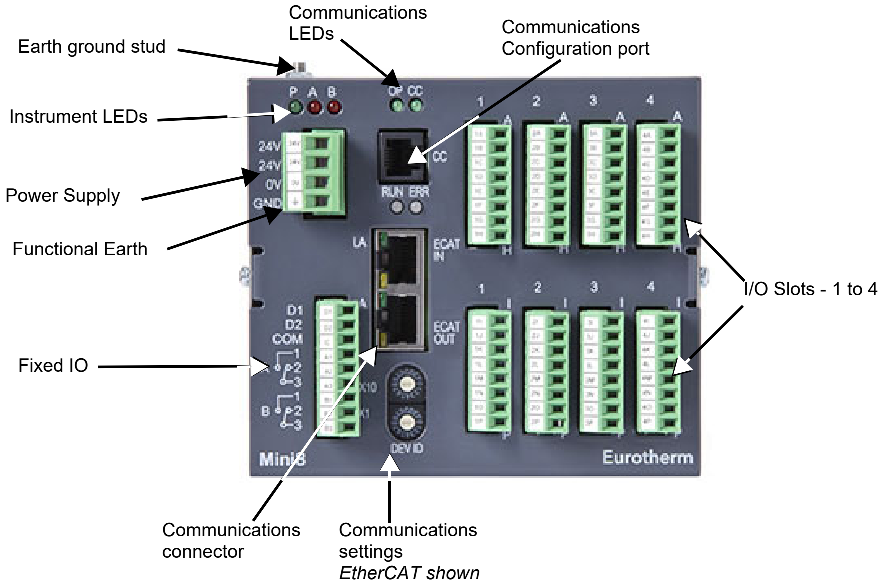
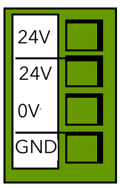


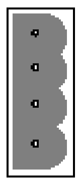
Figure 3 Terminal Layout for Mini8 Loop Controller

Power Supply

The power supply requires a supply between 17.8 to 28.8Vdc, 15W maximum.

24V	Ø	24V dc
24V	Ø	24V dc
0V	Ø	0V dc
GND	Ø	Functional Earth





Connector terminals will accept wire sizes from 0.2 to 2.5, 24 to 12 AWG.

The Power Supply Terminal marked GND should only be connected on older model units, which do not have a protective earth ground stud present. The GND terminal is a functional earth connection, which is used for EMC compliance purposes.

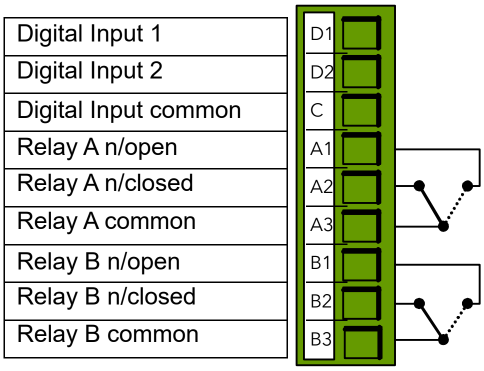
Protective Earth Ground Stud

A minimum cable gauge of 2.0 mm² CSA (14 AWG), fitted with an M4 ring terminal, not exceeding 50 cm in length must be used.

Connection should be made between the Mini8 loop controller protective earth ground stud and the steel DIN Rail. The steel DIN Rail must be bonded to protective earth ground in the application.

Fixed IO Connections

These I/O are part of the power supply board and are always fitted.



Digital Inputs:

- ON requires +10.8V to +28.8V.
- OFF requires -28.8V to +5V.
- +5V to +10.8V is undefined.
- Typical drive 2.5mA at 10.8V.

Relays contacts: 1A max, 42Vdc. These contacts are NOT rated for mains operation.

Digital Communications Connections

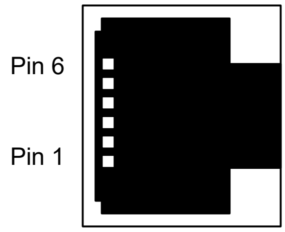
Two communications connections are fitted – a Modbus Configuration port (RJ11) and a Fieldbus port.

The Fieldbus is either Isolated Modbus EIA-485, DeviceNet, or Ethernet Modbus/TCP.

Configuration Communications Port (CC)

The Configuration Communications Port (CC) (Modbus) is on an RJ11 socket. It is always fitted just to the right of the power supply connections. It is a point to point EIA-232 connection. Eurotherm supply a standard cable to connect a serial COM port on a computer to the RJ11 socket, part no. SubMin8/cable/config.

9 pin DF to PC COM port (RS232)	RJ11 Pin	Function
-	6	N/C
3 (Tx)	5	Rx
2 (Rx)	4	Tx
5 (0v)	3	0v (gnd)
	2	N/C
	1	N/C (Reserved)



Note: RJ11 is for configuration only, not recommended for connection to display panels or other plant equipment.

See also "Configuration Communications Port" on page 126.

Shielded Communications Cables

Use shielded cables. To reduce the effects of RF interference, ground the transmission line at one end of the shielded cable. However, take care to remove differences in ground potentials allowing circulating currents to flow as these can induce common mode signals in the data lines. Where doubt exists it is recommended that the shield be grounded at only one section of the network. This applies to all communications protocols.

Note: Shielded cables as used for communication connections, such as EtherNet, are connected to the Mini8 loop controller case through the RJ45 connector. Take care to avoid ground loops as the body of the Mini8 loop controller is grounded.

Electrical Connections for Modbus RTU

For Modbus operation see "Modbus" on page 130.

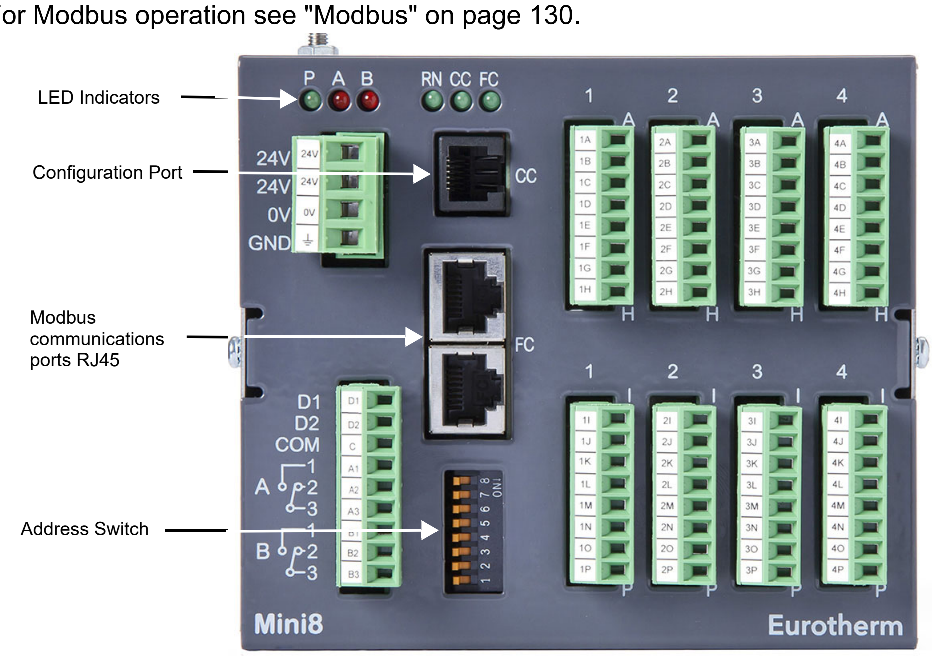


Figure 4 Front Panel Layout Modbus

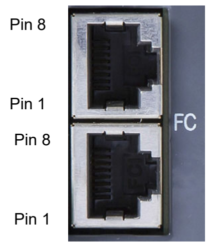
Isolated Modbus Connectors

In the Mini8 loop controller two RJ45 sockets are provided on the front panel for Isolated Modbus connections. One is for the incoming connection to a PC acting as a Client (Master), the second may be used either to loop onto the next instrument or for a line terminator, see Figure 10.

The wiring of the RJ45 plug allows both EIA-485 3-wire or EIA-485 4-wire or EIA-422 connections.

To construct a cable for EIA-485/EIA-422 operation use a shielded cable with twisted pairs plus a separate core for common.

RJ45 pin	3-wire	5-wire
8		RxA
7		RxB
6		Ground
5		
4		
3	Ground	Ground
2	A	TxA
1	B	TxB
Plug shroud to cable shield		



The 2000 series Communications Handbook, part number HA026230, gives further information on digital communications and is available on www.eurotherm.com.

EIA-485

EIA-485 is a standard defining the electrical characteristics of drivers and receivers for use in balanced digital multipoint systems. A balanced line consists of two identical conductors, other than ground, to transmit and receive the signal. This is usually referred to as a 2-wire system, or sometimes 3-wire. The two wires consist of a shielded twisted pair of equal length and equal impedances designed to reduce the effects of radiated and received electromagnetic interference. Terminating resistors are required at either end of the transmission line to reduce the effects of reflected signals. The EIA-485 standard is, thus, suited for use over long distances and in electrically noisy environments.

The Mini8 loop controller also provides connections for EIA-485 4-wire or EIA-422. This system consists of two shielded twisted pairs. One pair is used for transmit and the second for receive. A common is also provided.

One or more devices configured as network slaves (servers) may be connected to such a network in a linear, multi-drop configuration as described in "EIA-485 to EIA-232 Converter" on page 33 and "Client with Multiple Servers Short Network" on page 34.

Direct Connection – Client (Master) and One Server (Slave)

It is a common requirement to connect one Client (Master) and one Server (Slave). Termination resistors (RT) are to be installed at both the transmitter end and receiver end of the cable. These are particularly required for long cable runs (e.g. 2m to 200m) although for short local connections it may be found that these are not strictly necessary.

A Modbus terminator is available from your supplier which is designed to fit into the spare RJ45 connector on the Mini8 loop controller. The order code is SubMin8/RESISTOR/MODBUS/RJ45. It is colored black.

Example 1: Two-wire EIA-485 Connection

For 2-wire both Client (Master) and Server (Slave) ends act as Tx and Rx.

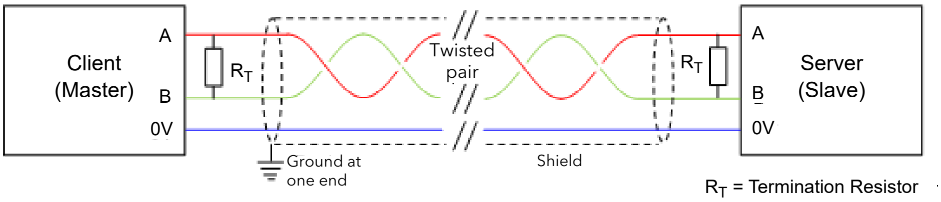


Figure 5 Two-wire EIA-485 Connection

Example 2: Four-wire EIA-485 Connection

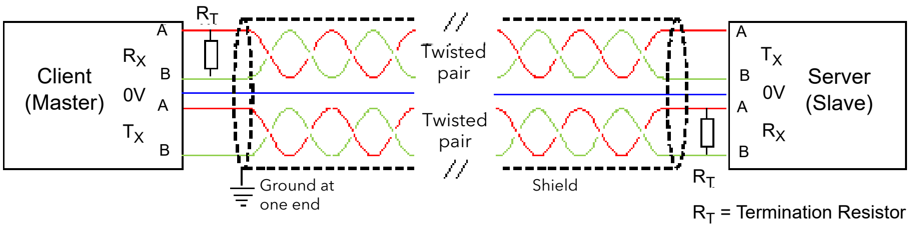


Figure 6 Four-wire EIA-485 Connection

EIA-485 to EIA-232 Converter

In practice it is often necessary to use a buffer to convert EIA-485 (or EIA-422) connections from the Mini8 loop controller to the serial port of the PC. The use of an EIA-485 board built into the computer is not recommended since this board may not be isolated and the Rx terminals may not be biased correctly for this application. This may cause electrical noise problems or damage to the computer.

To make the connections between the converter and the RJ45 connection on the Mini8 loop controller, either create a patch cable and connect the open end to the converter or, using twin shielded cable, crimp an RJ45 plug on the Mini8 loop controller end.

Connections for an EIA-485 to EIA232 converter are shown in the following diagrams.

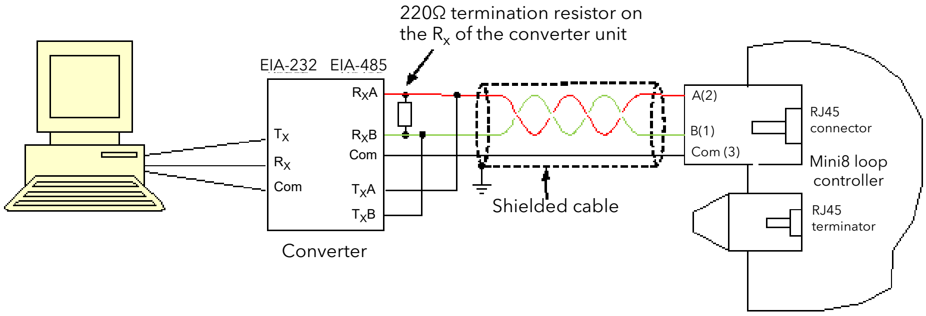


Figure 7 Communications Converter - 2-wire Connections

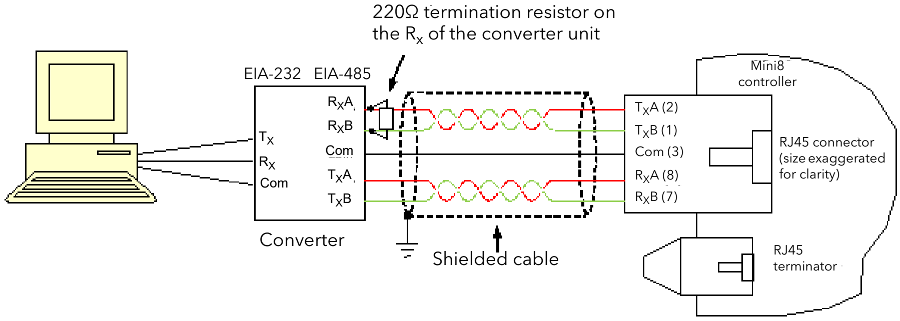


Figure 8 Communications Converter - 4-wire Connections

The above diagrams assume a serial port on the PC. For a PC using USB a USB to serial converter is required between the PC and the converter.

Client with Multiple Servers Short Network

The EIA-485 standard allows one or more instruments to be connected (multi dropped) using a 2-wire or 4-wire connection, with cable length of less than 1200m (3937ft). Up to 31 Servers (Slaves) and one Client (Master) may be connected. Servers (Slaves) may be Mini8 loop controllers or other instruments such as Eurotherm controllers or indicators.

NOTICE

COMMUNICATION LINE PARAMETERS

The communication line must be daisy chained from device to device and it must be correctly terminated. A Modbus terminator containing the correct termination resistors is available from Eurotherm, order code: SubMin8/RESISTOR/MODBUS/RJ45.

Failure to follow these instructions can result in equipment damage.

The Modbus terminator is colored BLACK.

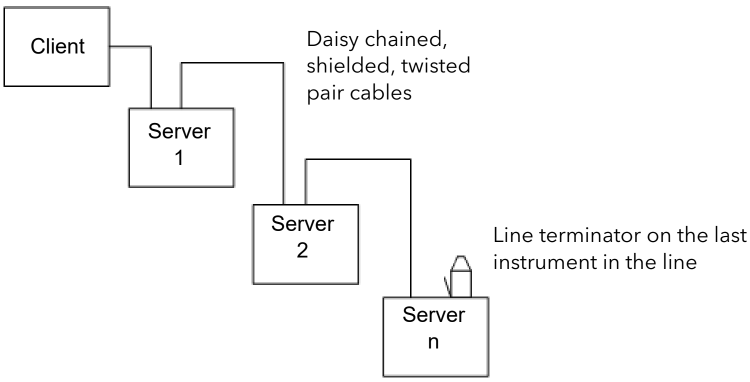


Figure 9 Multiple Servers (Slaves) - Overview

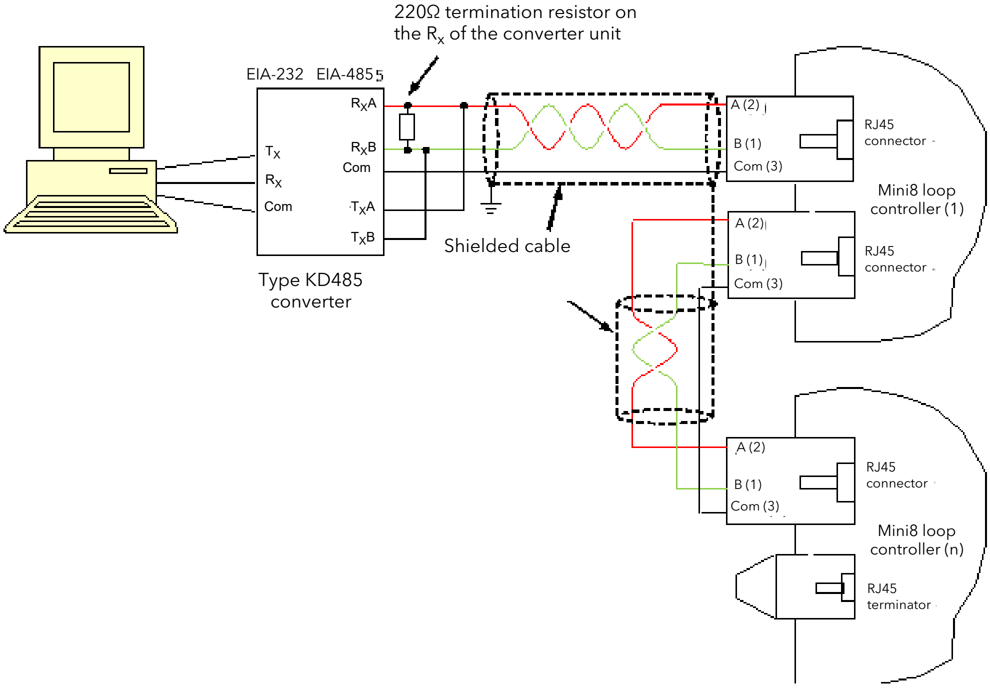


Figure 10 Multiple Servers (Slaves) - EIA-485 2-wire Connections

Wiring Connections for Modbus Broadcast Communications

The Digital Communications module for the broadcasting Mini8 loop controller must be the Field Comms and is only EIA-485/EIA-422. EIA-232 is not available.

Standard patch cables cannot be used, as the connections do not ‘cross over’. Wire using twisted pair(s) cable and crimp on the appropriate RJ45 or RJ11 plug.

EIA-485 2-wire

Connect A (+) to A (+).

Connect B (-) to B (-).

This is shown diagrammatically below:

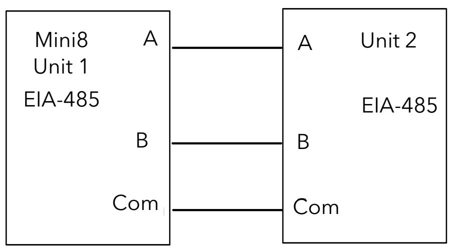


Figure 11 Rx/Tx Connections EIA-485 2-wire

EIA-422, EIA-485 4-wire

Rx connections in the Client (Master) are wired to Tx connections of the Server(s) (Slave(s)).

Tx connections in the Client (Master) are wired to Rx connections of the Server(s) (Slave(s)).

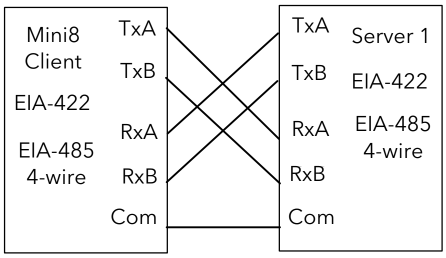


Figure 12 Rx/Tx Connections for EIA-422, EIA-485 4-wire

Electrical Connections for DeviceNet

DeviceNet uses a 5-way, 5.08mm pitch, connector/screw terminal. The DeviceNet bus is powered (24V) from the system network, not from the instrument. The Mini8 loop controller requirement is a load of around 100mA. For the address switch see "Ethernet (Modbus TCP)" on page 146.

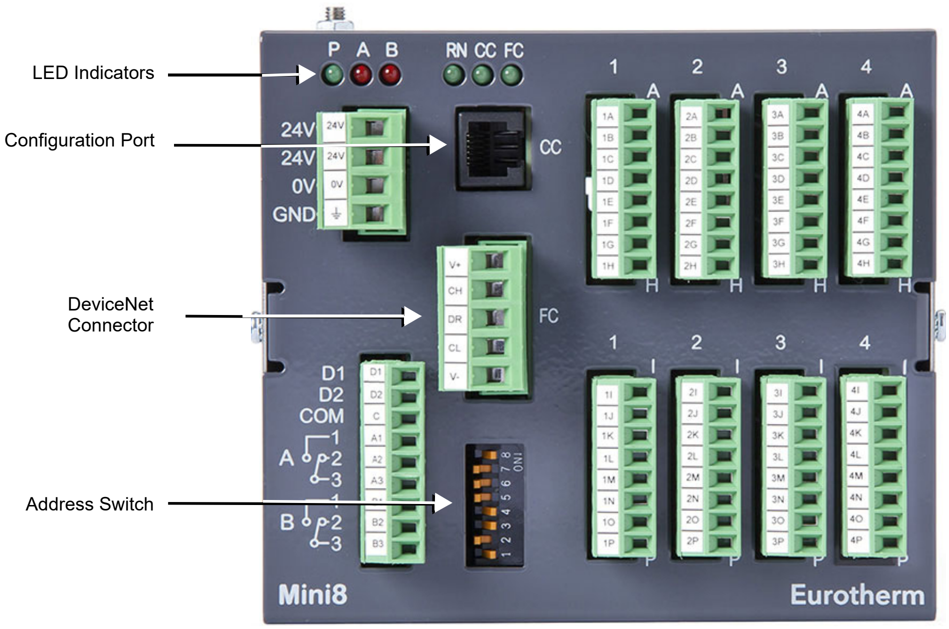
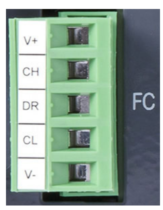


Figure 13 Front Panel Layout DeviceNet

DeviceNet Connector

Pin	Legend	Function
5	V+	V+
4	CH	CAN HIGH
3	DR	DRAIN
2	CL	CAN LOW
1	V-	V-



Mini8 loop controller Label	**Color**	**Description**

V+	Red	Network power positive terminal. Connect the red wire of the DeviceNet cable here. If the network does not supply the power, connect the positive terminal of an external 11-25Vdc power supply.
CAN_H	White	CAN_H data bus terminal. Connect the white wire of the DeviceNet cable here.
SHIELD	None	Shield/Drain wire connection. Connect the DeviceNet cable shield here. To avoid ground loops, ground the network in only one location.
CAN_L	Blue	CAN_L data bus terminal. Connect the blue wire of the DeviceNet cable here.
V-	Black	Network power negative terminal. Connect the black wire of the DeviceNet cable here. If the DeviceNet network does not supply the power, connect the negative terminal of an external 11-25Vdc power supply.

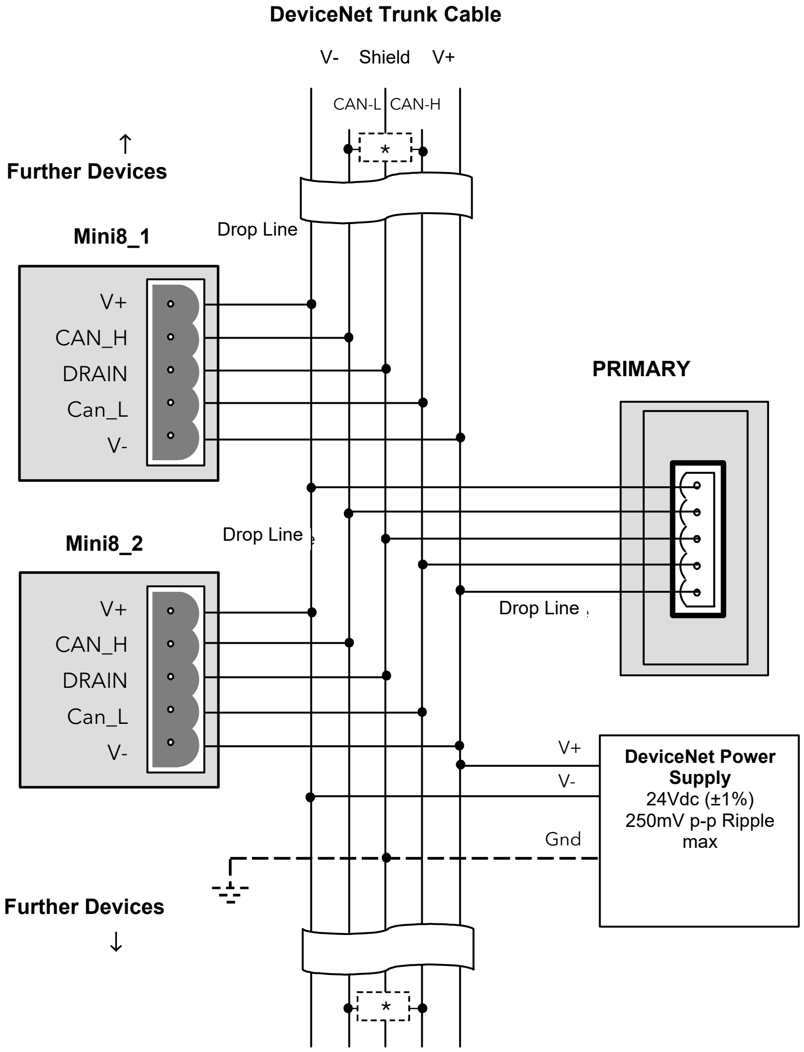
The DeviceNet specification states that the bus terminators of 121Ω should not be included as any part of a primary or secondary unit. They are not supplied but should be included in the cabling between CAN_H and CAN_L where required.

Network Length

Network length depends on Baud rate:

Network Length	**Varies w/speed, up to 4000m possible w/repeaters**		
Baud Rate	125bps	250bps	500bps
Thin trunk	100m (328ft)	100m (328ft)	100m (328ft)
Max drop	6m (20ft)	6m (20ft)	6m (20ft)
Cumulative drop	156m (512ft)	78m (256ft)	39m (128ft)

Typical DeviceNet Wiring Diagram



* 121Ω 1% 1/W terminating resistor must be connected across the blue and white wires at each end of the DeviceNet trunk cable.

Note: this resistor is sometimes included in the primary or other devices but should only be switched into circuit on the last device on the trunk cable.

Notes:

1. The DeviceNet network is powered by an external independent 24V supply which is separate from the internal powering of the individual controllers.
2. Power taps are recommended to connect the DC power supply to the DeviceNet trunk line.

Power taps include:

- A Schottky diode to connect the power supply V+ and allows for multiple power supplies to be connected.
- Two fuses or circuit breakers to help protect the bus from excessive current which could damage the cable and connectors.
- The ground connection, HF, to be connected to the main supply ground terminal at one point only.

See also the DeviceNet Communications Handbook HA027506.

Electrical Connections for Enhanced DeviceNet Interface

This version of DeviceNet has been added to use a standard connector widely used by Semiconductor machine builders. Configuration for both versions is the same and is described in the DeviceNet Handbook HA027506 which can be downloaded from www.eurotherm.com. The Enhanced DeviceNet interface uses a different connector, as described below, but cabling, cable specification and termination are the same as described in the previous section.

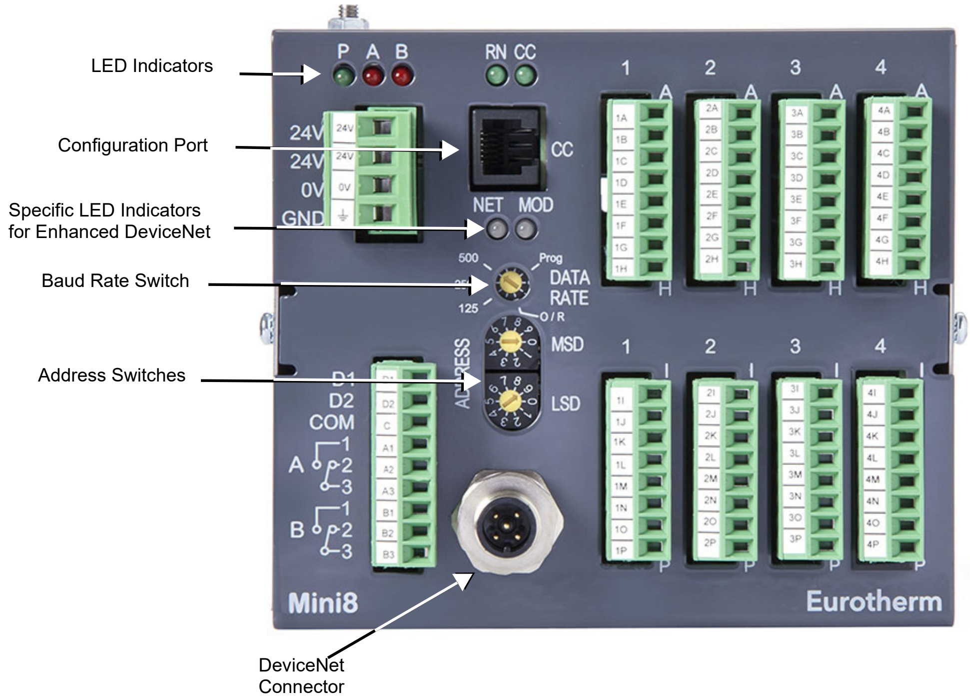
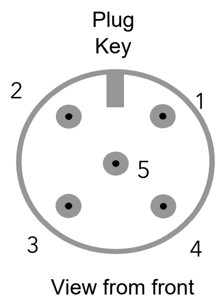


Figure 14 Enhanced DeviceNet Panel Layout

Enhanced DeviceNet Connector

The 5-way connector shown in the previous section is replaced by a ‘Micro-Connect’ circular 5-pin M12 male connector mounted in the module.

Pin	Legend	Function
5	CAN_L	CAN LOW
4	CAN_H	CAN HIGH
3	V-	V-
2	V+	V+
1	DR	DRAIN



Switches and LED Indicators

The Enhanced DeviceNet interface also uses different Module and Network Status indicators, address and baud rate switches. To set the Address and Baud Rate, see"Enhanced DeviceNet Connector" on page 39. For Module and Network Status indication see "Status Indication for Enhanced DeviceNet" on page 48.

Electrical Connections for Ethernet

The Ethernet connection uses standard Cat5e patch cables (RJ45). These would be used with a 10Base-T hub to create a network.

A crossover patch cable may be used ‘point-to-point’ that is, to connect a single instrument directly to a PC.

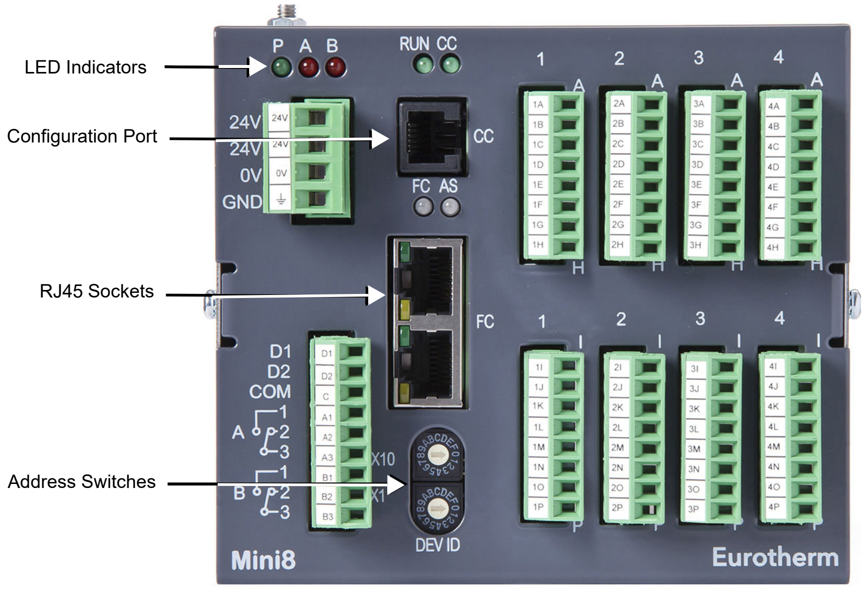
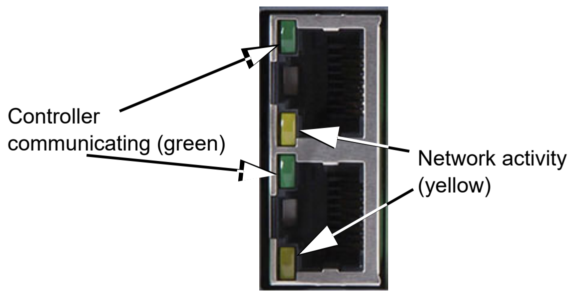


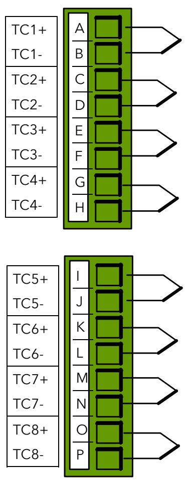
Figure 15 Ethernet Front Panel Layout

Connector: RJ45

Pin	Function
8	
7	
6	RX-
5	
4	
3	RX+
2	TX-
1	TX+



Electrical Connections for Thermocouple Input TC4, TC8 and ET8



The TC8 and ET8 thermocouple modules both take eight thermocouples (TC1 to TC8 on terminals A to P).

The TC4 module takes four thermocouples (TC1 to TC4 on terminals A to H).

They may be placed in any slot in the Mini8 loop controller.

Up to four thermocouple modules may be fitted in a Mini8 loop controller.

Each input can be configured to any thermocouple type or a linear mV input.

Notes:

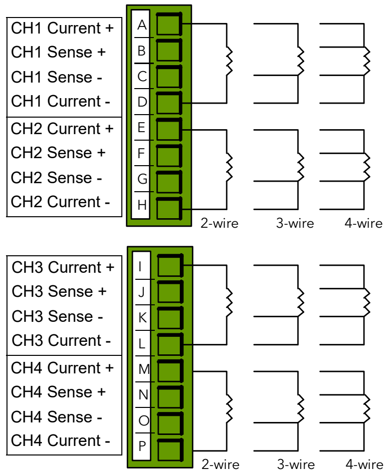
1.Configuration of Mini8 Controller is performed using ‘iTools’ configuration suite running on a PC.

2. If ET8 modules are fitted, also fit the supplied protective cover for thermal stability.

See subsequent chapters in this manual and specifically Example 1 given in "The I/O" on page 65 for further information.

Electrical Connections for RTD

The RT4 module provides four RTD / Pt100 or four RTD / Pt1000 inputs for 2, 3 or 4 wire connections.



Each input can be configured for Pt100 standard linearization or Pt1000 standard linearization. When configured for Pt100 the input will accept up to 420Ω. When configured for Pt1000 the input will accept up to 4200Ω.

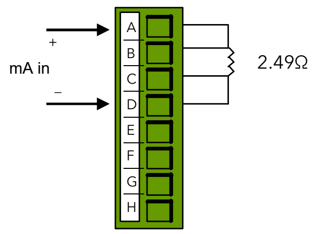
Up to four modules may be fitted in a Mini8 loop controller and they may be placed in any slot.

Note: Configuration of Mini8 Loop Controller is performed using ‘iTools’ configuration suite running on a PC.

See subsequent chapters in this manual and specifically Example 2 given in "The I/O" on page 65 for further information.

☺ Tip:

Spare RT4 input channels may be configured as mA inputs using a 2.49Ω resistor, order code: SubMini8/resistor/Shunt/249R.1 and setting the Resistance Range to Low (see "Using RT4 as mA input" on page 104.)

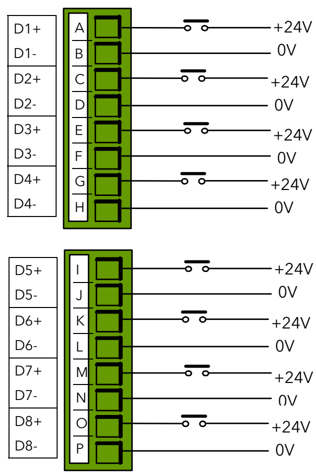


Electrical Connections for Logic Input DI8

The DI8 module provides eight logic inputs.

They may be placed in any slot in the Mini8 loop controller.

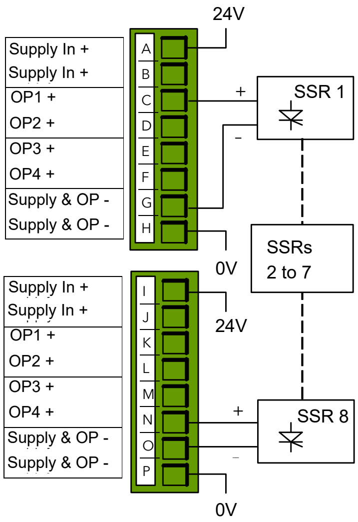
Up to four modules may be fitted in a Mini8 loop controller.



Digital Inputs:

- ON requires +10.8V to +28.8V.
- OFF requires -28.8V to +5V.
- +5V to +10.8V is undefined.
- Typical drive 2.5mA at 10.8V.

Electrical Connections for Logic Output DO8



The DO8 module provides eight logic outputs.

They may be placed in any slot in the Mini8 loop controller.

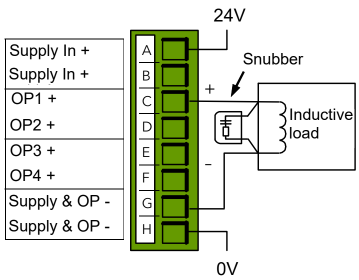
Up to four may be fitted in a Mini8 loop controller.

Each output can be configured to Time Proportioning or On/Off.

Supply In + (A,B,I,J) are all linked internally.

Supply In – (G,H,O,P) are all linked internally.

Electrical Connections for Inductive Loads



This section applies if logic outputs are used to switch inductive loads.

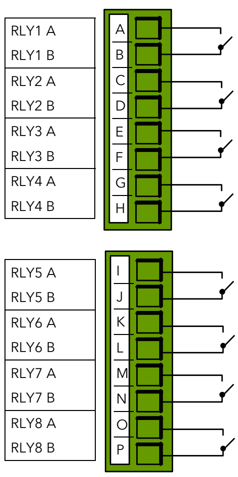
Some inductive loads may produce a large back EMF when switching off. If the back EMF is >30V this may cause damage to the switching transistor in the module.

For this type of load it is recommended to add transient suppressors or 'snubbers' across the coils as shown. A snubber typically consists of a 15nF capacitor in series with a 100Ω resistor.

Snubbers are available to order from your supplier by quoting part number SUB32-snubber.

It is the user's responsibility to determine the type of load which is to be used.

Electrical Connections for Relay Output RL8



The RL8 module provides eight relay outputs.

Note: Up to two modules may be fitted and in slots 2 and/or 3 only.

Relay contacts for full contact life:

- Maximum 264Vac 2A with snubber fitted.
- Minimum 5Vdc, 10mA

Snubbers are used to prolong the life of relay contacts and to reduce interference when switching inductive devices such as contactors or solenoid valves. If the relay is used to switch a device with a high impedance input, no snubber is necessary.

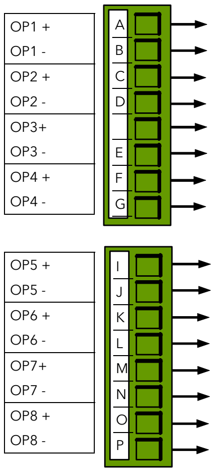
All relay modules are fitted internally with a snubber since these are generally required to switch inductive devices. However, snubbers pass 0.6mA at 110V and 1.2mA at 230Vac, which may be sufficient to hold on high impedance loads. If this type of device is used it will be necessary to remove the snubber from the circuit.

The relay module has to be removed from the instrument, see "Adding or Replacing an IO Module" on page 44. The snubber is removed from the relay module by inserting a screwdriver into one of the pair of slots either side of the track of each snubber network. Twist the screwdriver to break out this track between the slots.

This action is not reversible.

Electrical Connections for Analog Output AO4 and AO8

The AO8 modules provides eight analog outputs and the AO4 provides four analog outputs.



Each output is configurable within 0 to 20mA, max load 360Ω.

The AO4 offers OP1 to OP4 on terminals A to H.

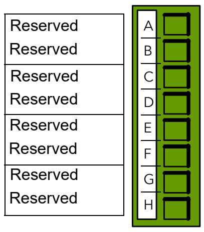
Note: Only one module may be fitted and in slot 4 only.

☺ Tip:

A 0 to 10V output can be obtained by scaling the drive to 0 to 10mA and fitting an external 1kΩ resistor (for example). Low load impedance may alter the results but this can be corrected by adjusting the output range accordingly.

Electrical Connections for Current Transformer Input Module CT3

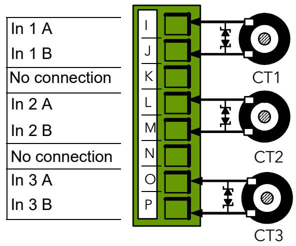
This provides inputs for three current transformers.



The heater load cables are threaded through the transformers.

Each input is 50mA max into 5Ω.

The current transformers provide channel isolation; there is no channel-to-channel isolation in the module.



It is recommended that the current transformer is fitted with a voltage limiting device such as two back-to-back zener diodes between 3 and 10V, rated for 50mA.

There are three CT inputs, one for each phase. Up to a maximum of 16 heaters may be threaded through the CTs but with a further limit of six heater wires through each individual CT.

See "Current Monitor" on page 107 for typical circuit arrangements.

Note: If a CT3 module is fitted in a controller, a DO8 module must also be fitted. Otherwise, the controller cannot be configured.

Adding or Replacing an IO Module

Refer to the Mini8 Module Changing Guide (HA033632ENG) for details of adding or replacing IO modules.

Mini8 Loop Controller LED Indicators

LED indicators P, A and B are common to all Mini8 loop controllers and indicate the power and the state of the output relays as shown in the following table.

	P	A	B
Color	Green	Red	Red
OFF	Power off	Relay A De-energized	Relay B De-energized
ON	Power on (24V)	Relay A Energized	Relay B Energized

LED indicators RN and CC are common to all Mini8 loop controllers and show the status of the Mini8 loop controller and communications activity.

FC is replaced by Network and Module Status LEDs when DeviceNet communications modules are fitted.

RN is replaced by RUN when an Ethernet communications module is fitted.

	RN/RUN	CC	FC (Not Ethernet)	
			Modbus	DeviceNet
Color	Green	Green	Green	Green
Function	Run mode	Configuration activity (EIA-232)	Field comms activity	Status
OFF	Not running	--	Offline	Offline
Flashing	Standby	Config traffic	Traffic	Ready
ON	Running	--		Connected

	FC	AS
Color	Green	Green
ON	Connected	DHCP is enabled and has obtained an IP address from a DHCP server.
Flashing	Comms traffic received on FC comms port	No DHCP connection, but auto-ip is assigned to the instrument
OFF	No traffic on the FC comms port	All other cases

Notes:

1. The Modbus/Ethernet connector itself has two in-built LEDs (see "Electrical Connections for Ethernet" on page 40, and "Electrical Connections for Thermocouple Input TC4, TC8 and ET8" on page 40).
2. The Mini8 loop controller is controlling normally ONLY if the green RN (RUN for Ethernet) LED is permanently ON.
3. In iTools the parameter ‘Comms Network Status’ is available enumerated as shown in the following table. The enumerations correspond to the FC indicator as shown in the final column:

Table 1: DeviceNet Status parameter

'Status' Parameter Enumeration	Meaning	Corresponding FC LED
Running (0)	Network connected and running	On
Init (1)	Network initializing	Off
Ready (2)	DeviceNet traffic detected but not for this address	Flashing
Offline (3)	No DeviceNet traffic detected	Off
Bad_GSD (4)		
Offline (10)	Apparent duplicates. Used for SemiSIG DeviceNet applications	
Ready (11)		
Online (12)		
IOTimeout (13)		
LinkFail (14)		
ComFault(15)		

Status Indication for Enhanced DeviceNet

If an Enhanced DeviceNet module is fitted (see "Electrical Connections for Enhanced DeviceNet Interface" on page 39), two bi-color LEDs are used to indicate Network and Module status.

These two LEDs replace the single LED shown as FC on other modules, see previous section.

Network Status Indication

The network status LED (NET) indicates the status of the DeviceNet communications link as shown in the table below.

Note: The final column shows the enumerated values for the ‘Comms Network Status’ parameter available in iTools.

LED State	Network State	Description	‘Status’ Parameter Enumerations
OFF	Off	Module is not on line	OFFLINE (10)
Green flashing	On-line, not connected	Module is on line but has no connections established	READY (11)
Green ON	On-line and connected	Module is on line and has connections established	ONLINE (12)
Red flashing	Connection timed out	One or more connections have timed out	IO TIMEOUT (13)
Red ON	Critical communication problem detected	Communication problem detected that results in the device being incapable of communicating on the network	‘LINK FAIL’ (14)
Green/Red	Communications problem detected	Communications problem detected but the device has received an ‘Identify Communication Faulted’ Request	‘COMM FAULT’ (15)

Module Status Indication

The module status LED (MOD) has the functionality shown below:

LED State	Device State	Description
OFF	Off	No power applied to DeviceNet network.
Green/Red flashing	Self test	Irregular flash: LED power-up test. Regular flash: Interface module initializing. If the LED remains in this flashing state indefinitely, check the Baud rate switch setting.
Green ON	Operational	DeviceNet interface is operational.
Red ON	Unrecoverable fault detected	Mini8 Loop Controller not powered. Non-volatile memory checksum incorrect.
Red/off flashing	Recoverable fault detected	Communication problem detected between the network and the DeviceNet module.

Using the Mini8 Loop Controller

The Mini8 loop controller does not have a display. The only means of configuring it, and of interfacing with it during normal operation, is via digital communications.

The auxiliary communications port CC (RJ11) gives a Modbus interface, connected to iTools, for configuration and commissioning.

The main communications port, FC, offers Modbus, DeviceNet, or EtherCAT and is normally connected to the system of which the Mini8 loop controller is a part. It is the means by which the Mini8 loop controller is operated.

Below are ways the Mini8 loop controller may be used in a system. iTools is the preferred PC-based solution. The Modbus single register addressing is preferred for Operator panels and PLCs where floating point may not be available or necessary. Some parameters may also be read this way as floats or long integers.

iTools

iTools is a PC-based solution. The iTools suite allows configuration, commissioning, trend graphs and logging with OPC Scope, Superloop, Recipes and User pages with View Builder.

iTools OPC Open server

With an OPEN OPC server running on a PC all the Mini8 loop controller parameters are available to any third party package with an OPC client. The advantage of this is that all the parameters are addressed by name – the iTools OPC server handles all the physical communication addresses. An example would be with Wonderware inTouch using OPCLink. In this situation the user would not have to know any of the parameter addresses, and would just select a parameter by browsing through the namespace.

For example, Eurotherm.ModbusServer.1.COM1.ID001-Mini8.Loop.1.Main.PV.

Modbus, Single Register, SCADA addressing

The key parameters of the Mini8 loop controller are available at a fixed single 16-bit register address, independent of its configuration. These can be used with any device with a serial Modbus master. The parameters are listed in full with their addresses in "Modbus SCADA Table" on page 351.

By default, iTools displays the SCADA address of the relevant parameters.

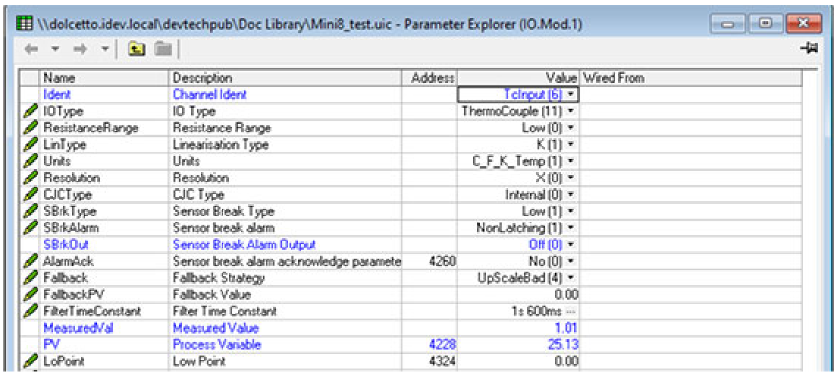


Figure 16 iTools Parameter Explorer showing SCADA addresses

As shown, not all the parameters within the instrument are available. If other parameters are required they can be obtained by using the Commstab folder. This allows up to 250 other parameters to be made available using indirection addressing. This is explained in "Modbus SCADA Table" on page 351.

Also note that in this area the resolution (number of decimal points) has to be configured and the serial Master has to scale the parameter correctly.

Modbus (Floating Point)

If the application requires the extra resolution, the Commstab folder also offers an alternative solution where a parameter can be indirectly addressed and communicated either as a floating point or as a double integer value – its ‘Native’ format. This can be used with any device, for example, PC or PLC, with a serial Modbus master, able to decode a double register for floating point numbers and long integers. See "Modbus SCADA Table" on page 351.

Fieldbus

The Mini8 loop controller may be ordered with the option of Isolated Modbus EIA-485, DeviceNet, or Ethernet Modbus/TCP or EtherCAT.

DeviceNet comes pre-configured with the key parameters of eight PID loops and alarms (60 input parameters process variables, alarm status, and so on and 60 output parameters – setpoints, and so on). Loops 9-16 and SuperLoops 9-24 are not included in the DeviceNet tables as there are insufficient attributes for the DeviceNet parameters. See "DeviceNet Parameter Tables" on page 353.

Mini8 Loop Controller Execution

The nominal update of all inputs and function blocks is 110ms.

The iTools Operator Interface

Much of this manual is about configuring the Mini8 loop controller with iTools. However iTools also provides a commissioning tool and can be used as a long-term operator view.

First it is necessary to go ‘on-line’ to the Mini8 loop controller(s). This assumes the communication ports have been wired up to the COM port on the iTools computer (see "Digital Communications" on page 126).

Scanning

Open iTools and, with the controller connected, press 

on the iTools menu bar. iTools will search the communications ports for recognizable instruments. Controllers connected using the RJ11 configuration port or with the configuration clip (CPI), may be found at address 255 (as a single point-to-point connection) or on a multidrop EIA-485 or EIA-422 network will be found at the address configured in the controller.

The iTools handbook, part no. HA028838, provides further step by step instructions on the general operation of iTools. This and the iTools software may be downloaded from www.eurotherm.com.

When an instrument is found on the network it will be shown as, for example:

‘COM1.ID001-Min8’ which represents <computer com port>.ID<instrument address>-<Instrument type>

Stop the scan once all the instruments have been found.

Figure 17 Synchronizing message

Once an instrument is found on the network a message ‘sync pending’ or ‘synchronizing’ is displayed next to it whilst iTools extracts the exact configuration from the instrument. Wait until this message disappears.

Browsing and Changing Parameter Values

Once the instrument is synchronized, the parameter navigation tree is displayed. The contents of this tree will vary depending on the actual configuration of the instrument.

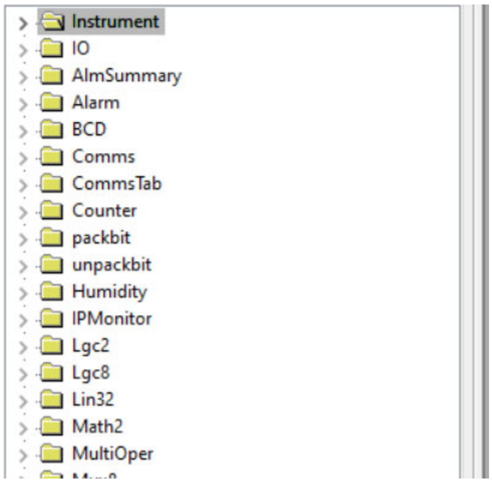


Figure 18 Parameter Navigation Tree

To view or change a parameter:

1. Highlight the folder
2. Press 

 to get the parameter window. Double-click on a folder to open the Parameter List View for the selected Block.
3. Clicking on a Folder when a Parameter List View is open, will update the Parameter List to the selected Block.
4. To change the value of a parameter,
    a. Click the parameter value.
    b. Type in the new value. Note a pop-up window indicates the current value, and the high and low limits.
    c. Press <Enter> to enter the new value or <Esc> to cancel.

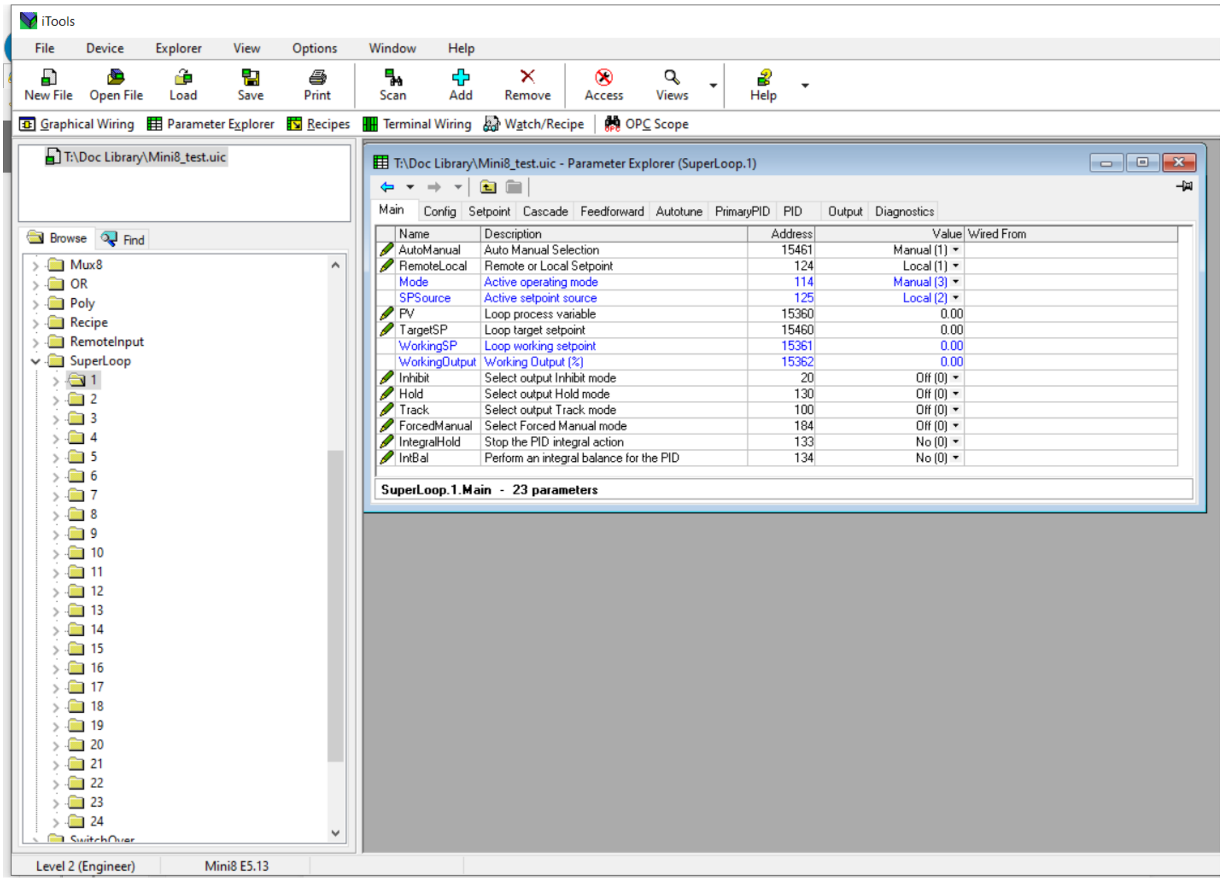


Figure 19 Parameter Values

The 'Access' button puts the controller into configuration mode. In this mode the controller can be set up without its outputs being active. Click 'Access' again to return to operating level.

To find a parameter use the 'Find' tab at the bottom of the folder list.

☺ Tip: In parameter lists:

- Parameters in BLUE are read-only.
- Parameters in BLACK are read/write.

☺ Tip: Every parameter in the parameter lists has a detailed description in the help file – just click on a parameter and press Shift-F1 on the keyboard or right-click and select parameter help.

Recipes

A recipe is a list of parameters whose values can be captured and stored in a dataset which can then be loaded at any time to restore the recipe parameters, thus providing a means of altering the configuration of an instrument in a single operation even in Operator mode. Recipes can be set up and loaded using iTools.

A maximum of five datasets are supported, referenced by name, and defaulted to be the dataset number i.e. 1…5.

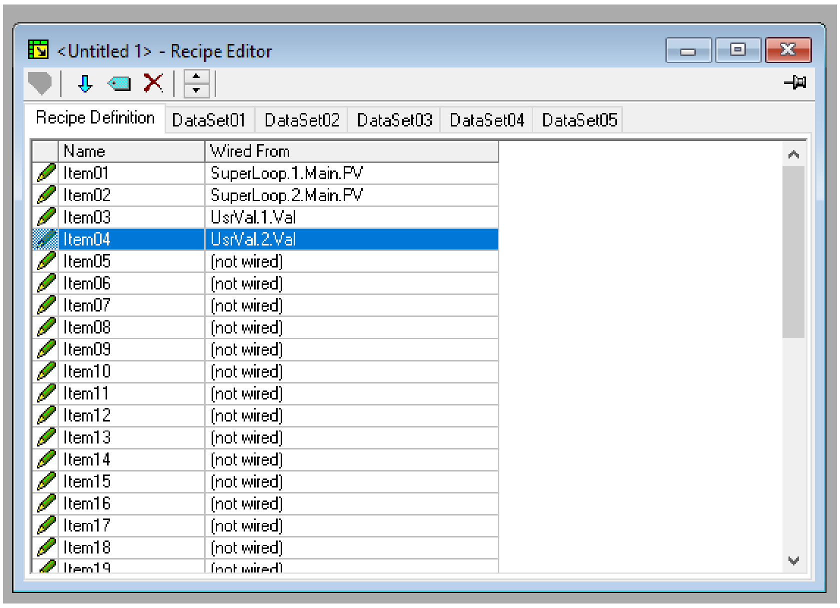
By default each dataset consists of 40 parameters which must be populated by the user. A recipe can take a snapshot of the current values and store these into a recipe dataset.

Each dataset can be given a name using iTools configuration software.

Recipe Definitions

To define a Recipe, press 

then, select the 'Recipe Definition' and 'Recipe Dataset' tabs as required.



The Recipe Definition table contains a set of 40 parameters. Not all 40 parameters need to be wired.

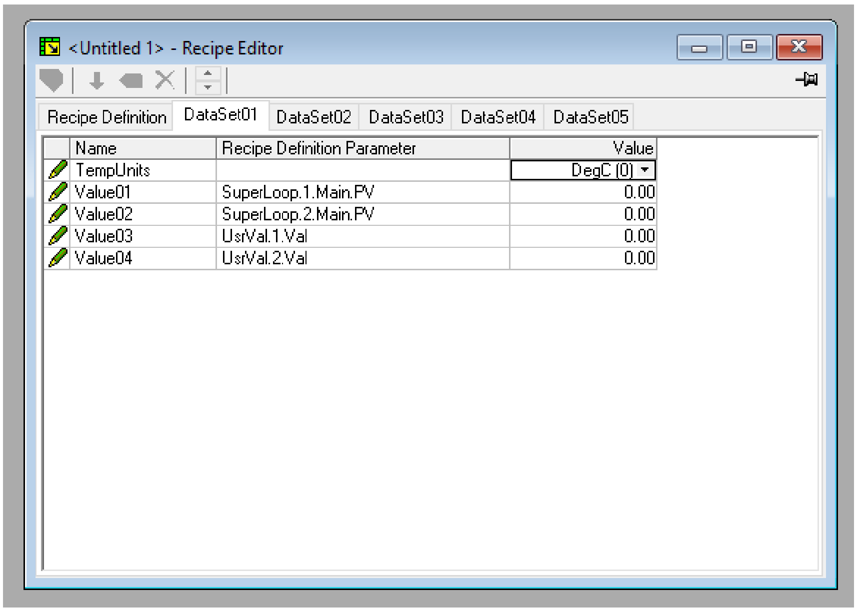
The Recipe Definition tab allows the user to produce a customized list.

To add parameters:

1. Double-click in the next empty item.
2. This opens the parameter list to choose from.
3. Adding a parameter to the list will automatically populate the five datasets with the current value of the added parameter.

Data Sets

Up to five DataSets are available, each being a recipe for a particular batch or process.

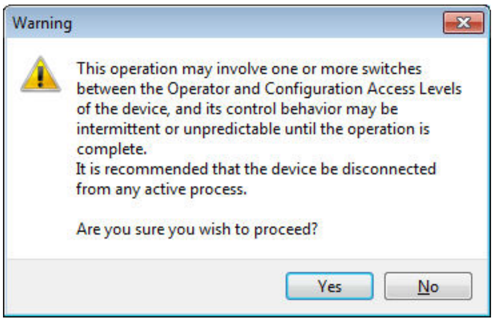


To Save the Data Set

1. Set up the required values in the selected data set—see the example above.
2. Press Enter.
3. Press the ‘Update device flash’ (Ctrl+F)’ button in the top left of the Flash Editor display to update the controller. This sets the values into all five of the controller datasets.

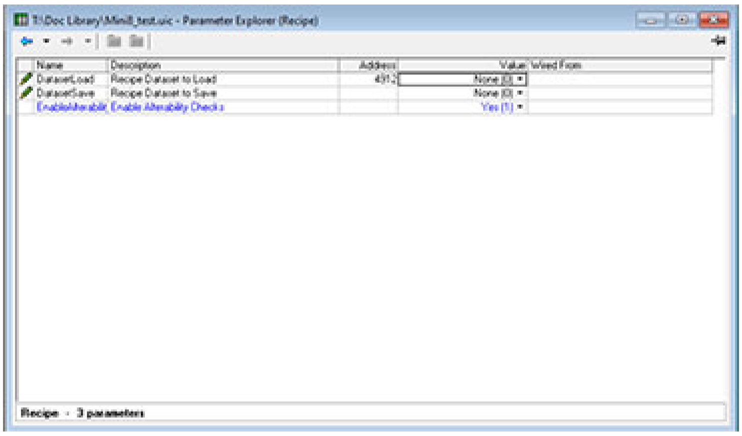
Note: Saving in the controller will save the current values into one dataset.

Since this operation may involve one or more switches between Operator Level and Configuration Level it is recommended that the controller is disconnected from the process. A warning message is shown.



To Load a Data Set

1. In the browser list select ‘Recipe’.



2. Select the required Dataset.

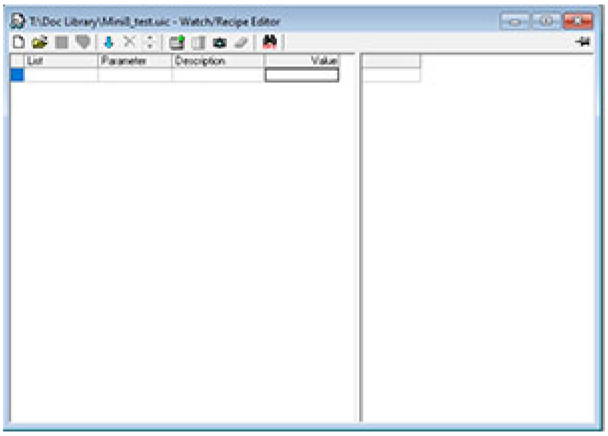
Watch Recipe Editor

Click on the Watch/Recipe tool button, by selecting 'Watch/Recipe' in the Views menu, or via the shortcut (Alt+A). The window is in two parts: the left part contains the watch list; the right-hand part contains a data set, initially empty and unnamed.

Watch Recipes are run from iTools, and are not stored or run from the device, i.e. iTools must be running and connected to a specific device.

The window is used:

1. To monitor a so-called 'watch list' of parameter values. The watch list can contain parameters from many different lists within the same device.
2. To create 'data sets' of parameter values which can be selected and downloaded to the device, in the sequence defined by the recipe. The same parameter may be used more than once in a recipe.



Creating a Watch List

After opening the window parameters can be added to it as described below. Parameters can be added only from the device to which the Watch/Recipe window relates (that is, parameters from more than one device cannot be placed in one Watch list). The values of the parameters update in real time, allowing the user to monitor, simultaneously, a number of parameters which might otherwise be unrelated.

Adding Parameters to the Watch List

To add parameters to the watch list, proceed as follows:

1. Parameters can be clicked and dragged into the watch list grid from elsewhere in iTools (for example: the main browse tree, the Parameter Explorer window, the Graphical Wiring Editor (if applicable)). The parameter is placed either in the empty row at the bottom of the list, or 'on top' of an existing parameter, in which case it is inserted above this parameter in the list, the remaining parameters moving down one place.
2. Parameters can be dragged from one position in the list to another. In such a case, a copy of the parameter is produced: the source parameter remains in place. Parameters can be also be copied by using the 'Copy Parameter' item in the Recipe or right mouse-click menu, or by using the shortcut (Ctrl+C). Data set values are not included in the copy.

3. The 'Insert item...' tool button, the 'Insert Parameter' Recipe menu item or the shortcut <Insert> can be used to open a browse window from which a parameter can be selected. The selected parameter is inserted above the currently active parameter.
4. A parameter can be 'copied' from (for example) the Graphical Wiring Editor and subsequently 'pasted' into the watch list using the 'Paste Parameter 'item in the Recipe menu, or the right mouse-click context menu (shortcut = Ctrl+V).

Creating a Data Set

All the parameters required for the recipe should be added to the watch list, described above.

Once this has been done, if the empty data set is selected (by clicking on the column header), the 'Snapshot' tool button (Ctrl+A) can be used to fill the data set with the current values. Alternatively, the 'Snapshot Values' item in the Recipe or context (right-click) menu or the shortcut + can be used to fill the data set.

Individual data values can now be edited by typing directly into the grid cells. Data values can be left blank or cleared, in which case, when the recipe is downloaded, no value will be written for those values. Data values can be cleared by deleting all the characters in the field, then, either moving to a new cell, or typing <Enter>.

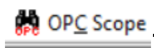
The set is called 'Set 1' by default. The name can be edited by using the 'Rename Data Set...' item in the Recipe or right mouse-click menu, or by using the shortcut (Ctrl+R).

New data sets can be added and edited in the same way, by using the 'Create a new empty....' tool button (Ctrl+W), or by selecting the 'New Data Set' item in the Recipe or right mouse-click menu, or by using the shortcut +.

Once all the data sets required for the Recipe have been created, and saved, they can be downloaded to the device, one at a time, using the download tool (Ctrl+D), or equivalent Recipe/context menu item.

OPC Scope

OPC Scope is a standalone OPC client that can be used to attach to the iTools OPC Server. It offers real-time trend charts and data logging to disk in a .csv (comma separated variable) format which can easily be opened in a spreadsheet such as Excel.

With iTools open, OPC Scope can be started using the icon 

.

But it can also be started on its own using the Windows Start/Programs/Eurotherm iTools/OPC Scope.

Select Server/Connect or click the icon and the OPC Server will start up (if it is not running) and will display the active ports on the computer. Opening the COM port will show the attached instruments as shown below.

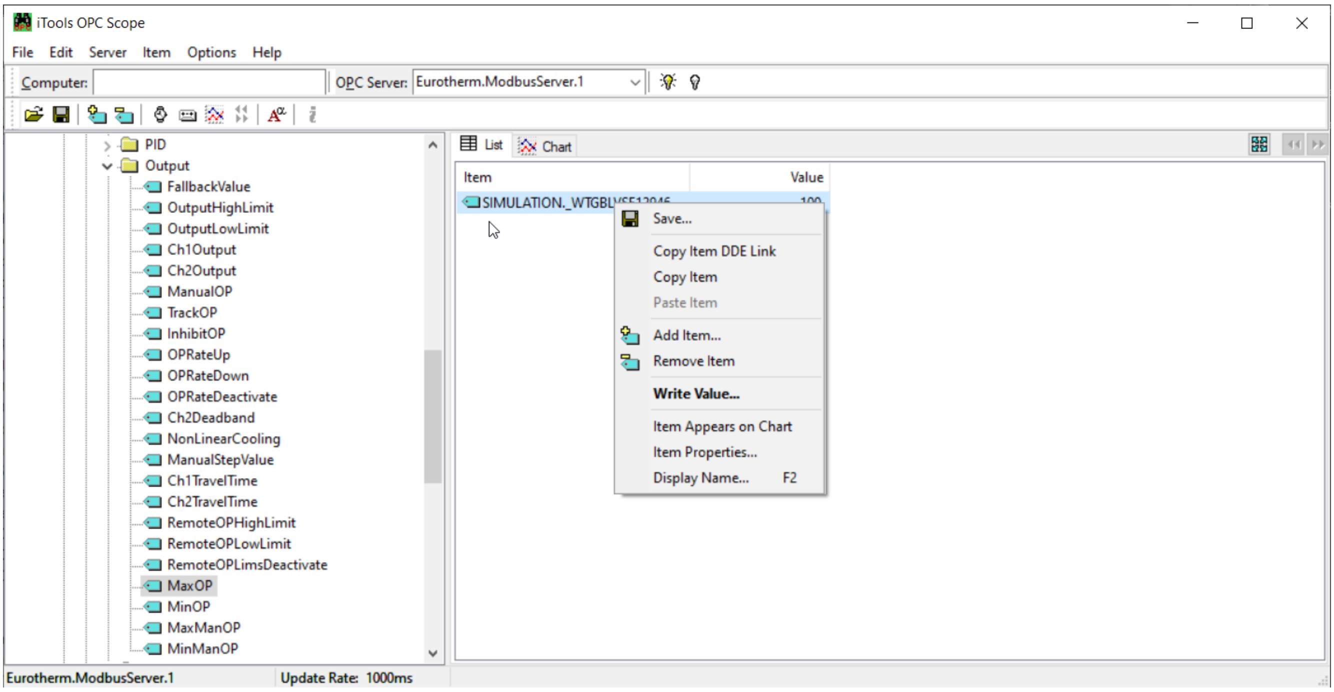


Figure 20 COM Port - Attached Instruments

The 'ID001-Mini8' folder will contain all the same folders for the instrument that would have been seen in iTools itself.

Expand the folder and double-click on the blue item tag to add to the List Window. The List Window shows all the selected parameters and their current value.

Right-click on a parameter to get the context menu.

OPC Scope List Window Context Menu

Command	Description
Save	Saves the OPC Scope configuration as <filename>.uix. See "OPC Server" on page 59.
Copy Item DDE link	Copies the DDE path to the clipboard. Use 'Paste Special' in an Excel cell and select 'Paste Link' and the current parameter value will be displayed in the cell.
Copy/Paste Item	Copy & Paste
Add Item	Add a new variable by name (easier to browse the navigation tree)
Remove Item	Remove the selected item.
Write Value	Write a new value (not if the item is Read Only).
Item appears on Chart	Up to eight items can be trended on the Chart Window
Item Properties	Gives the item properties as seen by OPC

The OPC List can contain parameters from any instrument attached to the Modbus network.

If you have iTools Open (not iTools Standard) then OPC Scope can run on a remote networked computer. Enter the name of the server computer (attached to the instruments) the 'Computer' window and browse for the 'Eurotherm.ModbusServer1'.

OPC Scope Chart Window

Click the Chart tab  at the bottom of the display window and select Chart Control Panel.

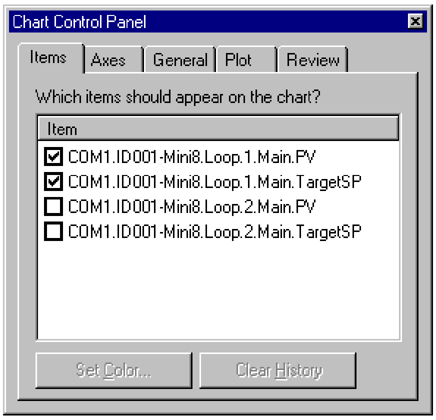


Figure 21 Chart Control Panel

1. **Items**. Includes all the items in the list window. Those items ticked (up to eight) will appear on the chart.
2. **Axes**. Allows time intervals from 1 minute to 1 month. Vertical axes can be 'auto' scaled or a fixed range may be entered.
3. **General**. Allows selection of colors, grid, legends and a data box.
4. **Plot**. Allows selection of line thickness and printing.
5. **Review**. Allows review of early history charts.

These are also available on the toolbar.

iTools Trend Graph showing Loop1 SP and PV

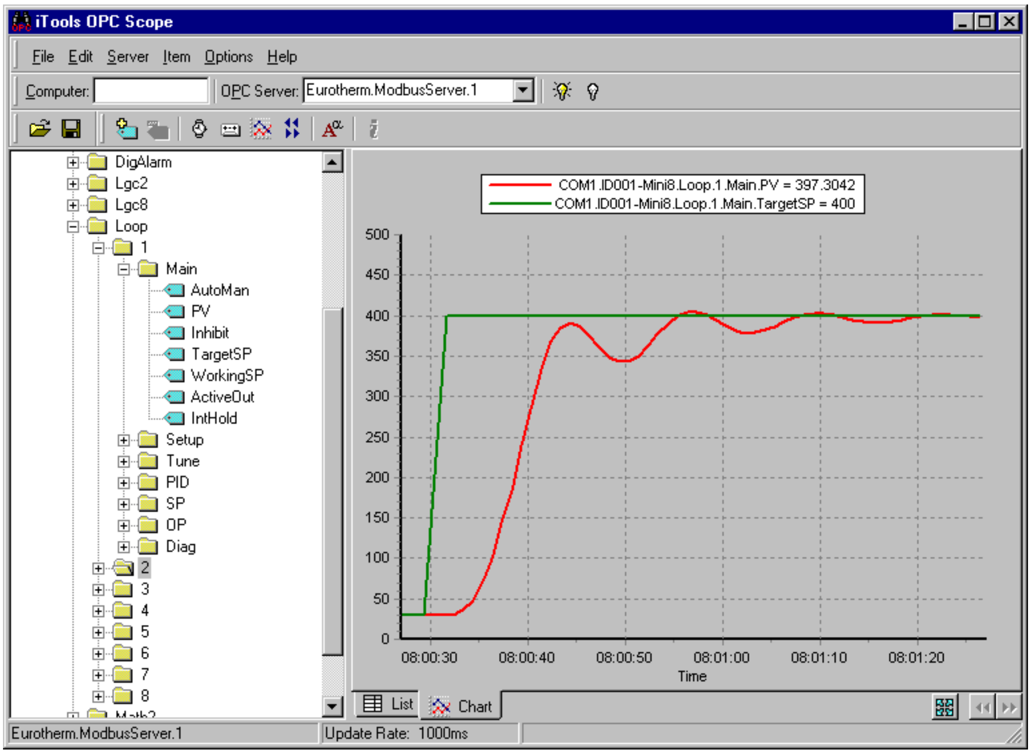


Figure 22 iTools Trend Graph

The  icon allows the chart to occupy all the window space.

OPC Server

iTools and OPC Scope all use the Eurotherm OPC Server to provide the connection between the instruments and the computer displays. When you ‘scan’ for instruments on iTools it is in fact the OPC Server that is actually doing the work in background (the window is not usually displayed).

OPC Scope can run on its own but for it to find the instruments on the network it is necessary to tell the server where they are.

1. Start OPC Server (Windows Start/Programs/Eurotherm iTools/OPC Server).
2. On the menu go to ‘Network’ and select ‘Start One-Shot Scan’.
3. Stop the scan when all the instruments have been found.

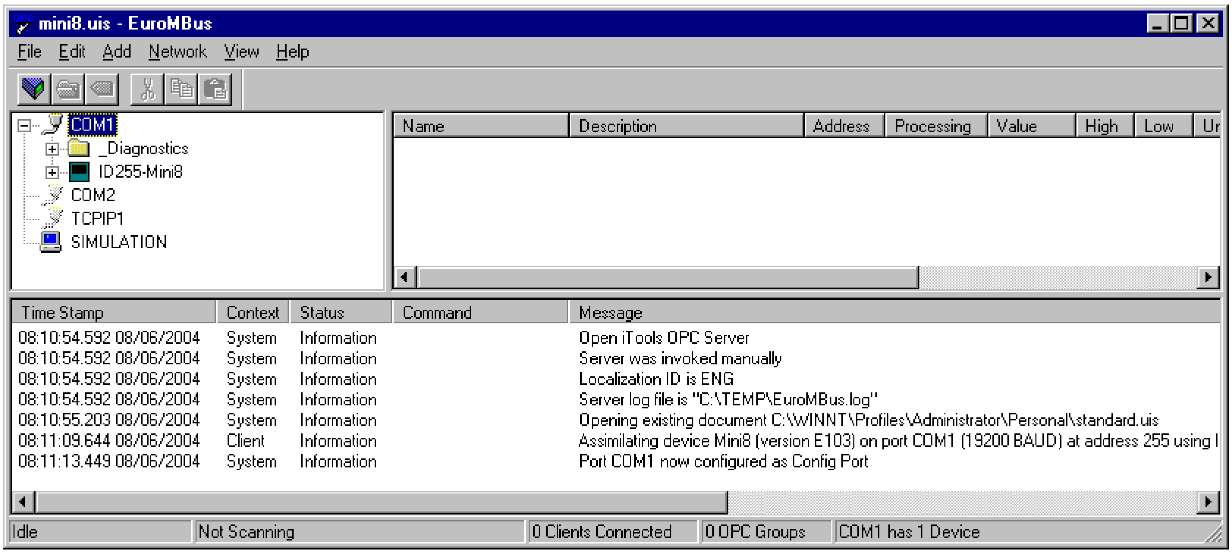


Figure 23 Running OPC Server

4. On the menu go to ‘File’ and select ‘Save As’ and save the file with a suitable name.
5. Once saved you will be asked ‘Would you like to make this file the default start server address file?’ – select ‘Yes’.
6. Close the server.

Now if you double-click on an OPC Scope file (for example, Mini8 Project.uix) then this file will open OPC Scope and in turn, in the background, OPC Scope will open the OPC Server with this instrument file loaded. OPC Scope will then be active with live data from the instrument(s).

Configuration Using iTools

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

It is the responsibility of the person commissioning the controller to ensure the configuration is correct.

The controller must not be configured while it is connected to a live process as entering Configuration Mode pauses all outputs. The controller remains in Standby until Configuration Mode is exited.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Configuration

The Mini8 loop controller is supplied unconfigured and has to be configured for use in an application. This is performed using iTools.

The iTools handbook, part no. HA028838 provides further step by step instructions on the general operation of iTools. This and the iTools software may be downloaded from www.eurotherm.com.

On-Line/Off-line Configuration

iTools may be used ‘off-line’ without a real Mini8 loop controller connected. This SIMULATION Mini8 loop controller can be created and configured in iTools. The configuration can be saved to disk as a clone file. This file can later be loaded into a real Mini8 loop controller live application. See "Cloning" on page 61.

If iTools is connected to a real Mini8 loop controller then all the parameter changes made will be written to the device immediately. Once the Mini8 loop controller is configured and working as required, its final configuration can be saved to disk as a ‘clone’ file of the format <name>.uic.

Connecting a PC to the Mini8 Loop Controller

Configuration Cable and Clip

The controller may be connected to the PC running iTools using the Eurotherm cable SubMin8/Cable/Config from the RJ11 port connecting to a serial port on the PC.

Alternatively a Configuration Clip is available from Eurotherm that can be fitted into the rear of the controller.

Note: It is only possible to use this Configuration Clip when the controller is NOT connected to a DIN rail.

The benefit of using this arrangement is that it is not necessary to power the controller, since the clip provides the power to the internal memory of the controller.

Scanning

Open iTools and, with the controller connected, press 

on the iTools menu bar. iTools will search the communications ports and TCP/IP connections for recognizable instruments. Controllers connected using the RJ11 configuration port or with the configuration clip (CPI), will be found at address 255 regardless of the address configured in the controller. These connections only work from iTools to a single controller.

The iTools handbook, part no. HA028838, provides further step by step instructions on the general operation of iTools. This and the iTools software may be downloaded from www.eurotherm.com.

In the following pages it is assumed that the user is familiar with iTools and has a general understanding of Windows.

Cloning

Saving a Clone File

On the iTools menu 'File – Save to File' allows the clone file of the attached Mini8 loop controller to be saved to disc as <user name>.UIC file. This can be loaded into another Mini8 loop controller.

Note: After synchronization, iTools uses a 'quick' save and will only resave parameters that have been changed through iTools itself. If there is any chance that parameters have been changed through the other port then it is necessary to resave all the parameters. On the menu bar under Options – Cloning, select **Reload**. The recommended option is to keep **Ask** selected.

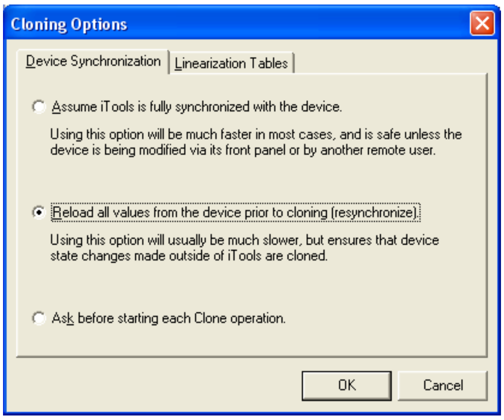


Figure 24 Cloning Options

Loading a Clone File

On the iTools menu 'File – Load values File' allows a clone file of the form <user name>.UIC to be loaded into an attached Mini8 loop controller. Whilst loading, the report window will indicate what is happening. It makes a number of attempts to load all the values and may report some problems. This is generally not an issue. If, for some reason, it cannot be loaded, iTools will report specifically that the load '**Failed**'.

Cloning of Communications Port Parameters

A Mini8 loop controller can be communicated to via the Config Comms (CC) or Field Comms (FC) ports.

All Mini8 loop controller clone operations require the device to be in Configuration Access Mode, during which the Mini8 loop controller will not control (see note 1 below).

Saving of a clone file from a Mini8 device can be performed on any port (CC or FC).

Loading of a clone file into a Mini8 device can be performed on any port (CC or FC) with the following limitations:

- FC port: No comms parameter (CC or FC) will be updated
  - The user will be notified via the Clone report that manual update is required
- CC port:
  - Connected using Modbus Server (Slave) Address 255 – full clone of all comms parameters
  - Connected using any other Modbus Server (Slave) Address (e.g. 1-254) – No comms parameters (CC or FC) will be updated
    - The user will be notified via the Clone report that manual update is required

Notes:

1. Only one comms session can have Configuration Access Mode. Other connections (serial or Ethernet) will be unable to put device into Config mode.
2. If target device's Module/IO Actual does not correspond with the Clone file's Module/IO Expected, cloning will abort, informing the user they need to resolve module/IO mismatch.

Configuring the Mini8 Loop Controller

Note: iTools does not need to be connected to a Mini8 loop controller to configure an application, configuration can be performed offline.

Configuration involves selection of the required elements called ‘function blocks’ and setting their parameters to the correct values. The next stage is to wire (or soft wire) all the function blocks together to create the required strategy of control for the application.

Function Blocks

The controller application consists of function blocks. A function block is a software algorithm. It may be represented as a rectangle as shown below, with Input parameters on the left, and Output parameters on the right.

Input parameters may be initialized by setting the parameter value, or by soft-wiring from another selected source parameter (see "Graphical Wiring Editor" on page 70).

A representation of a function block is shown below.

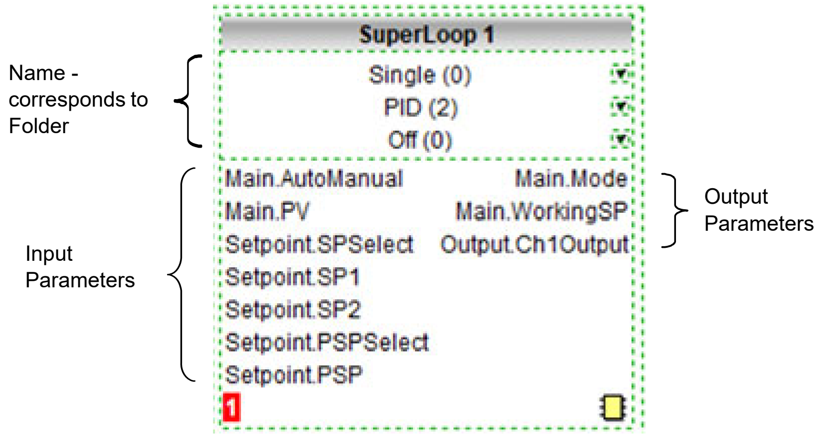


Figure 25 Representation of a Function Block

Parameters

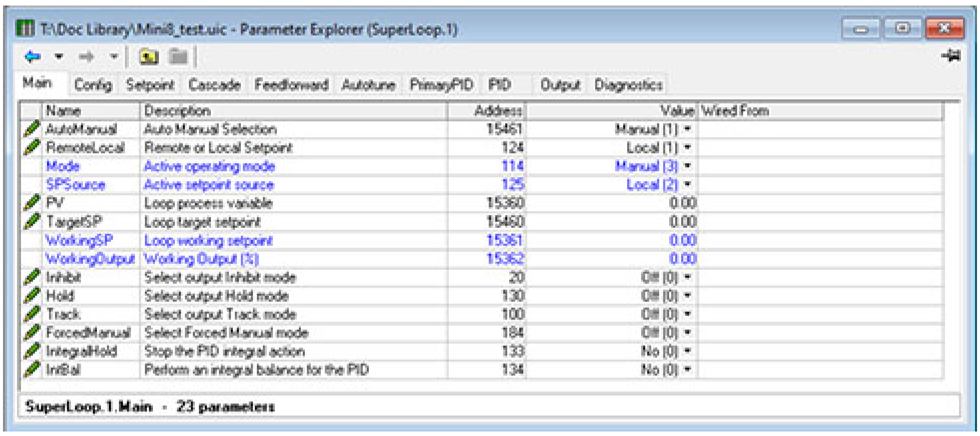


Figure 26 Example of a Function Block

Function Block parameters are shown in the Parameter List View. The name of the Function Block is shown in the Window title. For Function Blocks with a large number of Parameters, these are divided into sub-lists, each represented by a named tab.

Wiring

Wiring (sometimes known as User Wiring, Soft Wiring, or Graphical Wiring) refers to the connections that are made in software between function block parameters. This wiring is created during configuration using the iTools Graphical Wiring Editor.

In general, every function block has at least one input and one output. Parameters are used to specify where a function block reads its incoming data (the 'Input Source'). The source of a wire is the selected output parameter of Function Block. The destination of a wire is the selected Input parameter of another Function Block.

All parameters shown in the function block diagrams are also shown in the parameter tables or lists, in the relevant chapters. See "Complete list of Function Blocks" on page 85.

Figure 27 shows an example of how a LogicIn input is wired to the SuperLoop input and the SuperLoop channel 1 output is wired to the time proportioning logic output.

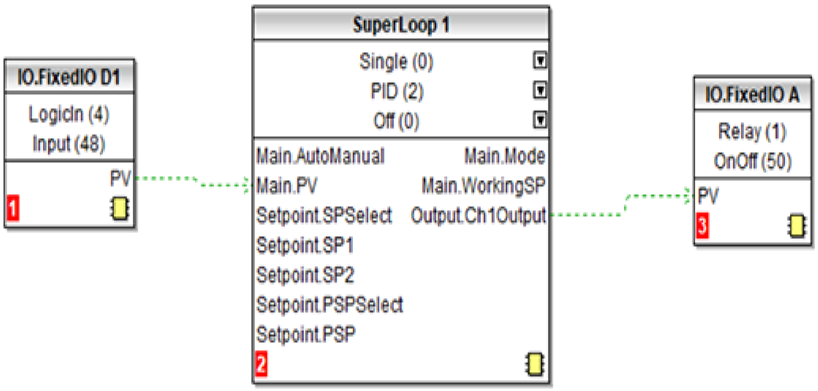


Figure 27 Function Block Wiring

Simple Worked Example

Using function blocks and wiring the following sections will show a blank Mini8 loop controller being configured to have one PID loop.

The I/O

With the Mini8 loop controller successfully connected to iTools, configuration can begin.

☺ Tip:

In parameter lists:

- Parameters in BLUE are read-only.
- Parameters in BLACK are read/write.

☺ Tip:

Every parameter in the parameter lists has a detailed description in the help file – just click on a parameter and press Shift-F1 on the keyboard or right-click and select parameter help.

The I/O will already have been installed in the Mini8 loop controller and can be checked in iTools.

Example 1: Thermocouple Input Configuration

In the IO list ModIDs select the type of module. Thermocouple modules may be four-input modules or eight-input modules.

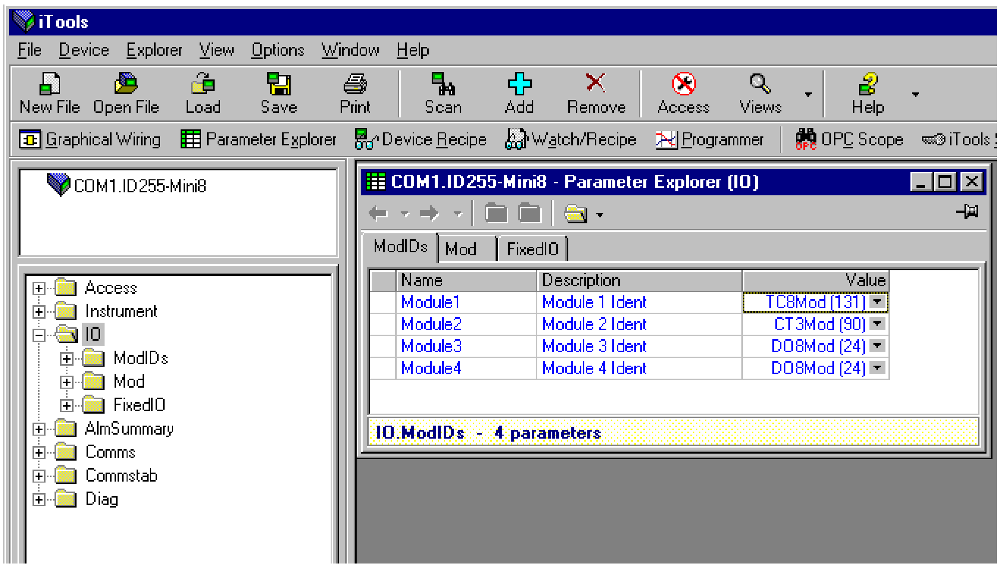


Figure 28 Mini8 loop controller I/O Modules

This unit has an eight thermocouple input board in slot 1, a CT3 input card in slot 2, and two DO8 output cards in slot 3 and slot 4. Clicking on the 'Mod' tab will enable the first channel of the thermocouple card to be configured. Firstly the Mini8 loop controller has to be put into configuration mode. Go to Device/Access/Configuration or click on the Access button:

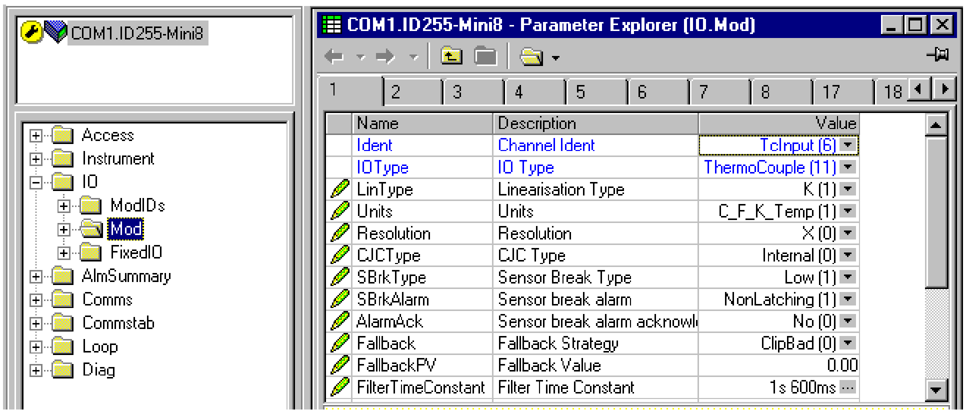


Figure 29 Thermocouple Input

Select the I/O type, linearization, units, resolution, and so on, required. Parameter details are in "Thermocouple Input" on page 95.

The other thermocouple channels can be found by using the 2, 3, 4…7, 8 tabs on the top of the parameter window.

Slot 2 in the Mini8 loop controller has a CT3 input card and this is configured elsewhere so the Tabs 9 to 16 are not shown.

Slot 3 has a DO8 output card and the first channel of this will be on tab 17 (to 24).

Slot 4 has a DO8 output card and the first channel of this will be on tab 25 (to 32).

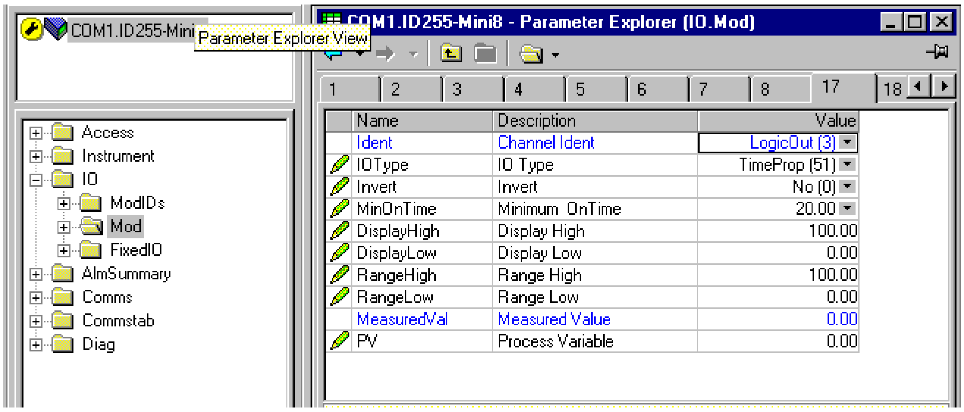


Figure 30 Digital Output Channel

Set this channel up as required, IOType, MinOnTime, and so on as required. The parameters are detailed in "Logic Output" on page 93.

The remaining channels on this slot will be found under the tabs 18 to 24.

Slot 4 also contains a DO8 output card with outputs under tabs 25 to 32.

The Fixed I/O is always there and there is nothing that has to be configured.

The Current Monitor is covered in "Current Monitor" on page 107.

Example 2: RTD Input Configuration

In the IO list ModIDs select the type of module. RTD modules are four-input modules [RT4Mod (173)].

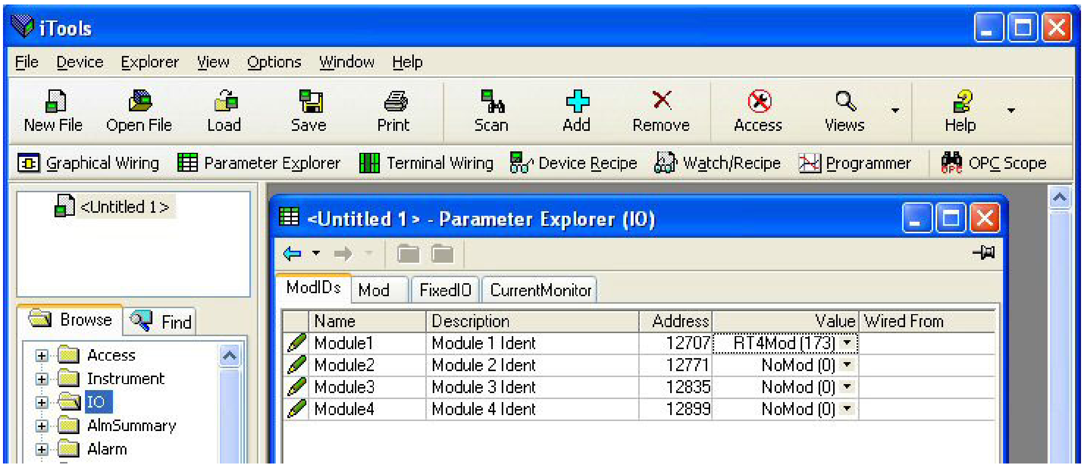


Figure 31 Mini8 Loop Controller IO Module1 Defined as RTD

RTDs can be defined as 2-wire [RTD2 (32)], 3-wire [RTD3 (33)] or 4-wire [RTD4 (34)] in the module definition list.

NOTICE
Configure the ‘IO Type’ and the ‘Resistance Range’ to match the RTD in use so that the correct lead compensation calculation is selected.

.

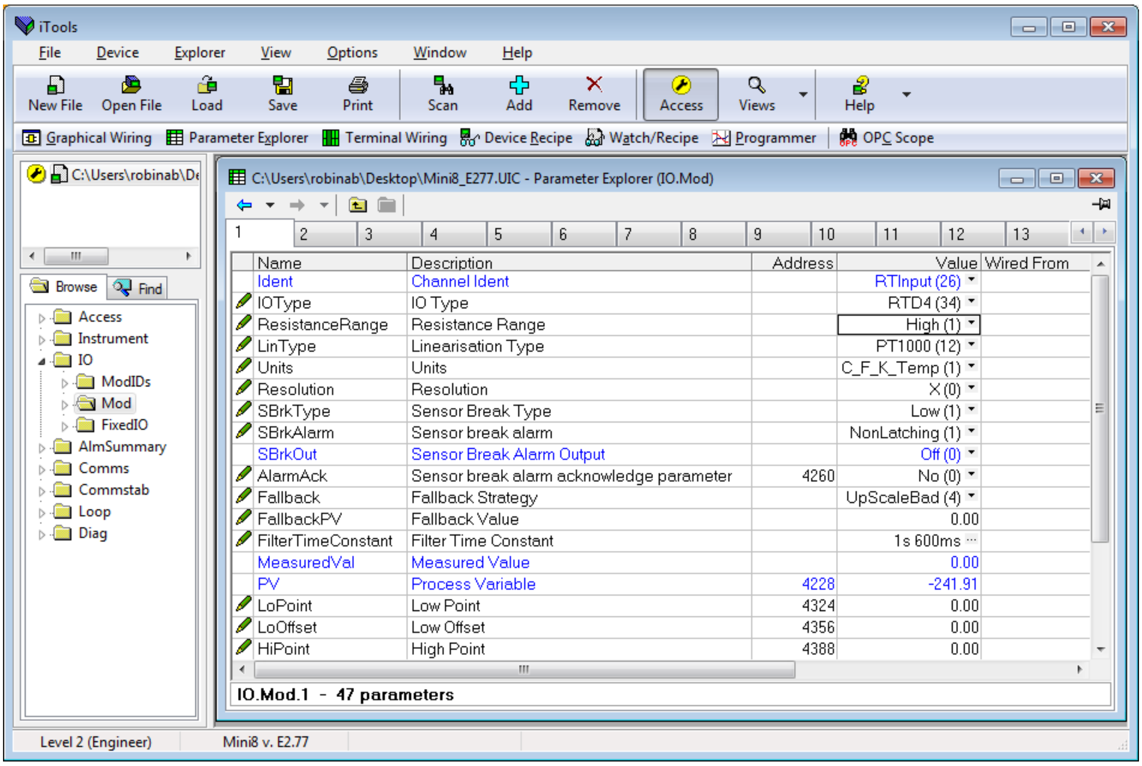


Figure 32 Module 1 defined as RTD4

Wiring

The IO that has been configured now needs to be wired to PID loops and other function blocks.

Select 

(GWE) to create and edit the Function Block wiring.

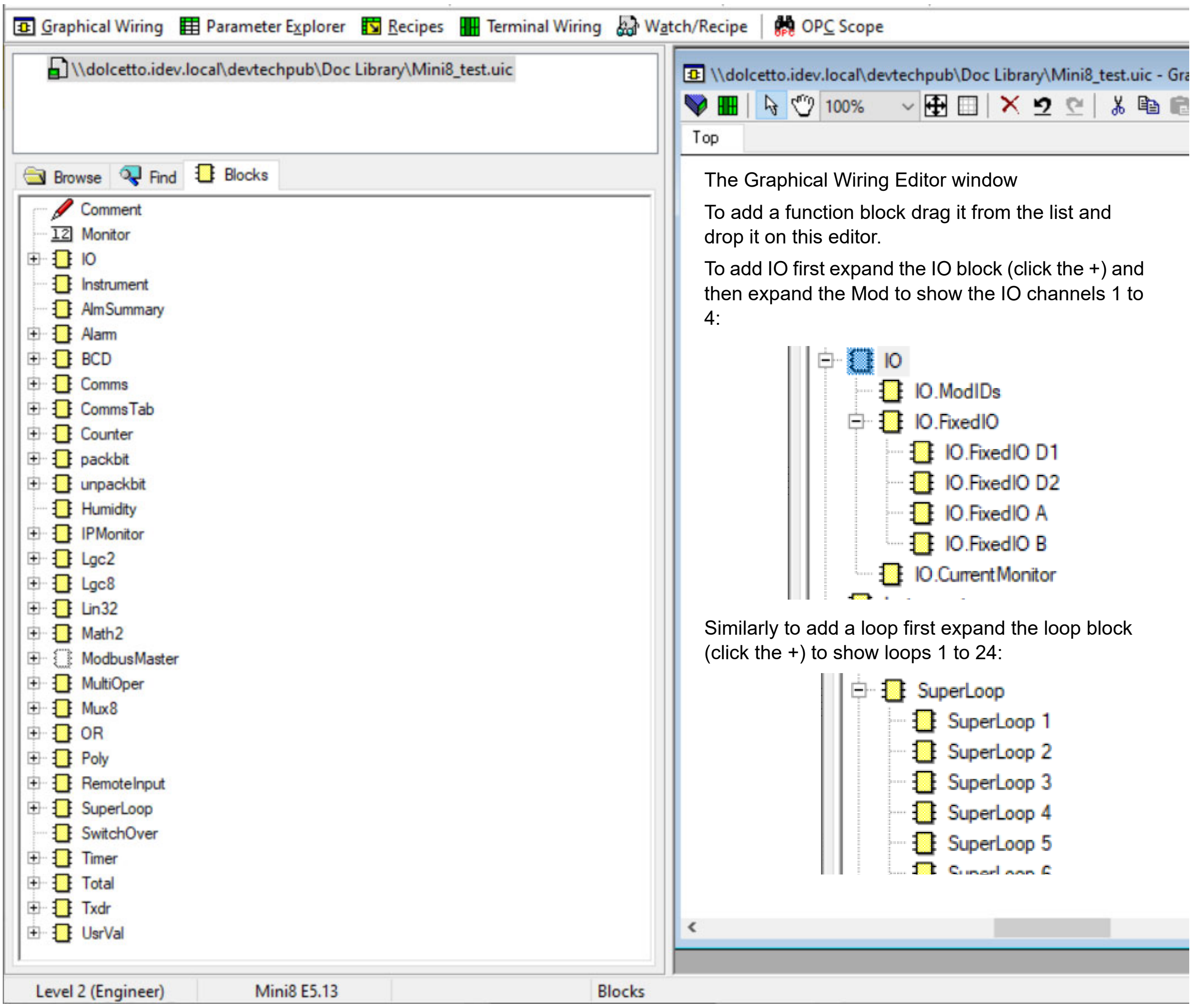


Figure 33 List of Function Blocks & Graphical Wiring Window

The left window now contains a list of the function blocks available.

Use drag and drop to select the first thermocouple from IOMod 1, the Cool output from IOMod 17 and the Heat output from IOMod 25 and drop them on the wiring window.

Finally take the first PID block from SuperLoop/Loop 1 and drop it on the wiring window.

Note: As each block is used, it grays out on the list.

There should now be four blocks on the window. Those blocks are shown with dotted lines, as they have not been loaded into the Mini8 loop controller.

First make the following wire connections.

1. Click on IO.Mod1.PV and move the pointer to SuperLoop 1.MainPV and click again. A dotted wire will have connected the two together.

2. Similarly join SuperLoop1.OP.Ch1Out to IOMod 25.PV (heat output).
3. Enable the Cool output by clicking the select arrow to the top of the SuperLoop block:

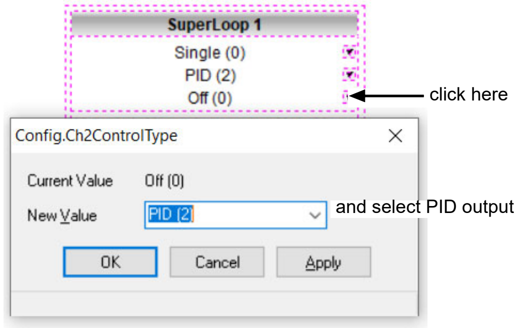


Figure 34 Enabling Cool Output

4. SuperLoop1.OP.Ch2Out to IOMod 23.PV (cool output)

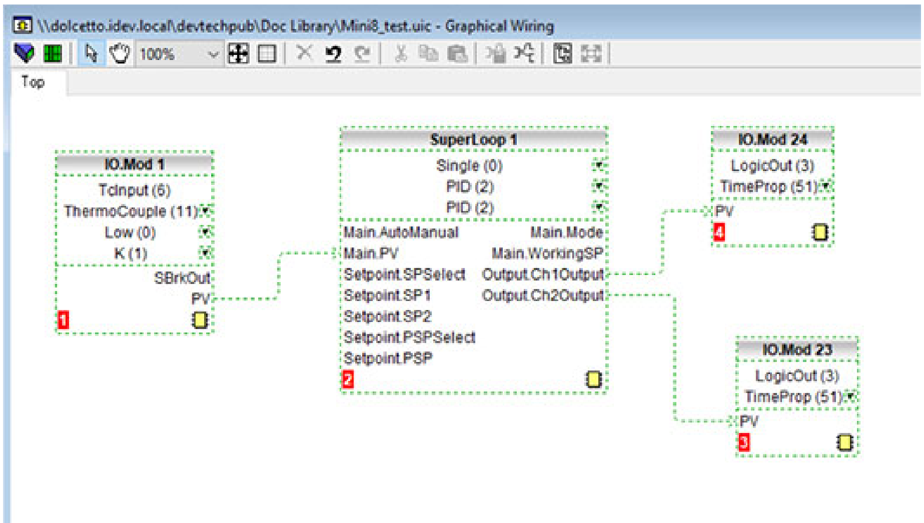


Figure 35 Wired Blocks before download

5. Right-click on the SuperLoop 1 function Block and select ‘Function Block View’. This opens the Loop parameter list on top of the wiring editor.

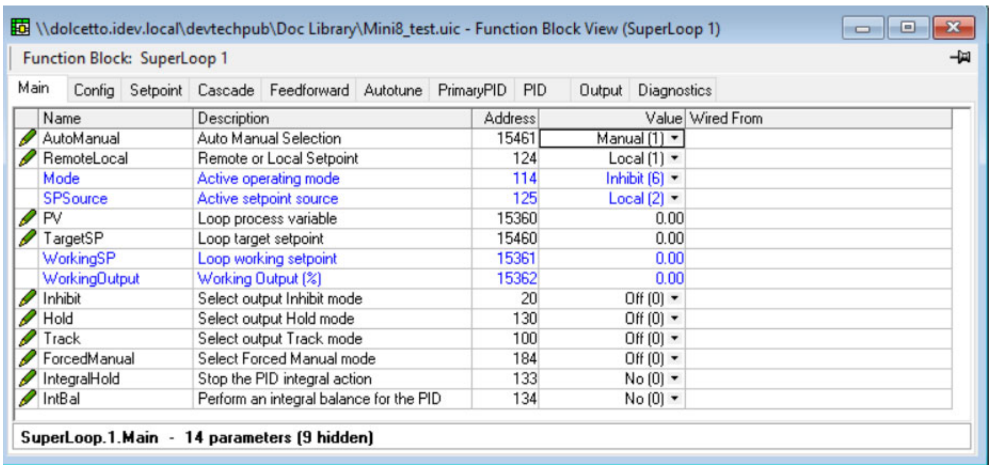


Figure 36 PID Function Block

This enables the PID function block to be set up to suit the required application. See "Control Loop Setup" on page 198 for details.

6. If connected to an instrument, click on the instrument button to download the wiring (otherwise, Save the wiring):

7. Once downloaded the dotted lines around the function blocks and the wires will become solid to show that the application is now in the Mini8 loop controller. The upper status line also shows that three wires have been used out of those available. Max is 250 but quantity depends on number of wires ordered (30, 60, 120 or 250).
8. Put the Mini8 loop controller back into Operating mode by clicking the Access button:

9. The Mini8 loop controller will now control the Loop1 as configured.

Graphical Wiring Editor

Select 

(GWE) to view and edit Function Block wiring. You can also add comments and monitor parameter values.

1. Drag and drop required function blocks into the graphical wiring from the list in the left pane.
2. Click on parameter to be wired from and drag the wire to the parameter to be wired to (do not hold mouse button down).
3. Right-click to edit parameter values.
4. Select parameter lists and switch between parameter and wiring editors.
5. Download to instrument when wiring completed.
6. Add comments and notes.
7. Dotted lines for wires and around Blocks show that the wiring requires saving or downloading to an instrument.

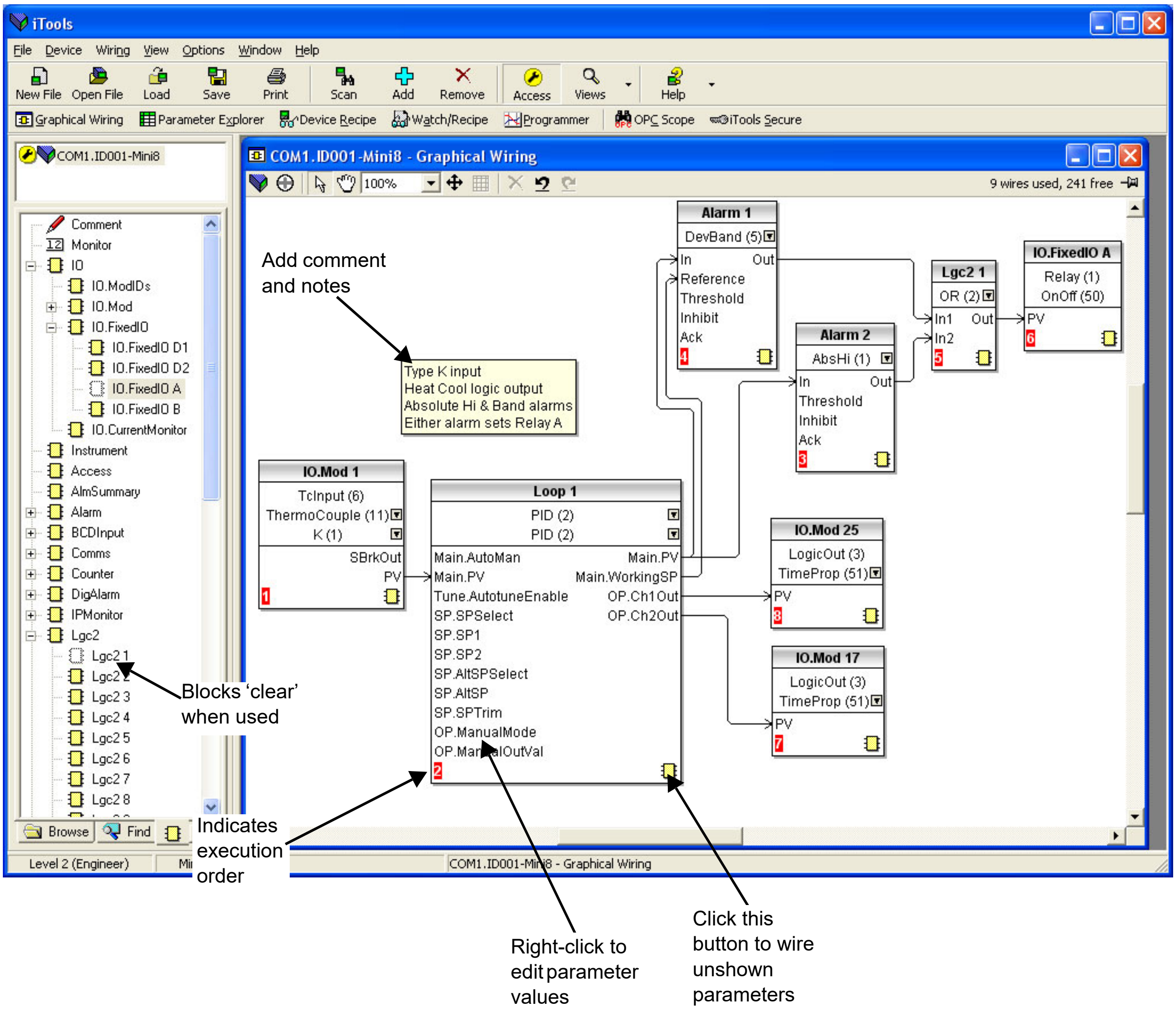


Figure 37 Graphical Wiring Editor

Graphical Wiring Toolbar

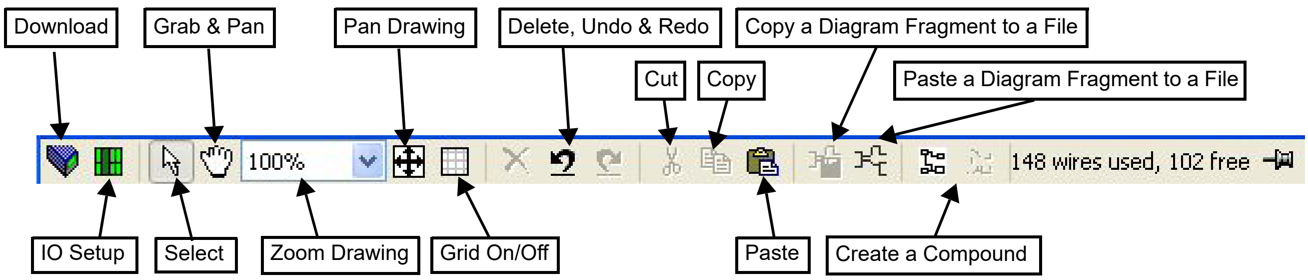


Figure 38 Graphical Wiring Toolbar

Function Block

A Function Block is an algorithm that may be wired to and from other function blocks to make a control strategy. Examples are: a control loop and a mathematical calculation.

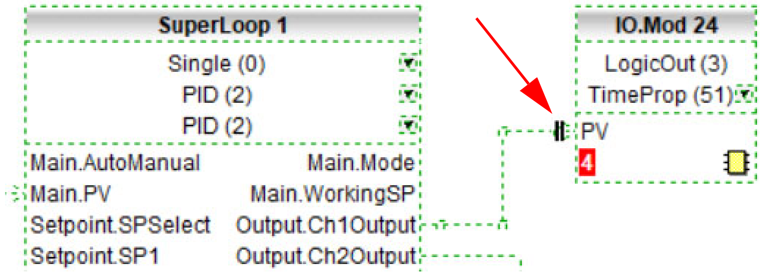
Each function block has inputs and outputs. Any parameter may be wired from, but only parameters that are alterable may be wired to.

A function block includes any parameters that are needed to configure or operate the algorithm.

Wire

A wire transfers a value from one parameter to another. They are executed by the instrument once per control cycle.

Wires are made from an output of a function block to an input of a function block. It is possible to create a wiring loop, in this case there will be a single execution cycle delay at some point in the loop. This point is shown on the diagram by a || symbol and it is possible to choose where that delay will occur.

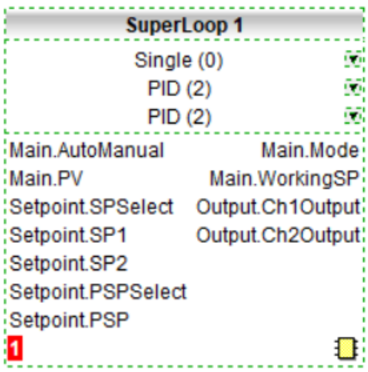


Block Execution Order

The order in which the blocks are executed by the instrument depends on the way in which they are wired.

The order is automatically worked out so that the blocks execute on the most recent data.

Using Function Blocks



If a function block is not faded in the tree then it can be dragged onto the diagram. The block can be dragged around the diagram using the mouse.

A labeled loop block is shown here. The label at the top is the name of the block.

When the block type information is alterable, click on the box with the arrow in it on the right to edit that value.

Figure 39 Function Block

The inputs and outputs that are considered to be of most use are always shown. In most cases all of these will need to be wired up for the block to perform a useful task. There are exceptions to this and the loop is one of those exceptions.

If you wish to wire from a parameter, which is not shown as a recommended output click on the icon in the bottom right, and a full list of parameters in the block will be shown, click on one of these to start a wire.

To start a wire from a recommended output just click on it.

Click the icon in the bottom right hand corner to wire other function block parameters not shown on the list on the right-hand side.

Function Block Context Menu

Right-clicking displays the context menu with the following entries.

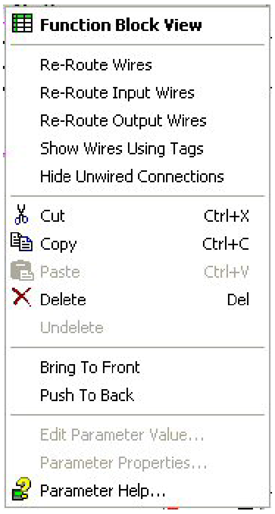


Figure 40 Function Block Context Menu

Function Block View
Displays an iTools parameter list which shows all the parameters in the function block. If the block has sub-lists these are shown in tabs.

Re-Route Wires Discard current wire route and do an auto-route of all wires connected to this block.

Re-Route Input Wires
Only do a re-route on the input wires

Re-Route Output Wires
Only do a re-route on the output wires

Show wires using tags
Shows the beginning and end of each wire with a descriptor showing the source or destination. Used to simplify a diagram with many wires.

Hide Unwired Connections
Hides function block pins that are not used.

Cut Cut the selected function block.

Copy Right-click over an input or output and copy will be enabled, this menu item will copy the iTools "url" of the parameter which can then be pasted into a watch window or OPC Scope.

Paste Add a new copy of the function block.

Delete If the block is downloaded, mark it for delete, otherwise delete it immediately.

Undelete This menu entry is enabled if the block is marked for delete and unmarks it and any wires connected to it for delete.

Bring To Front Bring the block to the front of the diagram. Moving a block will also bring it to the front.

Push To Back Push the block to the back of the diagram. Useful if there is something underneath it.

Edit Parameter Value
This menu entry is enabled when the mouse is over an input or output parameter. When selected it creates a parameter edit dialog so the value of that parameter can be changed.

Parameter Properties
Selecting this entry brings up the parameter properties window. The parameter properties window is updated as the mouse is moved over the parameters shown on the function block.

Parameter Help Selecting this entry brings up the help window. The help window is updated as the mouse is moved over the parameters shown on the function block. When the mouse is not over a parameter name the help for the block is shown.

Tooltips

Hovering over different parts of the block will bring up tool-tips describing the part of the block beneath the mouse.

If you hover over the parameter values in the block type information a tool-tip showing the parameter description, its OPC name, and, if downloaded, its value will be shown.

A similar tool-tip will be shown when hovering over inputs and outputs.

Function Block State

The blocks are enabled by dragging the block onto the diagram, wiring it up, and downloading it to the instrument.

When the block is initially dropped onto the diagram it is drawn with dashed lines.

When in this state the parameter list for the block is enabled but the block itself is not executed by the instrument.

Once the download button is pressed the block is added to the instrument function block execution list and it is drawn with solid lines.

If a block is deleted from the Graphical Wiring Diagram, when connected to a real instrument, it is shown on the diagram in a ghosted form until the download button is pressed.

This is because it and any wires to/from it are still being executed in the instrument. On download it will be removed from the instrument execution list and the diagram. A ghosted block can be undeleted using the context menu.

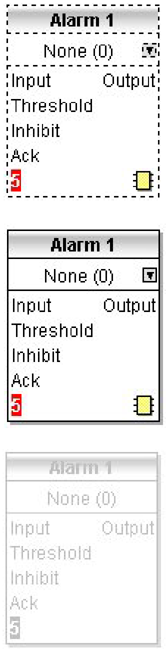


Figure 41 Function Block States

When a dashed block is deleted it is removed immediately.

Using Wires

Making a Wire between Two Blocks

To make a wire between two blocks:

1. Drag two blocks onto the diagram from the function block tree.
2. Start a wire by either clicking on a recommended output or clicking on the icon at the bottom right corner of the block to bring up the connection dialog. The connection dialog shows all the connectible parameters for the block, if the block has sub-lists the parameters are shown in a tree. If you wish to wire a parameter which is not currently available click the red button at the bottom of the connection dialog. Recommended connections are shown with a green plug, other parameters which are available are yellow and if you click the red button the unavailable parameters are shown red. To dismiss the connection dialog either press the escape key on the keyboard or click the cross at the bottom left of the dialog.
3. Once the wire has started the cursor will change and a dotted wire will be drawn from the output to the current mouse position.
4. To make the wire either click on a recommended input to make a wire to that parameter or click anywhere except on a recommended input to bring up the connection dialog. Choose from the connection dialog as described above.

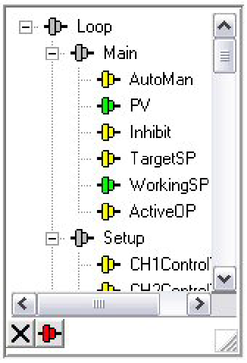


Figure 42
Wires Between Blocks

The wire will now be auto-routed between the blocks. New wires are shown dotted until they are downloaded.

Wire Context Menu

The wire block context menu has the following entries on it.

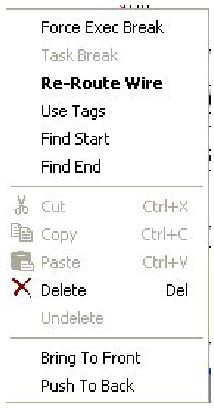


Figure 43
Wire Context Menu

Force Exec Break If wires form a loop, a break point has to be found where the value which is written to the block input comes from a block which was last executed during the previous instrument execute cycle thus introducing a delay. This option tells the instrument that if it needs to make a break it should be on this wire.

Re-Route Wire Discard wire route and generate an automatic route from scratch.

Use Tags If a wire is between blocks which are a long way apart, then, rather than drawing the wire, the name of the wired to/from parameter can be shown in a tag next to the block. Draw the wire first then use this menu to toggle this wire between drawing the whole wire and drawing it as tags.

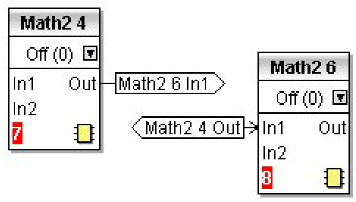


Figure 48
Use Tags

Find Start Find the source of the selected wire.

Find End Find the destination of the selected wire.

Delete If the wire is downloaded, mark it for deletion, otherwise delete it immediately.

Undelete This menu entry is enabled if the wire is marked for deletion and unmarks it for deletion.

Bring To Front Bring the wire to the front of the diagram. Moving a wire will also bring it to the front.

Push To Back Push the wire to the back of the diagram.

Wire Colors

Wires can be the following colors:

Black	Normal functioning wire.
Red	The wire is connected to an input which is not alterable when the instrument is in operator mode and so values which travel along that wire will be rejected by the receiving block.
Blue	The mouse is hovering over the wire, or the block to which it is connected it selected. Useful for tracing densely packed wires.
Purple	The mouse is hovering over a 'red' wire.

Routing Wires

When a wire is placed it is auto-routed. The auto routing algorithm searches for a clear path between the two blocks. A wire can be auto-routed again using the context menus or by double-clicking the wire.

If you click on a wire segment you can drag it to manually route it. Once you have done this it is marked as a manually routed wire and will retain its current shape. If you move the block to which it is connected the end of the wire will be moved but as much of the path as possible of the wire will be preserved.

If you select a wire by clicking on it, it will be drawn with small boxes on its corners.

Tool-tips

Hover the mouse over a wire and a tool-tip showing the names of the parameters which are wired and, if downloaded, their current values will also be shown.

Using Comments

Drag a comment onto the diagram and the comment edit dialog will appear.

Figure 49 Comment Edit Dialog

Type in a comment. Use new lines to control the width of the comment, it is shown on the diagram as typed into the dialog. Click OK and the comment text will appear on the diagram. There are no restrictions on the size of a comment. Comments are saved to the instrument along with the diagram layout information.

Comments can be linked to function blocks and wires. Hover the mouse over the bottom right of the comment and a chain icon will appear, click on that icon and then on a block or a wire. A dotted wire will be drawn to the top of the block or the selected wire segment.

Comment Context Menu

The comment context menu has the following entries on it.

Edit	Open the comment edit dialog to edit this comment.
Unlink	If the comment is linked to a block or wire this will unlink it.
Cut	Remove the comment.
Copy	Make a copy of the comment.
Paste	Paste a new copy of the comment.
Delete	If the comment is downloaded mark it for deletion, otherwise delete it immediately.
Undelete	This menu entry is enabled if the comment is marked for deletion and unmarks it for deletion.

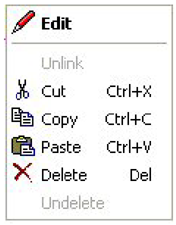


Figure 50 Comment Context Menu

Using Monitors

Drag a monitor onto the diagram and connect it to a block input or output or a wire as described in ‘Using Comments’.

The current value (updated at the iTools parameter list update rate) will be shown in the monitor. By default the name of the parameter is shown, double-click or use the context menu to not show the parameter name.

Monitor Context Menu

The monitor context menu has the following entries on it.

Show Names	Show parameter names as well as values.
Unlink	If the monitor is linked to a block or wire this will unlink it.
Cut	Remove the monitor.
Copy	Make a copy of the monitor.
Paste	Paste the copy of the monitor.
Delete	If the monitor is downloaded, mark it for deletion, otherwise delete it immediately.
Undelete	This menu entry is enabled if the monitor is marked for deletion and unmarks it for deletion.
Bring To Front	Bring the monitor to the front of the diagram. Moving a monitor will also bring it to the front.
Push To Back	Push the monitor to the back of the diagram. Useful if there is something underneath it.
Parameter Help	When a parameter is selected, this menu entry provides help on that parameter.

Figure 51 Monitor Context Menu

Downloading

A Graphical Wiring configuration needs to be saved. If connected to a real device then the Wiring definition will be downloaded to the instrument. When the wiring editor is opened the current wiring and diagram layout is read from the instrument. No changes are made to the instrument function block execution or wiring until the download button is pressed.

When a block is dropped on the diagram, instrument parameters are changed to make the parameters for that block available. If you make changes and close the editor without saving them there will be a delay while the editor clears these parameters.

When you download, the wiring is written to the instrument that then calculates the block execution order and starts executing the blocks. The diagram layout including comments and monitors is then written into instrument flash memory along with the current editor settings. When you reopen the editor the diagram will be shown positioned the same as when you last downloaded.

Selections

Wires are shown with small blocks at their corners when selected. All other items have a dotted line drawn round them when they are selected.

Selecting Individual Items

Clicking on an item on the drawing will select it.

Multiple Selection

Control-click an unselected item to add it to the selection, doing the same on a selected item unselects it.

Alternatively, hold the mouse down on the background and swipe it to create a rubber band, anything which isn't a wire inside the rubber band will be selected.

Selecting two function blocks also selects any wires which join them. This means that if you select more than one function block using the rubber band method any wires between them will also be selected.

Pressing Ctrl-A selects all blocks and wires.

Colors

Items on the diagram are colored as follows:

Red Function blocks, comments and monitors which partially obscure or are partially obscured by other items are drawn red. If a large function block like the loop is covering a small one like a math2, the loop will be drawn red to show that it is covering another function block. Wires are drawn red when they are connected to an input which is currently unalterable. Parameters in function blocks are colored red if they are unalterable and the mouse pointer is over them.

Blue Function blocks, comments and monitors which are not colored red are colored blue when the mouse pointer is over them. Wires are colored blue when a block to which the wire is connected is selected or the mouse pointer is over it. Parameters in function blocks are colored blue if they are alterable and the mouse pointer is over them.

Purple A wire which is connected to an input which is currently unalterable and a block to which the wire is connected is selected or the mouse pointer is over it is colored purple (red + blue).

Diagram Context Menu

Highlight an area of the graphical wiring by left-clicking the mouse button and dragging around the required area. Right-click in the area to show the Diagram Context Menu. The diagram context menu has the following entries:

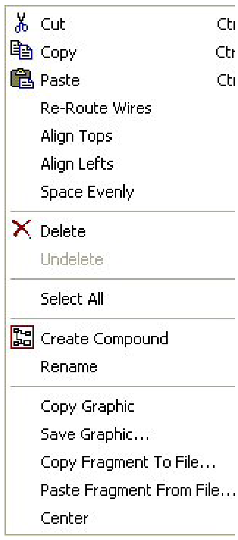


Figure 52 Diagram Context Menu

Cut To delete the selected area.

Copy To make a copy of the selected area.

Paste To paste the selected area.

Re-Route Wires Discard current wire route and do an auto-route of all selected wires. If no wires are selected this is done to all wires on the diagram.

Align Tops Line up the tops of all the selected items except wires.

Align Lefts Line up the left hand side of all the selected items except wires.

Space Evenly This will space the selected items such that their top left corners are evenly spaced. Select the first item, then select the rest by Control-clicking them in the order you wish them to be spaced, then choose this menu entry.

Delete Marks all selected items for deletion (will be deleted on next download).

Undelete This menu entry is enabled if any of the selected items are marked for deletion and unmarks them when selected.

Select All To select the complete graphical wiring.

Create Compound Create a new tab (Compound 1, 2, and so on) of the selected area.

Rename To customize the Compound name.

Copy Graphic If there is a selection, it is copied to the clipboard as a Windows metafile, if there is no selection, the whole diagram is copied to the clipboard. Paste into your favorite documentation tool to document your application.

Save Graphic Same as Copy Graphic but saves to a metafile rather than putting it on the clipboard.

Copy Fragment to File
To make a copy of the selected area and save it to file.

Paste Fragment from File
To paste the selected area from file.

Center To place the selected area in the center of the graphical wiring view.

Wiring Floats with Status Information

There are a number of parameters that support a floating point status. There are circumstances whereby these parameters may have a value that is inaccurate or incorrect for some reason, for example, due to a sensor break, or out-of-range value. In such cases, the Float Status provides indication of whether or not the value can be used.

This status information is made available to any block that is wired from such a parameter, enabling the block to take this status into account.

Block	Input Parameters	Output Parameters
IO.MOD	1.PV to 32.PV	1.PV to 32.PV
SuperLoop.Main	PV	PV
SuperLoop.SP		TrackPV
Math2	In1	Out
	In2	
Programmer.Setup	PVIn	
Poly	In	Out
Load		PVOut1
		PVOut2
Lin16	In	Out
Txdr	InVal	OutVal
IPMonitor	In	Out
SwitchOver	In1	
	In2	
Total	In	
Mux8	In1 to 8	Out
Multi-oper	In1 to 8	SumOut, MaxOut, MinOut, AverageOut
Lgc2	In1	
	In2	
UsrVal	Val	Val
Humidity	WetTemp	RelHumid
	DryTemp	DewPoint
	PsychroConst	
	Pressure	

Parameters appear in both lists where they can be used as inputs or outputs depending on configuration. The action of the block on detection of a ‘Bad’ input is dependent upon the block. For example, the loop treats a ‘Bad’ input as a sensor break and takes appropriate action; the Mux8 simply passes on the status from the selected input to the output, and so on.

The Poly, Lin16, SwitchOver, Multi-Operator, Mux8, IO.Mod.n.PV blocks can be configured to act on bad status in varying ways. The options available are as follows:

0: Clip Bad

The measurement is clipped to the limit it has exceeded and its status is set to ‘BAD’, so that any function block using this measurement can operate its own fallback strategy. For example, a control output may be held at its current value.

1: Clip Good

The measurement is clipped to the limit it has exceeded and its status is set to ‘GOOD’, such that any function block using this measurement may continue to calculate and not employ its own fallback strategy.

2: Fallback Bad

The measurement will adopt the configured fallback value that has been set by the user. In addition the status of the measured value will be set to ‘BAD’, such that any function block using this measurement can operate its own fallback strategy. For example, control loop may hold its output to the current value.

3: Fallback Good

The measurement will adopt the configured fallback value that has been set by the user. In addition the status of the measured value will be set to ‘GOOD’, such that any function block using this measurement may continue to calculate and not employ its own fallback strategy.

4: Up Scale

The measurement will be forced to adopt its high limit. This is like having a resistive pull up on an input circuit. In addition the status of the measured value will be set to ‘BAD’, such that any function block using this measurement can operate its own fallback strategy. For example, the control loop may hold its output to the current value.

5: Down Scale

The measurement will be forced to adopt its low limit. This is like having a resistive pull down on an input circuit. In addition the status of the measured value will be set to ‘BAD’, such that any function block using this measurement can operate its own fallback strategy. For example, the control loop may hold its output to the current value.

Edge Wires

If the Loop.Main.AutoMan parameter were wired from a logic input in the conventional manner it would be impossible to put the instrument into manual via communications. Other parameters need to be controlled by wiring but also need to be able to change under other circumstances, for example, Alarm Acknowledgements. for this reason some Boolean parameters are wired in an alternative way.

These are listed as follows:

Set Dominant

When the wired-in value is 1 the parameter is always updated. This will have the effect of overriding any changes through digital communications. When the wired-in value changes to 0 the parameter is initially changed to 0 but is not continuously updated. This permits the value to be changed through digital communications.

Loop.Main.AutoMan → Programmer.Setup.ProgHold → Access.StandBy

Rising Edge

When the wired-in value changes from 0 to 1, a 1 is written to the parameter. At all other times the wire does not update the parameter. This type of wiring is used for parameters that start an action and, when once completed, the block clears the parameter. When wired to, these parameters can still be operated via digital communications.

Loop.Tune.AutotuneEnable	Txdr.ClearCal	Alarm.Ack
	Txdr.StartCal	DigAlarm.Ack
Programmer.Setup.ProgRun	Txdr.StartHighCal	AlmSummary.GlobalAck
Programmer.Setup.AdvSeg	Txdr.StartTare	
Programmer.Setup.SkipSeg		Instrument.Diagnostics. ClearStats
IPMonitor.Reset		

Both Edge

This type of edge is used for parameters which may need to be controlled by wiring or but should also be able to be controlled through digital communications. If the wired-in value changes then the new value is written to the parameter by the wire. At all other times the parameter is free to be edited through digital communications.

Loop.SP.RateDisable → Loop.OP.RateDisable

Mini8 Loop Controller Overview

Input and output parameters of function blocks are wired together using software wiring to form a particular control strategy within the Mini8 loop controller. An overview of all the available functions and where to obtain more detail is shown below.

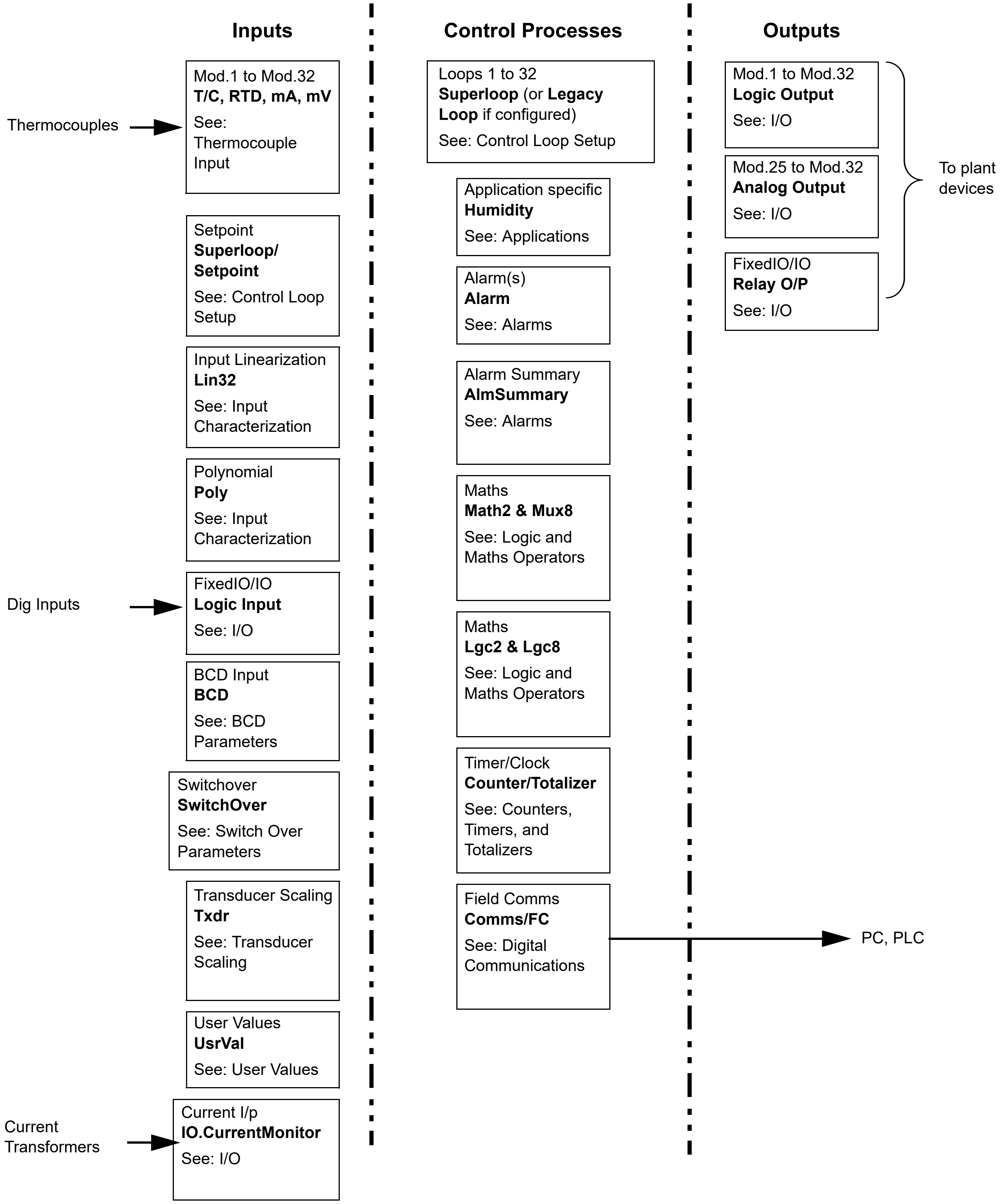


Figure 53 Controller Example

Mini8 loop controllers are supplied unconfigured, and with those blocks included in the order code.

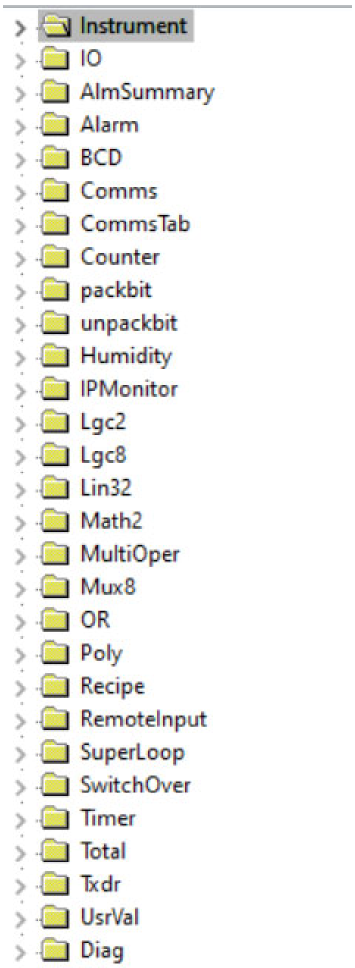
The purpose of the Loop control blocks, using a PID algorithm, is to reduce the difference between SP and PV (the deviation or ‘control error’ signal) to zero by providing a compensating output to the plant via the output driver blocks.

The timer and alarms blocks may be made to operate on a number of parameters within the controller, and digital communications provides an interface to data collection and control.

The controller can be customized to suit a particular process by ‘soft wiring’ between function blocks.

Complete list of Function Blocks

Note: In SIMULATION mode, all features are enabled. Before downloading an application to a REAL device, you should check that the appropriate features are enabled in the device via Feature Security.



The list opposite represents an unconfigured Mini8 loop controller that has been ordered with all features enabled.

If a particular block or blocks do not appear in your instrument then the option has not been ordered. Check the order code of your instrument and contact Eurotherm. They may have been restricted due to Feature Security.

Once a block is dragged and dropped onto the graphical wiring window, the block icon in the block list opposite will be grayed out. At the same time a folder containing the blocks parameters will have been created in the Browse List.

Note: Refer to "Technical Specification" on page 360 for details of these blocks, including the maximum number provided.

Figure 54 Complete List of Function Blocks

Instrument

Instrument / Info

Block: Instrument		**Sub-block: Info**		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
TempUnits	Temperature Units		DegC(0)	NONE
InstrumentNumber	Instrument Number			NONE
Type	Instrument Type			NONE
NativeType	Native instrument type for iTools			NONE
PSUType	PSU Type			NONE
Version	Instrument firmware version			NONE
NativeVersion	Native instrument firmware version for iTools			
CompanyID	Company identification			NONE
CustomerID	Customer identification			NONE
AppName	Name of Application			NONE

Instrument / Security

This list provides security information as follows:

Block: Instrument	**Sub-block: Security**
Name	**Parameter Description**
IM	Instrument Mode
MaxIM	Max Instrument mode (iTools use only)
CommsPassword	To Set Comms Password
CommsPasswordIsSet	Comms password has been Set
ConfigAccess	Indication that config mode can be accessed
CommsPasswordExpiry	Comms Password Expiry Days
PassLock Time	Password lockout time
FeaturePasscode1	Feature Passcode 1
FeaturePasscode2	Feature Passcode 2
FeaturePasscode3	Feature Passcode 3
FeaturePasscode4	Feature Passcode 4
FeaturePasscode5	Feature Passcode 5
ClearMemory	Clear Memory
ConfigLockPassword	Configuration Lock Password
ConfigLockEntry	Configuration Lock Password Entry
ConfigLockStatus	Configuration Lock Status
ConfigLockParamLists	Configuration Lock Parameter Lists
IMGlobal	Comms config locked (iTools only)
EnableUnencryptedLogin	Enable Unencrypted Comms Login
ClearCommsPassword	Clear Comms Password
HttpEnable	Enable Upgrade Mode
UpgradeMode	Enable Upgrade Mode

Instrument / Diagnostics

This list provides diagnostic information as follows:

Block: Instrument	Sub-block: Diagnostics
Name	**Parameter Description**
NotificationStatus	Notification Status Word
StandbyCondStatus	Standby Condition Status Word
SampleTime	Sample Time (in seconds)
DebugComms	Debug Comms
CommsPassUnsuccess	Comms Config Unsuccessful Password Entries
CommsPassSuccess	Comms Config Successful Password Entries
TimeFormat	TimeFormat
TimeDP	Time decimal place
ForceStandby	Force instrument into standby mode
ExecStatus	Execution Status
ResetCounter	Reset counter
IOOutputActiveStatus	IO Output Active Status

Instrument / Modules

This list provides module information as follows:

Block: Instrument	Sub-block: Modules
Name	**Parameter Description**
IO1Fitted	IO 1 Fitted Module
IO1Expected	IO 1 Expected Module
IO2Fitted	IO 2 Fitted Module
IO2Expected	IO 2 Expected Module
IO3Fitted	IO 3 Fitted Module
IO3Expected	IO 3 Expected Module
IO4Fitted	IO 4 Fitted Module
IO4Expected	IO 4 Expected Module
CommsFitted	Comms Module Fitted
CommsExpected	Comms Module Expected

Instrument / ConfigLockConfigList

This list provides information on configuration parameters that can be altered as follows:

Block: Instrument	Sub-block: ConfigLockConfigList
Name	**Parameter Description**
Parameter <1 to 100>	Parameter that is to be alterable.

Instrument / ConfigLockOperList

This list provides information on operational parameters that can be set to Read-only as follows:

Block: Instrument	**Sub-block: ConfigLockOperList**
Name	**Parameter Description**
Parameter <1 to 100>	Parameter that is to be read only.

Instrument / RemoteHMI

This list provides Remote HMI information as follows:

Block: Instrument	**Sub-block: RemoteHMI**
Name	**Parameter Description**
RemoteInterlock	Interlock for Remote HMI
HMIScratch <1 to 30>	HMI Scratch Register <1 to 30>

I/O

This lists the modules fitted into the instruments, all the IO channels, the fixed IO and the current monitoring.

The IO blocklists all the channels of each of the IO boards in the four available slots. Each board has up to eight inputs or outputs making a maximum of 32 channels. The channels are listed under Mod to Mod32.

Slot	Channels
	IO.Mod. to IO.Mod.8
2	IO.Mod.9 to IO.Mod.6
3	IO.Mod.7 to IO.Mod.24
4	IO.Mod.25 to IO.Mod.32

Note: The current transformer input, CT3, is not included in this arrangement. There is a separate sub-block for current monitoring under IO.CurrentMonitor. If this board is fitted into slot 2 the IO.Mod.9 to Mod.6 would not exist.

IO/ ModIDs

<table>
<tr><th colspan="2">Block: IO</th><th colspan="3">Sub-block: ModIDs</th></tr>
<tr><th>Name</th><th>Parameter Description</th><th>Value</th><th>Default</th><th>Access Level</th></tr>
<tr><td>Module</td><td>ModuleIdent</td><td rowspan="4">0 NoMod – No Module
24 DO8Mod – 8 logic outputs
36 RL8Mod – 8 relay outputs
60 DI8 – 8 logic inputs
90 CT3Mod – 3 current transformer inputs
3 TC8Mod – 8 thermocouple/mV inputs
33 TC4Mod – 4 thermocouple/mV inputs
47 - ET8Mod– 8 thermocouple/mV inputs
73 RT4 – 4 Pt00 or Pt000 inputs
20 AO8Mod – 8. 0-20 mA outputs (Slot 4 only)
203 AO4Mod – 4. 0-20 mA outputs (Slot 4 only)</td><td>0</td><td>Read Only</td></tr>
<tr><td>Module2</td><td>Module2Ident</td><td>0</td><td>Read Only</td></tr>
<tr><td>Module3</td><td>Module3Ident</td><td>0</td><td>Read Only</td></tr>
<tr><td>Module4</td><td>Module4Ident</td><td>0</td><td>Read Only</td></tr>
</table>

Modules

The content of the Mod folders depends on the type of IO module fitted in each slot. These will be covered in the following sections.

IO / FixedIO

Each DI8 card provides eight logic input channels (voltage controlled) to the system. These can be wired to provide digital inputs to any function block within the system.

IO / FixedIO / D

Block – IO		**Sub-block FixedIO.D**		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Ident	Channel Ident	Relay () LogicIn (4)	LogicIn (4)	
IOType	IO Type	Input (48) OnOff (50)	Input (48)	
Invert	Invert	No (0) Yes ()	No (0)	
MeasuredVal	Measured Value	Off (0) On ()	On ()	
PV	Process Variable	Off (0) On ()	On ()	
SbyAct	Standby Action	Off (0) On () Cont (2) Frz (3) Cont (4)		

IO / FixedIO / D2

Block – IO		**Sub-block FixedIO.D2**		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Ident	Channel Ident	4 LogicIn	LogicIn (4)	
IOType	IO Type	48 Input	Input (48)	
Invert	Invert	0 No Yes	No (0)	
MeasuredVal	Measured Value	0 Off On	On ()	
PV	Process Variable	0 Off On	On ()	

IO / FixedIO / A

Block – IO		Sub-block FixedIO.A		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Ident	Channel Ident	Relay () LogicIn (4)	Relay ()	
IOType	IO Type	Input (48) OnOff (50)	OnOff (50)	
Invert	Invert	No (0) Yes ()	No (0)	
MeasuredVal	Measured Value	Off (0) On ()	Off (0)	
PV	Process Variable	Off (0) On ()	Off (0)	
SbyAct	Standby Action	Off (0) On ()	Off (0)	

IO / FixedIO / B

Block – IO		Sub-block FixedIO.B		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Ident	Channel Ident	Relay () LogicIn (4)	Relay ()	
IOType	IO Type	Input (48) OnOff (50)	OnOff (50)	
Invert	Invert	No (0) Yes ()	No (0)	
MeasuredVal	Measured Value	Off (0) On ()	Off (0)	
PV	Process Variable	Off (0) On ()	Off (0)	
SbyAct	Standby Action	Off (0) On ()	Off (0)	

IO / CurrentMonitor / Config

Note: If a CT3 board is fitted, then a DO8 board must also be fitted to enable the controller to be configured.

Block – IO		Sub-block CurrentMonitor.Config		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Commission	Commission CT	0 No Auto 2 Manual	No (0)	
CommissionStatus	Commission Status	0 NotCommissioned Commissioning 2 NoD08orRL8Cards 3 NoLoopTPOuts 4 SSRFault 5 NotAccepted 6 Passed 7 ManuallyConfigured 8 MaxLoadsCT 9 MaxLoadsCT2 0 MaxLoadsCT3	Not Commissioned (0)	
Interval	Measurement Interval	Any valid time interval (h:m:s:ms)	0s	
Inhibit	Inhibit	0 No Yes	No (0)	
MaxLeakPh	Max Leakage Current Phase	0.25		
MaxLeakPh2	Max Leakage Current Phase 2	0.25		
MaxLeakPh3	Max Leakage Current Phase 3	0.25		
CTRange	CT input range	0.0		
CT2Range	CT input 2 range	0.0		
CT3Range	CT input 3 range	0.0		
CalibrateCT	Calibrate CT	Idle 2 0mA 3 -70mA 4 LoadFactCal 5 SaveUserCal	Idle ()	
CalibrateCT2	Calibrate CT2	Idle 2 0mA 3 -70mA 4 LoadFactCal 5 SaveUserCal	Idle ()	
CalibrateCT3	Calibrate CT3	Idle 2 0mA 3 -70mA 4 LoadFactCal 5 SaveUserCal	Idle ()	

Logic Input

If a slot is fitted with a DI8 board then eight channels will be available to be configured and connected to Loop inputs.

Logic In Parameters

Block – IO		Sub-block Mod. to .32			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Channel Identity	LogicIn			Read Only
IOType	IO Type	Input OnOff	Logic input On off input		Conf
Invert	Sets the sense of the logic input	No Yes	Normal logic applied Logic NOT applied	No	Conf
MeasuredVal	The current value of the input signal to the hardware including the effect of the Invert parameter.	0	Off On		Read only
PV	This is the input value, before the Invert parameter is applied	0 to 00 or 0 to (OnOff)			Oper

Logic Output

If a slot is fitted with a DO8 board then eight channels will be available to be configured and connected to Loop outputs, alarms or other logic signals.

Logic Out Parameters

Block – IO		Sub-block Mod. to .32			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Channel Identity	LogicOut			Read Only
IOType	IO Type	OnOff	On off output		Conf
		Time Prop	Time proportioning output		
Invert	Sets the sense of the logic output	No Yes	Normal logic applied Logic NOT applied	No	Conf
SbyAct	Action taken by output when instrument goes into Standby Mode	Off, On Continue	Switches On/Off Remains in its last state	Off	Conf
The next five parameters are only shown when 'IO Type' = 'Time Prop' outputs					
MinOnTime	Minimum output on/off time. Stops relays from switching too rapidly	Auto 0.0 to 50.00 seconds	Auto = 20ms. This is the fastest allowable update rate for the output	Auto	Oper
DisplayHigh	The maximum displayable reading	0.00 to 00.00		00.00	Oper
DisplayLow	The minimum displayable reading	0.00 to 00.00		0.00	Oper
RangeHigh	The maximum (electrical) input/output level	0.00 to 00.00		00	Oper
RangeLow	The minimum (electrical) input/output level	0.00 to 00.00		0	Oper
Always displayed					
MeasuredVal	The current value of the output demand signal to the hardware including the effect of the Invert parameter.	0	Off On		Read only
PV	This is the desired output value, before the Invert parameter is applied	0 to 00 or 0 to (OnOff)			Oper

PV can be wired from the output of a function block. For example if it is used for control it may be wired from the control loop output (Ch Output).

Logic Output Scaling

If the output is configured for time proportioning control, it can be scaled such that a lower and upper level of PID demand signal can limit the operation of the output value.

By default, the output will be fully off for 0% power demand, fully on for 00% power demand and equal on/off times at 50% power demand. You can change these limits to suit the process. It is important to note, however, that these limits are set to recommended values for the process. For example, for a heating process it may be required to maintain a minimum level of temperature. This can be achieved by applying an offset at 0% power demand which will maintain the output on for a period of time. Take care to ensure that this minimum on period does not cause the process to overheat.

If Range Hi is set to a value <00% the time proportioning output will switch at a rate depending on the value - it will not switch fully on.

Similarly, if Range Lo is set to a value >0% it will not switch fully off.

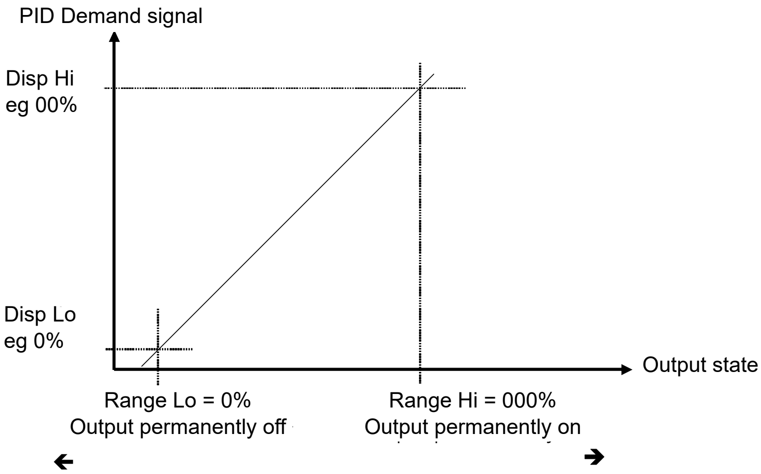
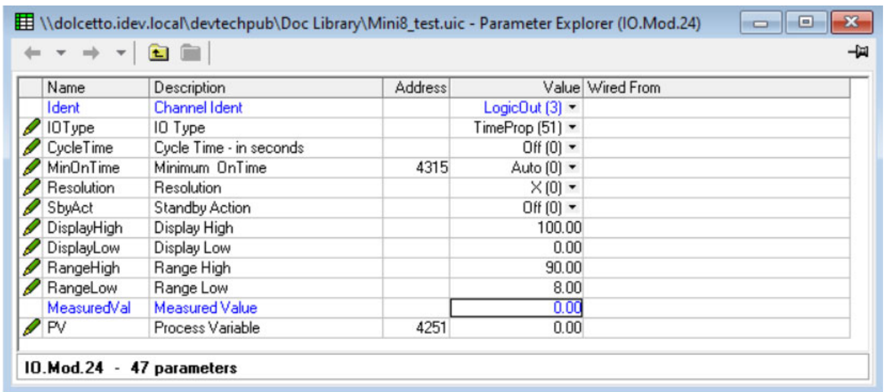


Figure 55 Time Proportioning Output

Example: To Scale a Proportioning Logic Output

Set Access level to ‘configuration’.



Name	Description	Address	Value	Wired From
Ident	Channel Ident		LogicOut (3)	
IOType	IO Type		TimeProp (51)	
CycleTime	Cycle Time - in seconds		Off (0)	
MinOnTime	Minimum OnTime	4315	Auto (0)	
Resolution	Resolution		X (0)	
SbyAct	Standby Action		Off (0)	
DisplayHigh	Display High		100.00	
DisplayLow	Display Low		0.00	
RangeHigh	Range High		90.00	
RangeLow	Range Low		8.00	
MeasuredVal	Measured Value		0.00	
PV	Process Variable	4251	0.00	

Figure 56 Example (Scaling Proportioning Logic Output)

In this example the output will switch on for 8% of the time when the PID demand wired to ‘PV’ signal is at 0%.

Similarly, it will remain on for 90% of the time when the demand signal is at 00%.

Relay Output

If slot 2 and/or 3 is fitted with a RL8 board then eight channels will be available to be configured and connected to Loop outputs, alarms or other logic signals.

Relay Parameters

Block – IO		**Sub-block Mod.9 to Mod.24**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Channel Identity	Relay			Read Only
IOType	IO Type	OnOff	On off output		Conf
		Time Prop	Time proportioning output		
Invert	Sets the sense of the logic input or output	No Yes	Normal logic applied Logical NOT applied	No	Conf
SbyAct	Action taken by output when instrument goes into Standby Mode	Off, On Continue	Switches On/Off Remains in its last state	Off	Conf
The next five parameters are only shown when 'IO Type' = 'Time Prop' outputs					
MinOnTime	Minimum output on/off time Stops relays from switching too rapidly	Auto 0.0 to 50.00 seconds	Auto = 220ms. This is the fastest allowable update rate for the output	Auto	Oper
DisplayHigh	The maximum displayable reading	0.00 to 00.00		00.00	Oper
DisplayLow	The minimum displayable reading	0.00 to 00.00		0.00	Oper
RangeHigh	The maximum (electrical) input/output level	0.00 to 00.00		00	Oper
RangeLow	The minimum (electrical) input/output level	0.00 to 00.00		0	Oper
Always displayed					
MeasuredVal	The current value of the output demand signal to the hardware including the effect of the Invert parameter.	0	Off On		Read only
PV	This is the desired output value, before the Invert parameter is applied	0 to 00 or 0 to (OnOff)			Oper

Thermocouple Input

A TC4 offers four channels and the TC8/ET8 boards offer eight channels which may be configured as thermocouple inputs or mV inputs.

Thermocouple Input Parameters

Block – IO		Sub-blocks: Mod. to Mod.32			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Channel Ident	TCinput			Read Only
IO Type	IO Type	Thermocouple	For direct T/C connection		Conf
		mV	For mV inputs, usually linear, scaled to engineering units.		
Lin Type	Input linearization	see "Linearization Types and Ranges" on page 104			Conf
Units	Display units used for units conversion	see "Input Linearization Parameters" on page 195			Conf
Resolution	Resolution	XXXXX to X.XXXX	Sets scaling for digital communications using the SCADA table		Conf
CJC Type	To select the cold junction compensation method	Internal 0°C (32°F) 45°C (3°F) 50°C (22°F) External Off	See description in "CJC Type" on page 98 for further details	Internal	Conf
SBrk Type	Sensor break type	Low	Sensor break will be detected when its impedance is greater than a 'low' value		Conf
		High	Sensor break will be detected when its impedance is greater than a 'high' value		
		Off	No sensor break		
SBrk Alarm	Sets the alarm action when a sensor break condition is detected	ManLatch	Manual latching — see also "Alarms" on page 117 Alarms		Oper
		NonLatch	No latching		
		Off	No sensor break alarm		
AlarmAck	Sensor Break alarm acknowledge	No Yes		No	Oper
DisplayHigh	The maximum display value in engineering units	-99999 to 99999	For IO Type mV only Limits apply to Linear and SqRoot linearization.	00	Oper
DisplayLow	The minimum display value in engineering units	-99999 to 99999		0	Oper
RangeHigh	The maximum (electrical) input mV	RangeLow to 70		70	Oper
RangeLow	The minimum (electrical) input mV	-70 to RangeHigh		0	Oper
Fallback	Fallback Strategy See also "Fallback" on page 99.	Downscale	Meas Value = Input range Lo - 5% of the mV signal received from the PV input.		Conf
		Upscale	Meas Value = Input range Hi + 5% of the mV signal received from the PV input.		
		Fall Good	Meas Value = Fallback PV		
		Fall Bad	Meas Value = Fallback PV		
		Clip Good	Meas Value = Input range Hi/Lo +/- 5%		
		Clip Bad	Meas Value = Input range Hi/Lo +/- 5%		
Fallback PV	Fallback value See also "Fallback" on page 99		Instrument range		Conf

Block – IO		Sub-blocks: Mod. to Mod.32			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Filter Time Constant	Input filter time. An input filter provides damping of the input signal. This may be necessary to mitigate the effects of excessive electrical noise on the PV input.		Off to 500:00 (hhh:mm) s:ms to hhh:mm	s600ms	Oper
Measured Val	The current electrical value of the PV input				Read Only
PV	The current value of the PV input after linearization		Instrument range		Read Only
LoPoint	Low Point		Lower cal point	0.0	Oper
LoOffset	Low Offset		Offset at lower point	0.0	Oper
HiPoint	High Point		Higher cal point	0.0	Oper
HiOffset	High Offset		Offset at Higher point	0.0	Oper
Offset	Used to add a constant offset to the PV see "PV Offset (Single Point)" on page 100		Instrument range	0.0	Oper
CJC Temp	Reads the temperature of the rear terminals at the thermocouple connection				Read Only
SBrk Value	Sensor break Value Used for diagnostics only, and displays the sensor break trip value				Read Only
Cal State	Calibration State. Calibration of the PV Input is described in "Calibration Parameters" on page 345	Idle			Conf
Status	PV Status The current status of the PV.	0 - OK - Startup 2 - SensorBreak 4 – Out of range 6 - Saturated 8 – Not Calibrated 25 – No Module	Normal operation Initial startup mode Input in sensor break PV outside operating limits Saturated input Uncalibrated channel No Module		Read Only
SbrkOutput	Sensor Break Output	Off /On			Read Only

Linearization Types and Ranges

Input Type		Min Range	Max Range	Units	Min Range	Max Range	Units
J	Thermocouple type J	-20	200	°C	-346	292	°F
K	Thermocouple type K	-200	372	°C	-328	250	°F
L	Thermocouple type L	-200	900	°C	-328	652	°F
R	Thermocouple type R	-50	768	°C	-58	324	°F
B	Thermocouple type B	0	820	°C	32	3308	°F
N	Thermocouple type N	-200	300	°C	-328	2372	°F
T	Thermocouple type T	-250	400	°C	-48	752	°F
S	Thermocouple type S	-50	768	°C	-58	324	°F
PL2	Thermocouple Platinel II	0	369	°C	32	2496	°F
C	Custom						
Linear	mV linear input	-70	70	mV			
SqRoot	Square root						
Custom	Customized linearization tables						

CJC Type

A thermocouple measures the temperature difference between the measuring junction and the reference junction. The reference junction, therefore, must either be held at a fixed known temperature or accurate compensation be used for any temperature variations of the junction.

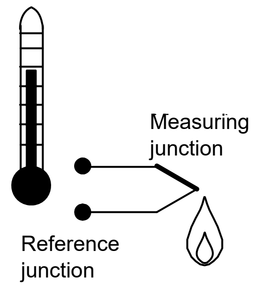


Figure 57 CJC Action

Internal Compensation

The controller is provided with a temperature sensing device which senses the temperature at the point where the thermocouple is joined to the copper wiring of the instrument and applies a corrective signal.

Where very high accuracy is needed and to accommodate multi-thermocouple installations, larger reference units are used which can achieve an accuracy of ±0.°C or better. These units also allow the cables to the instrumentation to be run in copper. The reference units are contained basically under three techniques, Ice-Point, Hot Box and Isothermal.

Ice-Point

There are usually two methods of feeding the EMF from the thermocouple to the measuring instrumentation via the ice-point reference, bellows type and temperature sensor type.

The bellows type utilizes the precise volumetric increase which occurs when a known quantity of ultra pure water changes state from liquid to solid. A precision cylinder actuates expansion bellows which control power to a thermoelectric cooling device. The temperature sensor type uses a metal block of high thermal conductance and mass, which is thermally insulated from ambient temperatures. The block temperature is lowered to 0°C (32°F) by a cooling element, and maintained there by a temperature sensing device.

Special thermometers are obtainable for checking the 0°C (32°F) reference units and alarm circuits that detect any movement from the zero position can be fitted.

Hot Box

Thermocouples are calibrated in terms of EMF generated by the measuring junctions relative to the reference junction at 0°C (32°F). Different reference points can produce different characteristics of thermocouples, therefore referencing at another temperature does present problems. However, the ability of the hot box to work at very high ambient temperatures, plus a good reliability factor has led to an increase in its usage. The unit can consist of a thermally insulated solid aluminium block in which the reference junctions are embedded.

The block temperature is controlled by a closed loop system, and a heater is used as a booster when initially switching on. This booster drops out before the reference temperature, usually between 55°C (3°F) and 65°C (49°F), is reached, but the stability of the hot box temperature is now important. Measurements cannot be taken until the hot box reaches the correct temperature.

Isothermal Systems

The thermocouple junctions being referenced are contained in a block which is heavily thermally insulated. The junctions are allowed to follow the mean ambient temperature, which varies slowly. This variation is accurately sensed by electronic means, and a signal is produced for the associated instrumentation. The high reliability factor of this method has favoured its use for long term monitoring.

CJC Options in Mini8 Loop Controller Series

0 – Internal	CJC measurement at instrument terminals
– 0C	CJC based on external junctions kept at 0°C (Ice Point)
2 – 45C	CJC based on external junctions kept at 45°C (Hot Box)
3 – 50C	CJC based on external junctions kept at 50°C (Hot Box)
4 – External	CJC based on independent external measurement
5 – Off	CJC switched off

Sensor Break Value

The controller continuously monitors the impedance of a transducer or sensor connected to any analog input. This impedance, expressed as a percentage of the impedance which causes the sensor break flag to trip, is a parameter called 'SBrkValue'.

The table below shows the typical impedance which causes sensor break to trip for various types of input and high and low SBrk Impedance readings. The impedance values are only approximate (±25%) as they are not factory calibrated.

TC4/TC8/ET8 Input Range -77 to +77mV	SBrk Impedance – High SBrk Impedance – Low	~ 2kΩ ~ 3kΩ

Fallback

A Fallback strategy may be used to configure the default value for the PV in case of any issues. These may be due an out of range value, a sensor break, lack of calibration or a saturated input.

The Status parameter would indicate the nature of the issue and could be used to diagnose the problem.

Fallback has several modes and may be associated with the Fallback PV parameter

The Fallback PV may be used to configure the value assigned to the PV in case of any issue. The Fallback parameter should be configured accordingly.

The Fallback parameter may be configured so as to force a Good or Bad status when in operation. This in turn allows the user to choose to override issues or to allow them to affect the process.

User Calibration (Two Point)

All ranges of the controller have been calibrated against traceable reference standards. However in a particular application it may be necessary to adjust the displayed reading to overcome other effects within the process. A two point calibration is offered allowing offset and slope adjustment. This is most useful where the setpoints used in a process cover a wide range. The Low and High points should be set on or near the extremities of the range.

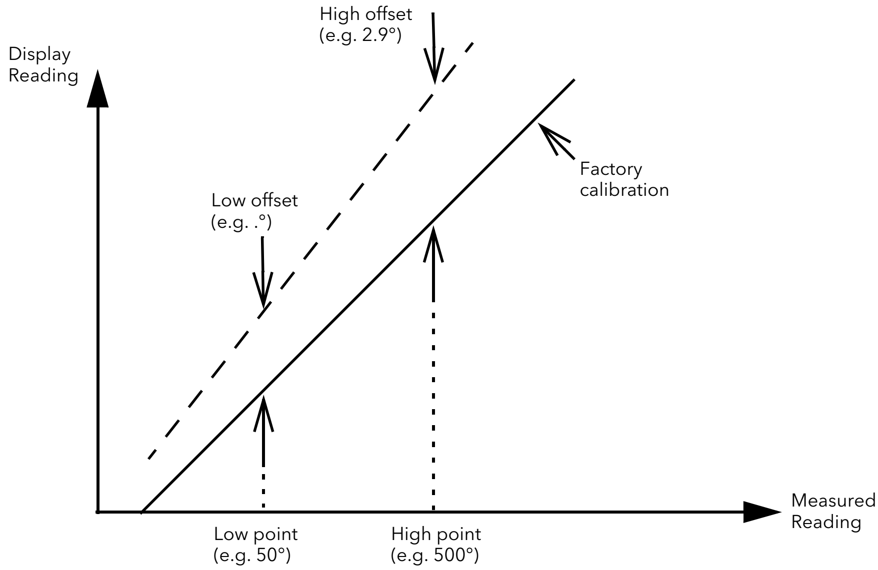


Figure 58 Two Point User Calibration

PV Offset (Single Point)

All ranges of the controller have been calibrated against traceable reference standards. This means that if the input type is changed it is not necessary to calibrate the controller. There may be occasions, however, when you wish to apply an offset to the standard calibration to take account of known issues within the process, for example, a known issue either with a sensor or its positioning. In these instances, it is not advisable to change the reference calibration, but to apply a user defined offset.

A single point offset is most useful where the process setpoint remains at nominally the same value.

PV Offset applies a single offset over the full display range of the controller and can be adjusted in Operator Mode. It has the effect of moving the curve up and down about a central point as shown in the example below:

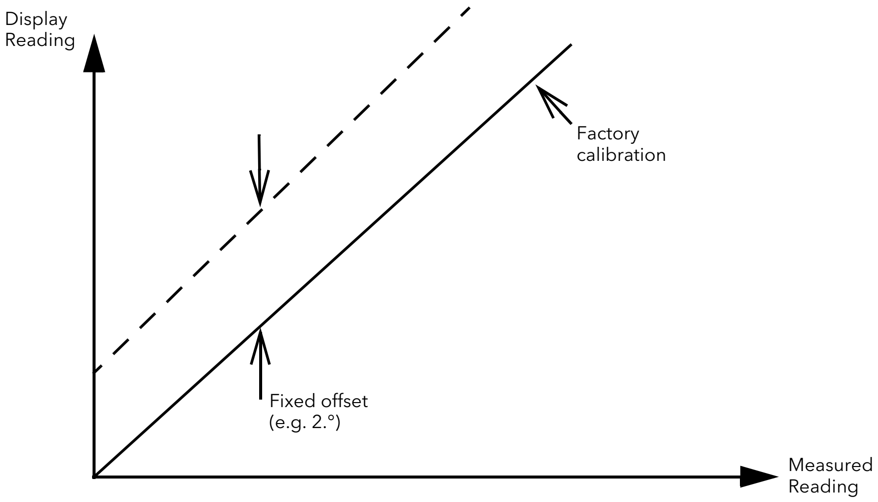


Figure 59 PV Offset Example

Example: To Apply an Offset

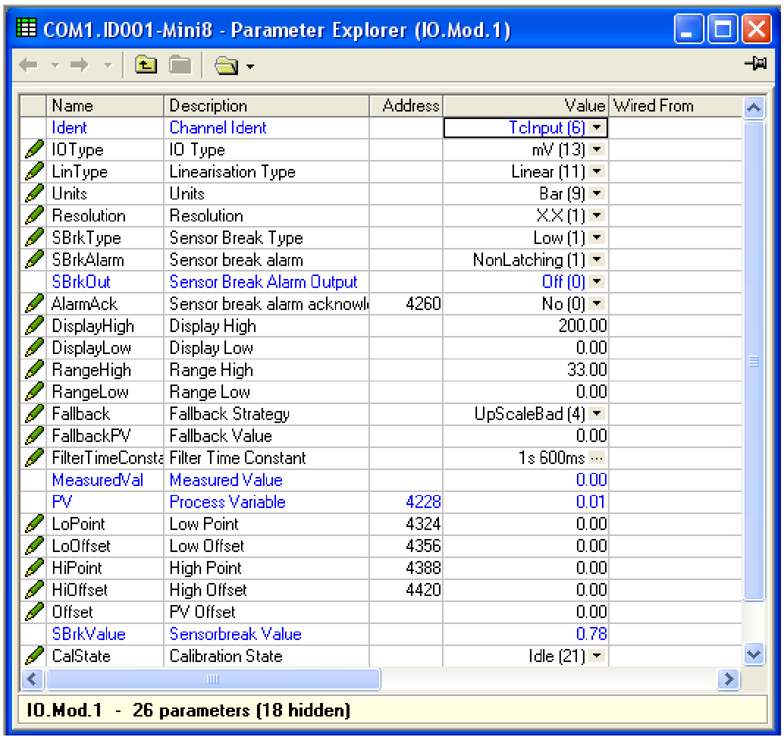
1. Connect the input of the controller to the source device which you wish to calibrate to.
2. Set the source to the desired calibration value. The controller will show the current measurement of the value.
3. If the value is correct, the controller is correctly calibrated and no further action is necessary. If you wish to offset the reading use the Offset parameter where:
   Corrected value (PV) = input value + Offset.

Using TC4 or TC8/ET8 channel as a mV input

Example – a pressure sensor provides 0 to 33mV for 0 to 200 bar.

1. Set IO type to 'mV'.
2. Set Linearisation Type to 'Linear'
3. Set DisplayHigh to '200' (bar).
4. Set DisplayLow to '0' (bar).
5. Set RangeHigh to '33mV'

6. Set RangeLow to ‘0mV’.



Name	Description	Address	Value	Wired From
Ident	Channel Ident		TcInput (6)	
IOType	IO Type		mV (13)	
LinType	Linearisation Type		Linear (11)	
Units	Units		Bar (9)	
Resolution	Resolution		X.X (1)	
SBrkType	Sensor Break Type		Low (1)	
SBrkAlarm	Sensor break alarm		NonLatching (1)	
SBrkOut	Sensor Break Alarm Output		Off (0)	
AlarmAck	Sensor break alarm acknowl	4260	No (0)	
DisplayHigh	Display High		200.00	
DisplayLow	Display Low		0.00	
RangeHigh	Range High		33.00	
RangeLow	Range Low		0.00	
Fallback	Fallback Strategy		UpScaleBad (4)	
FallbackPV	Fallback Value		0.00	
FilterTimeConsta	Filter Time Constant		1s 600ms	
MeasuredVal	Measured Value		0.00	
PV	Process Variable	4228	0.01	
LoPoint	Low Point	4324	0.00	
LoOffset	Low Offset	4356	0.00	
HiPoint	High Point	4388	0.00	
HiOffset	High Offset	4420	0.00	
Offset	PV Offset		0.00	
SBrkValue	Sensorbreak Value		0.78	
CalState	Calibration State		Idle (21)	

Figure 60 Result of Configuration Settings

Note: Maximum input range is ± 70mV.

Resistance Thermometer Input

The RT4 module offers four resistance inputs which can be linear or Pt00/Pt000.

RT Input Parameters

Block – IO		Sub-block: Mod . to .32			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Channel Ident	RTinput			Read Only
IO Type	IO Type	RTD2 RTD3 RTD4	For 2-wire, 3-wire or 4-wire connections.		Conf
ResistanceRange	Resistance Range	Low	Selects a Pt00	Low	Conf
		High	Selects a Pt000		
Lin Type	Linearization Type	See "Linearization Types and Ranges" on page 97			Conf
Units	Display units used for units conversion	See "Input Linearization Parameters" on page 195			Conf
Resolution	Resolution	XXXXX to X.XXXX	Sets scaling for digital communications using the SCADA table		Conf
SBrk Type	Sensor break type	Low	Sensor break will be detected when its impedance is greater than a 'low' value		Conf
		High	Sensor break will be detected when its impedance is greater than a 'high' value		
		Off	No sensor break		
SBrk Alarm	Sets the alarm action when a sensor break condition is detected	ManLatch	Manual latching (see also "Alarms" on page 117)		Oper
		NonLatch	No latching (see also "Alarms" on page 117)		
		Off	No sensor break alarm		
AlarmAck	Sensor Break alarm acknowledge	No Yes		No	Oper
Fallback	Fallback Strategy See also "Fallback" on page 99.	Downscale	Meas Value = Input range Lo - 5%		Conf
		Upscale	Meas Value = Input range Hi + 5%		
		Fall Good	Meas Value = Fallback PV		
		Fall Bad	Meas Value = Fallback PV		
		Clip Good	Meas Value = Input range Hi/Lo +/- 5%		
		Clip Bad	Meas Value = Input range Hi/Lo +/- 5%		
Fallback PV	Fallback value See also "Fallback" on page 99.		Instrument range		Conf
Filter Time Constant	Input filter time. An input filter provides damping of the input signal. This may be necessary to mitigate the effects of excessive electrical noise on the PV input.		Off to 500:00 (hhh:mm) s:ms to hhh:mm	.6 seconds	Oper
Measured Val	The current electrical value of the PV input				Read Only
PV	The current value of the PV input after linearization		Instrument range		Read Only
LoPoint	Low Point		Lower cal point (See "User Calibration (Two Point)" on page 100)	0.0	Oper
LoOffset	Low Offset		Offset at lower cal point	0.0	Oper
HiPoint	High Point		Higher cal point	0.0	Oper
HiOffset	High Offset		Offset at Higher cal point	0.0	Oper

Block – IO		Sub-block: Mod . to .32		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Offset	Used to add a constant offset to the PV see "PV Offset (Single Point)" on page 100	Instrument range	0.0	Oper
SBrk Value	Sensor break Value Used for diagnostics only, and displays the sensor break trip value			Read Only
Cal State	Calibration State. Calibration of the PV Input is described in "Calibration Parameters" on page 345	Idle		Conf
Status	PV Status The current status of the PV.	0 - OK: Normal operation - Startup: Initial startup mode 2 - SensorBreak: Input in sensor break 4 – Out of range: PV outside operating limits 6 - Saturated: Saturated input 8 – Not Calibrated: Uncalibrated channel 25 – No Module: No Module		Read Only
SbrkOutput	Sensor Break Output	Off /On		Read Only

Linearization Types and Ranges

Input Type		Min Range	Max Range	Units	Min Range	Max Range	Units
Pt00	00 ohm platinum bulb	-242	850	°C	-328	562	°F
Linear	Linear	0	420	ohms			
Pt000	000 ohm platinum bulb	-242	850	°C	-328	562	°F
Linear	Linear	0	4200	ohms			

Using RT4 as mA input

Wire the input with a 2.49Ω resistor as shown in "Electrical Connections for RTD" on page 41.

1. Set the Resistance Range to ‘Low’.
2. Set the Linearisation Type to ‘Linear’.

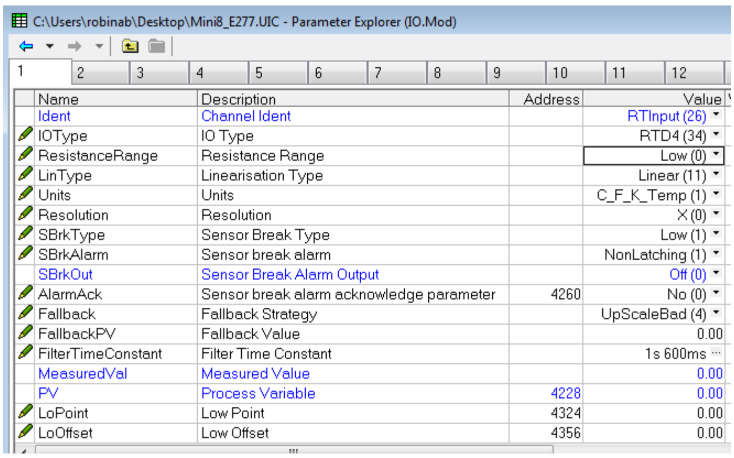


Figure 61 Result of RT4 Configuration Settings

The PV is mapped from the input using User Cal – see "User Calibration (Two Point)" on page 100.

Approximate Values for 4-20mA input with 2.49Ω resistor.

PV range	4 to 20	0 to 00
LoPoint	35.4	35.4
LoOffset	-3.4	-35.4
HiPoint	69.5	69.5
HiOffset	-49.5	-69.5

To achieve accuracy, calibrate the input against a reference. Resistor values up to 5Ω may be used.

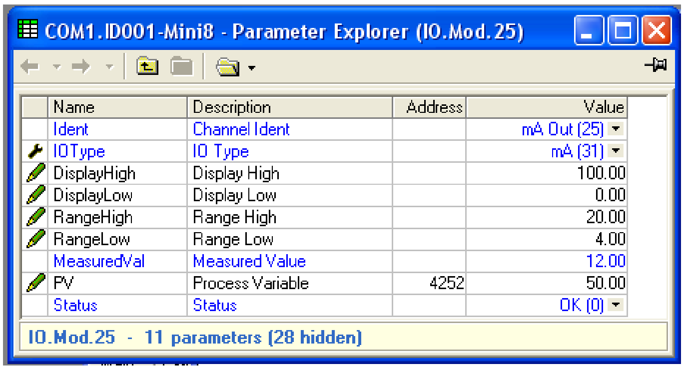
Analog Output

The AO4 offers four channels and the AO8 module eight channels which may be configured as mA outputs. An AO4 or AO8 may only be fitted in Slot 4.

Block – IO		Sub-block: Mod.25 to Mod.32			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Channel ident	mAout			Read Only
IO Type	To configure the output drive signal	mA	Milliamps dc		Conf
Resolution	Display resolution	XXXXX to X.XXXX	Determines scaling for SCADA communications		Conf
Disp Hi	Display high reading	-99999 to 99999 decimal points depend on resolution		00	Oper
Disp Lo	Display low reading			0	Oper
Range Hi	Electrical high input level	0 to 20		20	Oper
Range Lo	Electrical low input level			4	Oper
Meas Value	The current output value				Read Only
PV					Oper
Status	PV Status The current status of the PV.	0 - OK - Startup 2 - SensorBreak 4 – Out of range 6 - Saturated 8 – Not Calibrated 25 – No Module	Normal operation Initial startup mode Input in sensor break PV outside operating limits Saturated input Uncalibrated channel No Module		Read Only

Example: 4 to 20mA Analog Output

In this example 0% (=Display Low) to 00% (=Display High) from a Loop PID Output is wired to this output channel PV input which will give a 4mA (=Range Low) to 20mA (=Range High) control signal.


COM1.ID001-Mini8 - Parameter Explorer (IO.Mod.25)

Name	Description	Address	Value
Ident	Channel Ident		mA Out (25)
IOType	IO Type		mA (31)
DisplayHigh	Display High		100.00
DisplayLow	Display Low		0.00
RangeHigh	Range High		20.00
RangeLow	Range Low		4.00
MeasuredVal	Measured Value		12.00
PV	Process Variable	4252	50.00
Status	Status		OK (0)

IO.Mod.25 - 11 parameters (28 hidden)

Figure 62 Result of Analog Output Configuration Settings

Here the PID demand is 50% giving a MeasuredVal output of 2mA.

Fixed IO

There are two digital inputs, designated D and D2.

Block: IO		Sub-block: Fixed IO.D and IO.D2			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Channel Ident	LogicIn		LogicIn	Read Only
IO Type	IO Type	Input		Input	Read Only
Invert	Invert	No/Yes – input sense is inverted		No	Conf
Measured Val	Measured Value	On/Off	Value seen at the terminals	Off	Read Only
PV	Process Variable	On/Off	Value after allowing for Invert	Off	Read Only

There are two fixed relay outputs, designated A and B.

Block: IO		Sub-block: Fixed IO.A and IO.B			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Channel Ident	Relay		Relay	Read Only
IO Type	IO Type	OnOff		OnOff	Read Only
Invert	Invert	No/Yes = output sense is inverted.		No	Conf
Measured Val	Measured Value	On/Off	Value seen at the terminals after allowing for Invert.	Off	Read Only
PV	Process Variable	On/Off	Requested output before Invert	Off	Oper
SbyAct	Action taken by output when instrument goes into Standby Mode	Off, On Continue	Switches On/Off Remains in its last state	Off	Conf

Current Monitor

The Mini8 loop controller, with a CT3 card, has the capability of detecting external failures of up to 6 heater loads by measuring the current flowing through them via three current transformer inputs. The external failures that can be detected are:

'Solid State Relay (SSR) Fault'

If current is detected flowing through the heater when the controller is requesting it to be off then this indicates that the SSR is short-circuited. If current is not detected when the controller is requesting the heater to be on it indicates that the SSR is open-circuit.

'Partial Load Fault' (PLF)

If less current is detected flowing through the heater than the PLF threshold which has been set for that channel, then this indicates that the heater has a fault detected in it; in applications that use multiple heater elements in parallel then it indicates that one or more of the elements is open-circuit.

'Over Current Fault' (OCF)

If more current is detected flowing through the heater than the OCF threshold then this indicates that the heater has a fault detected in it; in applications that use multiple heater elements in parallel then it indicates that one or more of the elements has lower than expected resistance value.

Note: If the loop associated with a CT monitored output is inhibited, then that output will be excluded from the CT measurements and fault detection.

Heater failures are indicated via individual load status parameters and via four status words. In addition, a global alarm parameter will indicate when a new CT alarm has been detected, which, will also be registered in the alarm log.

Current Measurement

Individual LoadCurrent parameters indicate the current measured for each heater. The Current Monitor function block utilizes a cycling algorithm to measure the current flowing through one heater per measurement interval (default 0s, user alterable). Compensation within the control loop minimizes the disturbance to the PV when current through a load is being measured.

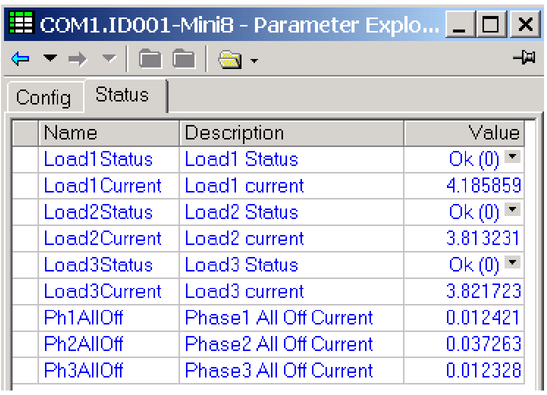


Figure 63 Result of Current Measurement Settings

The interval between successive measurements is dependent upon the average output power required to maintain SP. The recommended absolute minimum interval can be calculated as follows:

Minimum interval (s) > 0.25 * (00/average output power to maintain SP).

For example, if average output power to maintain SP is 0%, using the above rule, the recommended minimum interval is 2.5 seconds. The interval may need to be adjusted depending upon the response of the heaters being used.

Single Phase Configurations

Single SSR triggering

With this configuration, failures of individual heater loads can be detected. For example, if the current detected flowing through Heater 3 is less than its PLF threshold then this will be indicated as Load3PLF.

Example – Using one CT input

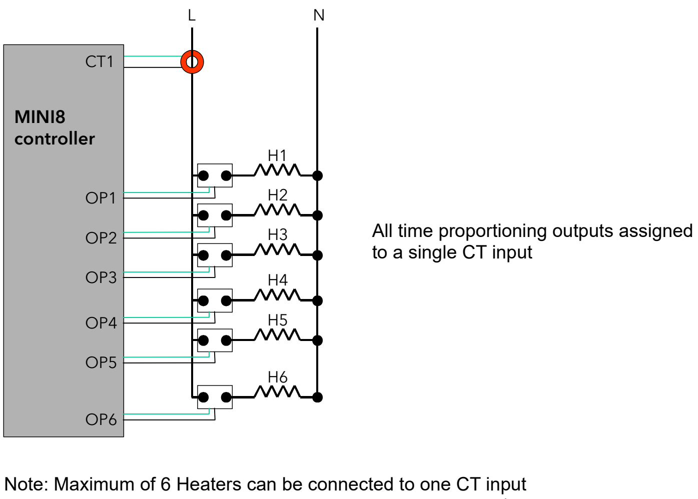


Figure 64 Using one CT Input

Example2 – Using three CT inputs

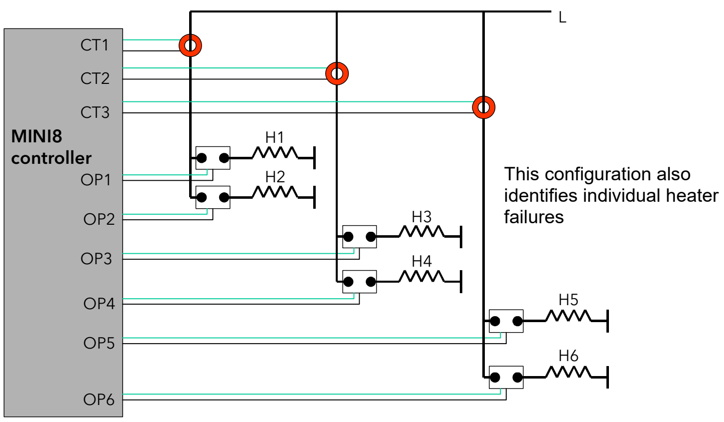


Figure 65 Using three CT Inputs

Multiple SSR triggering

With this configuration, failure of a set of heater loads can be detected. For example, if the current detected flowing through Heater Set is less than the PLF threshold of Load then this will be indicated as LoadPLF. Further investigation will then be required to determine which heater within Set has ceased to operate correctly.

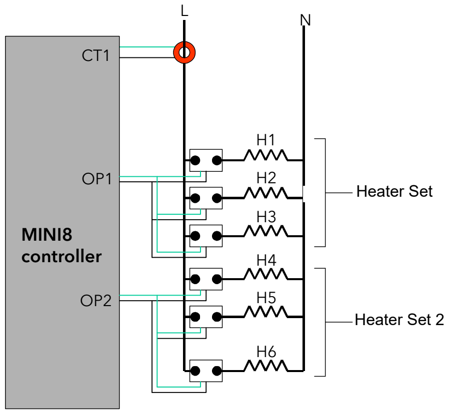


Figure 66 Multiple SSR Triggering

Split Time Proportioning Outputs

This is where a single power demand is split and applied to two time proportioning outputs, that have been scaled, allowing the loads to switch on incrementally as the output power increases. For example, Heater will deliver any demand from 0-50%, and Heater2 will deliver any demand from 50-00% (with Heater fully on).

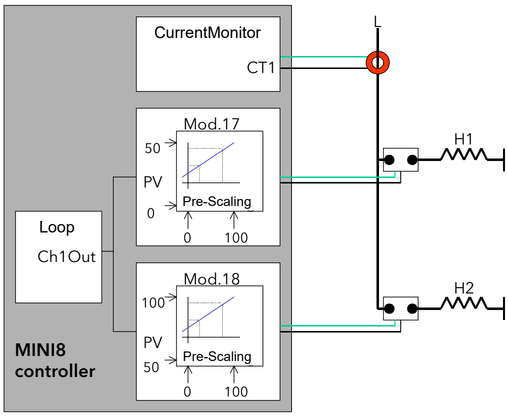


Figure 67 Split Time Proportioning Outputs

As the Mini8 loop controller has the capability of detecting faults with up to 6 heater loads it can handle this type of application even if all eight loops have split time proportioning outputs.

Three Phase Configuration

Configuration for Three Phase supply applications is similar to that for Single phase using three CT inputs.

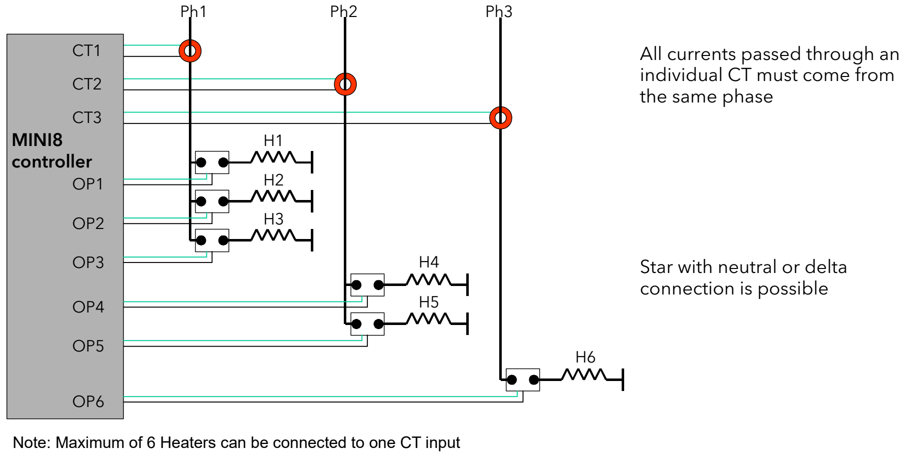


Figure 68 Three Phase Configuration

Parameter Configuration

If Current Monitor is enabled in the folder Instrument/Options/Current Monitor then the current monitor configuration folder appears as a subfolder in IO.

Block: IO		**Sub-block: CurrentMonitor/Config**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Commission	Commission CT	No Auto Manual Accept Abort	See "Commissioning" on page 111	No	Oper
CommissionStatus	Commission Status	Not commissioned	Not commissioned	0	Read Only
		Commissioning	Commissioning in progress		
		NoDO8orRL8cards	There are no DO8/RL8 cards installed in the instrument.		
		NoloopTPouts	The digital outputs are either not configured as time proportioning or are not wired from loop heater channels.		
		SSRfault	An SSR is detected as either short-circuit or open-circuit.		
		MaxLoadsCT/2/3	More than six heaters have been connected to CT input or 2 or 3.		
		NotAccepted	Commissioning did not succeed		
		Passed	Successfully auto commissioned		
		ManuallyConfigured	Configured manually		
Interval	Measurement Interval	sec to min		0s	Oper
Inhibit	Inhibit	No – current is measured Yes –current measurement is inhibited		No	Oper
MaxLeakPh	Max Leakage Current Phase	0.25 to A		0.25	Oper
MaxLeakPh2	Max Leakage Current Phase 2	0.25 to A		0.25	Oper

Block: IO		Sub-block: CurrentMonitor/Config		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
MaxLeakPh3	Max Leakage Current Phase 3	0.25 to A	0.25	Oper
CTRange (see Note)	CT input range	0 to 000A (Ratio to 50mA)	0	Oper
CT2Range (see Note)	CT input 2 range	0 to 000A (Ratio to 50mA)	0	Oper
CT3Range (see Note)	CT input 3 range	0 to 000A (Ratio to 50mA)	0	Oper
CalibrateCT	Calibrate CT	Idle See "Calibration" on page 113 0mA -70mA LoadFactorCal SaveUserCal	Idle	Oper
CalibrateCT2	Calibrate CT2	As CT	Idle	Oper
CalibrateCT3	Calibrate CT3	As CT	Idle	Oper

Note: The current rating of the CT used for each of the CT input channels should cover only the single largest load current proposed for its group of heaters. e.g. if CT has heaters of 5A, 5A & 25A it would need a CT capable of at least 25A.

Commissioning

Auto Commission

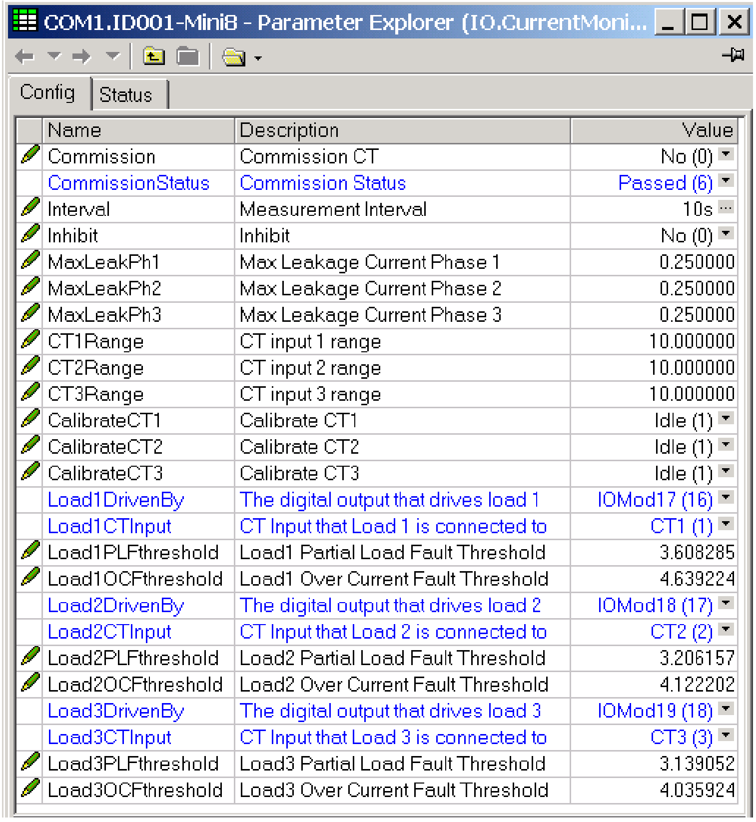
Auto commissioning of the Current Monitor is a feature that automatically detects which time proportioning outputs drive individual heaters (or heater sets), detects which CT input individual heaters are associated with and determines the Partial Load and Over Current thresholds using a :8 ratio. If auto commissioning does not succeed, a status parameter indicates the reason why.

Note: For the auto commissioning to operate successfully, the process must be enabled for full operation of the heating circuit with the digital outputs configured as Time Proportioning and ‘soft’ wired to the appropriate loop heater channels. During auto commissioning digital outputs will switch on and off.

How to Auto Commission

1. Put instrument into Operator Mode.
2. Set Commission to ‘Auto’ and CommissionStatus will display ‘Commissioning’.

3. If successful, CommissionStatus will display ‘Passed’ and configured load parameters will become available.


COM1.ID001-Mini8 - Parameter Explorer (IO.CurrentMoni...

Config | Status

Name	Description	Value
Commission	Commission CT	No (0)
CommissionStatus	Commission Status	Passed (6)
Interval	Measurement Interval	10s
Inhibit	Inhibit	No (0)
MaxLeakPh1	Max Leakage Current Phase 1	0.250000
MaxLeakPh2	Max Leakage Current Phase 2	0.250000
MaxLeakPh3	Max Leakage Current Phase 3	0.250000
CT1Range	CT input 1 range	10.000000
CT2Range	CT input 2 range	10.000000
CT3Range	CT input 3 range	10.000000
CalibrateCT1	Calibrate CT1	Idle (1)
CalibrateCT2	Calibrate CT2	Idle (1)
CalibrateCT3	Calibrate CT3	Idle (1)
Load1DrivenBy	The digital output that drives load 1	IOMod17 (16)
Load1CTInput	CT Input that Load 1 is connected to	CT1 (1)
Load1PLFthreshold	Load1 Partial Load Fault Threshold	3.608285
Load1OCFthreshold	Load1 Over Current Fault Threshold	4.639224
Load2DrivenBy	The digital output that drives load 2	IOMod18 (17)
Load2CTInput	CT Input that Load 2 is connected to	CT2 (2)
Load2PLFthreshold	Load2 Partial Load Fault Threshold	3.206157
Load2OCFthreshold	Load2 Over Current Fault Threshold	4.122202
Load3DrivenBy	The digital output that drives load 3	IOMod19 (18)
Load3CTInput	CT Input that Load 3 is connected to	CT3 (3)
Load3PLFthreshold	Load3 Partial Load Fault Threshold	3.139052
Load3OCFthreshold	Load3 Over Current Fault Threshold	4.035924

Figure 69 Result of Auto-Commissioning

If unsuccessful, CommissionStatus displays the reason:

NoDO8orRL8Cards
Indicates that there are no DO8 or RL8 cards installed in the instrument.

NoLoopTPOuts Indicates that the digital outputs are either not configured as time proportioning or are not wired from loop heater channels.

SSRFault Indicates that an SSR is either short-circuit or open-circuit.

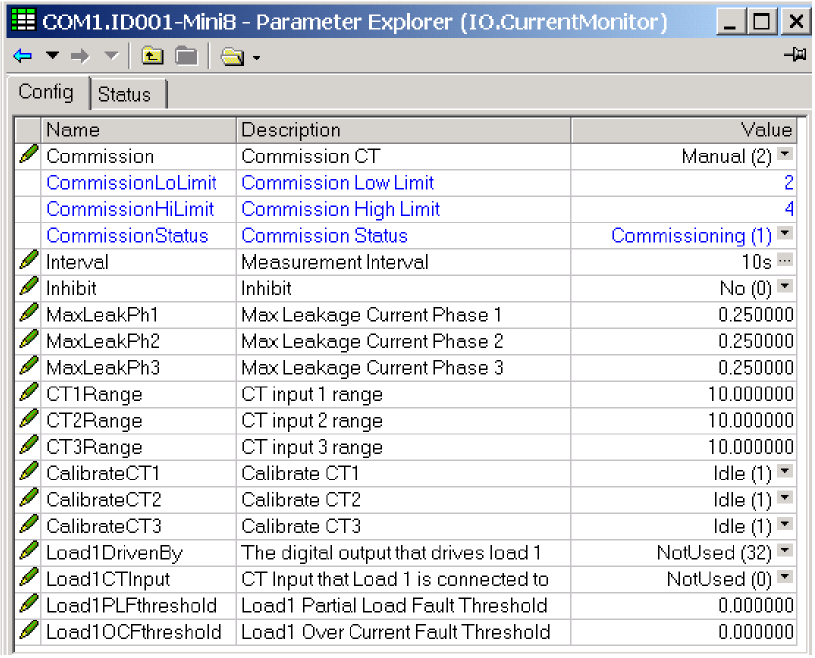
MaxLoadsCT (or 2,3)
Indicates that more than six heaters have been connected to CT input (or 2, 3)

Manual Commission

Manual Commissioning is also available for those users who want to commission the Current Monitor off-line or do not want to accept auto-commissioned settings.

How to Manual Commission

1. Set Commission to 'Manual'. CommissionStatus will display 'Commissioning' and Load configuration parameters will become available:



COM1.ID001-Mini8 - Parameter Explorer (IO.CurrentMonitor)

Config | Status

Name	Description	Value
Commission	Commission CT	Manual (2)
CommissionLoLimit	Commission Low Limit	2
CommissionHiLimit	Commission High Limit	4
CommissionStatus	Commission Status	Commissioning (1)
Interval	Measurement Interval	10s
Inhibit	Inhibit	No (0)
MaxLeakPh1	Max Leakage Current Phase 1	0.250000
MaxLeakPh2	Max Leakage Current Phase 2	0.250000
MaxLeakPh3	Max Leakage Current Phase 3	0.250000
CT1Range	CT input 1 range	10.000000
CT2Range	CT input 2 range	10.000000
CT3Range	CT input 3 range	10.000000
CalibrateCT1	Calibrate CT1	Idle (1)
CalibrateCT2	Calibrate CT2	Idle (1)
CalibrateCT3	Calibrate CT3	Idle (1)
Load1DrivenBy	The digital output that drives load 1	NotUsed (32)
Load1CTInput	CT Input that Load 1 is connected to	NotUsed (0)
Load1PLFthreshold	Load1 Partial Load Fault Threshold	0.000000
Load1OCFthreshold	Load1 Over Current Fault Threshold	0.000000

Figure 70 Load Parameters

2. Set LoadDrivenBy to the IO Module that is connected to the heater load.
3. Set LoadCTInput to the CT input number that is connected to the heater load.
4. Set LoadPLFthreshold and LoadOCFthreshold to appropriate values for the heater load.
5. Repeat for other loads.
6. To use the commissioned settings set Commission to 'Accept'. CommissionStatus will display 'ManuallyConfigured'.
7. To stop manual commissioning set Commission to 'Abort'. CommissionStatus will display 'NotCommissioned'.

Calibration

A Mini8 loop controller supplied from factory with the CT3 card already installed the CT inputs will have been factory calibrated. If the CT3 card is installed at a later date then default calibration values are automatically loaded into the instrument. However, three calibration parameters, one for each CT input, are provided to allow the inputs to be calibrated in the field.

Note: DC Current Source, capable of outputting a –70mA signal, is required to calibrate the inputs.

The three CT inputs are calibrated individually.

How to Calibrate

1. Apply the stimulus (0mA or –70mA) from the DC current source to the CT input to be calibrated.
2. Set CalibrateCT, to reflect the stimulus being applied to the input.
3. CalibrateCT displays ‘Confirm’. Select ‘Go’ to proceed with the calibration process.
4. After selecting Go, CalibrateCT displays ‘Calibrating’.
5. If calibration was successful, CalibrateCT displays ‘Passed’. Select ‘Accept’ to keep the calibration values.
6. If calibration was unsuccessful, CalibrateCT displays ‘Failed’. Select ‘Abort’ to reject the calibration.
7. Select ‘SaveUserCal’ to save the calibration values into non-volatile memory.
8. Select ‘LoadFactCal’ to restore calibration values to the factory calibrated or default settings.

Note: It is possible to stop the calibration process at anytime by selecting ‘Abort’.

Follow the same procedure for CT2 and CT3.

Alarm Summary

AlmSummary

This is a summary of all the alarms in the Mini8 loop controller. It provides global alarm and acknowledge flags as well as 16 bit status words which can be read over communications by the supervisory system.

Block: AlmSummary

Name	Parameter Description	Value		Default	Access Level
NewAlarm	A new alarm has occurred since the last reset (excludes CT alarms)	0 Off 1 On		Off (0)	Read Only
RstNewAlarm	Resets the NewAlarm flag	0 No 1 Yes		No (0)	Oper
NewCTAlarm	A new Current alarm has occurred since the last reset	0 Off 1 On		Off (0)	Read Only
RstNewCTAlarm	Resets the NewCTAlarm flag	0 No 1 Yes		No (0)	Oper
AnyAlarm	Any new alarm since the last reset	0 Off 1 On		Off (0)	Read Only
GlobalAck	Acknowledges every alarm in the Mini8 loop controller requiring acknowledgement. Also resets NewAlarm and NewCTAlarm flags.	0 No 1 Yes		No (0)	Oper
AlarmStatus1	16 bit word for alarms 1 to 8	Bit 0	Alarm 1 active		Read Only
		Bit 1	Alarm 1 not ack'd		
		Bit 2	Alarm 2 active		
		Bit 3	Alarm 2 not ack'd		
		Bit 4	Alarm 3 active		
		Bit 5	Alarm 3 not ack'd		
		Bit 6	Alarm 4 active		
		Bit 7	Alarm 4 not ack'd		
		Bit 8	Alarm 5 active		
		Bit 9	Alarm 5 not ack'd		
		Bit 10	Alarm 6 active		
		Bit 11	Alarm 6 not ack'd		
		Bit 12	Alarm 7 active		
		Bit 13	Alarm 7 not ack'd		
		Bit 14	Alarm 8 active		
		Bit 15	Alarm 8 not ack'd		
AlarmStatus2	16 bit word for alarms 9 to 16	Same format as above			Read Only
AlarmStatus3	16 bit word for alarms 17 to 24	Same format as above			Read Only
AlarmStatus4	16 bit word for alarms 25 to 32	Same format as above			Read Only
AlarmStatus5	16 bit word for alarms 33 to 40	Same format as above			Read Only
AlarmStatus6	16 bit word for alarms 41 to 48	Same format as above			Read Only
AlarmStatus7	16 bit word for alarms 49 to 56	Same format as above			Read Only
AlarmStatus8	16 bit word for alarms 57 to 64	Same format as above			Read Only

Block: AlmSummary					
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
SBrkAlarmStatus1	16 bit word for IO channels Mod.1 to 8	Bit 0	Mod.1 issue		Read Only
		Bit 1	Alarm 1 not ack'd		
		Bit 2	Mod.2 issue		
		Bit 3	Alarm 2 not ack'd		
		Bit 4	Mod.3 issue		
		Bit 5	Alarm 3 not ack'd		
		Bit 6	Mod.4 issue		
		Bit 7	Alarm 4 not ack'd		
		Bit 8	Mod.5 issue		
		Bit 9	Alarm 5 not ack'd		
		Bit 10	Mod.6 issue		
		Bit 11	Alarm 6 not ack'd		
		Bit 12	Mod.7 issue		
		Bit 13	Alarm 7 not ack'd		
		Bit 14	Mod.8 issue		
		Bit 15	Alarm 8 not ack'd		
SBrkAlarmStatus2	16 bit word for IO channels Mod.9 to 16	Same format as above			Read Only
SBrkAlarmStatus3	16 bit word for IO channels Mod.17 to 24	Same format as above			Read Only
SBrkAlarmStatus4	16 bit word for IO channels Mod.25 to 32	Same format as above			Read Only
CTAlarmStatus1	16 bit word for CT alarms 1 to 5	Bit 0	Load1 SSR issue		Read Only
		Bit 1	Load1 PLF		
		Bit 2	Load1 OCF		
		Bit 3	Load2 SSR issue		
		Bit 4	Load2 PLF		
		Bit 5	Load2 OCF		
		Bit 6	Load3 SSR issue		
		Bit 7	Load3 PLF		
		Bit 8	Load3 OCF		
		Bit 9	Load4 SSR issue		
		Bit 10	Load4 PLF		
		Bit 11	Load4 OCF		
		Bit 12	Load5 SSR issue		
		Bit 13	Load5 PLF		
		Bit 14	Load5 OCF		
		Bit 15	-		
CTAlarmStatus2	16 bit word for CT alarms 6 to 10	Same format as above			Read Only
CTAlarmStatus3	16 bit word for CT alarms 11 to 15	Same format as above			Read Only
CTAlarmStatus4	16 bit word for CT alarm 16	Same format as above			Read Only

Alarms

Alarms are used to alert the system when a pre-set level has been exceeded or a particular condition has changed state. As the Mini8 loop controller has no display to show alarms the alarm flags are all available over communications in status words (See "AlmSummary" on page 115. They may also be wired directly or via logic to an output such as a relay.

Alarms can be divided into three main types. These are:

- Analog alarms - operate by monitoring an analog variable such as the process variable and comparing it with a set threshold.
- Digital alarms – operate when the state of a boolean variable changes, for example, sensor break.
- Rate of Change alarms - operate when the rate at which the input increases (Rising Rate of Change) or decreases (Falling Rate of Change) at a rate that exceeds the maximum rate of change (per change time). The alarms remain active until the rising or falling rate of the input is below the configured rate of change.

Number of Alarms - up to 64 alarms may be configured.

Further Alarm Definitions

Hysteresis is the difference between the point at which the alarm switches 'ON' and the point at which it switches 'OFF'. It is used to provide a definite indication of the alarm condition and to minimize alarm relay chatter.

Latch used to hold the alarm condition once an alarm has been detected. It may be configured as:

None (Non latching)
A non-latching alarm will reset itself when the alarm condition is removed.

Auto (Automatic)
An auto-latching alarm requires acknowledgement before it is reset. The acknowledgement can occur BEFORE the condition causing the alarm is removed.

Manual
The alarm continues to be active until both the alarm condition is removed AND the alarm is acknowledged. The acknowledgement can only occur AFTER the condition causing the alarm is removed.

Event
Alarm output will activate.

Block The alarm may be masked during start up. Blocking inhibits the alarm from being activated until the process has first achieved a steady state. It is used, for example, to ignore start up conditions which are not representative of running conditions. A blocking alarm is not re-initiated after a setpoint change.

Delay A short time can be set for each alarm before the output goes into the alarm state. The alarm is still detected as soon as it occurs, but if it cancels before the end of the delay period then no output is triggered. The timer for the delay is then reset. It is also reset if an alarm is changed from being inhibited to uninhibited.

Note: Setting a new alarm threshold causes an action depending on the latching setting:

- If no latching then the alarm condition is re-evaluated and may change.
- If latching then the alarm condition persists until acknowledged.
- Blocking starts after acknowledgement for latching alarms and after setpoint write for non latching.

Analog Alarms

Analog alarms operate on variables such as PV, output levels, and so on. They can be soft wired to these variables to suit the process.

Analog Alarm Types

Absolute High an alarm occurs when the PV exceeds a set high threshold.

Absolute Low an alarm occurs when the PV exceeds a set low threshold.

Deviation High an alarm occurs when the PV is higher than the setpoint by a set threshold.

Deviation Low an alarm occurs when the PV is lower than the setpoint by a set threshold.

Deviation Band an alarm occurs when the PV is higher or lower than the setpoint by a set threshold.

These are shown graphically below for changes in PV plotted against time. (Hysteresis set to zero).

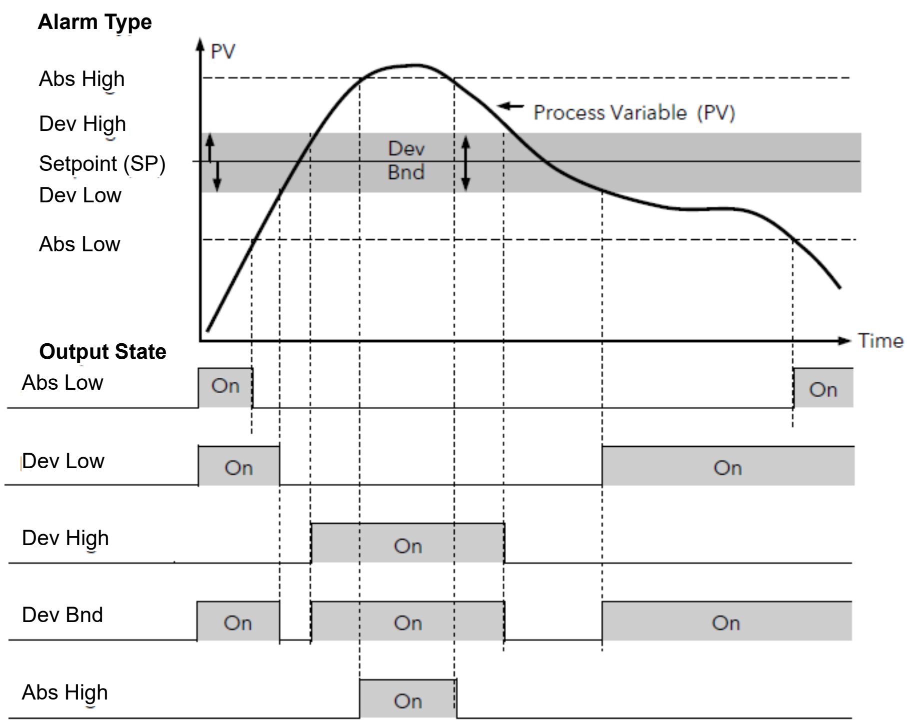


Figure 71 Analog Alarm Types

Digital Alarms

Digital alarms operate on Boolean variables. They can be soft wired to any suitable Boolean parameter such as digital inputs or outputs.

Digital Alarm Types

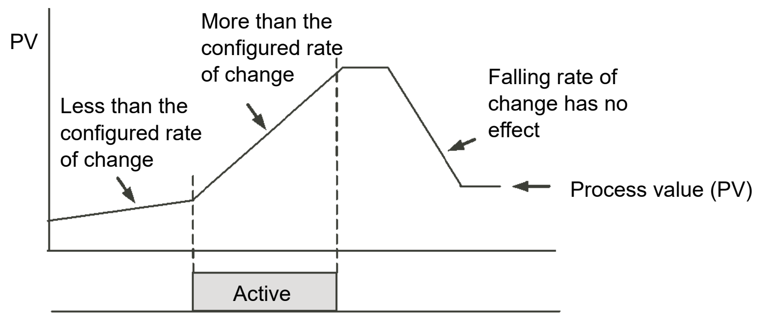
Pos Edge	The alarm will trigger when the input changes from a low to high condition.
Neg Edge	The alarm will trigger when the input changes from a high to low condition.
Edge	The alarm will trigger on any change of state of the input signal.
High	The alarm will trigger when the input signal is high.
Low	The alarm will trigger when the input signal is low.

Rate of Change Alarms

Rate of Change alarms operate on the rate at which the input increases or decreases with respect to the configured maximum rate of change (per change time). They are either Rising or Falling Rate of Change alarms.

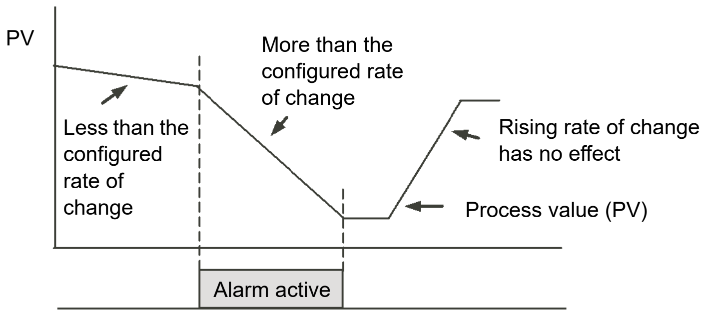
Rising Rate of Change

The Rising Rate of Change alarm sets the alarm active when the rate at which the input increases exceeds the configured maximum rate of change (per change time). It will remain active until the rising rate of the input falls below the configured rate of change.



Falling Rate of Change

The Falling Rate of Change alarm sets the alarm active when the rate at which the input decreases exceeds the configured maximum rate of change (per change time). It will remain active until the falling rate of the input falls below the configured rate of change.



Alarm Outputs

Alarms can operate a specific output (usually a relay). Any individual alarm can operate an individual output or any combination of alarms can operate an individual output. They are wired as required in configuration level.

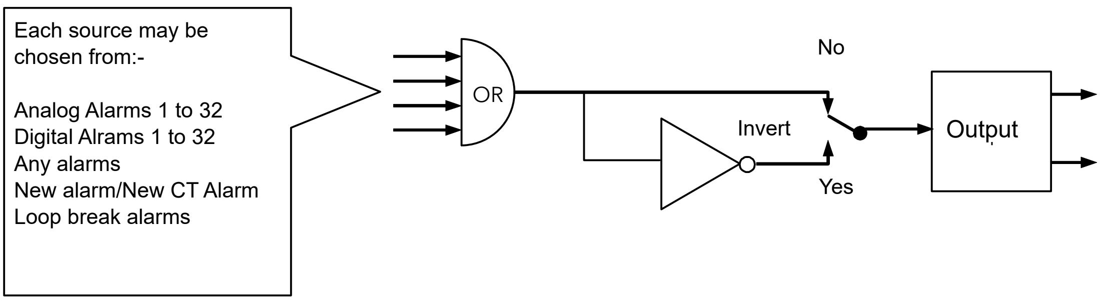


Figure 72 Alarm Outputs

How Alarms are Indicated

Alarm states are all embedded in 16-bit status words. See "AlmSummary" on page 115.

To Acknowledge an Alarm

Set the appropriate alarm acknowledge flag to acknowledge that particular alarm. Alternatively the GlobalAck in the AlmSummary folder can be used to acknowledge ALL alarms that require acknowledging in the instrument.

The action, which now takes place, will depend on the type of latching, which has been configured.

Non-Latched Alarms

If the alarm condition is present when the alarm is acknowledged, the alarm output will be continuously active. This state will continue for as long as the alarm condition remains. When the alarm condition clears the output will go off.

If the alarm condition clears before it is acknowledged the alarm output goes off as soon as the condition disappears.

Automatic Latched Alarms

The alarm continues to be active until both the alarm condition is removed AND the alarm is acknowledged. The acknowledgement can occur BEFORE the condition causing the alarm is removed.

Manual Latched Alarms

The alarm continues to be active until both the alarm condition is removed AND the alarm is acknowledged. The acknowledgement can only occur AFTER the condition causing the alarm is removed.

Alarm Parameters

Four groups of eight alarms are available. The following table shows the parameters to set up and configure alarms.

Block: Alarm Sub-blocks: 1 to 64

Name	Parameter Description	Value		Default	Access Level
Type	Selects the type of alarm	0 Off	Alarm not configured	Off (0)	Conf
		1 Abs Hi	Full Scale High		
		2 Abs Lo	Full Scale Low		
		3 Dev Hi	Deviation High		
		4 Dev Lo	Deviation Low		
		5 DevBnd	Deviation Band		
		6 RRoC	Rising Rate of Change		
		7 FRoC	Falling Rate of Change		
		8 DigHi	Digital High (1)		
		9 DigLo	Digital Low (0)		
		10 DigPosEdge	On rising edge		
		11 DigNegEdge	On falling edge		
		12 DigEdge	On change		
		13 AbsHiLo	Full Scale High or Low		
Status	The alarm status	Off (0)	The alarm is not active	Off (0)	Oper
		Active (1)	The alarm is active		
		InactiveNotAckd(2)	The alarm is inactive and has not been acknowledged		
		ActiveNotAckd(3)	The alarm is active and has not been acknowledged		
Input	This is the parameter that will be monitored and checked according to the AlarmType to see if an alarm condition has occurred.	0 to 1			Oper
Threshold	The alarm HIGH threshold	A value between -3.403E38 and +3.403E38		1.00	Conf
Hysteresis	The alarm hysteresis	A value between -3.403E38 and +3.403E38		0.00	Conf
Latch	Determine the type of latching the alarm will use, if any. Auto latching allows acknowledgement while the alarm condition is still active, whereas manual latching needs the condition to leave the alarm state before the alarm can be acknowledged. See also the description at the beginning of this chapter.	None	No latching is used		Oper
		Auto	Automatic		
		Manual	Manual		
		Event	Event		

Block: Alarm	Sub-blocks: 1 to 64				
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Block	Alarm Blocking is used to inhibit alarms from activating during start-up. In some applications, the measurement at start-up is in an alarm condition until the system has come under control. Blocking causes the alarms to be ignored until the system is under control, after this any deviations trigger the alarm.	No Yes	No blocking Blocking		Oper
Delay	This is a small delay between sensing the alarm condition and displaying it. If in the time between the two, the cause of the alarm is removed, then no alarm is shown and the delay timer is reset. It can be used on systems that are prone to electrical noise.	0:00.0 to 500 mm:ss.s hh:mm:ss hhh:mm		0:00.0	Oper
Output	The output indicates whether the alarm is on or off depending on the alarm condition, latching and acknowledge, inhibiting and blocking.	Off	Alarm output deactivated		Read Only
		On	Alarm output activated		
Ack	Used in conjunction with the latching parameter. It is set when the user responds to an alarm.	No Yes	Not acknowledged Acknowledged		Oper
Inhibit	Inhibit is an input to the Alarm function. It allows the alarm to be switched OFF. Typically the Inhibit is connected to a digital input or event so that during a phase of the process alarms do not activate. For Example, if the door to a furnace is opened the alarms may be inhibited until the door is closed again.	No Yes	Alarm not inhibited Inhibit function active		Oper
StandbyInhibit	Inhibit in Standby	Off (0)	No Inhibit in Standby	Off (0)	Conf
		On (1)	Inhibit in Standby active		

Example: To Configure Alarm 1 (as an Analog Alarm)

Change Access level to Configuration.

In this example the high alarm will be detected when the measured value exceeds 100.00.

The current measured value is 0.00 as measured by the ‘Input’ parameter. This parameter will normally be wired to an internal source such as a thermocouple input. In this example the alarm will set when the measured value exceeds the threshold 100.0 and will clear when the input decreases 0.50 units below the threshold level (i.e. at 99.5 units).

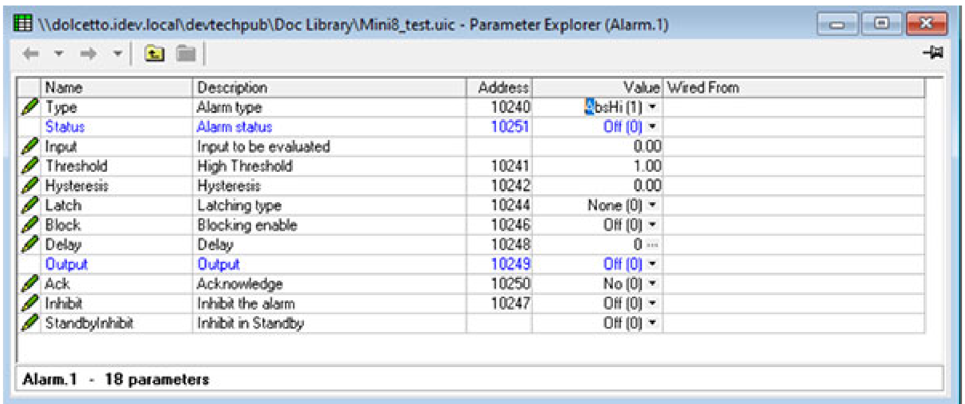


Figure 73 Configuring Alarm 1 as an Analog Alarm

Example: To Configure Alarm 2 (as a Digital Alarm)

Change Access level to Configuration.

In this example the digital alarm will come on if Timer 1 expires.

Timer.1.Out is wired to the alarm input. The Alarm.2.Out will turn on if the timer expires.

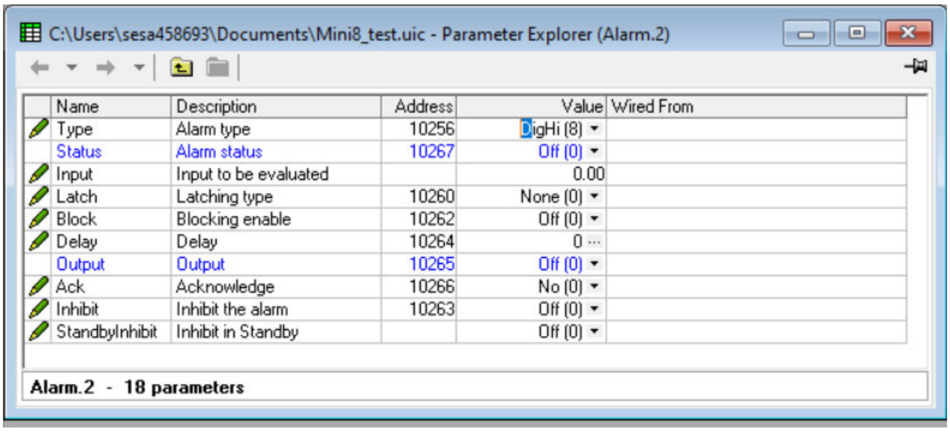


Figure 74 Configuring Alarm 2 as a Digital Alarm

BCD Input

The BCD Input Function block takes eight digital inputs and combines them to make a single numeric value, typically used to select a program or a recipe.

The block uses four bits to generate a single digit.

The following table shows how the input bits combine to make the output values.

Input 1	Units value (0 – 9)	BCD value (0 – 99)	Decimal value (0 – 255)
Input 2			
Input 3			
Input 4			
Input 5	Tens value (0 – 9)		
Input 6			
Input 7			
Input 8			

Since the inputs cannot all be relied upon to change simultaneously, the output will only update after all the inputs have been steady for two samples.

BCD Parameters

Block – BCDInput		**Sub-blocks: 1 and 2**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
BcdInput1	Digital Input 1	On or Off	Alterable from the operator interface if not wired	Off	Oper
BcdInput2	Digital Input 2	On or Off		Off	Oper
BcdInput3	Digital Input 3	On or Off		Off	Oper
BcdInput4	Digital Input 4	On or Off		Off	Oper
BcdInput5	Digital Input 5	On or Off		Off	Oper
BcdInput6	Digital Input 6	On or Off		Off	Oper
BcdInput7	Digital Input 7	On or Off		Off	Oper
BcdInput8	Digital Input 8	On or Off		Off	Oper
BcdOP	Reads the value (in BCD) of the switch as it appears on the digital inputs	0 – 99	See examples below		Read Only
BcdSettleTime	Settle Time				Oper

BCD Inputs								**BCD Output**	**Decimal Equivalent**
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**		
1	0	0	0	0	0	0	0	1	1
1	1	1	1	0	0	0	0	9	15
0	0	0	0	1	1	1	1	90	240
1	1	1	1	1	1	1	1	99	255

Example: To Wire a BCD Input

The BCD digital input parameters may be wired to digital input terminals of the controller. A DI8 module may be used and there are also two standard digital input terminals in FixedIO, D1 and D2.

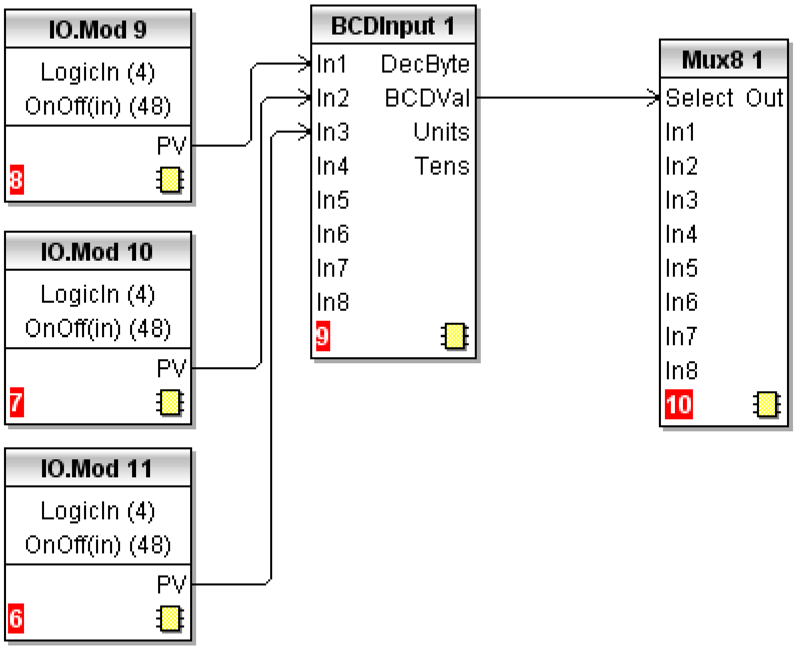


Figure 75 Example BCD Wiring

This example shows a BCD switch selecting one of eight values, In1 to In8 on the Mux8.

Digital Communications

Digital Communications (or ‘comms’ for short) allows the Mini8 loop controller to be part of a system by communicating with a PC or a programmable logic controller (PLC).

The Mini8 loop controller also has a configuration port for ‘cloning’ or saving/loading instrument configurations for future expansion of the plant or to allow you to recover a system if necessary.

Note: As the terms “Modbus Master” and “Modbus Slave” are now deprecated, they have been replaced in this Chapter with “Modbus Client” and “Modbus Server” respectively.

Configuration Communications Port

The Configuration Communications Port, known as ConfigComms (CC) is on an RJ11 socket, just to the right of the power supply connections. This will normally be connected to a PC running iTools. When connecting to iTools the instrument on this port will be found at address 255. iTools will also optimize the baud rate to suit the conditions.

Eurotherm supply a standard cable to connect a serial COM port on a computer to the RJ11 socket, part no. SubMini8/cable/config.

This port conforms to MODBUS RTU® protocol, a full description of which can be found at www.modbus.org.

Pin connections for the RJ11 connector are shown in "Configuration Communications Port (CC)" on page 31.

Note: The CC port is not isolated, and should not be used for connection to other devices. It should be used only for configuration and commissioning.

The baud rate of the CC port defaults to 19200bps. Set the comms port in the PC to the correct rate.

Configuration is also possible through the Field Communications port but ONLY if that port is Modbus or ModbusTCP. In that situation the Mini8 loop controllers can be multi-dropped to iTools.

Configuration Communications Parameters (Main)

Block - Comms		Sub-blocks: CC.Main (Config Comms Main)			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Interface	Comms Interface	None (0)	No Communications Interface		Read Only
		DeviceNet (63)	DeviceNet		
		Modbus non-iso (94)	Non-isolated Modbus		
		Modbus isolated (110)	Isolated Modbus		
		DeviceNet Enh (126)	Enhanced DeviceNet		
		EtherCAT (142)	EtherCAT		
		Ethernet (143)	Ethernet		
Protocol	Digital communications protocol	Modbus. The CC channel only supports Modbus RTU protocol.		Modbus RTU	Read Only
WDTimeout	Network Watchdog Timeout	0	De-activates the Watchdog	1	Conf
		1	Activates the Watchdog		
WDAction	Network Watchdog Action	0	Manual Recovery	1	Conf
		1	Automatic Recovery		
WDFlag	Network Watchdog Flag	0	Off	1	Conf
		1	On		
Delay	Comms delay	No	No delay	No	Conf
		Yes	Fixed delay. This inserts a delay between Rx and Tx to help ensure that the drivers used by intelligent EIA-232/EIA-485 converters have sufficient time to switch over.		
TimeFormat	Time format	0	milliseconds		Conf
		1	seconds		
		2	minutes		
		3	hours		

Configuration Communications Parameters (Network)

Block - Comms		Sub-blocks: CC.Network (Config Comms Network)			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Baud	Communications baud rate	4800		19200	Conf
		9600			
		19k2 (19200)			
Parity	Communications parity	None	No parity	None	Conf
		Even	Even parity		
		Odd	Odd parity		
Address	Instrument address	1 to 254		1	Oper

Field Communications Port (FC)

The Mini8 loop controller has a number of communication options. These have to be ordered from the factory as part of the instrument build. A change of protocol is not usually possible in the field. The physical port and the connections will vary depending on the field communications protocol. These are shown in the wiring section of the manual (see "Electrical Connections – Common to All Instruments" on page 29. Mini8 loop controller offers Modbus, DeviceNet, and Ethernet Modbus-TCP. These protocols are described in the following sections.

Communications Identity

The instrument recognizes the type of communication board fitted. The identity ‘Ident’ is displayed to show that the instrument is built as required.

Field Communications Parameters (Main)

Block - Comms		**Sub-blocks: FC.Main (Field Comms Main)**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Interface	Comms Interface	None (0) DeviceNet (63) Modbus non-iso (94) Modbus isolated (110) DeviceNet Enh (126) EtherCAT (142) Ethernet (143)	No Communications Interface DeviceNet Non-isolated Modbus Isolated Modbus Enhanced DeviceNet EtherCAT Ethernet		Read Only
Protocol	Digital communications protocol	ModbusSlave (11).	Modbus Slave	Modbus Slave	Read Only
		EtherNetIPAndModbus (12)	EtherNet/IP and Modbus		
		BacnetAndModSlv (13)	BacNet and Modbus Slave		
		ModMstAndASlv (15)	Modbus Master and Slave		
Status	Comms Network Status	Running (0) Init (1) Ready (2) Offline (3) Bad_GSD (4) Offline (10) Ready (11) Online (12) IOTimeout (13) LinkFail (14) Com Fault (15)	Network connected Network Initialising Network Ready Network Offline Device bad GSD (only Profibus) DeviceNet Offline DeviceNet Ready (no connections) DeviceNet Online DeviceNet IO Timeout DeviceNet Link Fail DeviceNet Comms Fault		Read Only
WDTimeout	Network Watchdog Timeout	0 1	De-activates the Watchdog Activates the Watchdog	1	Conf
WDAction	Network Watchdog Action	0 1	Manual Recovery Automatic Recovery	1	Conf
WDFlag	Network Watchdog Flag	0 1	Off On	1	Conf
TimeFormat	Time format	0 1 2 3	milliseconds seconds minutes hours		Conf

Field Communications Parameters (Network)

Block - Comms		Sub-blocks: FC.Network (Field Comms Network)			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
AutoDiscovery	Enables automatic discovery of instrument on a network	Off (0) On (1)		Off (0)	Conf
IP Mode	IP Mode	Static (0) DHCP (1)	Static IP number Dynamic IP number	Static (0)	Read Only
IPAddress1	1st byte of IP address	1 to 254			Conf
IPAddress2	2nd byte of IP address	1 to 254			
IPAddress3	3rd byte of IP address	1 to 254			
IPAddress4	4th byte of IP address	1 to 254			
SubnetMask1	1st byte of Subnet mask	0 to 255			Conf
SubnetMask2	2nd byte of Subnet mask	0 to 255			
SubnetMask3	3rd byte of Subnet mask	0 to 255			
SubnetMask4	4th byte of Subnet mask	0 to 255			
DefaultGateway1	1st byte of Default gateway	0 to 255			Conf
DefaultGateway2	2nd byte of Default gateway	0 to 255			
DefaultGateway3	3rd byte of Default gateway	0 to 255			
DefaultGateway4	4th byte of Default gateway	0 to 255			
MAC1	MAC address 1	0 to 255			Read Only
MAC2	MAC address 2	0 to 255			
MAC3	MAC address 3	0 to 255			
MAC4	MAC address 4	0 to 255			
MAC5	MAC address 5	0 to 255			
MAC6	MAC address 6	0 to 255			
BroadcastStormActive	Broadcast Storm Active	No (0) Yes (1)		No (0)	Read Only
RateProtectionActive	Rate Protection Active	No (0) Yes (1)		No (0)	Read Only
PrefMasterIPAddress1	1st byte of preferred master IP address	0 to 255			
PrefMasterIPAddress2	2nd byte of preferred master IP address	0 to 255			
PrefMasterIPAddress3	3rd byte of preferred master IP address	0 to 255			
PrefMasterIPAddress4	4th byte of preferred master IP address	0 to 255			

Modbus

This port conforms to MODBUS RTU® protocol a full description of which can be found on www.modbus.org.

Modbus Connections

This uses two parallel RJ45 connectors for use with shielded Cat5e patch cables. The connection is usually 2-wire but 4-wire is also available. This is selected by the top switch of the address switches below the RJ45 ports – OFF (to the left) 2-wire, ON (to the right) 4-wire.

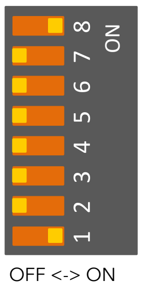
RJ45 pin connections are shown in "Electrical Connections for Modbus RTU" on page 31.

Modbus Address Switch

On a network of instruments an address is used to specify a particular instrument. Each instrument on a network MUST have a unique address. Address 255 is reserved for configuration using the configuration port or the configuration clip.

The switch is situated at the bottom of the Comms module. The switch gives addresses from 1 to 31. If Address 0 is set the Mini8 loop controller will then take the address and parity settings entered in the configuration of the instrument, see "Modbus Parameters" on page 145. This allows for addresses above 31.

Sw	OFF	ON
8	3-wire	4-wire
7	NO Parity	Parity
6	Even	Odd
5	-	Address 16
4	-	Address 8
3	-	Address 4
2	-	Address 2
1	-	Address 1



Example shows 4 wire and address 1

Baud Rate

The baud rate of a communications network specifies the speed that data is transferred between instrument and Client (Master). A baud rate of 9600 equates to 9600 bits per second. Since a single character requires 8 bits of data plus start, stop, and optional parity, up to 11 bits per byte may be transmitted. 9600 baud equates approximately to 1000 bytes per second. 4800 baud is half the speed – approx. 500 bytes per second.

In calculating the speed of communications in your system it is often the Latency between a message being sent and a reply being started that dominates the speed of the network.

For example, if a message consists of 10 characters (10ms at 9600 Baud) and the reply consists of 10 characters, then the transmission time would be 20ms. However, if the Latency is 20ms, then the transmission time has become 40ms. Baud rate is set in the parameter list see "Modbus Parameters" on page 145.

Parity

Parity is a method of confirming that the data transferred between devices has not been corrupted.

Parity is the lowest form of integrity in the message. It indicates that a single byte contains either an even or an odd number of ones or zeroes in the data.

In industrial protocols, there are usually layers of checking to confirm that the first byte transmitted is good. Modbus applies a CRC (Cyclic Redundancy Checksum) to the data to confirm that the package is correct.

Parity is set in the parameter list see "Modbus Parameters" on page 145.

Rx/Tx Delay Time

In some systems it is necessary to introduce a delay between the instrument receiving a message and its reply. This is sometimes caused by communications converter boxes which require a period of silence on the transmission to switch over the direction of their drivers.

Broadcast Client

NOTICE

POTENTIAL INSTRUMENT DAMAGE

When using broadcast client (master) communications, be aware that updated values are sent many times a second. Before using this facility, check that the instrument to which you wish to send values can accept continuous writes. Note that in common with many third party lower cost units, the Eurotherm 3200 series requires continuous writes to be directed to the remote setpoint rather than the working setpoint. For non-Eurotherm devices, damage to the internal non-volatile memory could result from the use of this function. If in any doubt, contact the manufacturer of the device in question for advice.

When using the 3200 series fitted software version 1.10 and greater, use the Remote Setpoint variable at Modbus address 26 if you need to write to a temperature setpoint. This has no write restrictions and may also have a local trim value applied. There is no restriction on writing to the EPC2000, EPC3000, 3500 or Mini8 loop controller series.

Failure to follow these instructions can result in equipment damage.

The Mini8 loop controller broadcast client (master) can be connected to up to 31 servers (slaves) if no segment repeaters are used. If repeaters are used to provide additional segments, 32 servers (slaves) are permitted in each new segment. The client (master) is configured by selecting a Modbus register address to which a value is to be sent. The value to send is selected by wiring it to the Broadcast Value. Once the function has been enabled, the instrument will send this value out over the communications link every control cycle typically every 110ms.

Notes:

1. The parameter being broadcast must be set to the same decimal point resolution in both client (master) and server (slave) instruments.
2. If iTools, or any other Modbus client, is connected to the port on which the broadcast client (master) is enabled, then the broadcast is temporarily inhibited. It will restart approximately 30 seconds after iTools is removed. This is to allow reconfiguration of the instrument using iTools even when broadcast client (master) communications is operating.

A typical example might be a multi zone application where the setpoint of each zone is required to follow, with digital accuracy, the setpoint of a client (master).

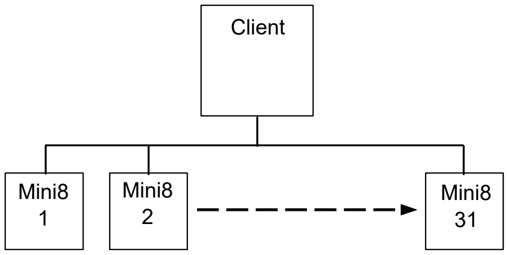


Figure 76 Broadcast Communications

Wiring Connections for Broadcast Communications are shown in "Wiring Connections for Modbus Broadcast Communications" on page 35.

Modbus TCP Client

Overview

Modbus TCP Client is protected by feature security.

Server profiles for Eurotherm products (EPCx (EPC3000 & EPC2000 generic), ePack, 3200, Mini8, and ePower devices are supported for ease of configuration.

A maximum of three Modbus TCP Server devices can be configured with timeouts and retries configurable per Server.

A maximum of 100 data points are supported to be shared among the three Server devices. These data points can be configured for writing to or reading from a configured Modbus Server.

Broadcast Client communications allow the Mini8 loop controllers to send a single value to any Server instruments using a Modbus broadcast using function code 6 (Write single value). This allows the Mini8 loop controller to link through digital communications with other products without the need for a supervisory PC to create a small system solution.

Example applications include multi-zone profiling applications or cascade control using a second controller. The facility provides an alternative to analog retransmission.

Configuration

Modbus Client can be configured using a PC using iTools software.

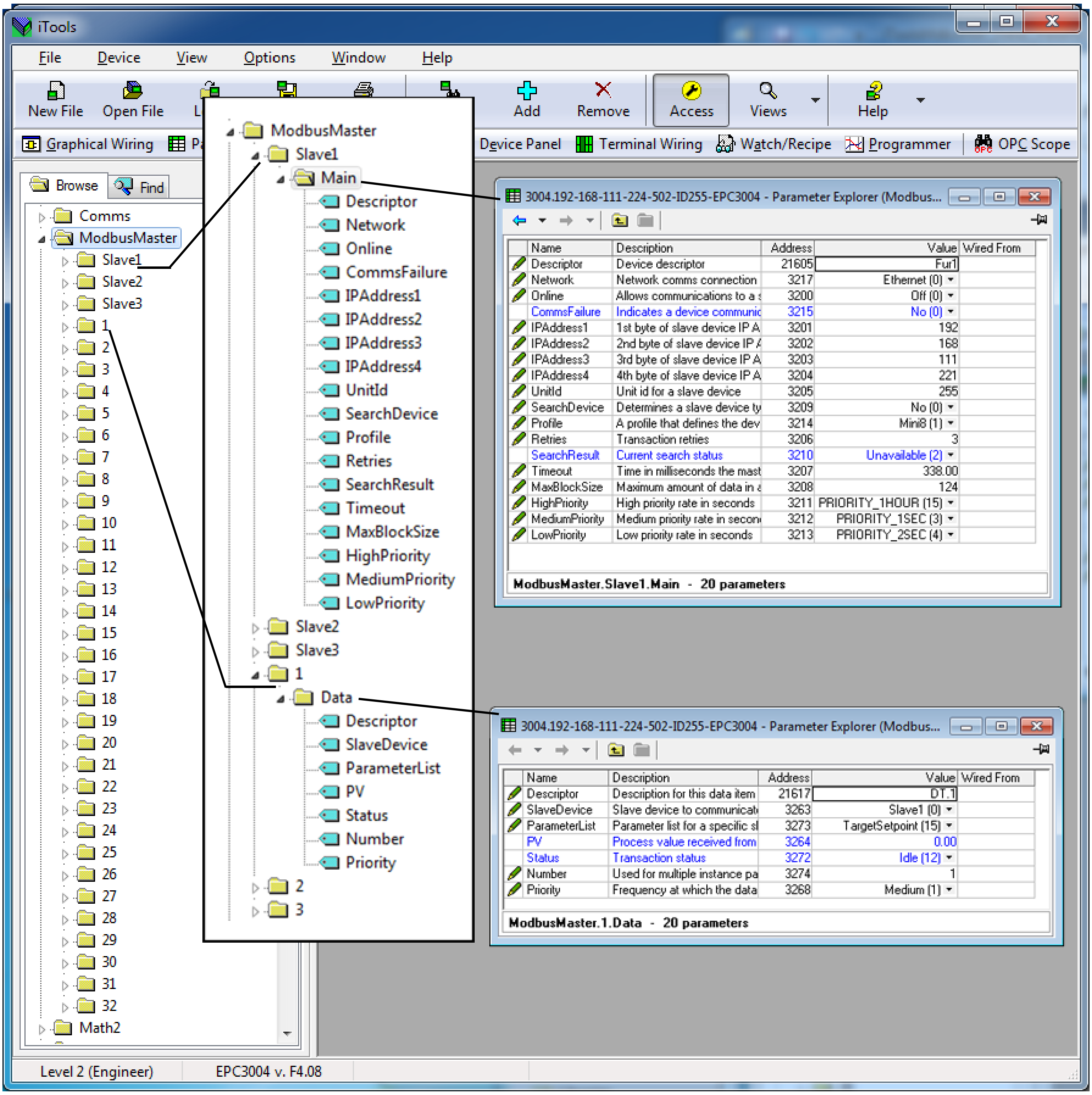
Once the Modbus Client feature is enabled via Feature Security, Comms.Option.Main.Protocol must be set to ModMstAndSlv(15). The instrument must then be restarted to reinitialise comms settings and make the ModbusMaster function block available.

Modbus Client configuration is divided into two parts:

- Setting up the Modbus Client Servers
- Defining the required server data that will be read from or written to the configured slave(s).

Notes:

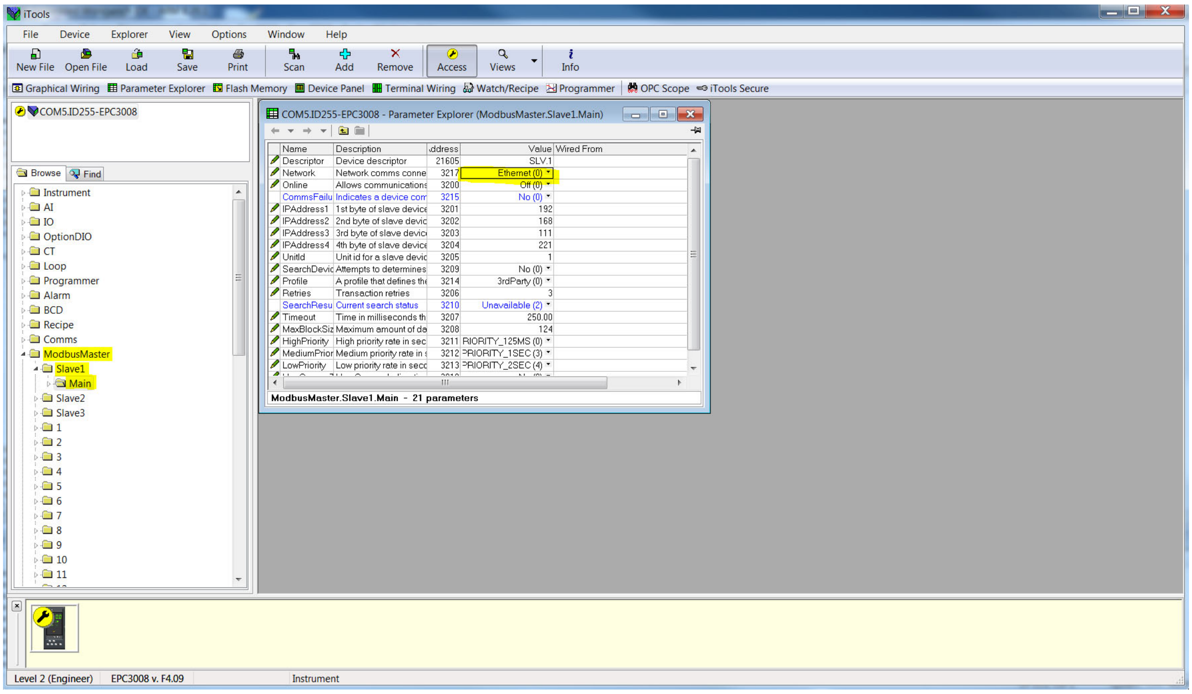
1. Server profiles are supported for some Eurotherm controllers. This simplifies the configuration and minimizes the need to know detailed data information, for example the Modbus address, data type and resolution for frequently used parameters.
2. The Network configuration of the Modbus TCP Client is the same as the Modbus TCP Server and can be found in Comms.Option.Network. Confirm that the IP address and subnet mask are configured correctly to be able to communicate with Modbus server devices within the subnet. If the server device is outside the subnet then the Comms.Option.Network.DefaultGateway must be configured correctly.


iTools
File Device View Options Window Help
New File Open File Add Remove Access Views Help
Graphical Wiring Device Panel Terminal Wiring Watch/Recipe Programmer OPC Scope
Browse Find
Comms
ModbusMaster
Slave1
Slave2
Slave3
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
Math2
ModbusMaster
Slave1
Main
Descriptor
Network
Online
CommsFailure
IPAddress1
IPAddress2
IPAddress3
IPAddress4
UnitId
SearchDevice
Profile
Retries
SearchResult
Timeout
MaxBlockSize
HighPriority
MediumPriority
LowPriority
Slave2
Slave3
1
Data
Descriptor
SlaveDevice
ParameterList
PV
Status
Number
Priority
2
3
3004.192-168-111-224-502-ID255-EPC3004 - Parameter Explorer (Modbus...
Name Description Address Value Wired From
Descriptor Device descriptor 21605 Fur1
Network Network comms connection 3217 Ethernet (0)
Online 3200 Off (0)
CommsFailure 3215 No (0)
IPAddress1 3201 192
IPAddress2 3202 168
IPAddress3 3203 111
IPAddress4 3204 221
UnitId Unit id for a slave device 3205 255
SearchDevice 3209 No (0)
Profile 3214 Mini8 (1)
Retries Transaction retries 3206 3
SearchResult Current search status 3210 Unavailable (2)
Timeout 3207 338.00
MaxBlockSize 3208 124
HighPriority High priority rate in seconds 3211 PRIORITY_1HOUR (15)
MediumPriority 3212 PRIORITY_1SEC (3)
LowPriority Low priority rate in seconds 3213 PRIORITY_2SEC (4)
ModbusMaster.Slave1.Main - 20 parameters
Descriptor Description for this data item 21617 DT.1
SlaveDevice 3263 Slave1 (0)
ParameterList 3273 TargetSetpoint (15)
PV 3264 0.00
Status Transaction status 3272 Idle (12)
Number 3274 1
Priority 3268 Medium (1)
ModbusMaster.1.Data - 20 parameters
Level 2 (Engineer) EPC3004 v. F4.08

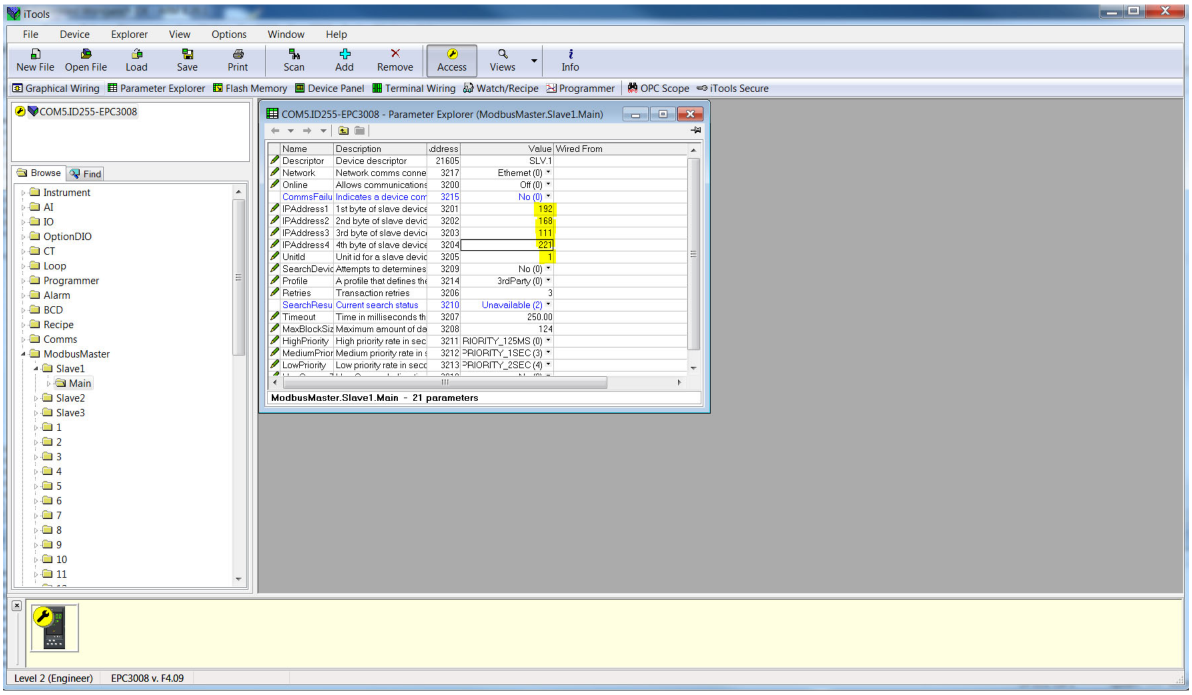
Configuring Modbus Servers

To configure communications to Modbus servers, proceed as follows:

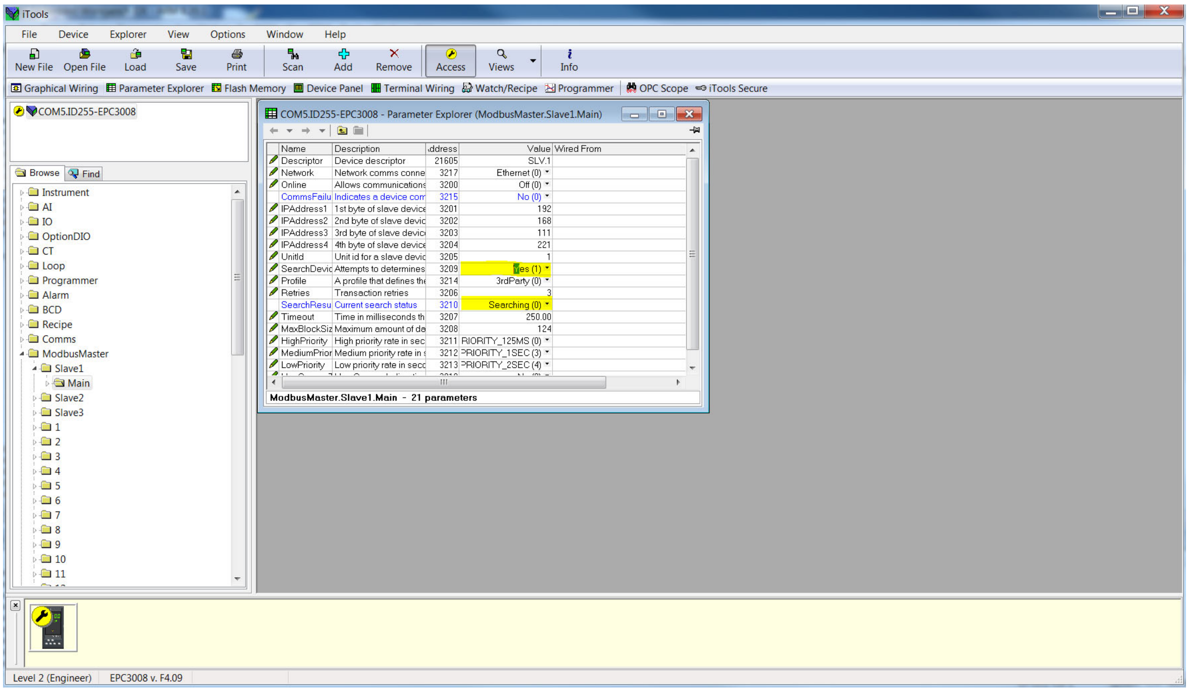
1. From iTools, place the instrument in Config mode and open ModbusMaster>Slave1>Main to configure the first server.



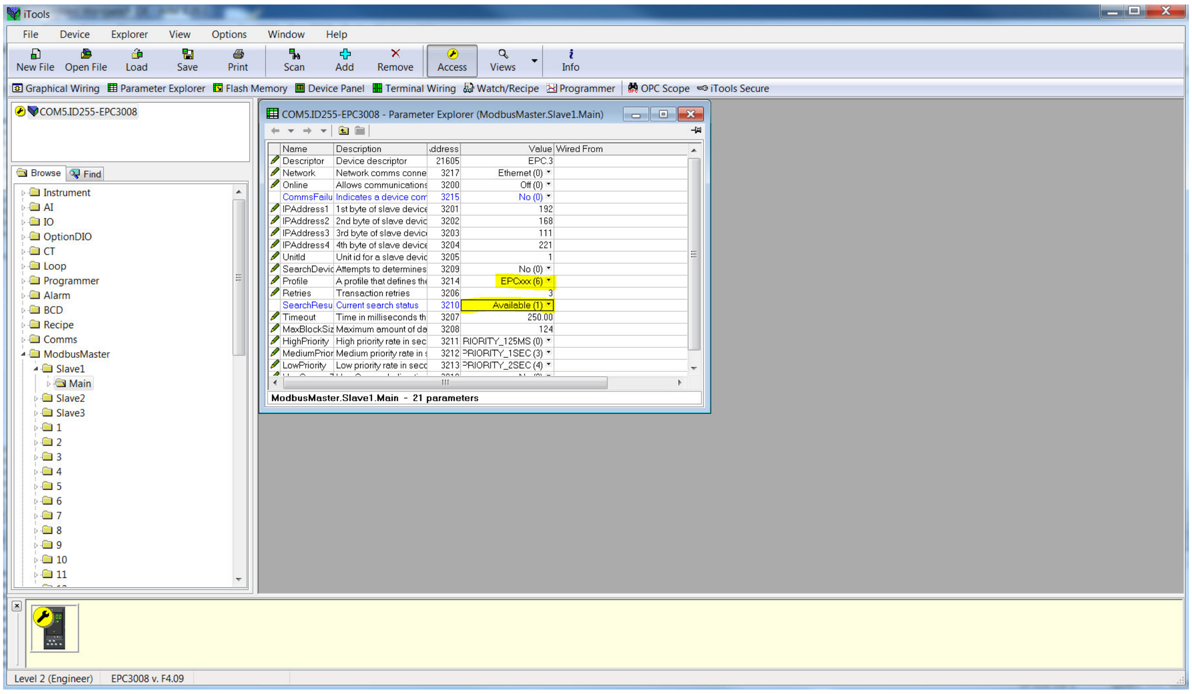
2. Configure the slave's (server's) IP Address and unit ID.



3. You can now check if the device is online via the “Search device” parameter by setting its value to “Yes”. The search status should change to “Searching(0)”.

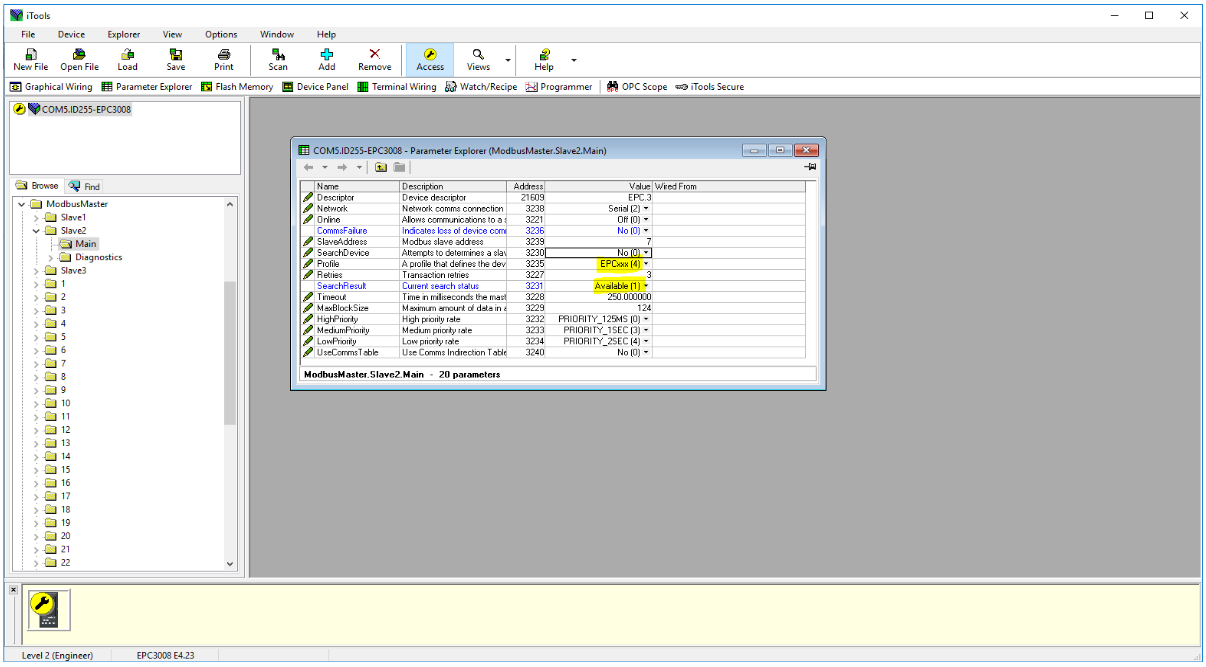


4. If the Modbus server is online then the search result will be “Available(1)” otherwise the result will be “Unreachable(3). If it is a Eurotherm instrument with a supported profile, the “Profile” parameter will display the Modbus server’s profile otherwise it will display “3rdParty(0)”.



5. We will now configure a second slave (server) (Slave2) using the same steps as described previously.

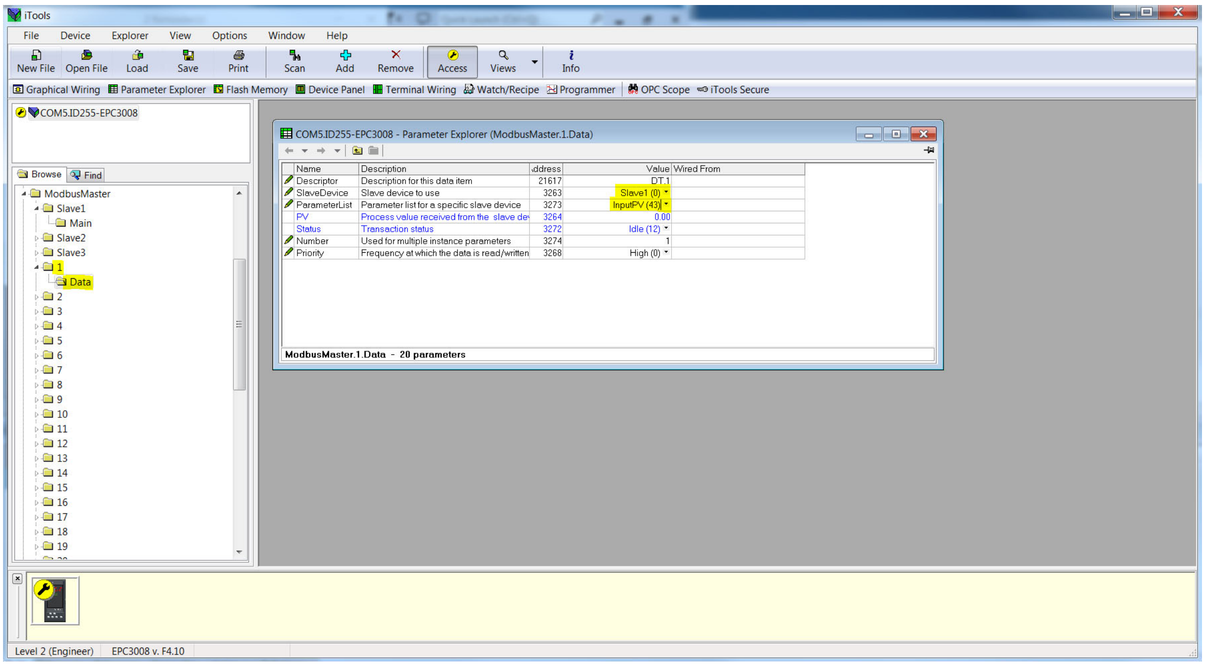
Note: Changes to the slave (server) profile will default previous data configured to be read from or written to the slave (server).



Data Configuration for Cyclic Read/Writes

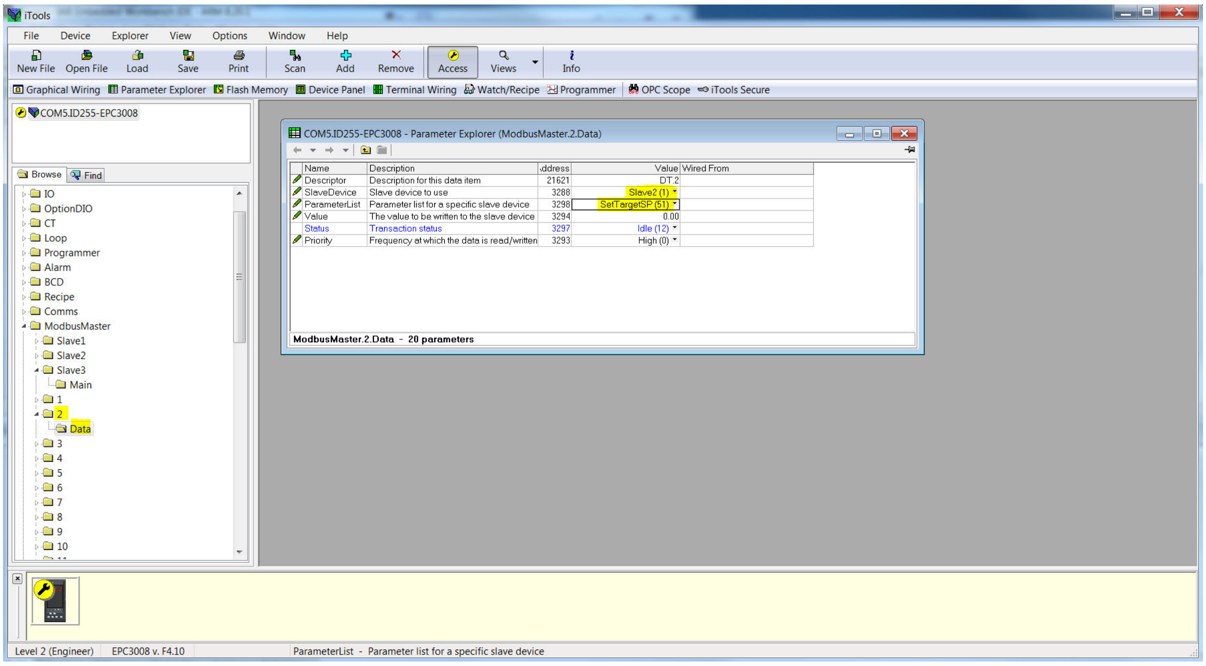
To configure data for cyclic read/writes:

1. A maximum of 100 data points can be configured. These data points can be shared among all three slaves (servers) or it can be used for a single slave (server).
2. For a slave (server) with a known profile, a data read can be configured by selecting the slave (server) and then select the required parameter from the Parameter list drop-down box. The register address, function code, data type and priority for the parameter will be automatically configured. You still have the option to change the recommended priority.

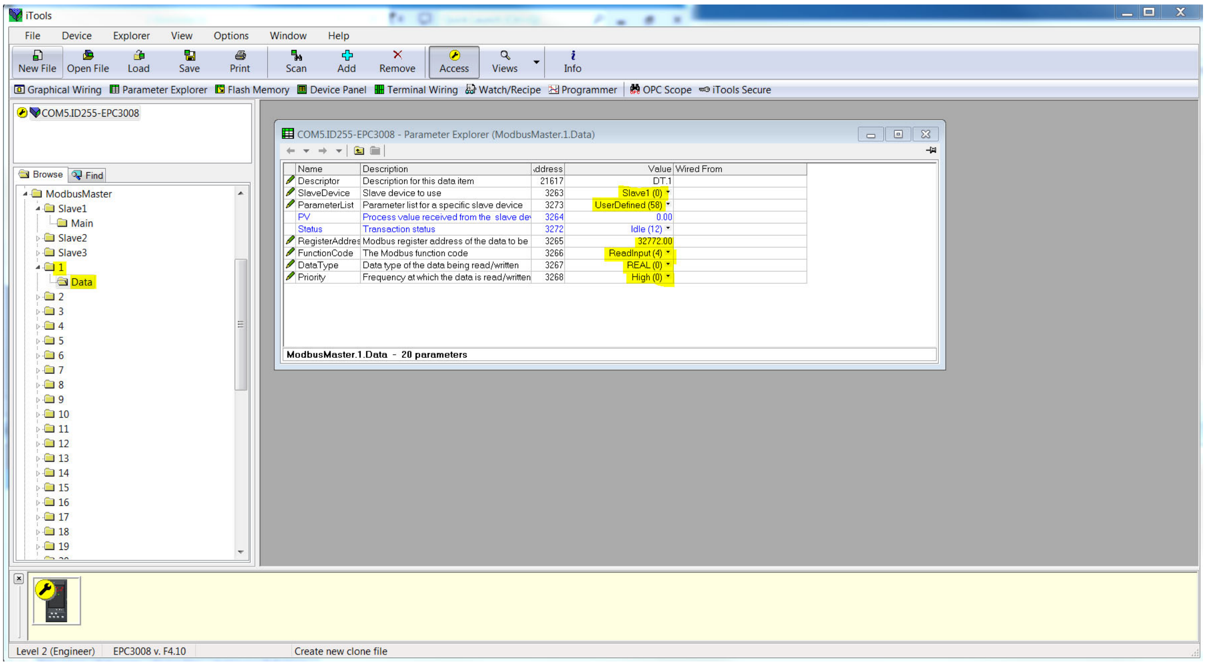


3. To configure a write for a known profile, select parameter to write from the Parameter List drop-down box.

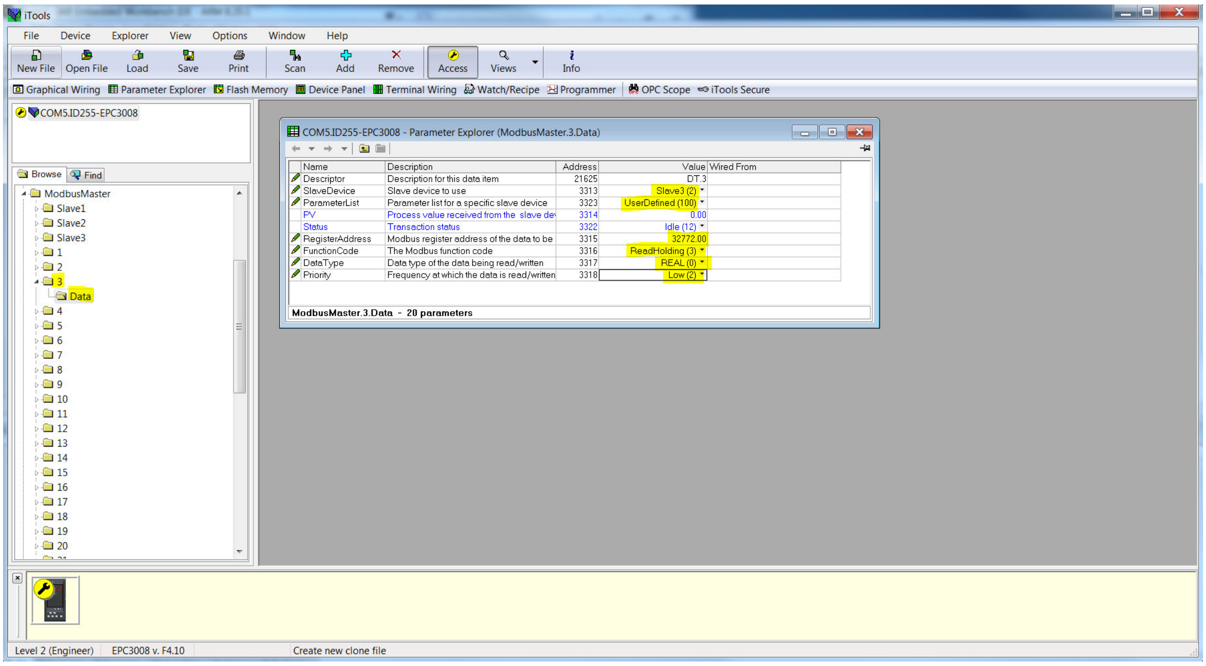
Note: The 'Value' parameter is usually wired from the source parameter of the values to be written to the slave (server).



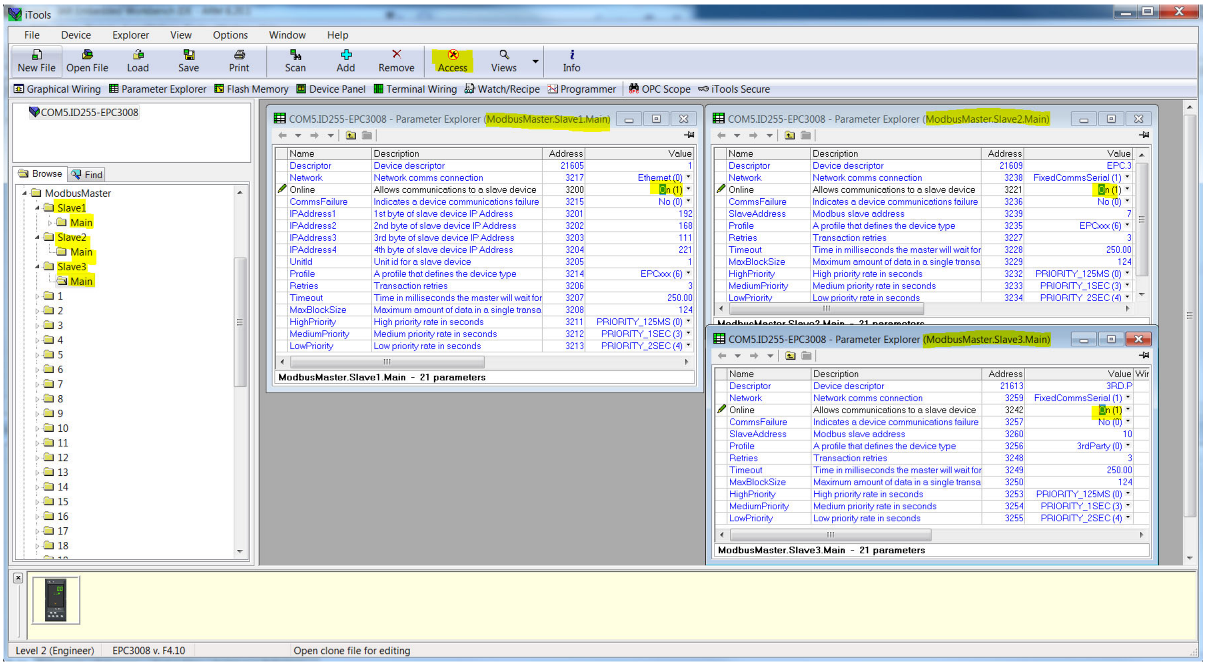
4. For a parameter that is not on the Parameter List. The data configuration has to be done manually. Select "UserDefined" from the Parameter List and configure the register address, function code, the data type and priority of data read/write.



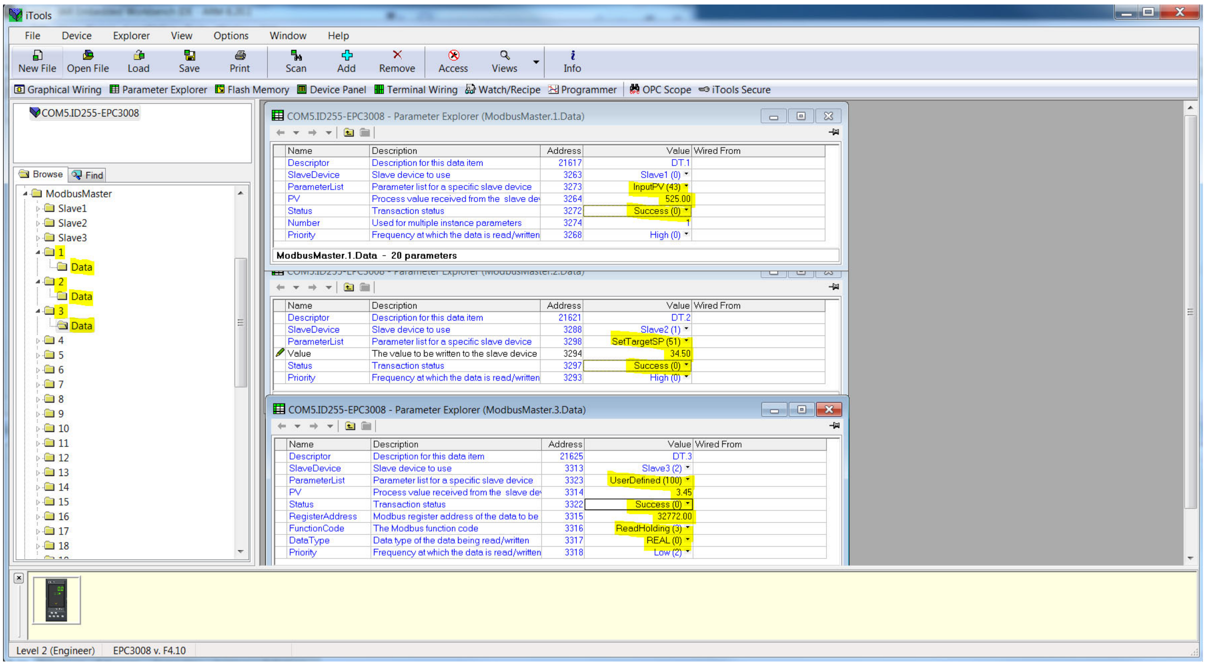
5. For a third party slave (server) (unsupported profile), select “UserDefined” from the Parameter List drop-down and configure the register address, function code, the data type and priority of data read/write.



6. To start cyclic communications to the slaves. Take the Modbus Client device out of Config mode and set the Online parameter for each of the servers.



7. The data read and write status should succeed if the wiring, comms configuration, slave (server) configuration and data configuration are correct. The PV read will be shown in the Data PV parameter.



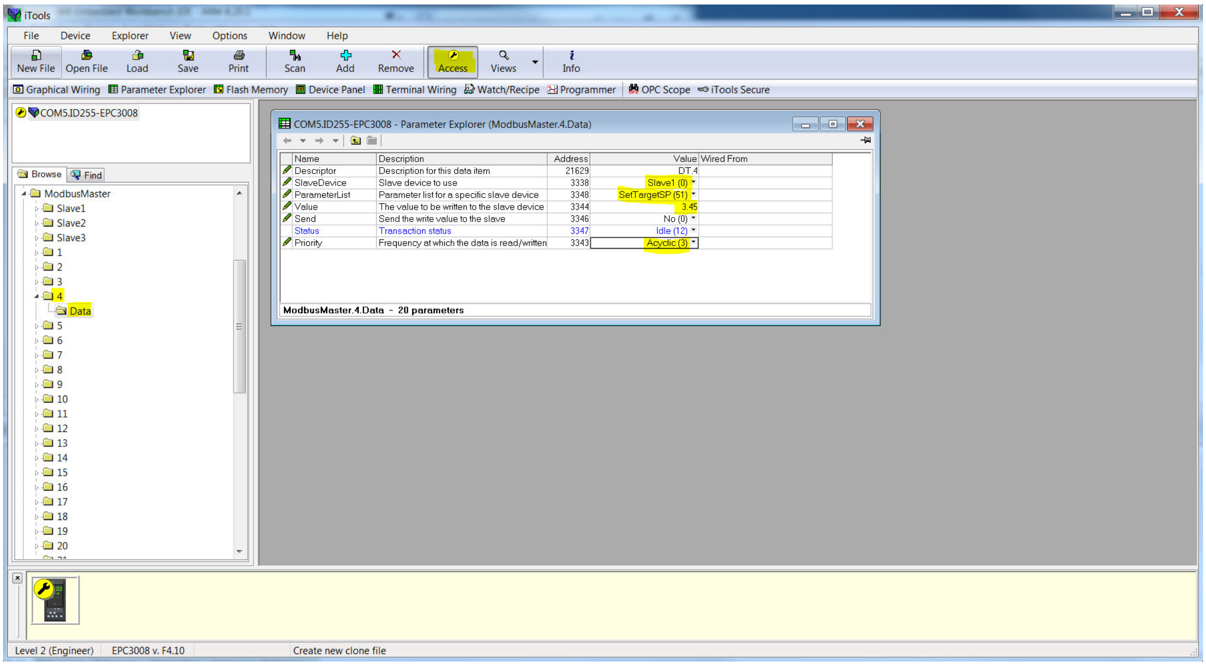
Data Configuration for Acyclic Data Writes

To configure data for Acyclic Data writes:

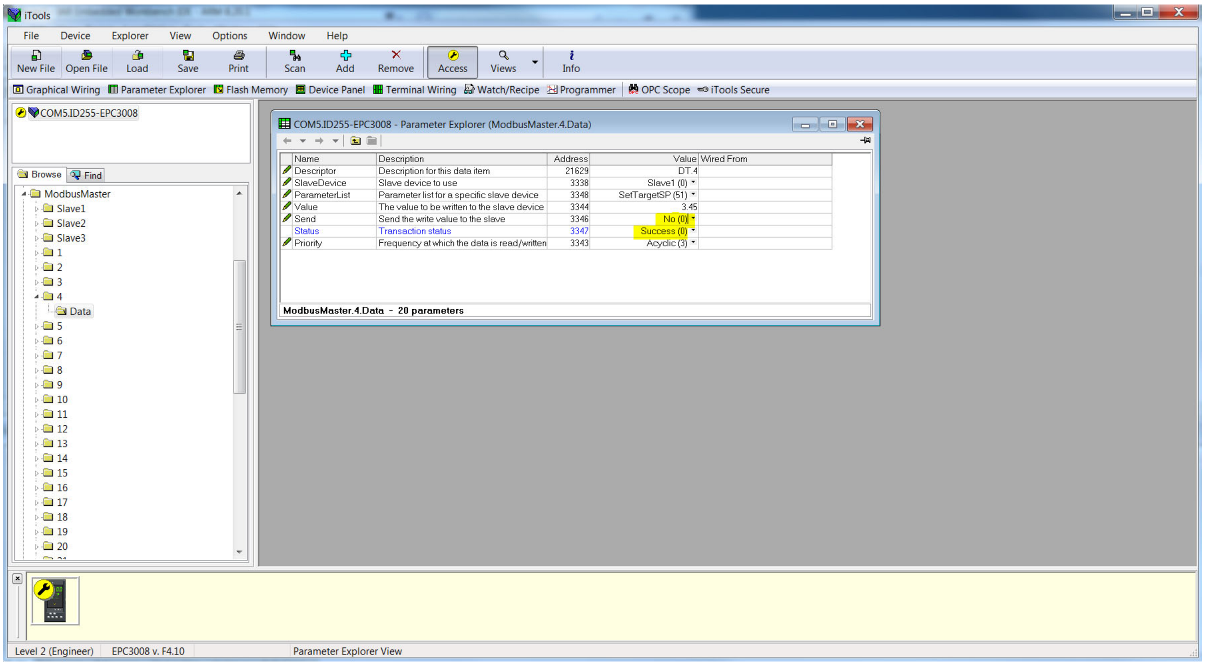
1. Place the Modbus Client device in Configuration mode.

Note: Cyclic communications to all servers will stop in Configuration mode. We can only set the server online parameter in Operator mode.

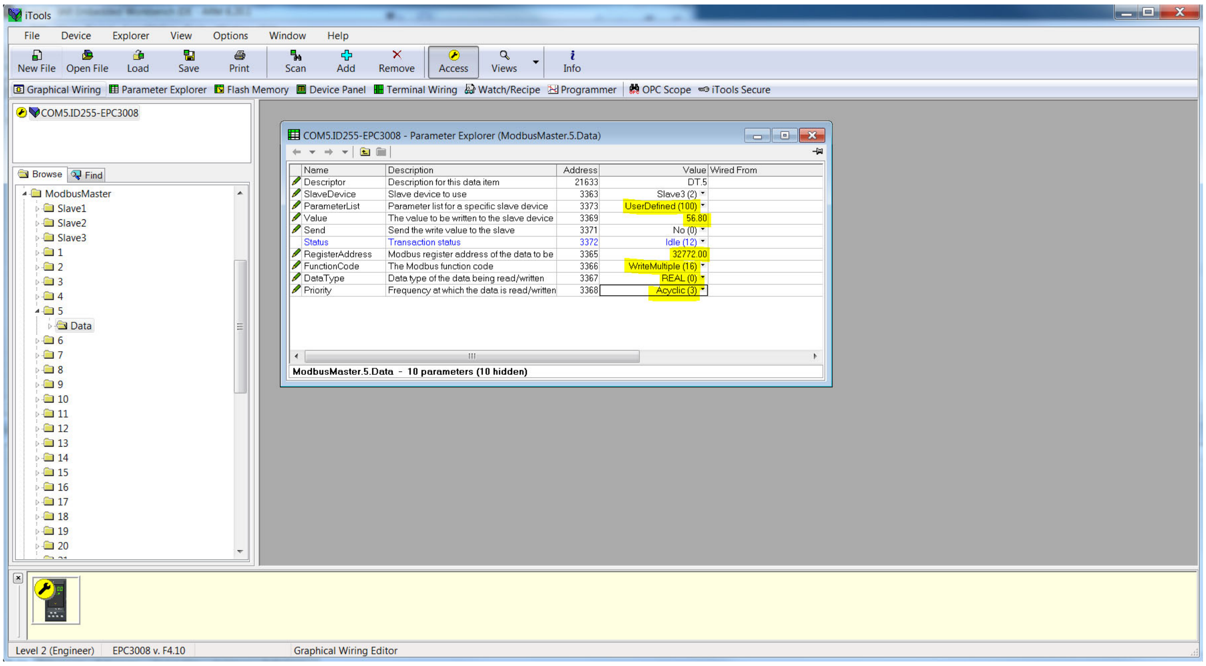
2. For a supported server profile select the server and parameter to write to as well as the value to write and then set the Priority to “Acyclic(3)”.



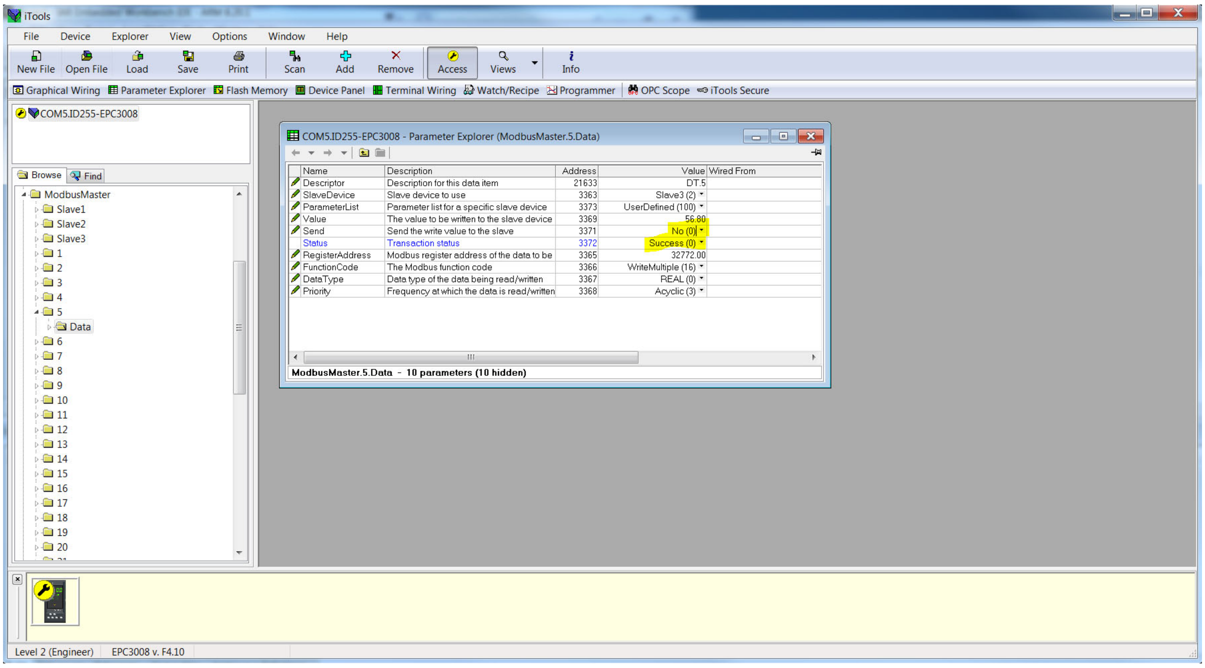
3. To send a write request, set the “Send” parameter. The Status will go to “Pending(13)” briefly before going to “Success” when the parameter has been written. If the write has failed then the Status will show the reason for the failure.



4. For an unsupported slave (server) profile (Third party) select the slave (server), select “UserDefined” from the Parameter List drop-down and configure the register address, function code (must be a write), the data type, the value to write and then set the Priority to “Acyclic(3)”.



5. To send a write request, set the “Send” parameter. The Status will go to “Pending(13)” briefly before going to “Success” when the parameter has been written. If the write has failed then the Status will show the reason for the failure.

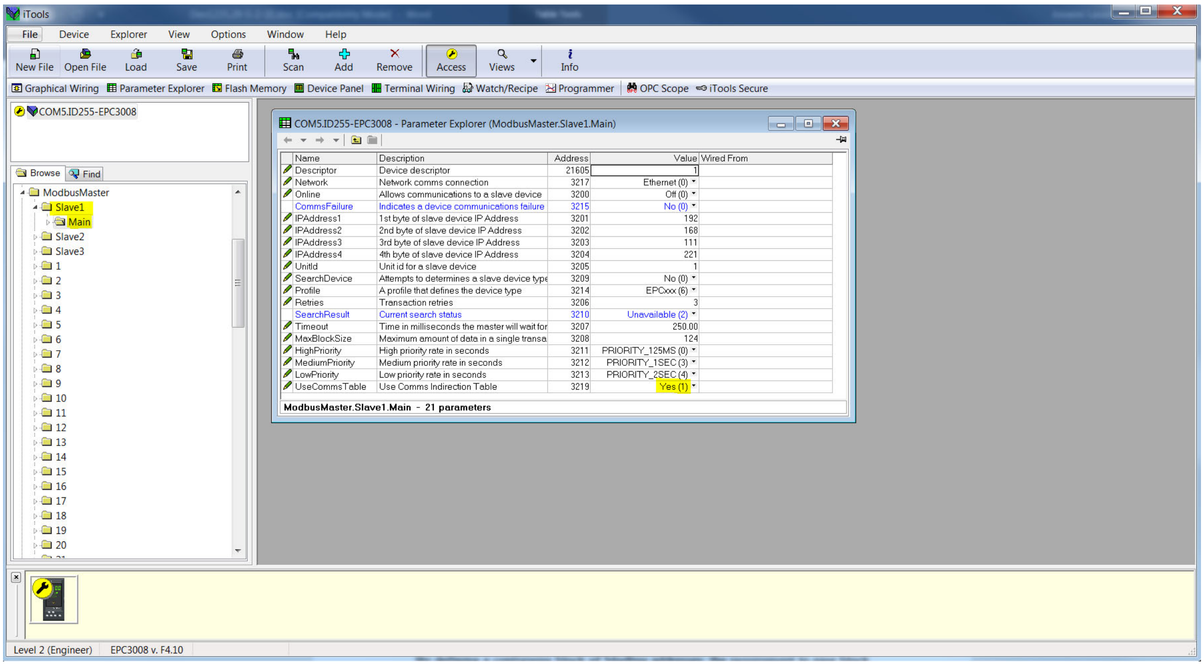


Accessing Modbus Client Data from Modbus Indirection Table

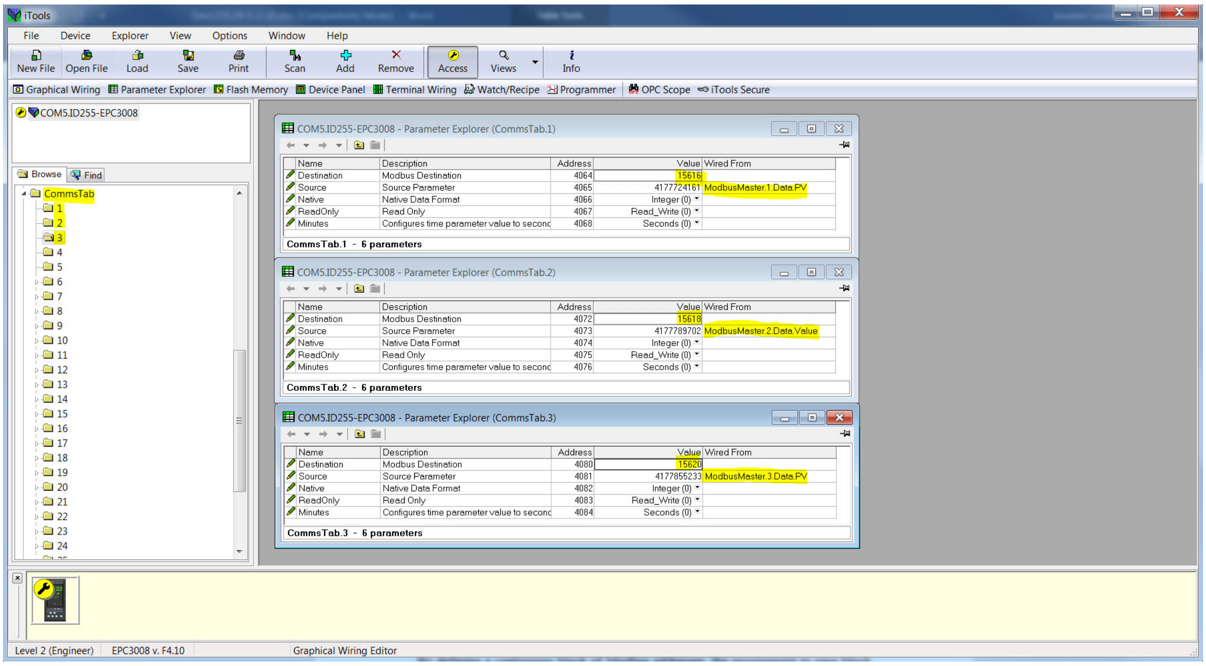
To allow efficient reads from and writes to Modbus Client data, CommsTab Function Block can be used to map Modbus Client data into a contiguous block of Modbus addresses in the range:

15360(0x3C00) to 15615(0x3CFF)

1. Modbus Client data can be auto-configured to be accessible from the Modbus Indirection table by placing the Modbus Client device into Configuration mode and setting the UseCommsTable parameter from any one of the server configuration windows and then taking the Modbus Client device out of Configuration mode to initialize the CommsTab Function Block settings.

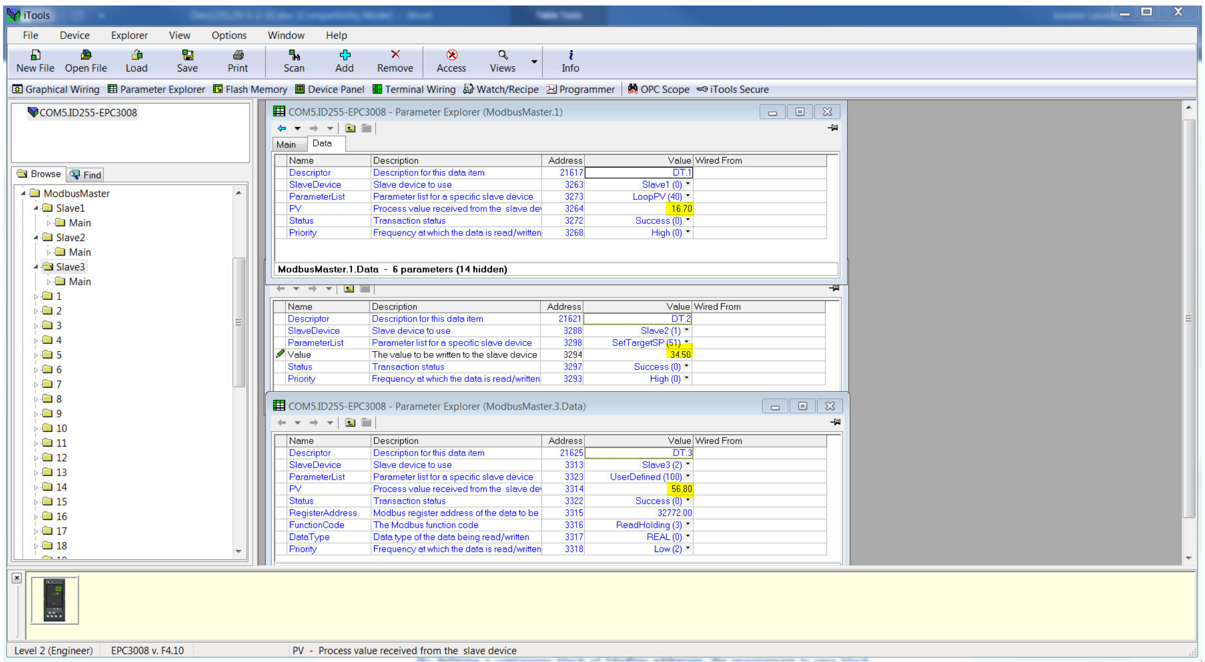


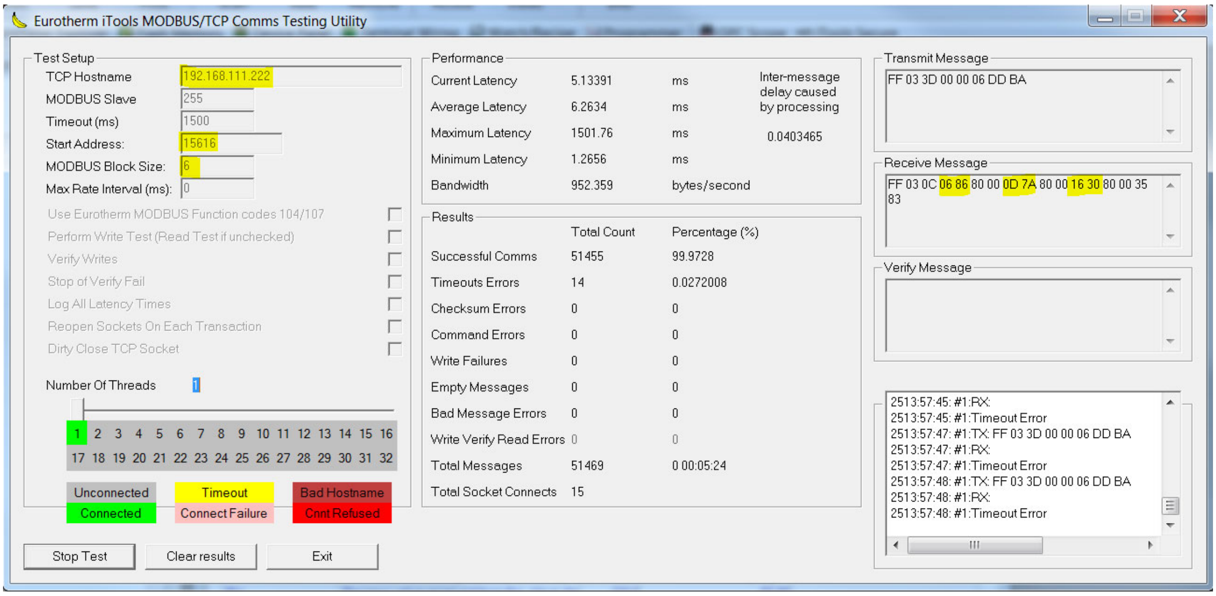
2. In Operator mode, the CommsTab Function Block should now show every configured Modbus Client data. The user can then change Native, ReadOnly and Minutes parameters from default to configure how the data is presented from the Modbus indirection table.



3. Screenshots below show Modbus Client data auto-configured to appear at the Modbus Indirection table and the values read by a 3rd party Modbus Client from our Modbus Client device:

Third party Modbus TCP Client read data	Modbus Master device data
0x0686	16.70
0x0D7A	34.50
0x1630	56.80





Note: There are 250 parameters available for configuration in the CommsTab Function Block. It is left to the user to partition the Modbus Indirection table for reads and writes for efficient data access.

Comms Indirection Table

Mini8 loop controllers make a fixed set of parameters available over digital communications using Modbus addresses. This is known as the SCADA Table. The SCADA Modbus address area is 0 to 15615 (0x3CFF).

Commstab function block allows a Source parameter value to be available (read/write) from a Destination Modbus address.

The following parameters however cannot be set as a Destination Modbus address:

- Instrument Number
- Instrument Type
- Instrument Firmware Version
- Company ID
- Feature Security Words

The following contiguous Modbus addresses have been reserved for use by the Commstab function block. By default the addresses have no associated parameters:

Modbus Range (Decimal)	Modbus Range (Hex)
15360 to 15615	3C00 to 3CFF

Modbus Parameters

The following table shows the parameters available for Modbus.

Block – Comms		Sub-block: FC (Field Communications)			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Comms Module Identity	Modbus		Modbus	Read only
Protocol	Digital communications protocol	Modbus		Modbus	Read only
Baud	Communications baud rate	Modbus: 4800, 9600 or 19k2 (19200)		9600	Conf
Parity	Communications parity	None Even Odd	No parity Even parity Odd parity	None	Conf
Address	Instrument address	1 to 254 Only writeable if DIP switches are set to Off.		1	Oper
Delay	Comms delay	No Yes	No delay Fixed delay. This inserts a delay between Rx and Tx to allow the drivers used by intelligent EIA-232/EIA-485 converters sufficient time to switch over.	No	Conf
Broadcast Enabled	To enable broadcast client (master) communications. (See "Broadcast Client" on page 131)	No Yes	Not enabled Enabled	No	
Broadcast Address	Address of the parameter being written to slaves.	0 to 32767	See "Modbus SCADA Table" on page 351 for addresses of all Mini8 loop controller parameters.	0	Only shown if Broadcast is enabled.
Broadcast Value	Value to be sent to instruments on the network. This would normally be wired to a parameter within the client (master).	Range of the parameter wired. In the case of a Boolean the value will be 0 or 1.		0.00	Only shown if Broadcast is enabled.
WDFlag	Network watchdog flag	Off On	This flag is ON when the Network communications have stopped addressing this instrument for longer than the Timeout time. It will be set by the Watchdog process and may be cleared Automatically or Manually according to the value of the Watchdog Action parameter.		
WDAct	Network watchdog action. The Watchdog Flag may be cleared Automatically upon reception of valid messages or Manually by a parameter write or a wired value.	Man	The Watchdog Flag must be cleared manually - either by a parameter write or a wired value.		Conf
		Auto	The Watchdog Flag will be automatically cleared when the Network Comms resume - according to the value in the Recovery Timer.		
WDTimeout	Network Watchdog Timeout If the Network communications stop addressing the instrument for longer than this value, the Watchdog Flag will become active.	h:m:s:ms A value of 0 disables the watchdog			Conf
TimeFormat	Time format	0 1 2 3	milliseconds seconds minutes hours		Conf

Ethernet (Modbus TCP)

Instrument setup

It is recommended that you setup the communications settings for each instrument before connecting it to any Ethernet network. This is not essential but network conflicts may occur if the default settings interfere with equipment already on the network. By default, the instruments are set to a fixed IP address of 192.168.111.222 with a default SubNet Mask setting of 255.255.255.0.

IP Addresses are usually presented in the form "xxx.xxx.xxx.xxx". In the instrument Comms folder each element of the IP Address is shown and configured separately.

"IP address 1" relates to the first set of three digits, IP address 2 to the second set of three digits and so on. This also applies to the SubNet Mask, Default Gateway and Preferred master IP Address.

Each Ethernet module contains a unique MAC address, normally presented as a 12-digit hexadecimal number in the format "aa-bb-cc-dd-ee-ff".

In the Mini8 loop controllers MAC addresses are shown as six separate decimal values in iTools. MAC1 shows the first pair of digits in decimal, MAC2 shows the second pair of digits and so on.

Note: Mini8 loop controller is shipped with Static IP, and switches set to 01.

Dynamic Host Configuration Protocol (DHCP) Settings

IP addresses may be ‘static’ – set by the user, or dynamically allocated by a DHCP server on the network.

This is set by the address switch, situated at the bottom of the Comms slot, as follows:

- 00 = DHCP (Dynamic Address) enabled
- 01 to FE = Static IP (use most recently obtained/configured address)
- FF = Reserved

If the IP Addresses are to be dynamically allocated the server uses the instrument MAC address to uniquely identify it.

For fixed IP Addresses set the IP address as well as the SubNet Mask. These must be configured into the instrument using iTools. Remember to note the allocated addresses.

Fixed IP Addressing

The Address Switch must be set to a non-zero value. In the "Comms.FC.Network" folder of the instrument the "IPMode" parameter will be set to "Static". Set the IP address and SubNet Mask as required.

Dynamic IP Addressing

The Address Switch must be set to zero. In the "Comms.FC.Network" folder of the instrument the "IPMode" parameter will be set to "DHCP". Once connected to the network and powered, the instrument will acquire its "IP address", "SubNet Mask" and "Default gateway" from the DHCP Server and display this information within a few seconds.

Default Gateway

The "Comms" tab also includes configuration settings for "Default Gateway", these parameters will be set automatically when Dynamic IP Addressing is used. When fixed IP addressing is used these settings are only required if the instrument needs to communicate wider than the local area network.

Preferred Master

The "Comms" tab also includes configuration settings for "Preferred Master". Setting this IP address to the IP Address of a particular PC will reserve one of the four available Ethernet sockets for that PC (reducing the number of available sockets for anonymous connections to three).

- 00 = DHCP (Dynamic Address) enabled
- 01 to FE = Static IP (use most recently obtained/configured address)
- FF = Reserved

iTools Setup

iTools configuration package, version V9.85 or later, may be used to configure Ethernet communications.

The following instructions configure Ethernet.

To include a Host Name/Address within the iTools scan:

1. Ensure iTools is NOT running before taking the following steps.
2. Within Windows, click ‘Start’, then ‘Settings’, then ‘Control Panel’.
3. In control panel select ‘iTools’.
4. Within the iTools configuration settings select the ‘TCP/IP’ tab.
5. Click the ‘Add’ button to add a new connection.
6. Enter a name for this TCP/IP connection.
7. Click the ‘Add’ button to add the IP address of the instrument in the ‘Host Name/ Address’ section.
8. Click ‘OK’ to confirm the new IP Address you have entered.
9. Click ‘OK’ to confirm the new TCP/IP port you have entered.
10. You should now see the TCT/IP port you have configured within the TCP/IP tab of the iTools control panel settings.

iTools is now ready to communicate with an instrument at the IP Address you have configured.

Ethernet Parameters

These are listed in the ‘Comms’ > ‘FC’ list in iTools.

Block - Comms		**Sub-block: FC**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Interface	Identifies that the Ethernet comms module is fitted.	Ethernet		Ethernet	Read only
Protocol	Digital communications protocol	ModbusSlave		ModbusSlave	Read only
Status	Ethernet Network Status	Running Offline Init Ready	Network connected and working Network not connected or working Network initialising Network ready to accept connection		Read only
WDTimeout	Network Watchdog Timeout	If the Network communications stop addressing the instrument for longer than this value, the Watchdog Flag will become active. h:m:s:ms A value of 0 de-activates the watchdog			Conf
WDAction	Network Watchdog Action	Man	The Watchdog Flag must be cleared manually - either by a parameter write or a wired value.		Conf
		Auto	The Watchdog Flag will be automatically cleared when the Network Comms resume - according to the value in the Recovery Timer.		
WDRecovery	Network Watchdog Recovery	When the Watchdog Action is set to Auto, this timer determines the delay after reception recommences before the Watchdog Flag is cleared. A value of 0 will reset the Watchdog flag upon the first valid message received. Other values will wait for at least 2 valid messages to be received within the set time before clearing the Watchdog flag.			Conf
WDFlag	Network Watchdog Flag	Off On	This flag is ON when the Network communications have stopped addressing this instrument for longer than the Timeout time. It will be set by the Watchdog process and may be cleared Automatically or Manually according to the value of the Watchdog Action parameter		
Delay	Comms Delay	0 1	No Yes	0	Conf
TimeFormat	Time format	0 1 2 3	milliseconds seconds minutes hours		Conf
AutoDiscovery	Auto Discovery	0 1	Off On	0	Conf
IP Mode	IP Mode	0 1	Static DHCP		Read-only
IP Address 1	1st Byte IP address	IP address format is xxx.xxx.xxx.xxx. 1^{st} Byte. 2^{nd} Byte. 3^{rd} Byte. 4^{th} Byte. Range 0 to 255		192	Conf
IP Address 2	2nd Byte IP address			168	
IP Address 3	3rd Byte IP address			111	
IP Address 4	4th Byte IP address			222	

Block - Comms		Sub-block: FC		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Subnet Mask 1	1st Byte Subnet Mask	Subnet mask format is xxx.xxx.xxx.xxx. 1st Byte. 2nd Byte. 3rd Byte. 4th Byte. Range 0 to 255	255	Conf
Subnet Mask 2	2nd Byte Subnet Mask		255	
Subnet Mask 3	3rd Byte Subnet Mask		255	
Subnet Mask 4	4th Byte Subnet Mask		0	
Default Gateway 1	1st Byte Default Gateway	Default gateway format is xxx.xxx.xxx.xxx. 1st Byte. 2nd Byte. 3rd Byte. 4th Byte. Range 0 to 255	0	Conf
Default Gateway 2	2nd Byte Default Gateway			
Default Gateway 3	3rd Byte Default Gateway			
Default Gateway 4	4th Byte Default Gateway			
MAC1	MAC address 1	A unique MAC address is allocated to every Ethernet device MAC addresses are six bytes in length and are shown in HEX format, for example: AA-BB-CC-DD-EE-FF 1st Byte 2nd Byte 3rd Byte 4th Byte 5th Byte 6th Byte	0	Read only
MAC2	MAC address 2			
MAC3	MAC address 3			
MAC4	MAC address 4			
MAC5	MAC address 5			
MAC6	MAC address 6			
BroadcastStormActive	Broadcast Storm Active	0 No 1 Yes		Conf
RateProtectionActive	Rate Protection Active	0 No 1 Yes		Conf
PrefMasterIPAddress1	1st byte of preferred master IP address	IP address format is xxx.xxx.xxx.xxx. 1st Byte. 2nd Byte. 3rd Byte. 4th Byte. Range 0 to 255		
PrefMasterIPAddress2	2nd byte of preferred master IP address			
PrefMasterIPAddress3	3rd byte of preferred master IP address			
PrefMasterIPAddress4	4th byte of preferred master IP address			

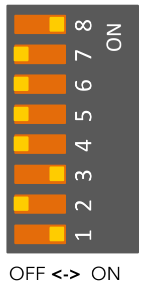
CommsTab

See "Comms Table" on page 351 for a description of the CommsTab.

DeviceNet

Only two parameters have to be set on the Mini8 loop controller for use with DeviceNet - Baud rate and address. Both can be set on the hardware address switch situated under the DeviceNet connector. Each Mini8 loop controller must have a unique address on the DeviceNet network and all units must be set to the same Baud rate. The switch gives addresses from 0 to 63.

Sw	OFF	ON
8	Baud rate	Baud rate
7	Baud rate	Baud rate
6	-	Address 32
5	-	Address 16
4	-	Address 8
3	-	Address 4
2	-	Address 2
1	-	Address 1



Example shows 500k baud rate and address 5

Address 0 is a valid DeviceNet address, but Mini8 loop controller addresses can be set via iTools, when all switches are set to 0.

Sw	Baud rate		
	125k	250k	500k
8	OFF	OFF	ON
7	OFF	ON	OFF

Note: Use a Baud rate of 500k unless the total length of the DeviceNet network is longer than 100m (328ft).

Enhanced DeviceNet Interface

See also "Electrical Connections for Enhanced DeviceNet Interface" on page 39. In this version of DeviceNet the slider switch is replaced by rotary BCD switches to set Node ID (address) and Baud Rate.

Address Switch

The Node ID (address) is set via two BCD rotary switches, one for each digit.

For example, an address of 13 is configured by setting the MSD to 1 and LSD to 3.

Valid DeviceNet address range is 0 - 63. If the switches are set in the range 64 - 99 the value will be ignored and the node address will be configured by the Mini8 controller via iTools.

When the address is changed the DeviceNet interface will automatically restart.

Baud Switch

The baud rate is selected by a single BCD rotary switch, and can be set to 125K, 250K or 500K.

The ‘Prog’ position is selected when it is required to upgrade the Mini8 controller firmware.

The O/R position is selected when it is required to set Baud Rate using iTools configuration software.

When the baud rate is changed or the ‘Prog’ position is selected power cycle the instrument to activate the change.

Check that the switch is set to valid positions as marked on the panel.

Switch Position in iTools

The value of the Baud Rate and Address is returned so that it can be read by iTools.

Note: If the DeviceNet network is unpowered for any reason, any changes to the Baud Rate and Address will NOT be seen in iTools even though the Mini8 controller is powered and communicating normally via the CC port or config clip.

DeviceNet Parameters

Block – Comms		Sub-block: FC (Field Communications)			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ident	Comms Module Identity	DeviceNet	DeviceNet Enhanced	DeviceNet	Read only
Protocol	Digital communications protocol	DeviceNet		DeviceNet	Read only
Baud	Communications baud rate	125k, 250k, 500k		125k	Conf
Address	Instrument address	0 to 63 Only writeable if DIP switches are set to Off.		1	Oper
Status	Comms network status	Offline	Network Offline		Read only
		Init	Network Initialising		
		Ready	Network ready to accept connection		
		Running	Network connected and running		
		Online	The device is on line and has connections in the Established state.		
		IO Timeout	One or more IO connections have timed out.		
		Link fail	Critical link issue: a comms issue has been detected, making the module incapable of communicating.		
		Comms fault	Comms port is in the 'faulted' condition and has accepted an 'Identify Comms Fault' Request		
WDFlag	Network watchdog flag	Off On	This flag is ON when the Network communications have stopped addressing this instrument for longer than the Timeout time. It will be set by the Watchdog process and may be cleared Automatically or Manually according to the value of the Watchdog Action parameter.		
WDAction	Network watchdog action. The Watchdog Flag may be cleared Automatically upon reception of valid messages or Manually by a parameter write or a wired value.	Man	The Watchdog Flag must be cleared manually - either by a parameter write or a wired value.		Conf
		Auto	The Watchdog Flag will be automatically cleared when the Network Comms resume - according to the value in the Recovery Timer.		
WDTimeout	Network Watchdog Timeout If the Network communications stop addressing the instrument for longer than this value, the Watchdog Flag will become active.	h:m:s:ms A value of 0 disables the watchdog			Conf
SafeMode Enable	'Safe Mode' Activate	Off On	If activated, 'safe-mode' will activate at power up and when the comms watchdog is latched. While in 'safe-mode', all loops will be set to manual, all powers will be set to SafeModePower value and all SPs will be set to SafeModeSP value.	Off	Conf
SafeModePower	'Safe Mode' power		When in 'safe-mode', the power output level of all loops will be set to this value.	0	Conf
SafeModeSP	'Safe Mode' setpoint		While in 'safe-mode', the Setpoint of all loops will be set to this value. It will be set immediately with no ramp or servo action.		Conf
DeviceNet Shutdown	DeviceNet Shutdown Activate	Enable Disable	If an unrecoverable issue occurs on the internal DeviceNet port, the module is able to send a DeviceNet Shutdown message. Some Clients are unable to process this message, so this parameter gives the ability for it to be deactivated.	Enable	Conf

EtherCAT

EtherCAT (Ethernet for Control Automation Technology) is an open real-time technology that realizes the specific transfer of data. It offers real-time performance and is aimed to maximize the utilization of high-speed full-duplex Ethernet data transfer through a twisted pair cable for industrial process control needs.

EtherCAT is based on the Ethernet technology and possesses advantages such as ease of implementation, cost of ownership and standardization. This makes it an excellent solution for industrial applications, to maximize the performance of control systems.

Medium access control employs the MainDevice/SubordinateDevice principle, where the MainDevice node (typically the control system) sends the Ethernet frames to the SubordinateDevice nodes, which extract data from and insert data into these frames on the fly. A complete range of topologies can be used for EtherCAT applications.

An EtherCAT segment is a single Ethernet device, from an Ethernet point of view, which receives and sends standard ISO/IEC 802-3 Ethernet frames. This Ethernet device may consist of a large number of EtherCAT SubordinateDevices, which process the incoming frames directly and extract the relevant user data, or insert data and transfer the frame to the next EtherCAT SubordinateDevice. The last EtherCAT SubordinateDevice within the segment sends the fully processed frame back, so that it is returned by the first SubordinateDevice to the MainDevice as a response frame.

This procedure utilizes the full duplex mode of Ethernet, which allows communication in both directions independently. Direct communication without a switch between a MainDevice and an EtherCAT segment consisting of one or several SubordinateDevices may be established.

EtherCAT slave (adapter) is implemented as a Mini8 gateway communications option card.

NOTICE

POTENTIAL BROADCAST STORM

EtherCAT SubordinateDevice controllers will reflect any frame back onto the network, therefore, it should not be connected to an office network as this may result in a broadcast storm.

Failure to follow these instructions can result in equipment damage.

EtherCAT Configuration

The EtherCAT Editors support the EtherCAT Semiconductor Device Profile (SDP):

- Temperature Controller Document: ETG.5003.2060 S ® V1.2.0.
- ETG.5003.2060 S ® V1.2.0 specifies the components of a semiconductor device of type Temperature Controller that will be visible over the EtherCAT network.

The following devices currently support this version of EtherCAT;

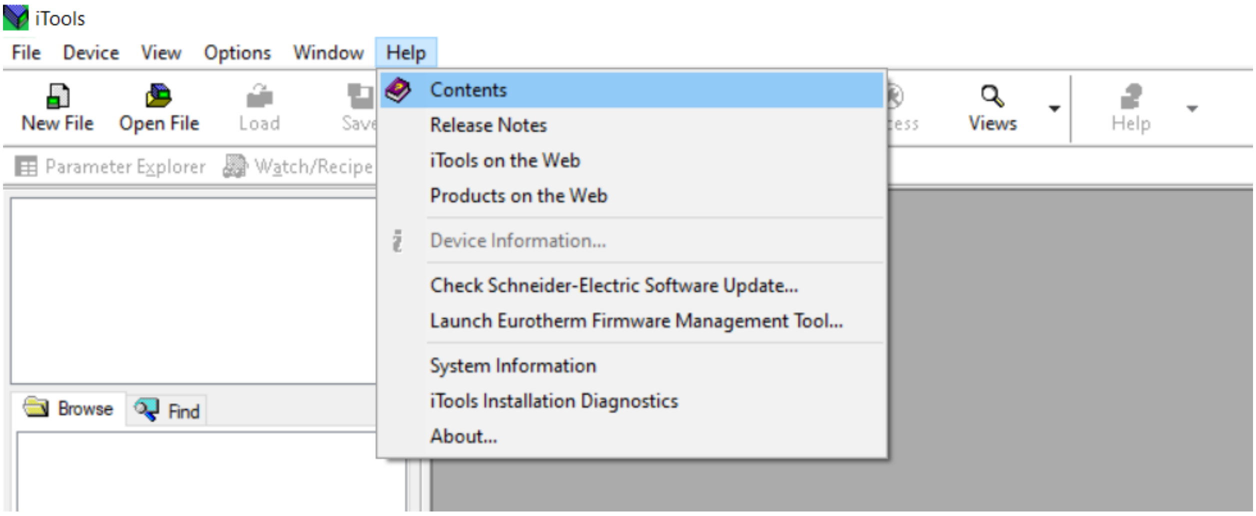
- Mini8 (Firmware version greater than V5.0)

The following editors are available to support configuration of the EtherCAT feature on the above supported devices;

- Temperature Control (TC) Editor
- Object Dictionary (OD) Editor.

Using iTools

iTools help is available that provides details on how to configure the EtherCAT feature using iTools and its associated editors.



EtherCAT Feature Switch

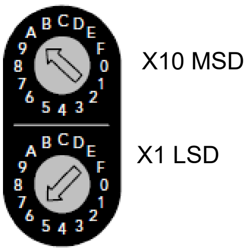


Figure 77 EtherCAT Feature Switch

The feature switch consists of two HEX rotary switches. The upper switch is the most significant digit and the lower the least significant digit.

There are two conditions where the switches can be set:

- 0x01 to 0xFE: MainDevice will use this value as the “Requesting ID”. The example shown in the diagram sets the Explicit Device ID of A6 (166), configured by setting the MSD to A and LSD to 6.
- 0x00: Invalid setting

EtherCAT Parameters

Folder – Field Comms (Comms.FC.EtherCAT)				
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
ApplicationState	EtherCAT application state	INIT (1) PREOP (2) BOOT (3) SAFEOP (4) OP (8)		Read Only
Device ID	EtherCAT device ID	As selected by the module switches		Read Only
Disabled	Disable EtherCAT application	No (0) Yes (1)	No (0)	Conf
EnableUpgrade	Enable FW Upgrade	No (0) Yes (1)	Yes (1)	Conf
ConfigAccessException	Config Access Exception Flag	No (0) Yes (1)	No (0)	Conf
ApplicationVersion	EtherCAT application version			Read Only
DefaultESIVersion	ESI version			Read Only
RxPdoSize	EtherCAT RxPDO Size			Read Only
TxPdoSize	EtherCAT TxPDO Size			Read Only
NotificationStatus	EtherCAT Notification			Read Only

Filetransfer over EtherCAT (FOE)

The Mini8 device supports Filetransfer Over EtherCAT (FOE) primarily for upgrading the firmware and Slave Information Interface (SII) binary data in the Mini8 device.

An upgrade file 'Eurotherm_MINI8_ECAT_xxx_configVxx.efw' will be available from the following site.

https://www.eurotherm.com/en/products/temperature-controllers-en/multi-loop-temperature-controllers-en/mini8-loop-controller/#download-tab

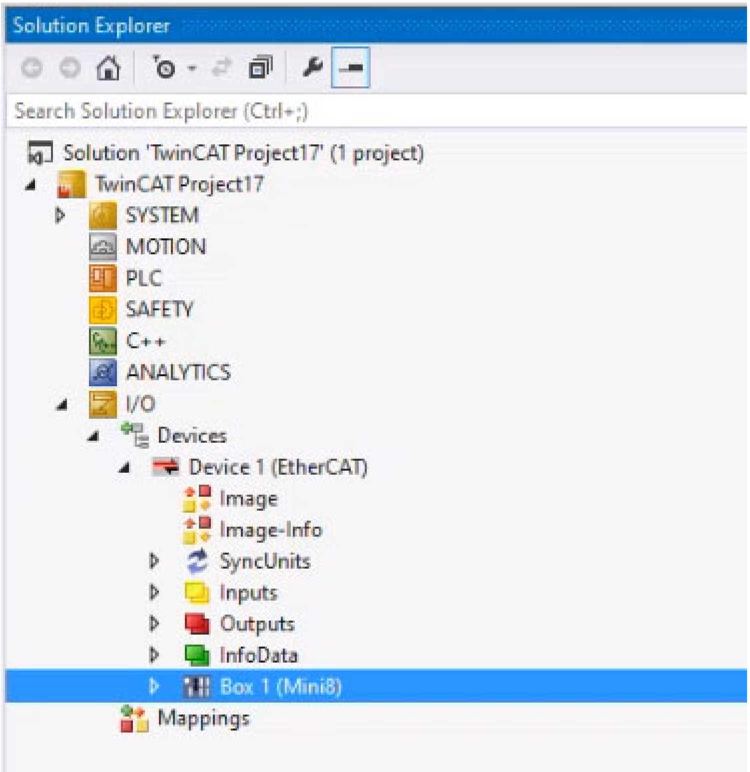
The upgrade file will contain the embedded firmware for the Mini8 and the SII file (Default 24 TCLoops) for the EtherCAT ASIC.

Both items will be downloaded during the download process.

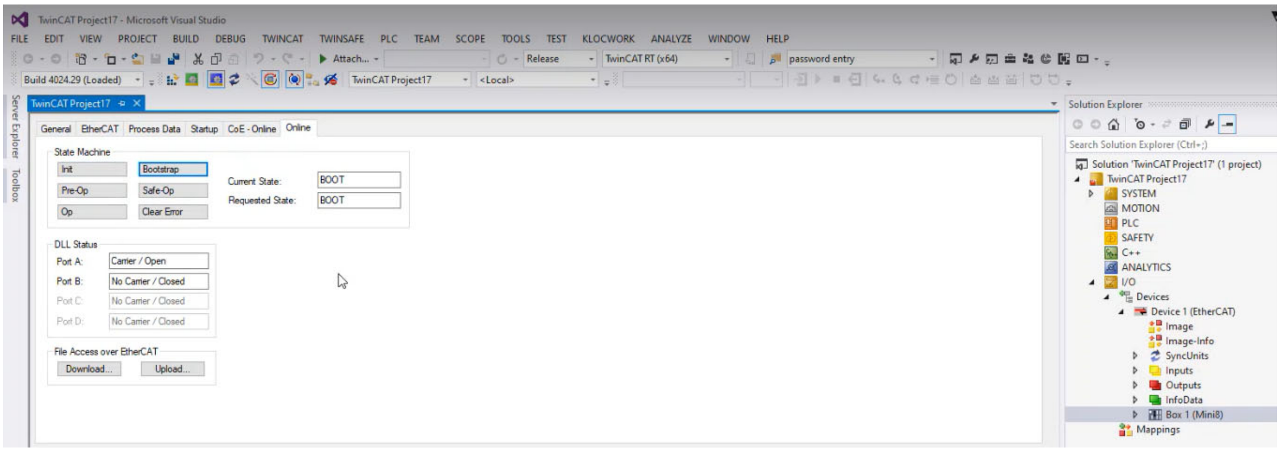
FOE - Downloading Upgrade File

The following example shows how TwinCAT can be used to download the upgrade file to the Mini8 using the FOE interface.

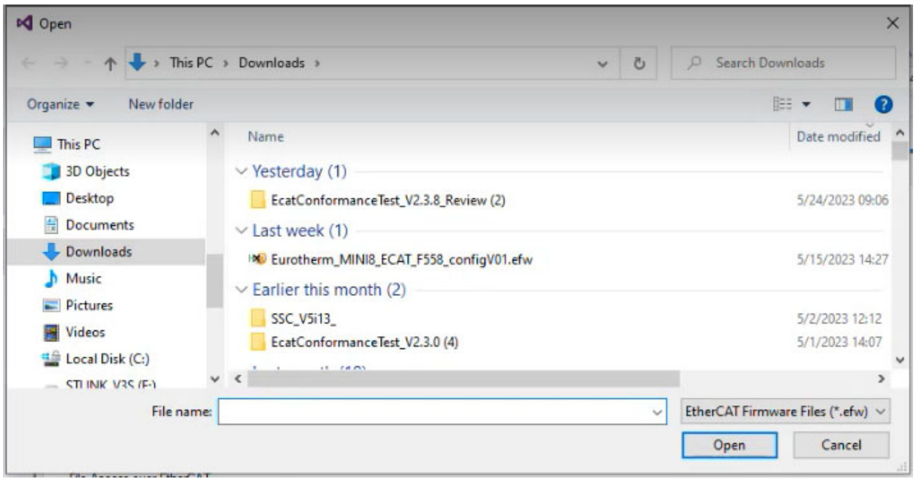
1. The TwinCat master must be online to the Mini8 device
2. Select the Mini8 device from the explorer pane.



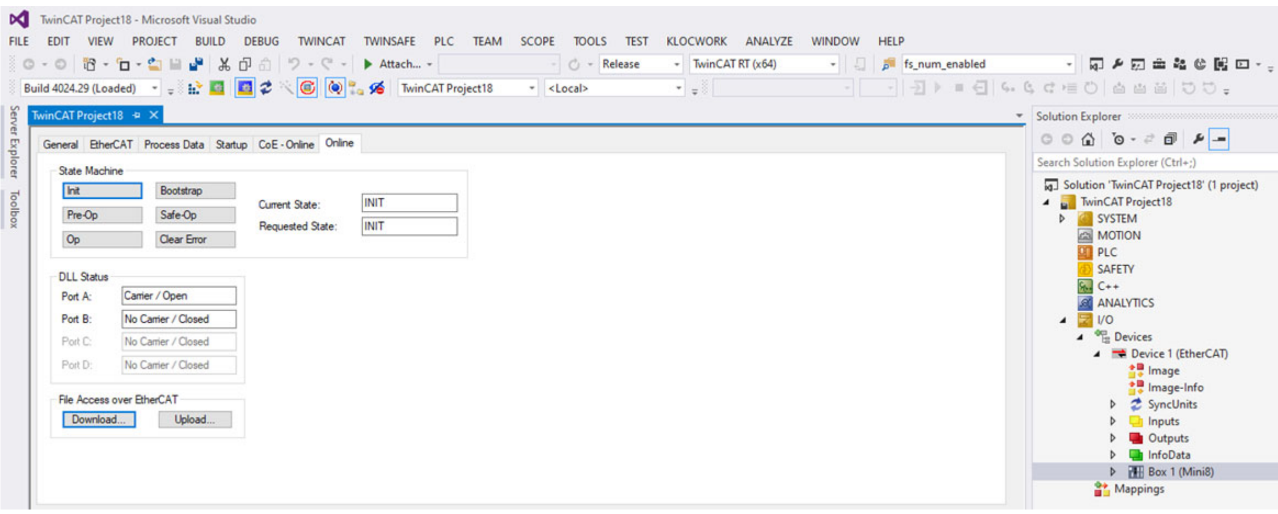
3. Place the Mini8 device in ‘Init’ and ‘Bootstrap’ mode.



4. Select the ‘Download’ button and an explorer dialog will appear, select the required ‘Eurotherm_MINI8_ECAT_xxx_configVxx.efw’ file to be downloaded to the Mini8 device.



5. Once the download is complete, the Mini8 device needs to be power cycled, this is done by selecting ‘Init’ mode.



Once power cycled, the Mini8 will boot up and upgrade its firmware which will take a few seconds. (This is indicated by the front panel “ERR” LED illuminated red).

Once the upgrade completes, the Mini8 will go online again with the TwinCAT client using the new firmware and SII data.

Ethernet over EtherCAT (EOE)

The Mini8 device supports Ethernet Over EtherCAT (EOE) feature, as per the ETG.5003.2060 S ® V1.2.0.standard.

Please consult, refer to your devices' EtherCAT Server/ Client support documentation for further details regarding Ethernet of EtherCAT (EOE).

Trademark

Terms of trademark for EtherCAT

- English: "EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany."
- German: „EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizensiert durch die Beckhoff Automation GmbH, Deutschland."
- French: „EtherCAT® est une marque déposée et une technologie brevetée sous licence de Beckhoff Automation GmbH, Allemagne."
- Italian: „EtherCAT® è un marchio registrato, la tecnologia è brevettata ed è concessa in licenza da Beckhoff Automation GmbH, Germania."
- Spanish: „EtherCAT® es una marca registrada y una tecnología patentada, bajo licencia de Beckhoff Automation GmbH, Alemania."
- Japanese: „EtherCAT®は、ドイツBeckhoff Automation GmbHによりライセンスされた特許取得?み技術であり登?商標です。"
- Korean: „EtherCAT® 독일 Beckhoff Automation GmbH의 허가를 받은 등록 상표이자 특허 기술입니다."
- Chinese: „EtherCAT®是注册商标和专利技术，由德国倍福自动化有限公司授权。"

Counters, Timers, and Totalizers

A series of function blocks are available which are based on time/date information. These may be used as part of the control process.

Counters

Up to two counters are available. They provide a synchronous edge triggered event counter.

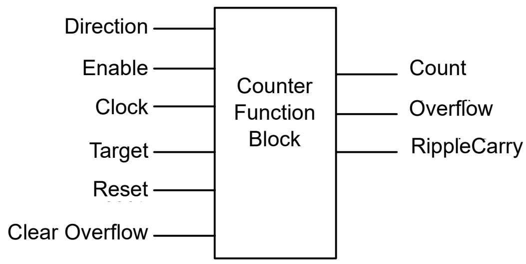


Figure 78 Counter Function Block

When configured as an Up counter, Clock events increment Count until reaching the Target. On reaching Target RippleCarry is set true. At the next clock pulse, Count returns to zero. Overflow is latched true and RippleCarry is returned false.

When configured as a down counter, Clock events decrement Count until it reaches zero. On reaching zero RippleCarry is set true. At the next clock pulse, Count returns to the Target count. Overflow is latched true and RippleCarry is reset false.

Counter blocks can be cascaded as shown in the diagram below.

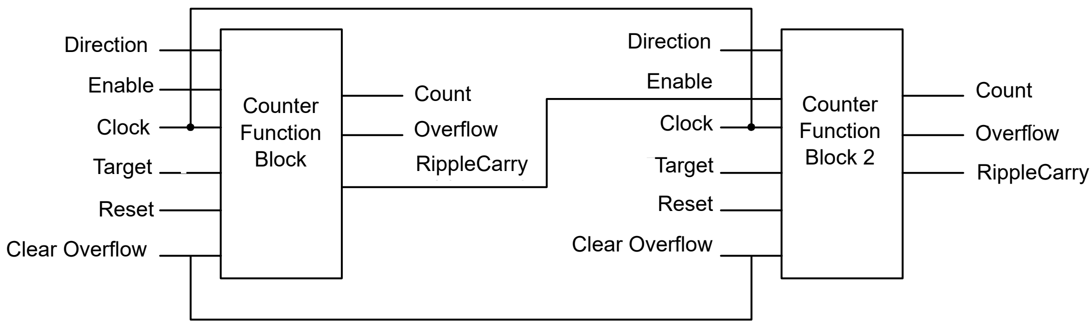


Figure 79 Cascading Counters

The RippleCarry output of one counter acts as an enabling input for the next counter. In this respect the next counter in sequence can only detect a clock edge if it was enabled on the previous clock edge. This means that the Carry output from a counter must lead its Overflow output by one clock cycle. The Carry output is, therefore, called a RippleCarry as it is NOT generated on an Overflow (i.e. Count > Target) but rather when the count reaches the target (i.e. Count = Target). The timing diagram in Figure 80 illustrates the principle for the Up Counter.

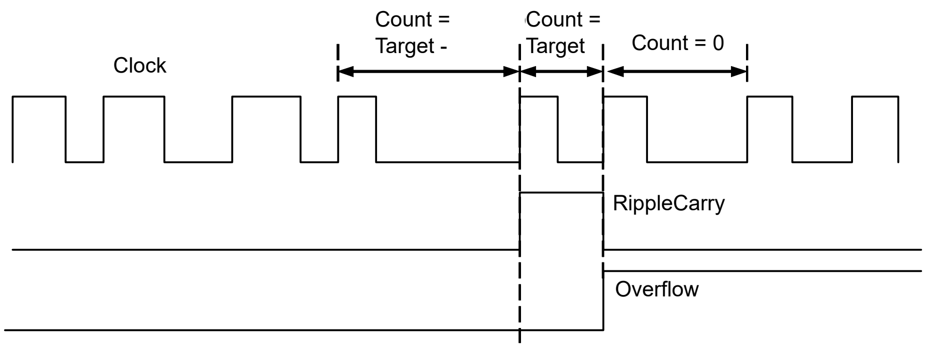


Figure 80 Timing Diagram for an Up Counter

Counter Parameters

Block - Counter		**Sub-blocks: to 2**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Enable	Counter enable. Counter or 2 is enabled in the Instrument Options folder but they can also be turned on or off in this list	Yes No	Enabled Disabled	No	Oper
Direction	Defines count up or count down. This is not intended for dynamic operation (i.e. subject to change during counting). It can only be set in configuration level.	Up Down	Up counter Down counter	Up	Conf
RippleCarry	Ripple carry to act as an enabling input to the next counter. It is turned On when the counter reaches the target set	Off			Read Only
Overflow	Overflow flag is turned on when the counter reaches zero				Read Only
Clock	Tick period to increment or decrement the count. This is normally wired to an input source such as a digital input.	0	No clock input Clock input present	0	Read Only if wired
Target	Level to which the counter is aiming	0 to 99999		9999	Oper
Count	Counts each time a clock input occurs until the target is reached.	0 to 99999			Read Only
Reset	Resets the counter	No Yes	Not in reset Reset	No	Oper
ClearOverflow	Clear overflow flag	No Yes	Not cleared Cleared	No	Oper

Timers

Up to eight timers can be configured. Each one can be configured to a different type and can operate independently of one another.

Timer Types

Each timer block can be configured to operate in four different modes. These modes are explained below.

On Pulse Timer Mode

This timer is used to generate a fixed length pulse from an edge trigger.

- The output is set to On when the input changes from Off to On.
- The output remains On until the time has elapsed.
- If the 'Trigger' input parameter recurs while the Output is On, the Elapsed Time will reset to zero and the Output will remain On.
- The triggered variable will follow the state of the output.

The diagram illustrates the behavior of the timer under different input conditions.

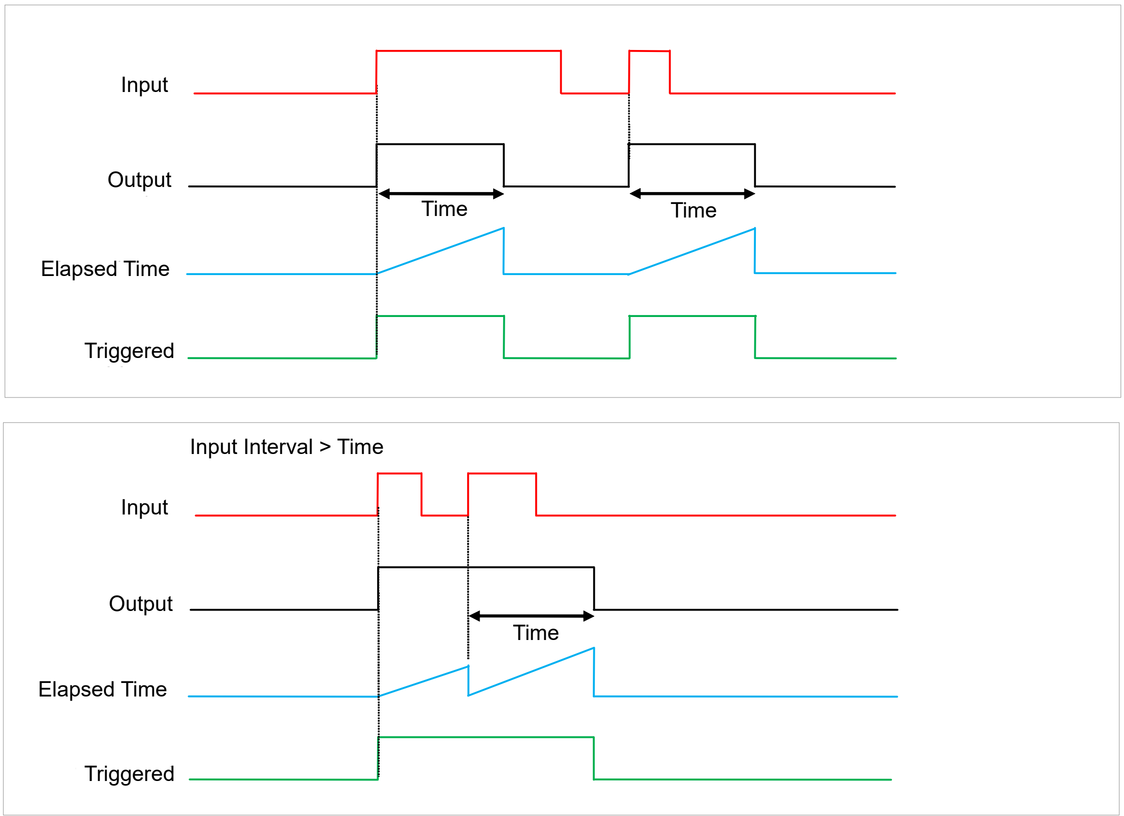


Figure 81 On Pulse Timer Under Different Input Conditions

On Delay Timer Mode

This timer provides a delay between the trigger event and the Timer output. If the input pulse is less than the set delay time there is no output pulse.

- The Output is set to Off when the Input changes from Off to On.
- The Output remains Off until the Time has elapsed.
- If the Input returns to Off before the time has elapsed, the Timer will cease and there will be no output.
- If the Input remains on until the Time has elapsed, the Output will be set to On.
- The Output will remain On until the Input is cleared to Off.
- The Triggered variable will be set to On by the Input changing from Off to On. It will remain On until both the Time has elapsed and the Output has reset to Off.

The diagram illustrates the behavior of the timer under different input conditions.

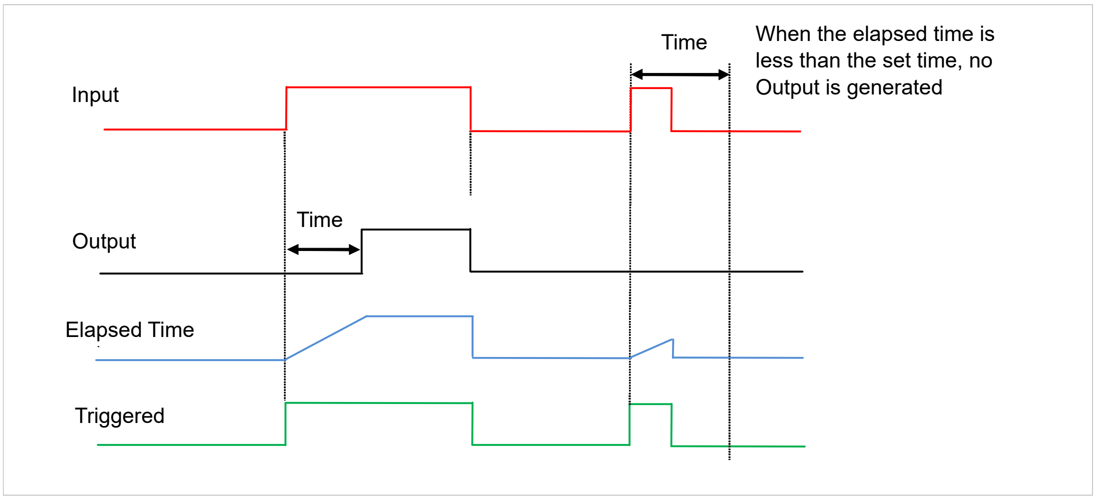


Figure 82 On Delay Timer under Different Input Conditions

This type of timer is used to make certain that the output is not set unless the input has been valid for a pre-determined period of time, thus acting as a kind of input filter.

One Shot Timer Mode

This timer behaves like a simple oven timer.

- When the Time is edited to a non-zero value the Output is set to On.
- The Time value is decremented until it reaches zero. The Output is then cleared to Off.
- The Time value can be edited at any point to increase or decrease the duration of the On time.
- Once set to zero, the Time is not reset to a previous value, it must be edited by the operator to start the next On-Time.
- The Input is used to gate the Output. If the Input is set, the time will count down to zero. If the Input is cleared to Off, then the Time will hold and the Output will switch Off until the Input is next set.

Note: Since the Input is a digital wire, it is possible for the operator to NOT wire it, and set the Input value to On which permanently enables the timer.

- The Triggered variable will be set to On as soon as the Time is edited. It will reset when the Output is cleared to Off.

The behavior of the timer under different input conditions is shown below.

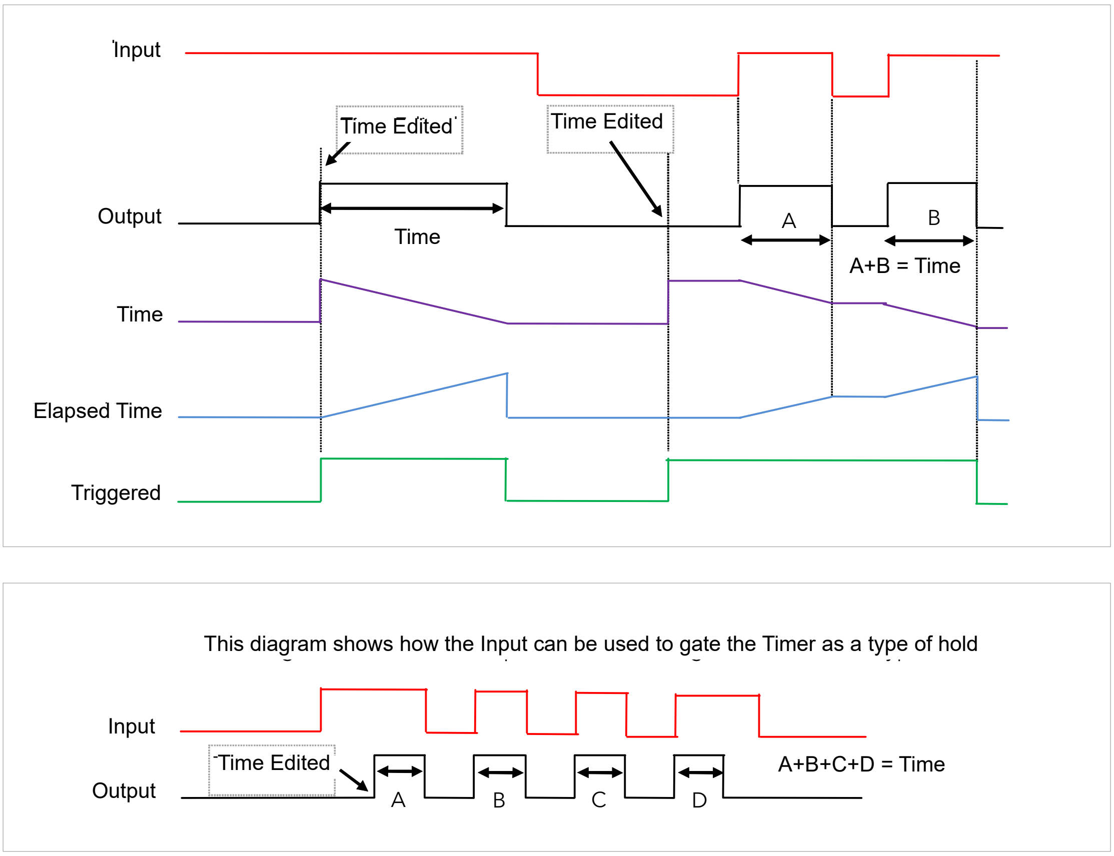


Figure 83 One Shot Timer

Minimum On Timer or Compressor Mode

This type of timer may also be known as an ‘Off Delay’ function where the output goes ‘on’ when the input goes active and remains on for a specified period after the input goes inactive.

It may be used, for example, to help prevent a compressor being cycled excessively.

- The output will be set to On when the Input changes from Off to On.
- When the Input changes from On to Off, the elapsed time will start incrementing towards the set Time.
- The Output will remain On until the elapsed time has reached the set Time. The Output will then switch Off.
- If the Input signal returns to On while the Output is On, the elapsed time will reset to 0, ready to begin incrementing when the Input switches Off.
- The Triggered variable will be set while the elapsed time is >0. It will indicate that the timer is counting.

The diagram illustrates the behavior of the timer under different input conditions.

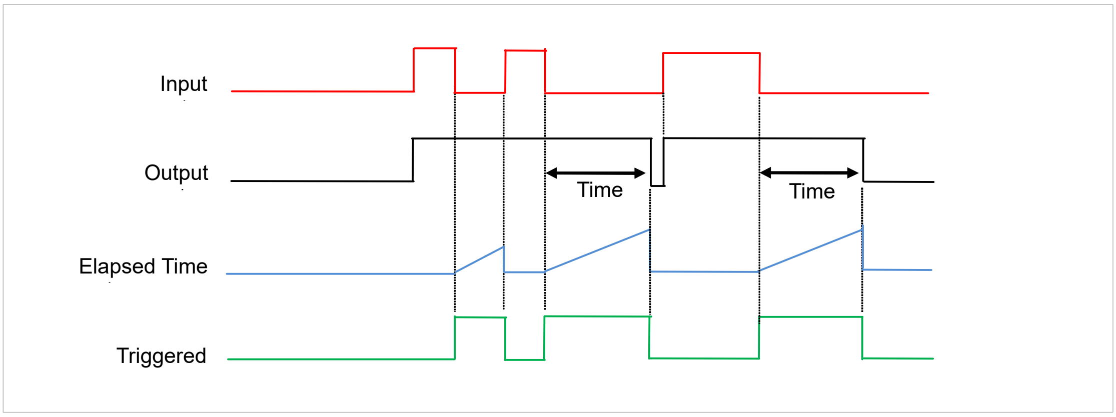


Figure 84 Minimum On Timer under Different Input Conditions

Timer Parameters

Block – Timer		Sub-blocks: to 8			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Type	Timer type	Off	Timer not configured	Off	Conf
		On Pulse	Generates a fixed length pulse from an edge trigger		
		Off Delay	Provides a delay between input trigger event and timer output		
		One Shot	Simple oven timer which reduces to zero before switching off		
		Min-On Ti	Compressor timer keeps the output ON for a time after the input signal has been removed		
Time	Duration of the timer. For re-trigger timers this value is entered once and copied to the time remaining parameter whenever the timer starts. For pulse timers the time value itself is decremented.	0:00.0 to 99:59:59		0:00.0	Oper
ElapsedTime	Timer elapsed time	0:00.0 to 99:59:59			R/O
In	Trigger/Gate input. Turn On to start timing	Off On	Off Start timing	Off	Oper
Out	Timer output	Off On	Output off Timer has timed out		R/O
Triggered	Timer triggered (timing). This is a status output to indicate that the timers input has been detected	Off On	Not timing Timer timing		R/O

The above table is repeated for Timers 2 to 8.

Totalizers

There are two totalizer function blocks which are used to measure the total quantity of a measurement integrated over time. A totalizer can, by soft wiring, be connected to any measured value. The outputs from the totalizer are its integrated value and an alarm state. The user may set a setpoint which causes the alarm to activate once the integration exceeds the setpoint.

The totalizer has the following attributes:

Run/Hold/Reset

In **Run** the totalizer will integrate its input and continuously test against an alarm setpoint.

In **Hold** the totalizer will stop integrating its input but will continue to test for alarm conditions.

In **Reset** the totalizer will be zeroed, and alarms will be reset.

Alarm Setpoint

If the setpoint is a positive number, the alarm will activate when the total is greater than the setpoint.

If the setpoint is a negative number, the alarm will activate when the total is lower (more negative) than the setpoint.

If the totalizer alarm setpoint is set to 0.0, the alarm will be off. It will not detect values above or below.

The alarm output is a single state output. It may be cleared by resetting the totalizer, or by changing the alarm setpoint.

Limits

The total is limited to a maximum of 9,999,999,999 and a minimum of -9,999,999,999.

Resolution

The totalizer maintains resolution when integrating small values onto a large total.

Totalizer Parameters

Block – Total		Sub-blocks: to 2			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
TotalOut	The totalized value	±9,999,999,999			Read Only
In	The value to be totalized	-9999.9 to 9999.9. Note:- the totalizer stops accumulating if the input is 'Bad'.			Oper
Units	Totalizer units	None AbsTemp V, mV, A, mA, pH, mmHg, psi, Bar, mBar, %RH, %, mmWG, inWG, inWW, Ohms, PSIG, %O2, PPM, %CO2, %CP, %/sec, RelTemp mBar/Pa/T sec, min, hrs,			Conf
Resolution	Totalizer resolution	XXXXX XXXX.X XXX.XX XX.XXX X.XXXX		XXXXX	Conf
AlarmSP	Sets the totalized value at which an alarm will occur	±9,999,999,999			Oper
AlarmOut	This is a read-only value which indicates the alarm output On or Off. The totalized value can be a positive number or a negative number. If the number is positive the alarm occurs when Total > + Alarm Setpoint If the number is negative the alarm occurs when Total > - Alarm Setpoint	Off On	Alarm inactive Alarm output active	Off	Oper
Run	Runs the totalizer	No Yes	Totalizer not running Select Yes to run the totalizer	No	Oper
Hold	Holds the totalizer at its current value Note: The Run & Hold parameters are designed to be wired to (for example) digital inputs. Run must be 'on' and Hold must be 'off' for the totalizer to operate.	No Yes	Totalizer not in hold Hold totalizer	No	Oper
Reset	Resets the totalizer	No Yes	Totalizer not in reset Totalizer in reset	No	Oper

Applications

packbit and unpackbit

Packbit - packs 16 individual bits into a 16 bit integer.

Unpackbit - unpacks a 16 bit integer into 16 individual bits.

packbit Parameters

Block – packbit		Sub-blocks: .1 to .8		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
In1 to In16	Input 1 to Input 16			Conf
Output	Output	0.00 to 10.00	0.00	Oper
Status	Status	Good (0) ChannelOff (1) OverRange (2) UnderRange (3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7) HWExceeded (8) NoData (9)	Good (0)	Oper
FallbackType	Fallback Type	FallGood (0) FallBad (1)	FallGood (0)	
Fallback	Fallback value	0.00 to 65535.00	0.00	Oper

unpackbit Parameters

Block – unpackbit		Sub-blocks: .1 to .8		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Input	Input	0.00 to 65535.00		
Out1 to Out16	Output 1 to Output 16	Off (0) On (1)	Off (0)	Conf
Status	Status	Good (0) ChannelOff (1) OverRange (2) UnderRange (3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7) HWExceeded (8) NoData (9)	Good (0)	Oper
FallbackType	Fallback Type	FallGood (0) FallBad (1)	FallGood (0)	
Fallback	Fallback value	0.00 to 65535.00	0.00	Oper

Humidity

Overview

Humidity (and altitude) control is a standard feature of the Mini8 loop controller. In these applications the controller may be configured to measure humidity using either the traditional Wet/Dry bulb method or it may be interfaced to a solid state sensor.

The controller output may be configured to turn a refrigeration compressor on and off, operate a bypass valve, and possibly operate two stages of heating and/or cooling.

Temperature Control of an Environmental Chamber

The temperature of an environmental chamber is controlled as a single loop with two control outputs. The heating output time proportions electric heaters, usually via a solid state relay. The cooling output operates a refrigerant valve which introduces cooling into the chamber. The controller automatically calculates when heating or cooling is required.

Humidity Control of an Environmental Chamber

Humidity in a chamber is controlled by adding or removing water vapor. Like the temperature control loop two control outputs are required, i.e. Humidify and Dehumidify.

To humidify the chamber, water vapor may be added by a boiler, an evaporating pan or by direct injection of atomized water.

If a boiler is being used, adding steam increases the humidity level. The humidify output from the controller regulates the amount of steam from the boiler that is allowed into the chamber.

An evaporating pan is a pan of water warmed by a heater. The humidify output from the controller humidity regulates the temperature of the water.

An atomization system uses compressed air to spray water vapor directly into the chamber. The humidify output of the controller turns on or off a solenoid valve.

Dehumidification may be accomplished by using the same compressor used for cooling the chamber. The dehumidify output from the controller may control a separate control valve connected to a set of heat exchanger coils.

Humidity Parameters

Blocks – Humidity		Sub-blocks: .1			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Resolution	Resolution of the relative humidity	X (0) XX (1) XXX (2) XXXX (3) XXXXX (4)			Conf
PsychroConst	The psychrometric constant at a given pressure (6.66E-4 at standard atmospheric pressure). The value is dependent on the speed of air-flow across the wet bulb, and hence the rate of evaporation. 6.66E-4 is for the ASSMANN ventilated Psychrometer.	0.0 to 10.0		6.66	Oper
Pressure	Atmospheric Pressure	0.0 to 2000.0		1013.0 mbar	Oper
WetTemp	Wet Bulb Temperature	Range units			
WetOffset	Wet Bulb Temperature offset	-100.00 to 100.00		0.00	Oper
DryTemp	Dry Bulb Temperature	Range units			
RelHumid	Relative Humidity is the ratio of actual water vapor pressure (AVP) to the saturated water vapor pressure (SVP) at a particular temperature and pressure.	0.00 to 100.00		100	Read Only
DewPoint	The dew point is the temperature to which air would need to cool (at constant pressure and water vapor content) in order to reach saturation.	-19999 to 99999			Read Only
Sbrk	Indicates that one of the probes is broken.	No (0) Yes (1)	No sensor break detection Sensor break detection enabled		Conf

Input Monitor

Description

There are two Input monitors. Each input monitor may be wired to any variable in the controller. It then provides three functions:

- Maximum detect
- Minimum detect
- Time above threshold

Maximum Detect

This function continuously monitors the input value. If the value is higher than the previously recorded maximum, it becomes the new maximum.

This value is retained following a power interruption.

Minimum Detect

This function continuously monitors the input value. If the value is lower than the previously recorded minimum, it becomes the new minimum.

This value is retained following a power interruption.

Time Above Threshold

This function increments a timer whenever the input is above a threshold value. If the timer exceeds 24 hours per day, a counter is incremented. The maximum number of days is limited to 255. A time alarm can be set on the timer so that once the input has been above a threshold for a period, an alarm output is given.

Applications include:

- Service interval alarms. This sets an output when the system has been running for a number of days (up to 255 days).
- Material stress alarms - if the process cannot tolerate being above a level for a period. This is a style of ‘policeman’ for processes where the high operating point reduces the life of the machine.
- In internal wiring applications in the controller.

Input Monitor Parameters

Block - IPMonitor		**Sub-blocks: 1 or 2**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
In	The input value to be monitored.	May be wired to an input source. The range will depend on the source.			Oper Read Only if wired
Max	The maximum measured value recorded since the last reset.	As above			Read Only
Min	The minimum measured value recorded since the last reset.	As above			Read Only
Threshold	The input timer accumulates the time the input PV spends above this trigger value.	As above			Oper
Days Above	Accumulated days the input has spent above threshold since the last reset.	Days is an integer count of the 24 hour periods only. The Days value should be combined with the Time value to make the total time above threshold.			Read Only
TimeAbove	Accumulated time above the 'Threshold' since last reset.	The time value accumulates from 00:00.0 to 23:59.9. Overflows are added to the days value.			Read Only
AlarmDays	Days threshold for the monitors time alarm. Used in combination with the AlarmTime parameter. The 'Out' is set to true if the inputs accumulated time above threshold is higher than the timer high parameters.	0 to 255		0	Oper
AlarmTime	Time threshold for the monitors time alarm. Used in combination with the AlarmDays parameter. The 'Out' is set to true if the inputs accumulated time above threshold is higher than the timer high parameters.	0:00.0 to 99:59:59		0:00.0	Oper
Out	Set true if the accumulated time that the input spends above the trigger value is higher than the alarm threshold.	Off (0) On (1)	Normal operation Time above setpoint exceeded		Read Only
Reset	Resets the Max and Min values and resets the time above threshold to zero.	No (0) Yes (1)	Normal operation Reset values	No	Oper
InStatus	Monitors the status of the input.	Good (0) ChannelOff (1) OverRange(2) UnderRange(3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7)	Normal operation The input channel is turned off The input is over range The input is under range The hardware status cannot be determined The input value has overflowed The input may be incorrectly wired		Read Only Oper

Logic and Maths Operators

Logic Operators

Logic Operators allow the controller to perform logical calculations on two input values. These values can be sourced from any available parameter including Analog Values, User Values and Digital Values.

The parameters to use, the type of calculation to be performed, use of logical NOT on the input value and 'fallback' value are determined in Configuration level.

There are 24 separate calculations – they do not have to be in sequence. When logic operators are enabled a Folder 'Lgc2' exists where the '2' denotes two input logic operators.

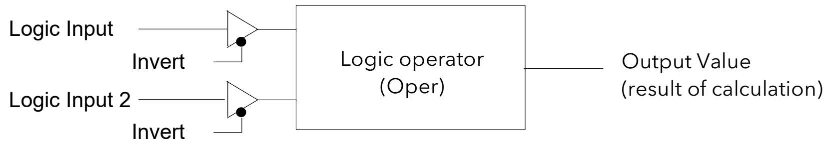


Figure 85 Two-Input Logic Operators

Logic Operators are found under the folder 'Lgc2'. Note that the logic operators can also be enabled by dragging a block onto the graphical wiring screen in iTools.

Logic 8

Logic 8 operators can perform logic calculations on up to eight inputs. The calculations are limited to AND, OR, and XOR. Up to two eight-input operators can be enabled. The block is labeled 'Lgc8' to denote eight-input logic operators.

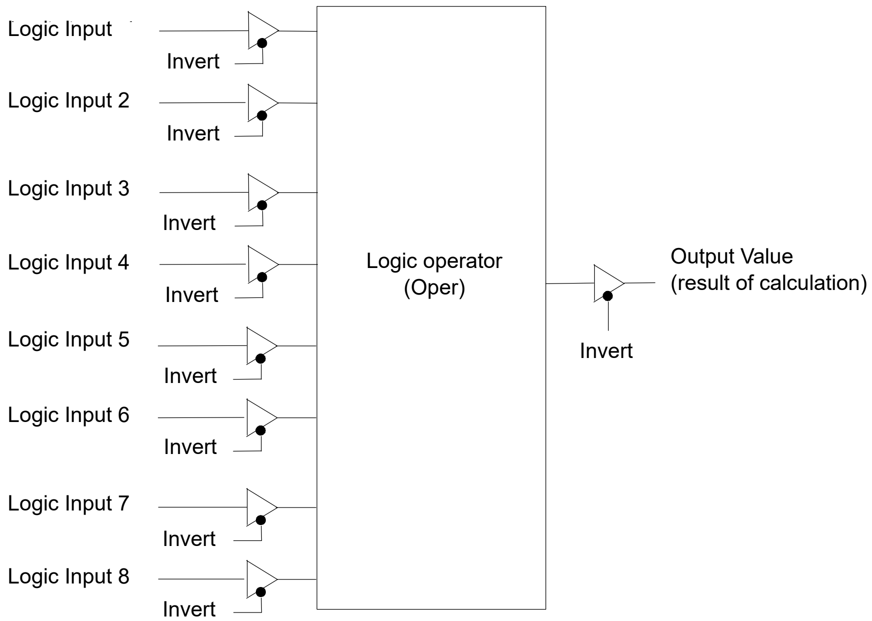


Figure 86 Eight-Input Logic Operators

Two-input Logic Operations

The following calculations can be performed:

Oper	Operator description	Input	Input 2	Output Invert = None
0: OFF	The selected logic operator is turned off			
: AND	The output result is ON when both Input and Input 2 are ON	0	0	Off
			0	Off
		0		Off
				On
2: OR	The output result is ON when either Input or Input 2 is ON	0	0	Off
			0	On
		0		On
				On
3: XOR	Exclusive OR. The output result is true when one and only one input is ON. If both inputs are ON the output is OFF.	0	0	Off
			0	On
		0		On
				Off
4: Latch	Input sets the latch, Input 2 resets the latch.	0	0	Off
			0	On
		0		Off
				Off
5: Equal (==)	The output result is ON when Input = Input 2	0	0	On
			0	Off
		0		Off
				On
6: Not equal (<>)	The output result is ON when Input does not equal Input 2	0	0	Off
			0	On
		0		On
				Off
7: Greater than (>)	The output result is ON when Input > Input 2	0	0	Off
			0	On
		0		Off
				Off
8: Less than (<)	The output result is ON when Input < Input 2	0	0	Off
			0	Off
		0		On
				Off
9: Equal to or Greater than (=>)	The output result is ON when Input ≥ Input 2	0	0	On
			0	On
		0		Off
				On
0: Less than or Equal to (<=)	The output result is ON when Input ≤ Input 2	0	0	On
			0	Off
		0		On
				On

Notes:

1. The numerical value is the value of the enumeration.
2. For options to 4 an input value of less than 0.5 is considered FALSE and greater than or equal to 0.5 is considered TRUE.

Logic Operator Parameters

Block – Lgc2 (2 Input Operators)		**Sub-blocks: to 40**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Oper	To select the type of operator	See previous table		None	Conf
In	Input	Normally wired to a logic, analog or user value. May be set to a constant value if not wired.		0	Oper
In2	Input 2				
FallbackType	The fallback state of the output if one or both of the inputs is bad	FalseBad (0)	The output value is FALSE and the status is BAD.		Conf
		TrueBad ()	The output value is TRUE and the status is BAD		
		FalseGood (2)	The output value is FALSE and the status is GOOD		
		TrueGood (3)	The output value is TRUE and the status is GOOD.		
Invert	The sense of the input value, may be used to invert one or both of the inputs	None (0)	Neither input inverted		Conf
		Input ()	Invert input		
		Input2 (2)	Invert input 2		
		Both (3)	Invert both inputs		
Out	The output from the operation is a boolean (true/false) value.	Off (0) On ()	Output not activated Output activated		Read Only
Status	The status of the result value	Good (0) ChannelOff () OverRange (2) UnderRange (3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7)			Read Only

Eight-Input Logic Operators

The eight-input logic operator may be used to perform the following operations on eight inputs.

Oper	Operator description
0: OFF	The selected logic operator is turned off
: AND	The output result is ON when ALL eight inputs are ON
2: OR	The output result is ON when one or more of the eight inputs are ON
3: XOR	Exclusive OR – the output is true if an odd number of inputs are true. (In ⊕ In2) ⊕ (In3 ⊕ In4) ⊕ (In5 ⊕ In6) ⊕ (In7 ⊕ In8)

Eight Input Logic Operator Parameters

Block – Lgc8 (8 Input Operators)		Sub-blocks: to 4			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Oper	To select the type of operator	OFF (0) AND () OR (2) XOR (3)	Operator turned off Output ON when all inputs are ON Output ON when one input is ON Exclusive OR	OFF	Conf
NumIn	This parameter is used to configure the number of inputs for the operation	to 8		2	Conf
InInvert	Used to invert selected inputs prior to operation. This is a status word with one bit per input, the left hand bit inverts input .	The invert parameter is interpreted as a bitfield where: (0x) - input 2 (0x2) - input 2 4 (0x4) - input 3 8 (0x8) - input 4 6 (0x0) - input 5 32 (0x20) - input 6 64 (0x40)- input 7 28 (0x80)- input 8 (e.g. 255 = all eight)		0	Oper
Out Invert	Invert the output	No (0) Yes ()	Output not inverted Output inverted	No	Oper
In to In8	Input state to 8	Normally wired to a logic, analog or user value. When wired to a floating point, values less than or equal to –0.5 or greater than or equal to .5 will be rejected (e.g. the value of the lgc8 block will not change). Values between –0.5 and .5 will be interpreted as ON when greater than or equal to 0.5 and OFF when less than 0.5. May be set to a constant value if not wired.		Off	Oper
Out	Output result of the operator	Off (0) Off ()	Output not activated Output activated		Read Only

Maths Operators

Maths Operators (sometimes known as Analog Operators) allow the controller to perform mathematical operations on two input values. These values can be sourced from any available parameter including Analog Values, User Values and Digital Values. Each input value can be scaled using a multiplying factor or scalar.

The parameters to use, the type of calculation to be performed and the acceptable limits of the calculation are determined in Configuration level. In normal operation the values of each of the scalars may be changed via communications or iTools.

There are 24 separate calculations – they do not have to be in sequence. When maths operators are enabled (in Instrument/Options folder) a Folder ‘Math2’ exists (where the ‘2’ denotes two-input maths operators).

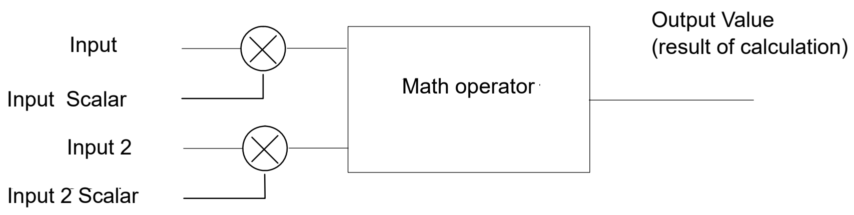
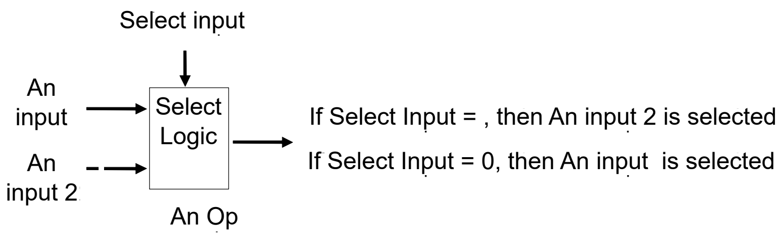


Figure 87 Two-Input Math Operators

Eight-input multiplexers are also available and are described in "Eight-Input Analog Multiplexers" on page 184.

Math Operations

The following operations can be performed:

0: Off	The selected analog operator is turned off
: Add	The output result is the addition of Input and Input 2
2: Subtract (Sub)	The output result is the difference between Input and Input 2 where Input > Input 2
3: Multiply (Mul)	The output result is the Input multiplied by Input 2
4: Divide (Div)	The output result is Input divided by Input 2
5: Absolute Difference (AbsDif)	The output result is the absolute difference between Input and 2
6: Select Max (SelMax)	The output result is the maximum of Input and Input 2
7: Select Min (SelMin)	The output result is the minimum of Input and Input 2
8: Hot Swap (HotSwp)	Input appears at the output provided input is 'good'. If input is 'bad' then input 2 value will appear at the output. An example of a bad input occurs during a sensor break condition.
9: Sample and Hold (SmpHld)	Normally input will be an analog value and input B will be digital. The output tracks input when input 2 = (Sample). The output will remain at the current value when input 2 = 0 (Hold). If input 2 is an analog value then any non zero value will be interpreted as 'Sample'.
0: Power	The output is the value at input raised to the power of the value at input 2. I.e. input $^{input\ 2}$
: Square Root (Sqrt)	The output result is the square root of Input . Input 2 has no effect.
2: Log	The output is the logarithm (base 0) of Input . Input 2 has no effect
3: Ln	The output is the logarithm (base n) of Input . Input 2 has no effect
4: Exp	The output result is the exponential of Input . Input 2 has no effect
5: 0 x	The output result is 0 raised to the power of Input value. I.e. 0^{input} . Input 2 has no effect
5: Select	Select input is used to control which Analog Input is switched to the output of the Analog Operator. If the select input is true input 2 is switched through to the output. If false input is switched through to the output. See example below: 

When Boolean parameters are used as inputs to analog wiring, they will be cast to 0.0 or .0 as appropriate. Values <= -0.5 or >= .5 will not be wired. This provides a way to stop a Boolean updating. Analog wiring (whether simple re-routing or involving calculations) will always output a real type result, whether the inputs were booleans, integers or reals.

Note: The numerical value is the value of the enumeration.

Math Operator Parameters

Block – Math2 (2 Input Operators)		**Sub-blocks: to 32**		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Oper	To select the type of operator	See previous table	None	Conf
InMul	Scaling factor on input	Limited to max float*	.0	Oper
In2 Mul	Scaling factor on input 2	Limited to max float*	.0	Oper
Units	Units applicable to the output value	None (0) C_F_K_Temp () V (2) mV (3) A (4) mA (5) PH (6) mmHg (7) psi (8) Bar (9) mBar (0) PercentRH () Percent (2) mmWG (3) inWG (4) inWW (5) Ohms (6) PSIG (7) PercentO2 (8) PPM (9) PercentCO2 (20) PercentCarb (2) PercentPerSec (22) RelTemperature (24) Vacuum (25) Secs (26) Mins (27) Hours (28)	None	Conf
Resolution	Resolution of the output value	X, X.X. X.XX, X.XXX, X.XXXX		Conf
LowLimit	To apply a low limit to the output	Max float* to High limit (decimal point depends on resolution)		Conf
HighLimit	To apply a high limit to the output	Low limit to Max float* (decimal point depends on resolution)		Conf
Fallback	The state of the Output and Status parameters in case of a detected fault condition. This parameter could be used in conjunction with fallback value	ClipBad (0) ClipGood () FallBad (2) FallGood (3) UpScaleBad (4) DownScaleBad (5) Descriptions, see "Fallback" on page 99		Conf
Fallback Val	Defines (in accordance with Fallback) the output value during detected fault conditions.	Limited to max float* (decimal point depends on resolution)		Conf
In	Input value (normally wired to an input source – could be a User Value)	Limited to max float* (decimal point depends on resolution)		Oper
In2	Input 2 value (normally wired to an input source – could be a User Value)	Limited to max float* (decimal point depends on resolution)		Oper
Out	Indicates the analog value of the output	Between high and low limits		Read Only

Block – Math2 (2 Input Operators)		Sub-blocks: to 32		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Status	This parameter is used in conjunction with Fallback to indicate the status of the operation. Typically, status is used to flag detected fault conditions and may be used as an interlock for other operations.	Good (0) ChannelOff () OverRange (2) UnderRange (3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7)		Read Only

* Max float in this instrument is ±9,999,999,999

Sample and Hold Operation

The diagram below shows the operation of the sample and hold feature.

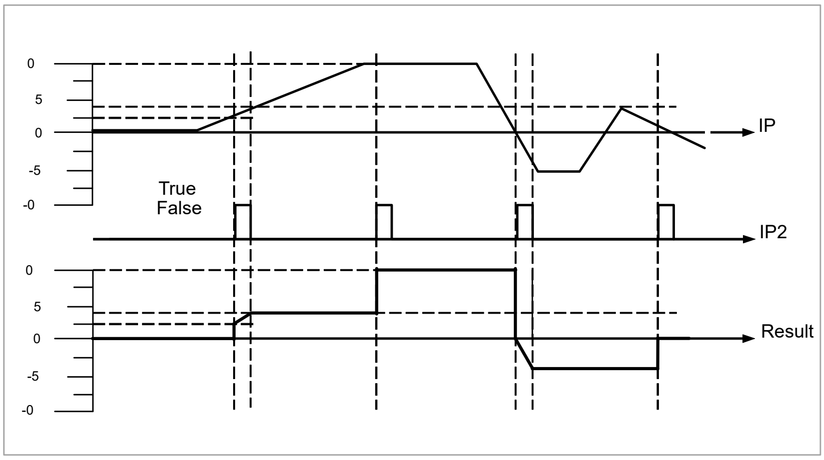


Figure 88 Sample and Hold

Multiple Input Operator Block

The Multiple Input Operator Block simultaneously outputs the Sum, Average, Minimum and Maximum values of up to eight valid inputs. The outputs will be clipped to user-defined limits or be replaced by a fallback value based on the selected fallback strategy.

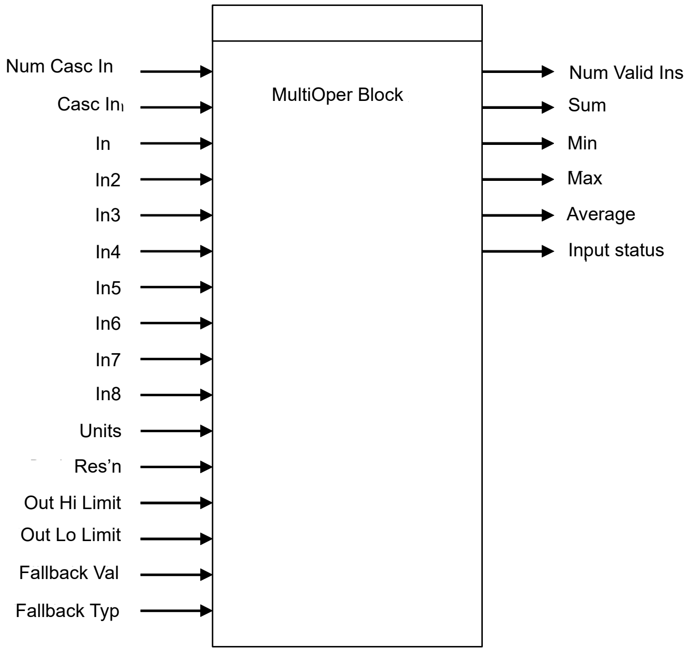


Figure 89 Multi Operator Function Block

'Num In' determines the number of inputs made available for use. This is settable by the user and is defaulted to two. Be careful not to set this number to a value higher than the desired number of inputs as any unused inputs are seen as valid inputs to the block (zero value by default). Num Casc In and Casc In will always be available.

'Input Status' gives an indication of the status of the inputs in priority order. Casc In has the highest priority, In the next highest up to In8 the lowest. If more than one input is bad then the input with the highest priority is shown as 'bad'. When the highest priority bad status is cleared the next highest priority bad status is shown. When all inputs are OK a status of OK is shown.

'Number of valid inputs' provides a count of the number of inputs used to perform the calculation within the block. This is required for cascaded operation and is detailed below.

Cascaded Operation

Multiple input operator blocks may be cascaded to allow operations on more than eight inputs (33 max for four instances of the block). Figure 90 shows how two blocks should be configured to find the average of more than eight inputs. If required the second block could then be cascaded to a third to provide up to eight more inputs.

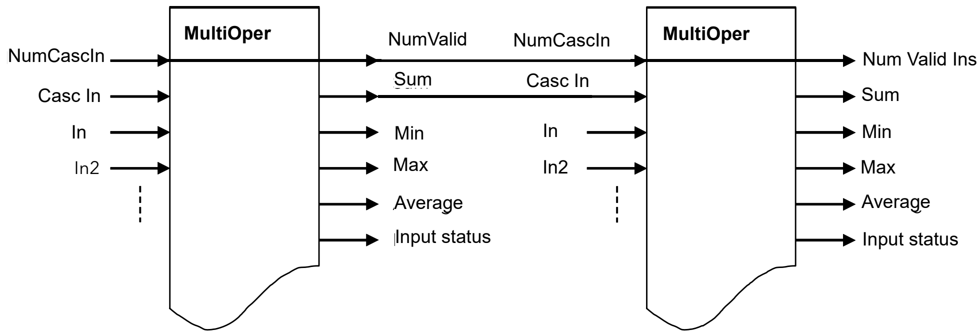


Figure 90 Cascaded Multi Operator Function Blocks

If 'CascIn' is has Good status, and 'NumCascIn' is not equal to zero we can assume that the block is in cascade and these values are used for calculations within the block, and the value given by 'NumCascIn' is added to 'NumValidIns'. When in cascade the sum, min, max and average outputs treat Casc In as an additional input to the block. For example if Casc In is greater than any number on the rest of the inputs then its value will be output as the max.

Fallback Strategy

The user is able to select the fallback strategy during configuration. The options are:

Clip Good

- The status of the outputs is always good.
- If an output is out of range then it is clipped to limits.
- If all inputs are bad, all outputs = 0 (or clipped to limits if 0 is not within the output range).

Clip Bad

- The status of all outputs is bad if one or more of the inputs are bad.
- If an output is out of range then it is clipped to limits and the status of that output is set to bad.
- If all inputs are bad, all outputs = 0 and all status' are set to bad (or clipped to limits if 0 is not within the output range).

Fall Good

- The status of the outputs is always good.
- If an output is out of range then it is set to the fallback value.
- If all inputs are bad, all outputs = fallback value.

Fall Bad

- The status of all outputs is bad if one or more of the inputs are bad.
- If an output is out of range then it is set to the fallback value and the status is set to bad.
- If all inputs are bad, all outputs are set to the fallback value and all statuses are set to bad.

Multiple Input Operator Block Parameters

Block – MultiOper (Multi Operator)		**Sub-blocks: to 4**		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
NumIn	Number of inputs selected to use.	2 to 8	2	Conf
CascNumIn	Number of cascaded inputs from the previous block	0 to 255	0	Read Only
CascIn	The cascaded input from a previous block	-99999 to 99999 (decimal point depends on Resolution)	0	Read Only
In to In 8	Input to Input 8	-99999 to 99999 (decimal point depends on Resolution)	0	Read Only
Units	Selected units for the I/O	None (0) C_F_K_Temp () V (2) mV (3) A (4) mA (5) PH (6) mmHg (7) psi (8) Bar (9) mBar (0) PercentRH () Percent (2) mmWG (3) inWG (4) inWW (5) Ohms (6) PSIG (7) PercentO2 (8) PPM (9) PercentCO2 (20) PercentCarb (2) PercentPerSec (22) RelTemperature (24) Vacuum (25) Secs (26) Mins (27) Hours (28) Days (29) Mb (30) Mb (3) ms (32)	None	Conf
Resolution	Selected resolution of the Outputs	X to X.XXXX	X	Conf
OutHiLimit	Upper limit of the outputs.	-99999 to 99999 (decimal point depends on Resolution). Minimum setting is limited by ‘OutLoLimit’.	0	Conf

Block – MultiOper (Multi Operator)		**Sub-blocks: to 4**		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
OutLoLimit	Lower limit of the outputs.	-99999 to 99999 (decimal point depends on Resolution). Maximum setting is limited by ‘OutHiLimit’.	0	Conf
FallbackTyp	Fallback Type selected.	ClipBad (0) ClipGood () FallBad (2) FallGood (3) See "Fallback Strategy" on page 181.	Clip Good	Conf
FallbackVal	Value to be output depending on Input status and fallback type selected.	-99999 to 99999 (decimal point depends on Resolution)	0	Conf
NumValidIn	Number of inputs used in the calculated outputs (Output)	2 to 8	0	Read Only
SumOut	Sum of the valid inputs (Output)	-99999 to 99999 (decimal point depends on Resolution)	0	Read Only
MaxOut	Maximum value of the valid inputs (Output)	-99999 to 99999 (decimal point depends on Resolution)	0	Read Only
MinOut	Minimum value of the valid inputs (Output)	-99999 to 99999 (decimal point depends on Resolution)	0	Read Only
AverageOut	Average value of the valid inputs (Output)	-99999 to 99999 (decimal point depends on Resolution)	0	Read Only
InputStatus	Status of the inputs (Output)	Good (0) CascInBad () InBad (2) In2Bad ((3) In3Bad ((4) In4Bad ((5) In5Bad ((6) In6Bad ((7) In7Bad ((8)	Good (0)	Read Only

Eight-Input Analog Multiplexers

The eight-input analog multiplexers may be used to switch one of eight inputs to an output. It is usual to wire inputs to a source within the controller that selects that input at the appropriate time or event.

Multiple Input Operator Parameters

Block – Mux8 (8 Input Multiplexers)		**Sub-blocks: to 8**		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
HighLimit	The high limit for all inputs and the fall back value.	Low limit to 99999 (decimal point depends on resolution)		Conf
LowLimit	The low limit for all inputs and the fall back value.	-99999 to High limit (decimal point depends on resolution)		Conf
Fallback	The state of the Output and Status parameters in case of a detected fault condition. This parameter could be used in conjunction with FallbackVal.	ClipBad (0) ClipGood () FallBad (2) FallGood (3) UpScaleBad (4) DownScaleBad (5) Descriptions see "Fallback Strategy" on page 181.		Conf
FallbackVal	Used (in accordance with Fallback) to define the output value during detected fault conditions	-99999 to 99999 (decimal point depends on resolution)		Conf
Select	Used to select which input value is assigned to the output.	Input to Input8		Oper
In to In8	Input values (normally wired to an input source)	-99999 to 99999 (decimal point depends on resolution)		Oper
Out	Indicates the analog value of the output	Between high and low limits		Read Only
Status	Used in conjunction with Fallback to indicate the status of the operation. Typically, status is used to flag detected fault conditions and may be used as an interlock for other operations.	Good (0) ChannelOff () OverRange (2) UnderRange (3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7) HWExceeded (8) NoData (9)		Read Only
Resolution	Selected resolution of the Outputs	X to X.XXXX	X.X ()	

Fallback

The fallback strategy will come into effect if the status of the input value is bad or if the input value is outside the range of Input Hi and Input Lo.

In this case the fallback strategy may be configured as:

Fall Good If the input value is above ‘High Limit’ or below ‘Low Limit’, then the output value is set to the ‘Fallback’ value, and the ‘Status’ is set to ‘Good’.

Fall Bad If the input value is above ‘High Limit’ or below ‘Low Limit’, then the output value is set to the ‘Fallback’ value, and the ‘Status’ is set to ‘Bad’.

Clip Good If the input value is above ‘High Limit’ or below ‘Low Limit’, then the output value is set to the appropriate limit, and

	'Status' is set to 'Bad'. If the input signal is within the limits, but its status is bad, the output is set to the 'Fallback' value.
Clip Bad	If the input value is above 'High Limit' or below 'Low Limit', then the output value is set to the appropriate limit, and 'Status' is set to 'Good'. If the input signal is within the limits, but its status is bad, the output is set to the 'Fallback' value.
Upscale	If the input status is bad, or if the input signal is above 'High Limit' or below 'Low Limit', the output value is set to the 'High Limit'.
Downscale	If the input status is bad, or if the input signal is above 'High Limit' or below 'Low Limit', the output value is set to the 'Low Limit'.

Input Characterization

Input Linearization

The linearization block converts an analog input into an analog output through a user-defined table. This linearization table consists of a series of 32 points defined by input breakpoints (In1 to In32) and output values (Out1 to Out32). In other words, the linearization block implements a piecewise linear curve (a connected sequence of line segments) defined by a series of input coordinates (In1 to In32) and associated output coordinates (Out1 to Out32).

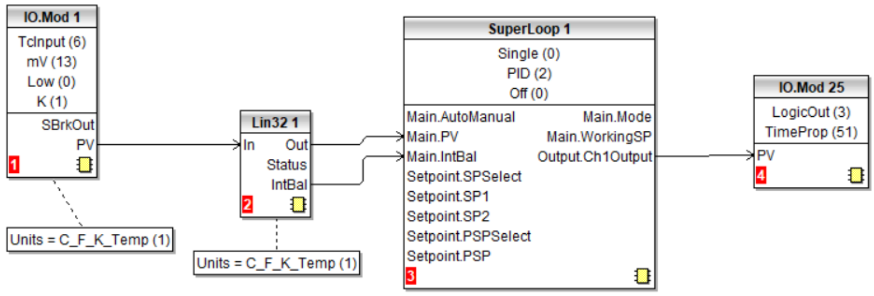
Two of the most typical applications for the LIN32 function block are:

1. Custom linearization of a sensor input.
2. Adjustment of the process variable to account for differences introduced by the overall measurement system or to derive a different process variable.

Custom Linearization

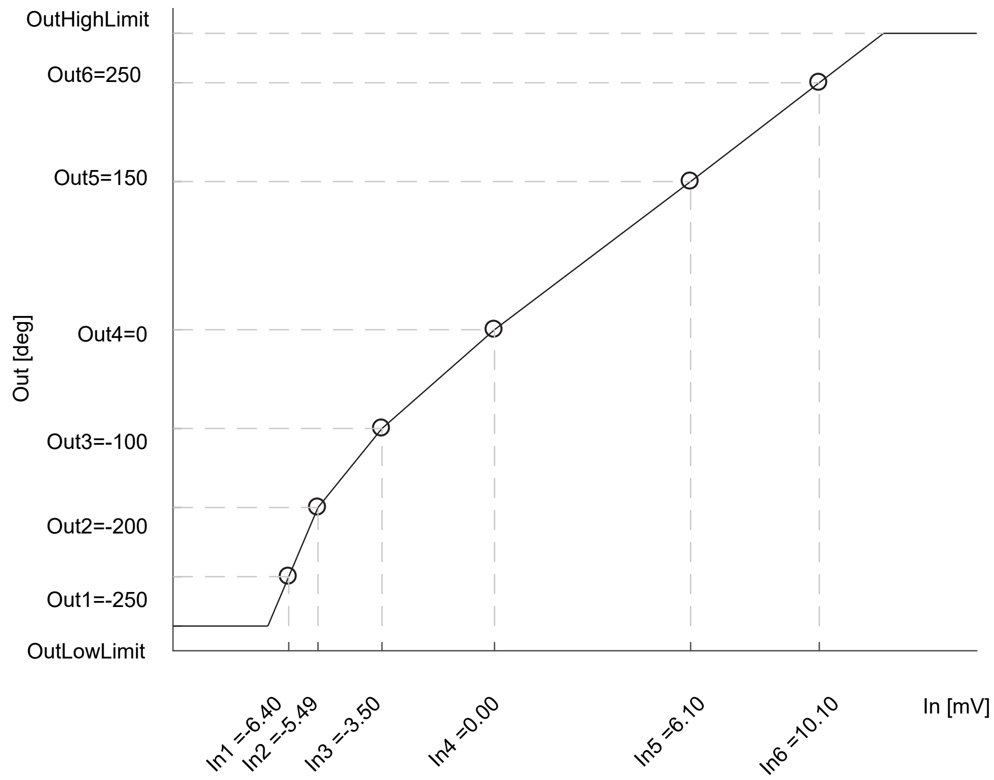
This application allows the user to create their own linearization table.

In the following example the LIN32 block is placed between the SuperLoop block and an Analog Input set to linear and Linearization Type to mV, V, mA, Ohms, etc. In the following example the AI block is set to mV.



The following graph shows a typically increasing linearization curve. The decision of the actual number of points depends of the required accuracy in converting the input electrical signal into the required output value: the higher the number of points, the higher accuracy can be obtained; conversely a lower number of points requires less time to configure the function block. If less than 32 points are used, set the 'NumPoints' parameter to the required number. Points not selected will then be ignored, the curve will continue in a straight line fit to the levels set in 'OutHighLimit' or 'OutLowLimit' and the 'CurveForm' output will be 'Increasing'.

Example 1: Custom Linearization - Increasing Curve

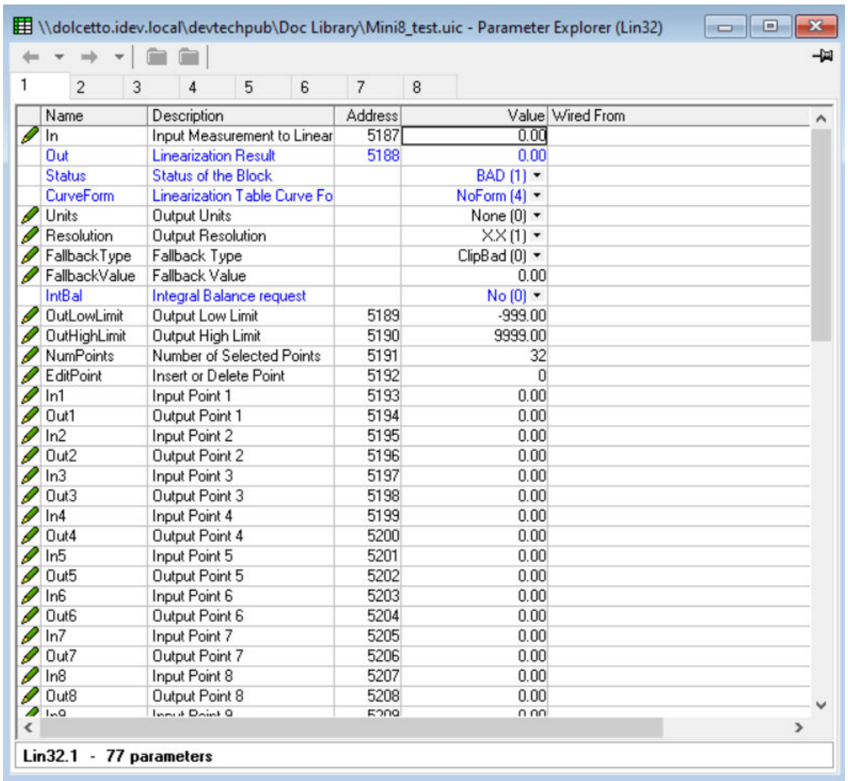


To Setup the Parameters

1. Set the appropriate Fallback type and value, Output units and resolution (editable only in Config mode); Units and resolution of the input and the input breakpoints will be derived by the source wired to 'In'.
2. Set the ‘OutHighLimit’ and ‘OutLowLimit’ to restrict the output of the linearization curve. The ‘OutHighLimit’ must be greater than the ‘OutLowLimit’.
3. Set the 'NumPoints' (6 in this example) to the required number of points for the linearization table. This is an important and required step and the effects of skipping it are reported in the Example 2.
4. Enter values of the first Input breakpoint ‘In1’ and Output value ‘Out1’.
5. Continue with the remaining Input breakpoints and Output values.
6. Wire the ‘IntBal’ parameter to the ‘Loop.Main.IntBal’ parameter. This prevents any proportional or derivative kick in the controller output when any change occurs in the LIN16 configuration parameters.

Points on the linearization curve can be derived from reference tables or can be found by associating the measurements of an external reference (e.g. temperature in degrees Celsius) to the AI electrical readings (e.g. mV or mA).

The iTools view reproduced below shows how the parameters are set up in LIN block 1 for the above example. Parameter help is also available by right clicking the parameter in the iTools list.

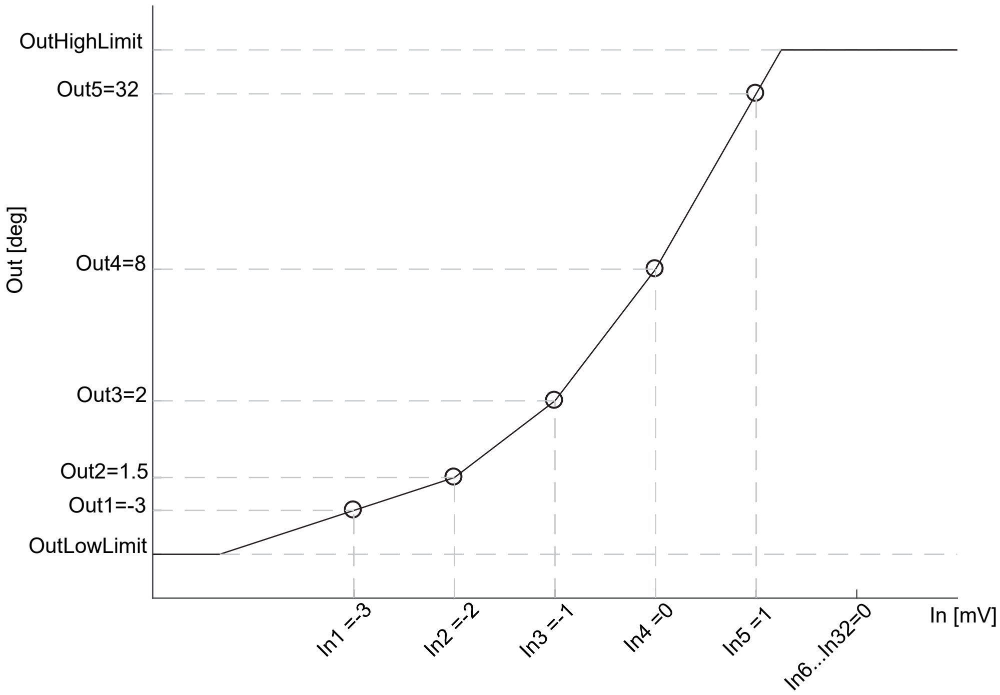


Name	Description	Address	Value	Wired From
In	Input Measurement to Linear	5187	0.00	
Out	Linearization Result	5188	0.00	
Status	Status of the Block		BAD (1)	
CurveForm	Linearization Table Curve Fo		NoForm (4)	
Units	Output Units		None (0)	
Resolution	Output Resolution		X.X (1)	
FallbackType	Fallback Type		ClipBad (0)	
FallbackValue	Fallback Value		0.00	
IntBal	Integral Balance request		No (0)	
OutLowLimit	Output Low Limit	5189	-999.00	
OutHighLimit	Output High Limit	5190	9999.00	
NumPoints	Number of Selected Points	5191	32	
EditPoint	Insert or Delete Point	5192	0	
In1	Input Point 1	5193	0.00	
Out1	Output Point 1	5194	0.00	
In2	Input Point 2	5195	0.00	
Out2	Output Point 2	5196	0.00	
In3	Input Point 3	5197	0.00	
Out3	Output Point 3	5198	0.00	
In4	Input Point 4	5199	0.00	
Out4	Output Point 4	5200	0.00	
In5	Input Point 5	5201	0.00	
Out5	Output Point 5	5202	0.00	
In6	Input Point 6	5203	0.00	
Out6	Output Point 6	5204	0.00	
In7	Input Point 7	5205	0.00	
Out7	Output Point 7	5206	0.00	
In8	Input Point 8	5207	0.00	
Out8	Output Point 8	5208	0.00	
In9	Input Point 9	5209	0.00	

The function block will automatically skip points that do not follow strictly monotonically increasing order of the 'In' coordinates. If at least one point has been skipped the 'CurveForm' parameter will show 'SkippedPoints'. If no valid interval is found the 'CurveForm' parameter will show 'NoForm' and the Fallback strategy will be applied. Other conditions when the Fallback strategy is applied are input source bad status (e.g. sensor break or sensor over-range) and calculated LIN32 output over-range (i.e. less than OutLowLimit or greater than InHighLimit).

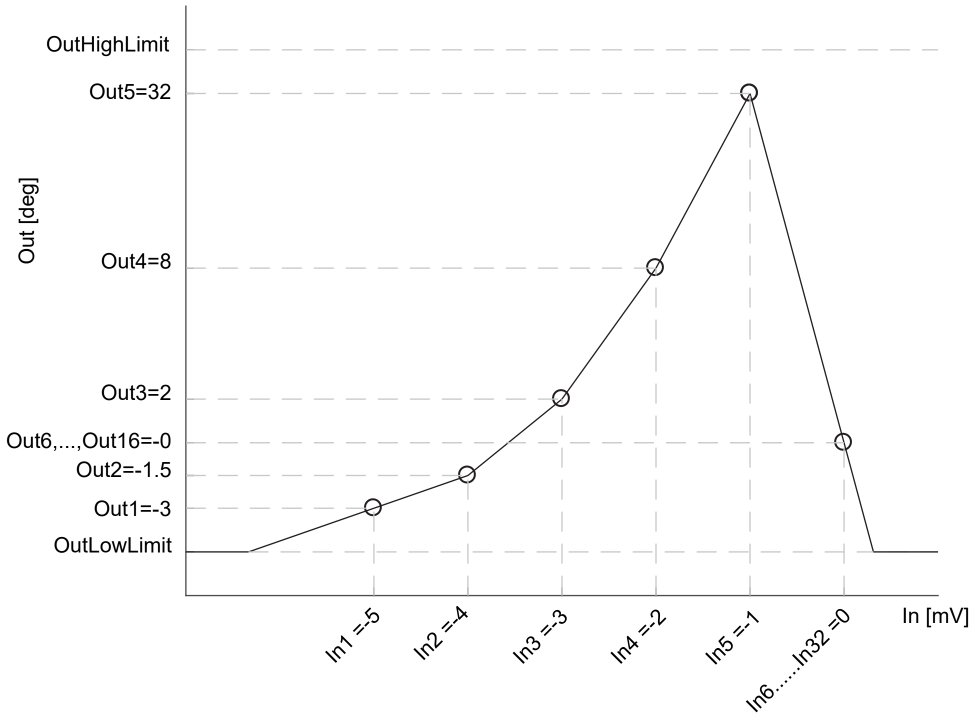
Example 2: Custom Linearization - Skipped Points Curve

If points defaulted to zero have not been deactivated, by reducing 'NumPoints', - AND assuming that at least one of the previous input breakpoints is positive (see the curve below) - then those points will be automatically skipped. The output characteristics will be the same as those obtained by deactivating the points defaulted to zero but the 'CurveForm' will be 'SkippedPoints'.



In1 to In5 will be used. In6 to In32 will be ignored. 'CurveForm' will be 'SkippedPoints'

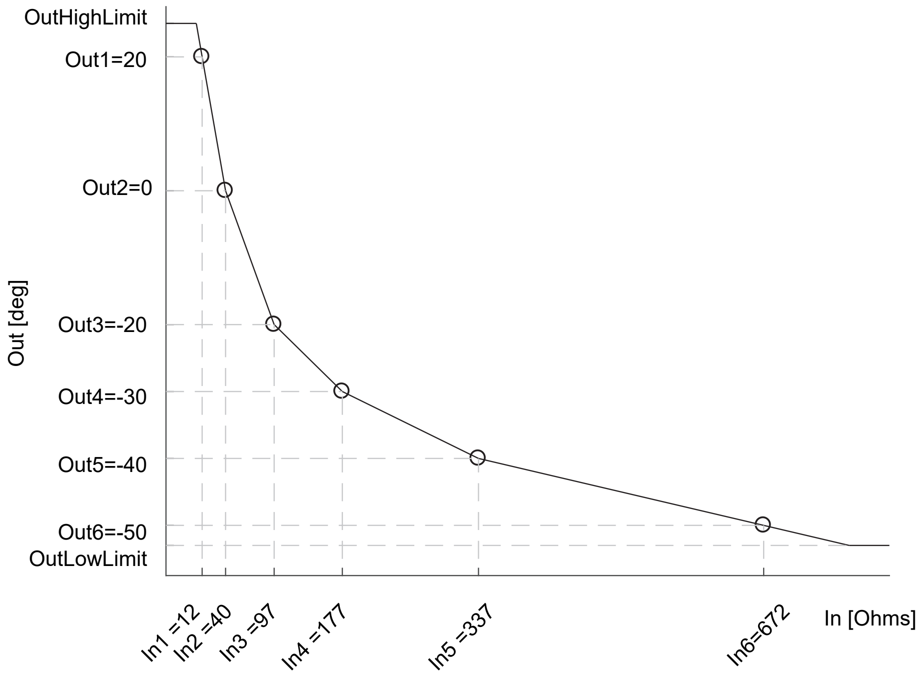
However, when the 'CurveForm' parameter is 'SkippedPoints' (because the number of points 'NumPoints' has not been reduced to the required set) it is not guaranteed that the output characteristics will be increasing or decreasing. In fact, for example, if the input breakpoints are all negative and the final points are zero, then the first "zero" point will be included in the characteristics - see the following picture. Therefore, always set 'NumPoints' to the required value in order to get the expected sensor linearization curve type - increasing, decreasing or free form.



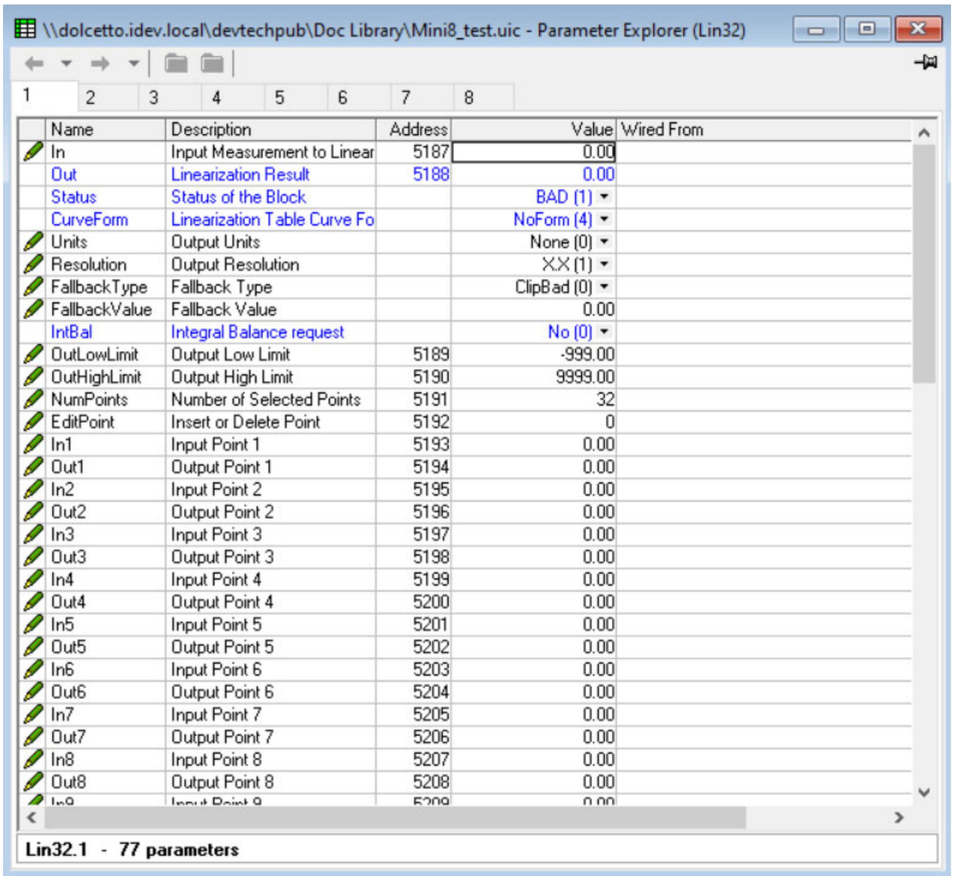
In1 to In5 will be used as well as In6, possibly resulting in a not expected curve. In7, ..., In32 will be ignored. CurveForm will be SkippedPoints.

Example 3: Custom Linearization - Decreasing Curve

The curve may also be a decreasing form as shown below.



The procedure to setup the parameters is the same as in the previous example.

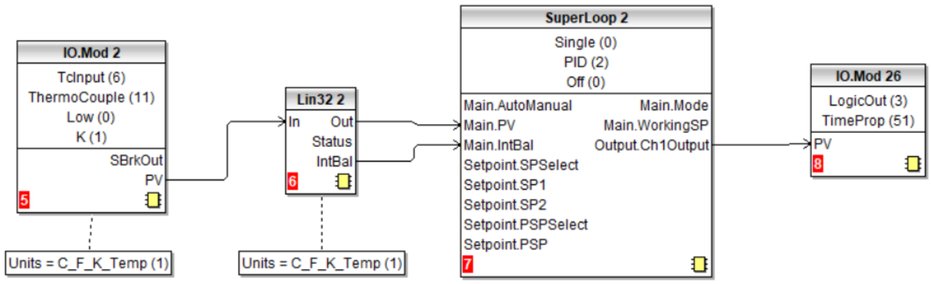


Name	Description	Address	Value	Wired From
In	Input Measurement to Linear	5187	0.00	
Out	Linearization Result	5188	0.00	
Status	Status of the Block		BAD (1)	
CurveForm	Linearization Table Curve Fo		NoForm (4)	
Units	Output Units		None (0)	
Resolution	Output Resolution		XX (1)	
FallbackType	Fallback Type		ClipBad (0)	
FallbackValue	Fallback Value		0.00	
IntBal	Integral Balance request		No (0)	
OutLowLimit	Output Low Limit	5189	-999.00	
OutHighLimit	Output High Limit	5190	9999.00	
NumPoints	Number of Selected Points	5191	32	
EditPoint	Insert or Delete Point	5192	0	
In1	Input Point 1	5193	0.00	
Out1	Output Point 1	5194	0.00	
In2	Input Point 2	5195	0.00	
Out2	Output Point 2	5196	0.00	
In3	Input Point 3	5197	0.00	
Out3	Output Point 3	5198	0.00	
In4	Input Point 4	5199	0.00	
Out4	Output Point 4	5200	0.00	
In5	Input Point 5	5201	0.00	
Out5	Output Point 5	5202	0.00	
In6	Input Point 6	5203	0.00	
Out6	Output Point 6	5204	0.00	
In7	Input Point 7	5205	0.00	
Out7	Output Point 7	5206	0.00	
In8	Input Point 8	5207	0.00	
Out8	Output Point 8	5208	0.00	

Adjustment of the Process Variable

This application allows the user to compensate for known inaccuracies introduced by the overall measurement system. This not only includes the sensor but also the overall measurement chain. Furthermore, this can also be used to derive a different process variable, for instance, a temperature measured in a different place from where the actual sensor is positioned. The adjustment is made directly on the value, and in the units, of the process variable measured by the controller.

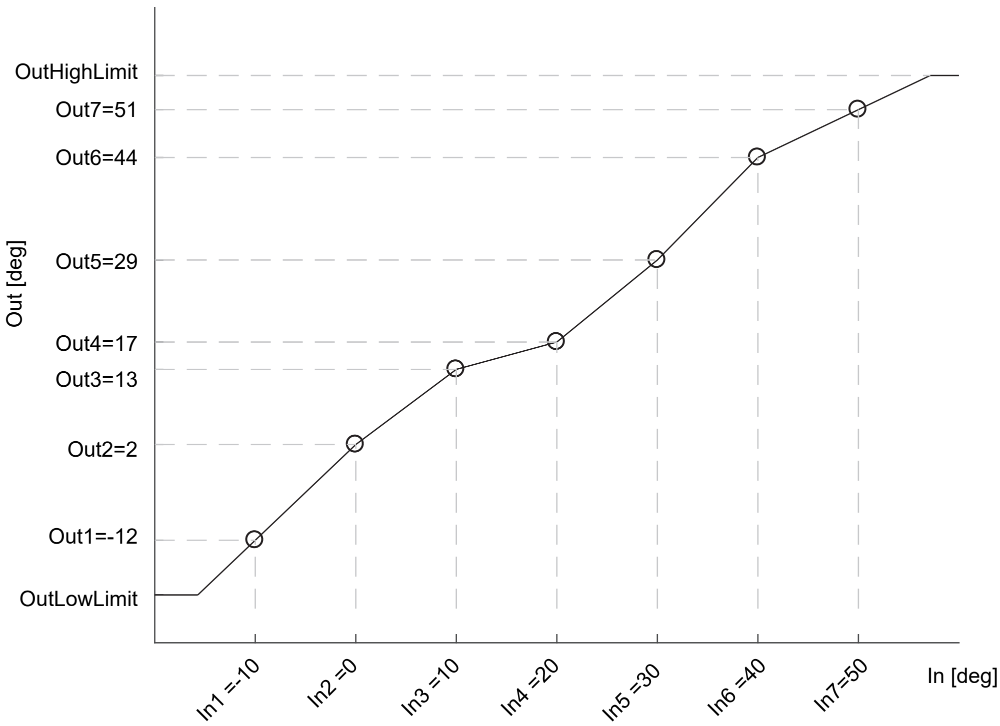
The process variable can be adjusted in different operating conditions (e.g. different temperatures), by using the LIN32 multiple point adjustment curve: this extends the simple PV Offset feature present in the AI block, which just adds or subtracts a single value to the measured PV in all operating conditions.



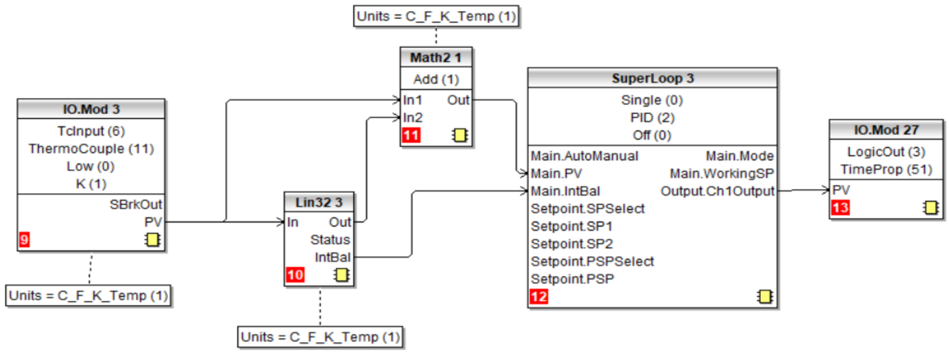
Two alternative configurations can be used:

In the first case the LIN32 table contains the process variable values ‘In1’ to ‘In32’, measured by the controller, and the reference values, ‘Out1’ to ‘Out32’, measured by an external reference.

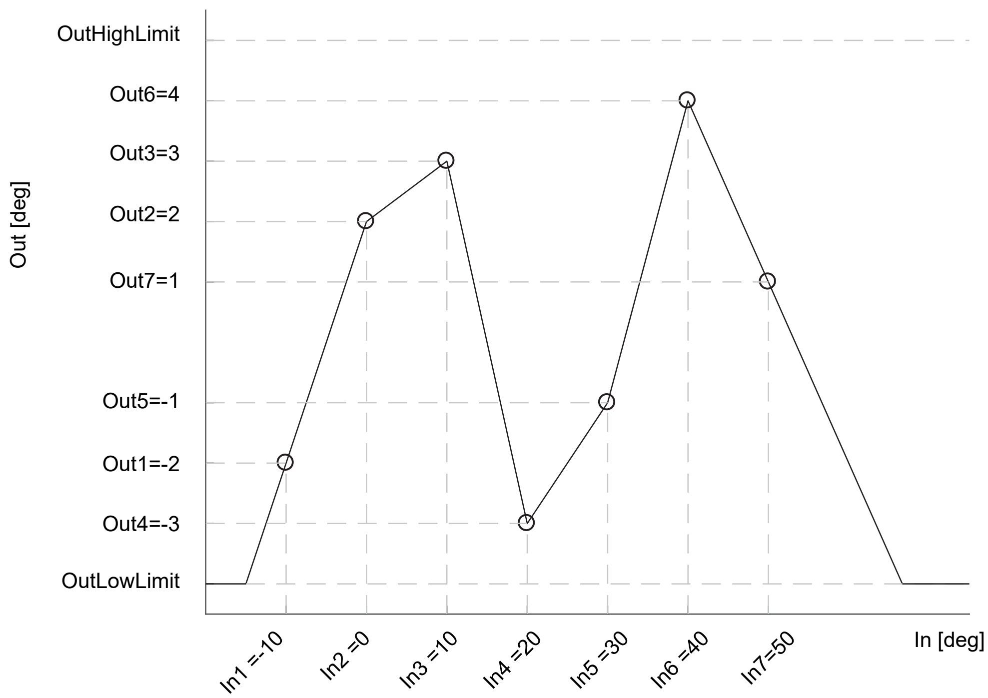
An example is shown below. The same setup procedure detailed before also applies here apart from the different configuration of the AI block. As shown in the graph and in the wiring diagram, the units of both the input and the output of LIN32 are absolute temperatures.



In the second case, for the same application, the LIN32 table stores the offsets between the process variable values measured in the controller and a Math block, set to Add, placed between the Analog Input (AI) and the SuperLoop block. The adjustment is made by adding the offset calculated by the LIN32 block to the measured process variable. In the case of temperature adjustment (and differently from the previous case) the output units of LIN32 should be set to relative temperature. This is in order to select the correct conversion equation when a temperature units change is applied to the offsets (e.g. from degrees Celsius to Fahrenheit).



Because offsets do not follow in general a continuously increasing or decreasing trend, then the ‘CurveForm’ parameter will be ‘FreeForm’, ‘Increasing’ or ‘Decreasing’ depending on their values: see the following graph as an example of a free form offset curve.



Both the two above mentioned configurations provide the control Loop function block with the same adjusted PV. The values are reported in table for the two examples. The high values of the offsets are only to accentuate in the pictures the action of the adjustment.

Input Breakpoints	Output values: absolute temperature	Alternative output values: relative temperature
-10 deg	-12 deg	-2 deg
0 deg	2 deg	2 deg
10 deg	13 deg	3 deg
20 deg	17 deg	-3 deg
30 deg	29 deg	-1 deg
40 deg	44 deg	4 deg
50 deg	51 deg	1 deg

Input Linearization Parameters

Block – Lin32		Sub-blocks: 1 to 8			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
In	Input measurement to linearize. Wire to the source for the custom linearization	Between InLowLimit and InHighLimit		0	Oper
Out	Linearization Result	Between OutLowLimit and OutHighLimit			Read Only
Status	Status of the block. A value of zero indicates a healthy conversion.	Good (0) Bad (1)	Within operating limits A bad output may be caused by a bad input signal (perhaps the input is in sensor break) or an output which is out of range		Read Only
CurveForm	Linearization Table Curve Form	Freeform (0) Increasing (1) Decreasing (2) SkippedPoints (3) NoForm (4)		NoForm	
Units	Units of the linearized output	None (0) C_F_K_Temp (1) V (2) mV (3) A (4) mA (5) PH (6) mmHg (7) psi (8) Bar (9) mBar (10) PercentRH (11) Percent (12) mmWG (13) inWG (14) inWW (15) Ohms (16) PSIG (17) PercentO2 (18) PPM (19) PercentCO2 (20) PercentCarb (21) PercentPerSec (22) RelTemperature (24) Vacuum (25) Secs (26) Mins (27) Hours (28) Days (29) Mb (30 Mb (31) ms (32)			Conf
Resolution	Resolution of the output value	X, X.X, X.XX, X.XXX, X.XXXX			Conf

Block – Lin32		Sub-blocks: 1 to 8			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
FallbackType	Fallback Type The fallback strategy will come into effect if the status of the input value is bad or if the input value is outside the range of input high scale and input low scale. In this case the fallback strategy may be configured as shown:	ClipBad (0)	If the input is outside a limit the output will be clipped to the limit and the status will be BAD	ClipBad	Oper
		ClipGood (1)	If the input is outside a limit the output will be clipped to the limit and the status will be GOOD		
		FallBad (2)	The output value will be the fallback value and the output status will be BAD		
		FallGood (3)	The output value will be the fallback value and the output status will be GOOD		
		UpScaleBad (4)	The output value will be output high scale and the output status will be BAD		
		DownScaleBad (5)	The output value will be the output low scale and the output status will be BAD		
Fallback Value	In the event of a bad status, the output may be configured to adopt the fallback value. This allows the strategy to dictate a 'safe' output in the event of a detected fault.			0	Oper
IntBal	Integral Balance Request	No (0) Yes (1)		No	
OutLowLimit	Adjust to correspond to the low input value	-99999 to OutHighLimit		0	Conf
OutHighLimit	Adjust to correspond to the high input value	OutLowLimit to 99999		0	Conf
NumPoints	Number of Selected Points				
EditPoint	Insert or Delete Points				
In1	Adjust to the first break point			0	Oper
Out1	Adjust to correspond to input 1			0	Oper
…etc up to				0	
In32	Adjust to the last break point			0	Oper
Out32	Adjust to correspond to input 32			0	Oper

The 32 point linearization does not require you to use all 32 points. If fewer points are required, then the curve can be terminated by setting the first unwanted value to be less than the previous point.

Conversely if the curve is a continuously decreasing one, then it may be terminated by setting the first unwanted point above the previous one.

Polynomial

Block – Poly		Sub-blocks: 1 to 2			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
LinType	To select the input type. The linearization type selects which of the instruments linearization curves is applied to the input signal. The instrument contains a number of thermocouple and RTD linearizations as standard. In addition there are a number of custom linearizations that may be downloaded using iTools to provide linearizations of non-temperature sensors.	J (0) K (1) L (2) R (3) B (4) N (5) T (6) S (7) PL2 (8) C (9) PT100 (10) Linear (11) PT1000 (12) SqRoot (14) Cust1 (20) Cust2 (21) Cust3 (22)		J	Conf
Resolution	Resolution of the output value	X, X.X, X.XX, X.XXX, X.XXXX		X	Conf
In	Input Value The input to the linearization block	Range of the input wired from			Oper
Out	Output value	Between Out Low and Out High			Read Only
InHighScale	Input high scale	In Low to 99999		0	Oper
InLowScale	Input low scale	-99999 to In High		0	Oper
OutHighScale	Output high scale	Out Low to 99999		0	Oper
OutLowScale	Output low scale	-99999 to Out High		0	Oper
FallbackValue	Value to be adopted by the output in the event of Status = Bad				Oper
Status	Indicates the status of the linearized output:	Good (0)	Good indicates the value is within range and the input is not in sensor break.		Read Only
		ChanneOff (1)			
		OverRange (2)	Indicates the Value is over range		
		UnderRange (3)	Indicates the Value is under range.		
		HardwareStatusInvalid (4)			
		Ranging (5)			
		Overflow (6)			
		Bad (7)	Indicates the Value is out of range or the input is in sensor break. **Note:** This is also effected by the configured fallback strategy		
		HWExceeded (8)			
		NoData(9)			

Control Loop Setup

What is a Control Loop?

An example of a heat-only temperature control loop is shown below:

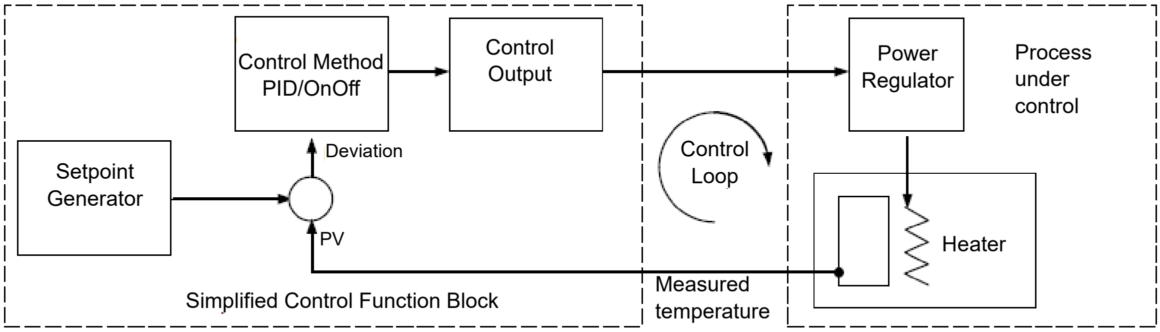


Figure 91 Single Loop Single Channel

The actual temperature measured at the process (PV) is connected to the input of the controller. This is compared with a setpoint (or required) temperature (SP). If there is a deviation between the set and measured temperature the controller calculates an output value to call for heating or cooling. The calculation depends on the process being controlled but normally uses a PID algorithm. The output(s) from the controller are connected to devices on the plant which cause the heating (or cooling) demand to be adjusted which in turn is detected by the temperature sensor. This is referred to as the control loop.

Types of Control Loop (SuperLoop and Legacy Loop)

SuperLoop

SuperLoop is the latest Eurotherm control loop and provides single and cascade loops in a single function block. It is the default control loop in the Mini8 loop controller Firmware 5.0+.

Legacy Loop

The legacy ‘Loop’ is provided for compatibility with older Mini8 loop controller applications. It may be specified at time of order. Cascade functions are not available for legacy loop.

SuperLoop - Single Loop Control

The Eurotherm SuperLoop can be configured to operate in Single Loop mode by setting the **LoopType** parameter as Single.

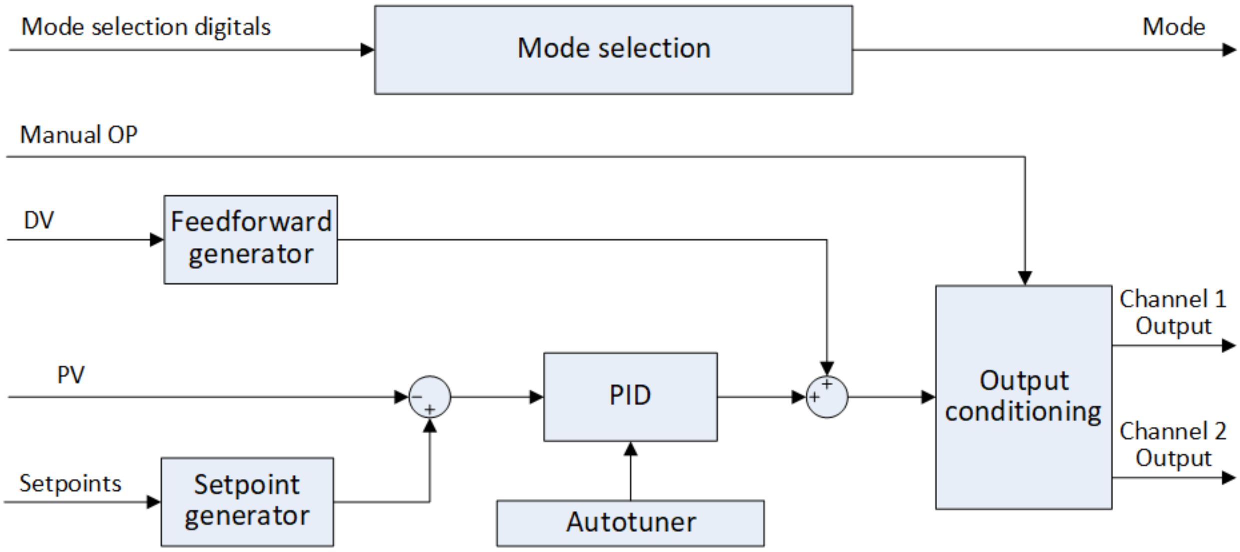


Figure 92 SuperLoop in Single loop control configuration (LoopType = Single).

In this configuration:

- The PID control algorithm drives the controller output to minimize the difference between the selected setpoint and the process variable (PV).
- The possible Loop modes go from Hold Inhibit to Auto (Cascade, Primary Tune and Forced Auto are not selectable). Refer to "Start-up and Recovery" on page 219 for the mode transition mechanism.
- The PID control algorithm drives the controller output to minimize the difference between the selected setpoint and the process variable (PV).
- The possible Loop modes go from Hold Inhibit to Auto (Cascade, Primary Tune and Forced Auto are not selectable). Refer to "Start-up and Recovery" on page 219 for the mode transition mechanism.
- Setpoint generator produces the target for the PV, process variable from a set of setpoint sources - e.g. local setpoints, remote setpoint, programmer setpoint.
- The Output conditioning block processes the target controller output by applying various algorithms and criteria and splits it into two channels - typically in temperature control applications heat and cool channels. It also manages manual, track and hold output modes.
- For automatic fine-tune of the PID terms, the Eurotherm autotune algorithm can be used in commissioning.
- Through the Feedforward generator an additional open-loop component can be added to the target output, which depends on a selectable disturbance variable.

SuperLoop - Cascade Loop Control

The Eurotherm SuperLoop can be configured to operate in Cascade Loop mode by setting the **LoopType** parameter as Cascade. In this configuration it can control a process with two functionally and dynamically interdependent process variables – primary PV and secondary PV – via one or two output channels:

- The primary PV is typically characterized by the slowest dynamics, such as the temperature of a furnace or the temperature of a workload in the furnace.
- The secondary PV is typically associated to an actuator such as a heating element.
- In temperature control applications the output channels are typically heat and cool channels, driving the demand to the actuators.

The simultaneous automatic control of the two PVs is achieved by a cascade of two PID loops:

- A primary loop where primary PID controls the primary PV to the user selected setpoint by driving the secondary loop;
- A secondary loop where the secondary PID controls the secondary PV to the setpoint driven by the primary PID.

Figure 93 shows a simplified view of the internal functional blocks of the SuperLoop in cascade configuration.

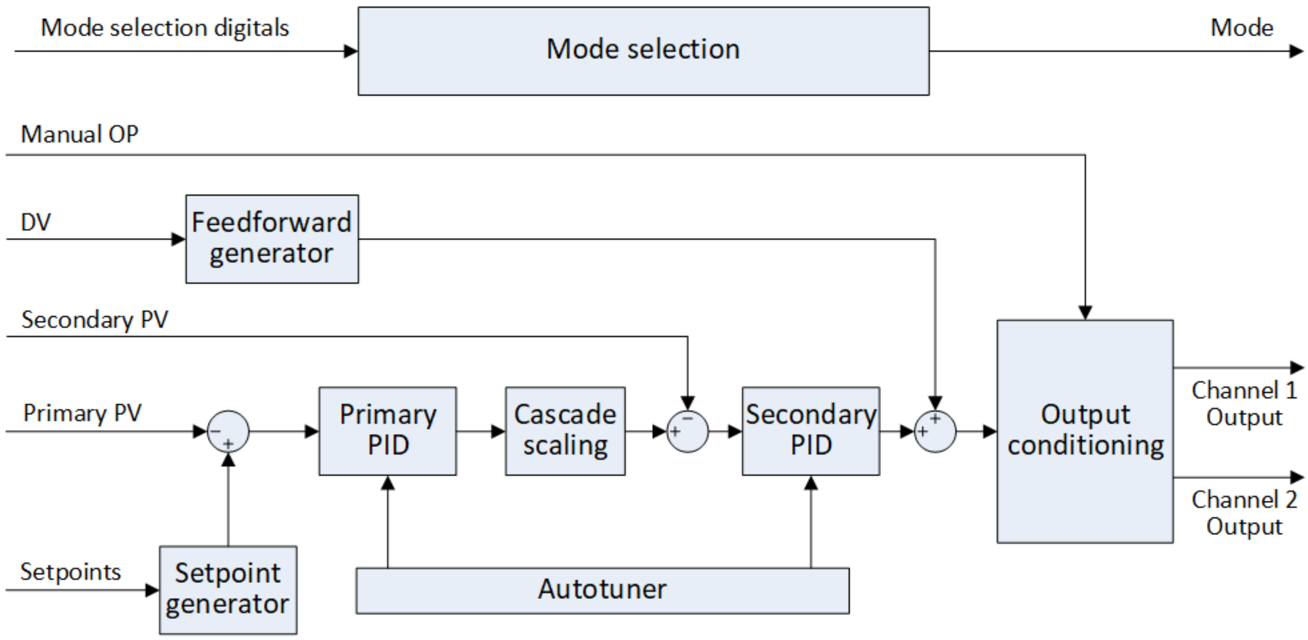


Figure 93 SuperLoop in Cascade Mode

- The Mode selection manages the transition between operative modes depending on mode selection digitals (e.g. AutoManual, CascadeMode, Inhibit) and other input flags and statuses. Refer to "Start-up and Recovery" on page 219 for the mode transition mechanism.
- The Setpoint generator produces the setpoint - Primary Working SP - for the primary process variable from a set of setpoint sources - e.g. local, remote, programmer setpoint.

- The Primary PID minimizes the difference between the selected setpoint and the Primary process variable by driving the Secondary setpoint.
- The Cascade scaling block converts Primary PID output into Secondary process variable units and generates the Secondary setpoint.
- The Secondary PID minimizes the difference between the Secondary process variable and the automatically generated Secondary setpoint by producing the target output.
- For automatic fine-tune of the PID terms, the Eurotherm autotune algorithm can be used both for Primary PID tuning and for Secondary PID tuning.
- Output conditioning blocks operate as in Single Loop type, described in the previous section.

There are two types of cascade control: full-scale and trim cascade types. The cascade configuration can be set via the **CascadeType** parameter.

Full Scale Cascade Type

If the engineering units used in the primary and secondary loops are not the same, full-scale mode is usually the most suitable solution. It is simple to set up, because the secondary setpoint range is already defined by the secondary range limits, **RangeHighLimit** and **RangeLowLimit**.

The following block diagram shows a simplified structure for a full-scale type cascade control system.

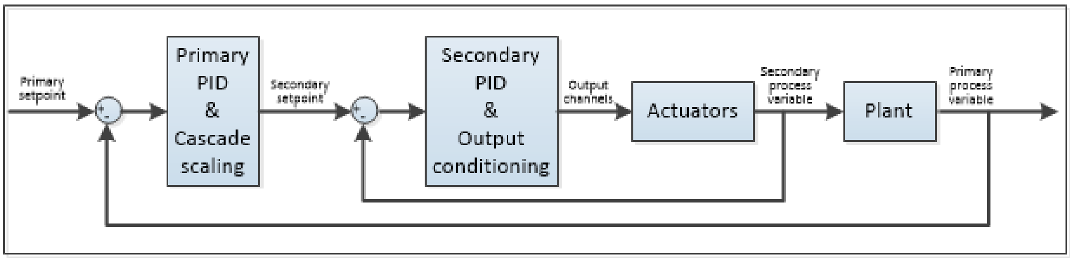


Figure 94 Full-scale Type Cascade Control System

Trim Cascade Type

If the engineering units used in the primary and secondary loops are the same, for instance in heating applications, trim cascade type mode is typically adopted.

In trim cascade type configuration, one of either the Primary SP, the Primary PV, or a Remote SP can be selected as main component of the secondary setpoint, via the **SecondarySPType** parameter. The Primary PID trims the main component to minimize the deviation between primary PV and its setpoint by adding an adjustment component to the secondary SP which spans between the trim range: **TrimRangeLow**, **TrimRangeHigh**.

The following block diagrams show the simplified structure for Primary SP and Primary PV trim type cascade control system.

- **SecondarySPType** = **PrimarySP** is selected in applications where the response speed is the priority and the actuators can be driven at maximum power without causing any damage to the plant. The response is accelerated by passing directly the primary SP to the secondary PID, on top of which the primary PID adds its adjustment component.

- **SecondarySPType** = **PrimaryPV** is selected in applications where the secondary process variable must change gradually to avoid damage to the plant, e.g. where thermal shock must be avoided. The actuator speed is controlled automatically by the dynamics of the plant itself, by deriving the main component of the secondary SP from the plant primary PV. The user can further limit the primary PID trim component added to the secondary SP within the trim range: **TrimRangeLow**, **TrimRangeHigh**.

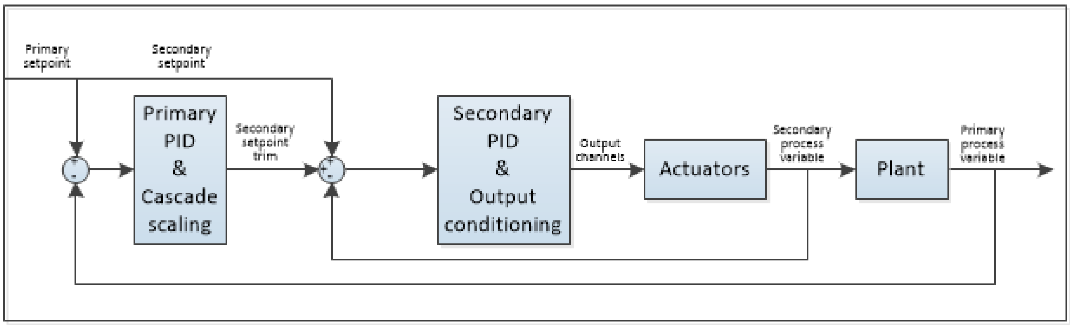


Figure 95 Cascade control system in Trim configuration
(CascadeType = Trim, SecondarySPType = PrimarySP)

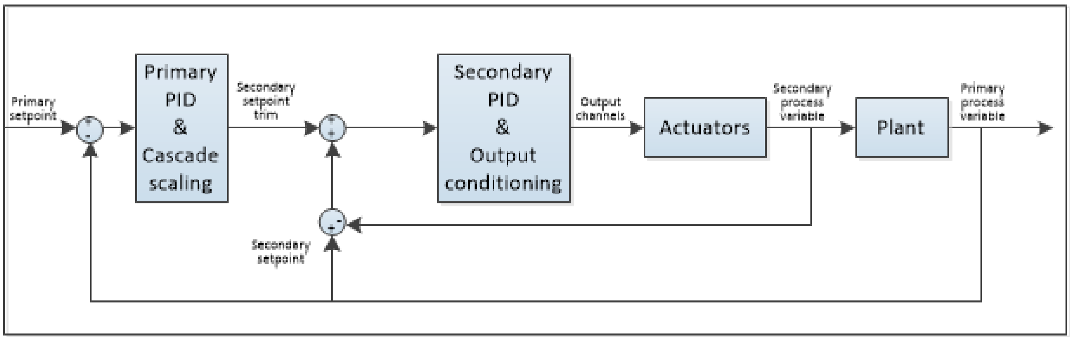


Figure 96 Cascade control system in Trim configuration
(CascadeType = Trim, SecondarySPType = PrimaryPV)

For applications where the two PVs have same units but an external source makes the steady-state deviation between secondary PV and primary PV not easily predictable, it might be challenging to establish the amount of SP trim that needs to be added to the main secondary SP component to reach the primary SP operating point. In these specific situations, e.g. in case of interactive multi-zone furnaces, Full-scale cascade type can be selected to make the primary loop drive the secondary SP within the whole secondary range.

Operating Modes

SuperLoop has a number of possible operating modes. It is quite possible for several modes to be requested by the application at once. The active mode is therefore determined by a priority model, whereby the mode with the highest priority will always win.

The Operating Modes are listed and detailed in the description of the parameter **Main.Mode** listed in "Main Parameters" on page 234. Figure 97 to Figure 99 show the selection criteria of modes and their priority:

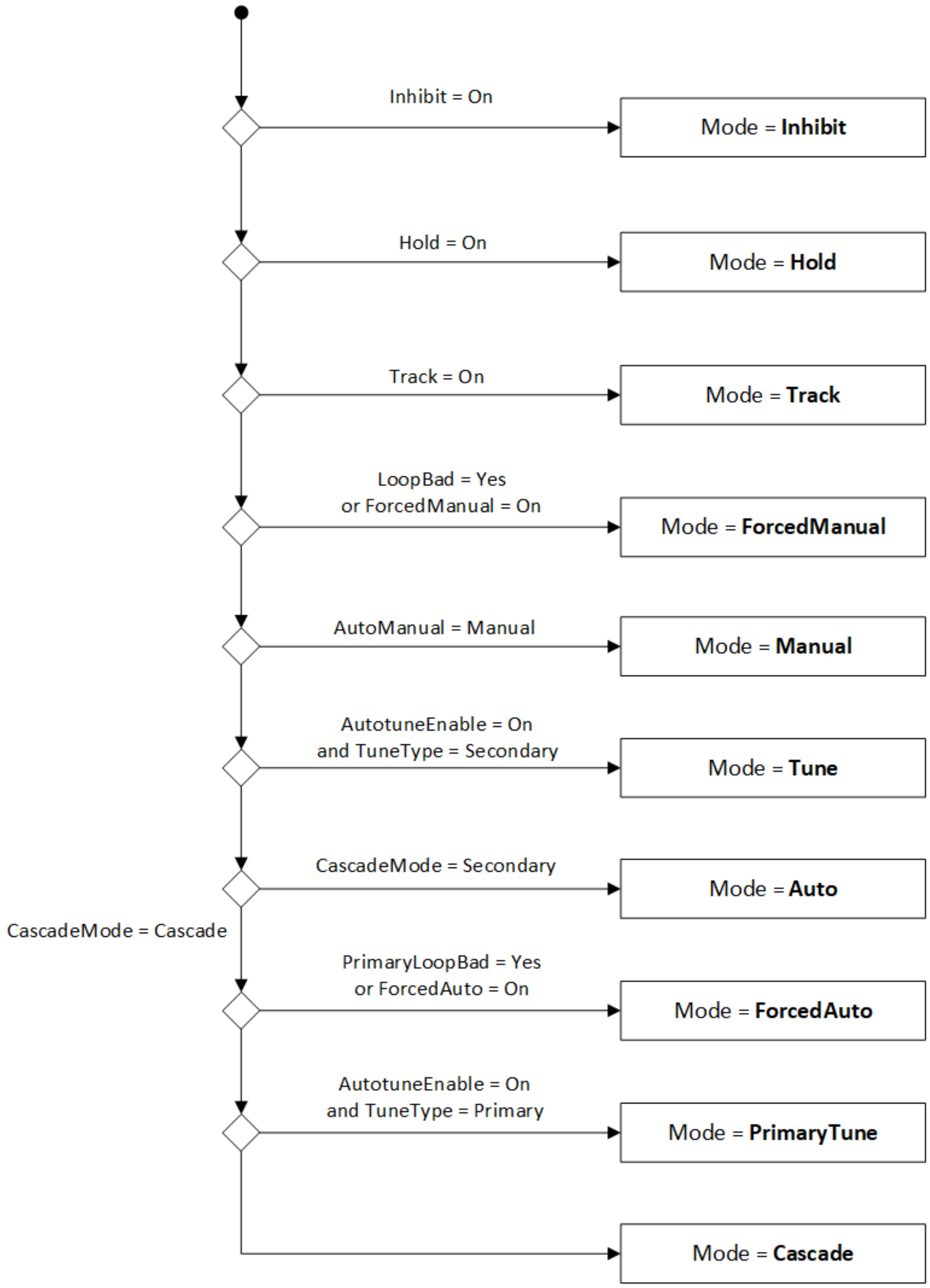


Figure 97 Mode selection diagram of SuperLoop in Cascade loop configuration (**LoopType** = Cascade) during operation.

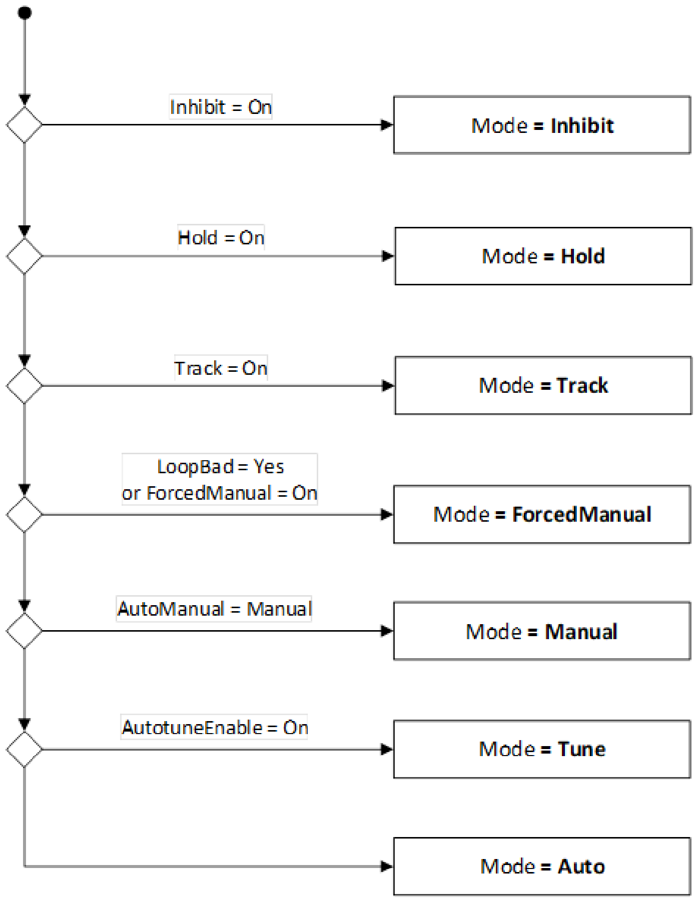


Figure 98 Mode selection diagram of SuperLoop in Single loop configuration (**LoopType** = Single) during operation

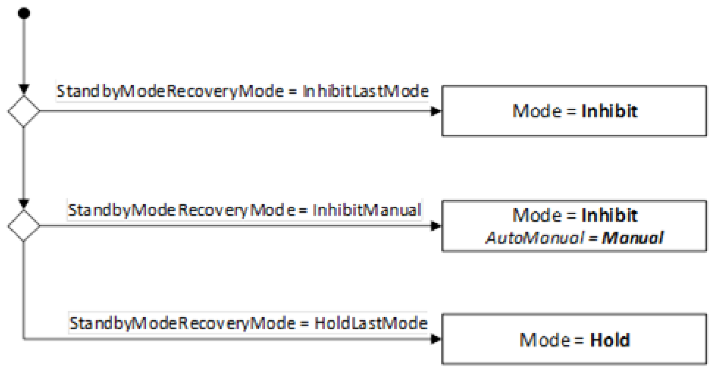


Figure 99 Mode selection diagram of SuperLoop during Config and Standby modes

Types of Control

Two types of channel output may be configured. These are PID control and On/Off control.

PID Control

Primary controller and Secondary controller feature the Eurotherm PID control algorithm.

PID, also referred to as 'Three Term Control', is an algorithm which continuously adjusts the output, according to a set of rules, to compensate for changes in the process variable. It provides more stable control than On/Off control but the parameters need to be set up to match the characteristics of the process under control.

Output term	Depends on:	Tuning parameter
ProportionalOP	PV deviation from WorkingSP	Proportional band (Engineering units or percent)
IntegralOP	Duration of the PV deviation	Integral time (seconds)
DerivativeOP	Rate of change of PV (default) or PV deviation	Derivative time (seconds)

The PID tuning parameters can be

- gain scheduled activating one of the available Gain Scheduler strategies (Manual set, automatic set dependent on an internal or remote scheduling variable, etc.).
- autotuned using the autotune algorithm.

The following forms can be activated by manually changing the tuning parameters:

Controller type	Proportional band	Integral time	Derivative time
PID	> 0	> 0	> 0
PI	> 0	> 0	= 0
PD	> 0	= 0	> 0
P	0	= 0	= 0

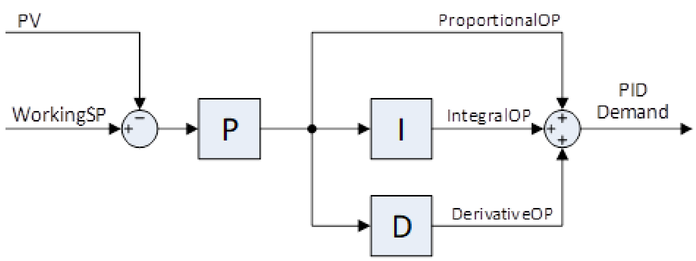


Figure 100 Eurotherm PID algorithm with derivative on deviation (**DerivativeType** = Deviation)

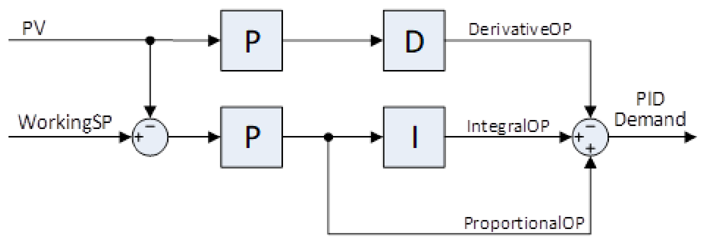


Figure 101 Eurotherm PID algorithm with derivative on PV (**DerivativeType** = PV)

The Eurotherm PID algorithm is based upon an ISA type algorithm in its positional (non-incremental) form. The ISA form is a gain-dependent parallel form where the proportional term (the proportional band) defines the gain of the overall controller. The ISA form is not to be confused with a gain-independent form where the three terms are completely independent.

It is possible to turn off integral and derivative terms and control on proportional only (P), proportional plus integral (PI) or proportional plus derivative (PD).

An example of where PI control might be used i.e. D is turned off, is process plants (flows, pressures, liquid levels), which are inherently turbulent and noisy, causing valves to fluctuate wildly.

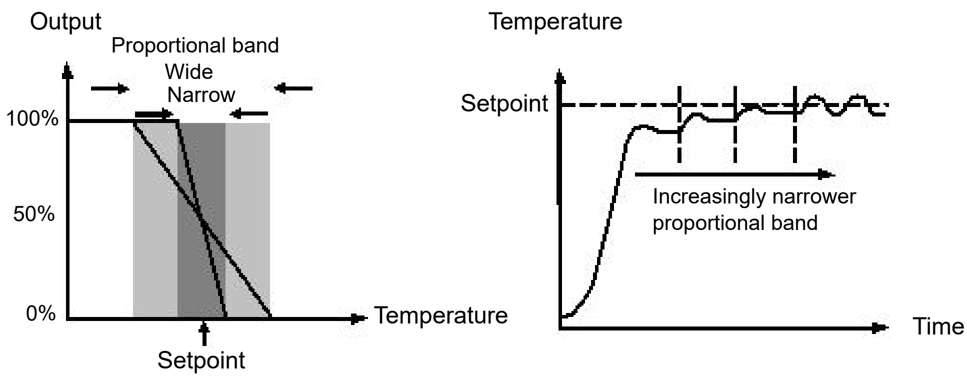
PD control may be used, for example, on servo mechanisms.

In addition to the three terms described above, there are other parameters which determine how well the control loop performs. These include High and Low Cutback and Manual Reset and are described in detail in subsequent sections.

Proportional Term 'PB'

The proportional term, or gain, delivers an output which is proportional to the size of the difference between SP and PV. It is the range over which the output power is continuously adjustable in a linear fashion from 0% to 100% (for a heat only controller). Below the proportional band the output is full on (100%), above the proportional band the output is full off (0%) as shown in the diagram below.

The width of the proportional band determines the magnitude of the response to the deviation. If it too narrow (high gain) the system oscillates by being over responsive. If it is too wide (low gain) the control is sluggish. The ideal situation is when the proportional band is as narrow as possible without causing oscillation.



The diagram also shows the effect of narrowing proportional band to the point of oscillation. A wide proportional band results in straight line control but with an appreciable initial deviation between setpoint and actual temperature. As the band is narrowed the temperature gets closer to setpoint until finally becoming unstable.

The proportional band can be specified in engineering units or in percent of span (**RangeHigh** – **RangeLow**). Engineering Units is recommended for its ease of use.

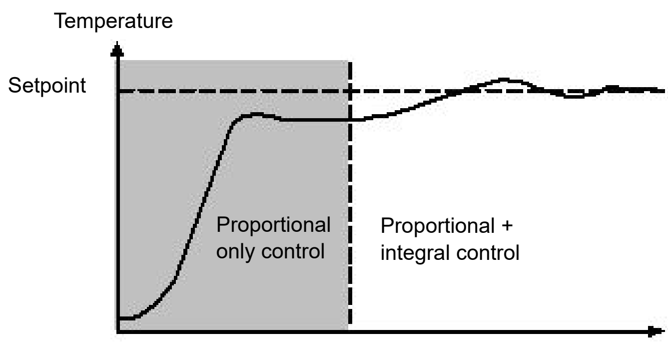
Previous controllers had the parameter Relative Cool Gain (R2G) to adjust the cool proportional band relative to the heat. This has been replaced by separate proportional bands for Channel 1 (Heat) and Channel 2 (Cool).

Integral Term 'TI'

In a proportional only controller, a difference between setpoint and PV must exist for the controller to deliver power. Integral is used to reduce this to a zero steady state control.

The integral term slowly shifts the output level as a result of a difference between setpoint and measured value. If the measured value is below setpoint the integral action gradually increases the output in an attempt to correct the difference. If it is above setpoint, integral action gradually decreases the output or increases the cooling power to correct the difference.

The diagram below shows the result of introducing integral action.



The units for integral are measured in time. The longer the integral time constant, the more slowly the output is shifted and results in a sluggish response. Too small an integral time will cause the process to overshoot and even oscillate. The integral action may be deactivated by setting its value to Off(0), in which case manual reset will be made available.

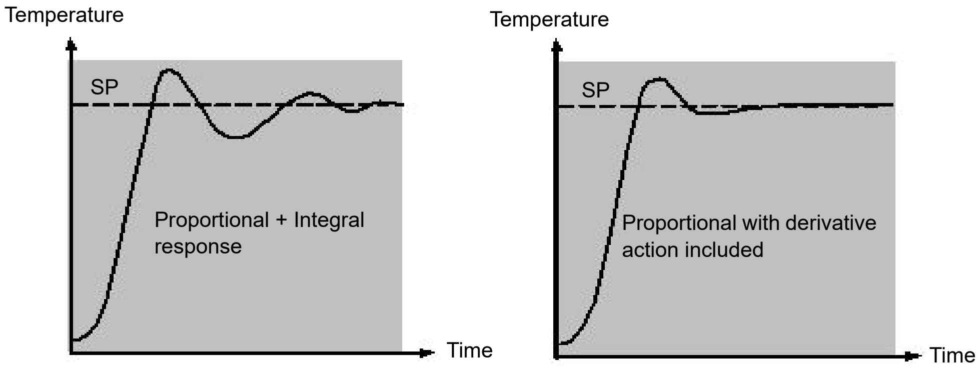
The integral time is always specified in seconds. In US nomenclature, the integral time is equivalent to ‘seconds per repeat’.

Integral Hold

When the **IntegralHold** parameter is turned on, the output value contained in the integrator will be frozen. It will be maintained even through mode changes. This can sometimes be useful e.g. in a cascade to stop the primary integral winding up when the secondary is saturated.

Derivative Term 'TD'

Derivative action, or rate, provides a sudden shift in output as a result of a rapid change in deviation. If the measured value falls quickly, derivative provides a large change in output in an attempt to correct the perturbation before it goes too far. It is most beneficial in recovering from small perturbations.



The derivative modifies the output to reduce the rate of change of the difference. It reacts to changes in the PV by changing the output to remove the transient. Increasing the derivative time will reduce the settling time of the loop after a transient change.

Derivative is often mistakenly associated with overshoot inhibition rather than transient response. In fact, derivative should not be used to curb overshoot on start up since this will inevitably affect the steady state performance of the system. Overshoot inhibition is best left to the approach control parameters, High and Low Cutback, described below.

Derivative is generally used to increase the stability of the loop, however, there are situations where derivative may be the cause of instability. For example, if the PV is electrically noisy, then derivative can amplify that electrical noise and cause excessive output changes, in these situations it is often better to deactivate the derivative and re-tune the loop.

Derivative time is always specified in seconds. Derivative action can be turned off by setting the derivative time to Off(0).

Derivative on PV or Deviation (SP - PV)

By default, derivative action is applied to the PV only and not to the deviation (SP - PV). This helps to prevent large derivative kicks when the setpoint is changed.

If required, derivative can be switched to deviation using the DerivativeType parameter. This is not usually recommended but can, for example, reduce overshoot at the end of SP ramps.

Manual Reset (PD Control)

In a full three-term controller (that is, a PID controller), the integral term automatically removes the steady state deviation from the setpoint. Turn off the integral term to set the controller to PD Under these conditions the measured value may not settle precisely at setpoint. The **ManualReset** parameter (MR) represents the value of the power output that will be delivered when the deviation is zero.

This value must be set manually in order to remove the steady state deviation.

Cutback

Cutback is a system of approach control for process start-up and for large setpoint changes. It allows the response to be tuned independently of the PID controller, thereby allowing optimum performance for both large and small setpoint changes and disturbances. It is available for all control types except OnOff.

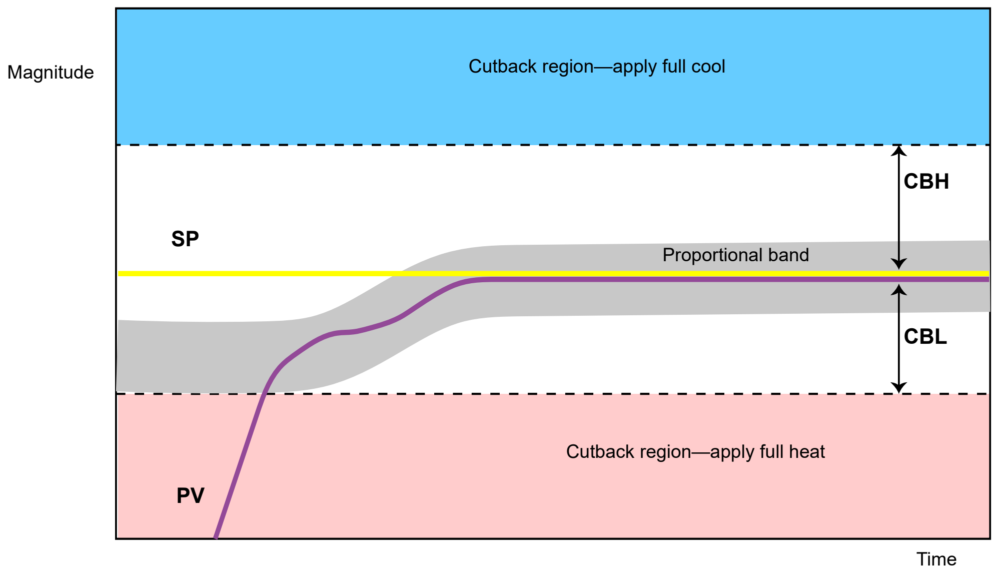
The cutback high and low thresholds, CBH and CBL, define two regions above and below the working setpoint (WSP). They are specified in the same units as the proportional band. Operation can be explained in three rules:

1. When the PV is more than *CBL* units *below* WSP, *maximum* output is always applied.
2. When the PV is more than *CBH* units *above* WSP, *minimum* output is always applied.
3. When the PV exits a cutback region, the output is returned *bumplessly* to the PID algorithm.

The effect of rule 1 and 2 is to bring the PV towards the WSP as rapidly as possible whenever there exists a significant deviation, just as an experienced operator might do manually.

The effect of rule 3 is to allow the PID algorithm to immediately start 'cutting back' the power from maximum or minimum when the PV passes the cutback threshold. Remember that, due to 1 and 2, the PV should be moving rapidly towards WSP, and it is this that causes the PID algorithm to start cutting back the output.

By default, CBH and CBL are set to *Auto (0)*, which means that they are automatically taken to be three times the proportional band. This is a reasonable starting point for most processes, but rise time to setpoint on start-up or large setpoint changes may be improved by tuning them manually.



Notes:

1. Because cutback is a type of non-linear controller, a set of CBH and CBL values that are tuned for one particular operating point may not be satisfactory for another operating point. It is always advisable therefore not to try to tune the cutback values *too* tightly, or otherwise to use gain scheduling to schedule different values of CBH and CBL at different operating points. All the PID tuning parameters can be gain scheduled.
2. Cutback is available for both primary and secondary PID algorithms.

Reverse/Direct Action

For single-channel loops, the concept of reverse and direct action is important.

The **ControlAction** parameter should be set appropriately:

1. If an increase in control output causes a corresponding increase in PV, such as in a heating process, then set **ControlAction** to Reverse (0).
2. If an increase in control output causes a corresponding decrease in PV, such as in a refrigeration process, then set **ControlAction** to Direct (1).

The **ControlAction** parameter is not available for split-range configurations, where channel 1 is always reverse acting and channel 2 is always direct acting.

Notes:

1. The **PrimaryControlAction** parameter should also be set.
2. Reverse/Direct Action setting is also available for the primary loop, via **PrimaryControlAction** setting.

Loop Break

The loop is considered to be broken if the PV does not respond to a change in the output. An alarm may be initiated but in Mini8 loop controllers this must be explicitly wired using the **LoopBreak** parameter. Since the time of response will vary from process to process the **LoopBreakTime** parameter allows a time to be set before a Loop Break Alarm is initiated. In these circumstances the output power will drive to high or low limit. For a PID controller, two parameters under diagnostics are used to determine if the loop is broken, **LoopBreakTime** and **LoopBreakDeltaPV**.

If the control loop is broken, the output will tend to wind up and eventually hit a limit.

Once the output is at the limit, the loop break detection algorithm will monitor the PV. If the PV has not moved by a specified amount (**LoopBreakDeltaPV**) in twice the specified time (**LoopBreakTime**), then a loop break will be flagged.

There are also equivalent parameters for Primary Loop:

- **PrimaryLoopBreak**
- **PrimaryLoopBreakTime**
- **PrimaryLoopBreakDeltaPV**

Gain scheduling

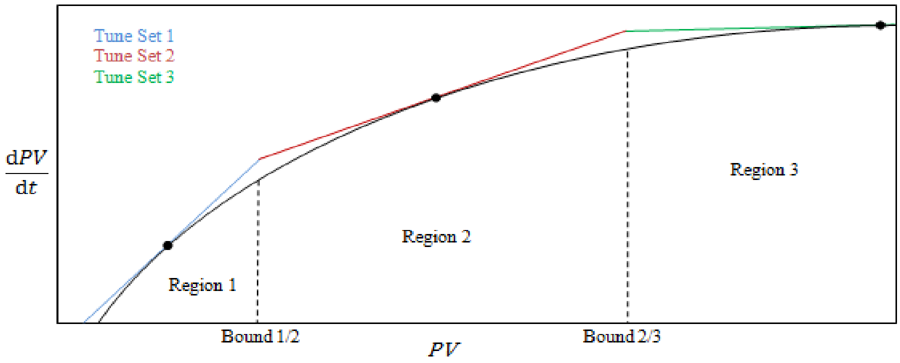
Note: Valid for both Primary and Secondary PID in Cascade Loop type.

Some processes exhibit non-linear dynamics. For example, a heat treatment furnace may behave quite differently at low temperatures than at high temperatures. This is commonly due to the effects of radiant heat transfer, which start to appear above about 700℃. This is illustrated in the diagram below.

It is often not feasible, then, for a single set of PID tuning constants to perform well over the entire process operating range. To combat this, several sets of tuning constants can be used and ‘scheduled’ according to the process operating point.

Each set of constants is called a ‘gain set’ or a ‘tune set’. The gain scheduler selects the active gain set by comparing the value of the Scheduling Variable (SV) against a set of boundaries.

An integral balance is issued whenever the active gain set changes. This helps to prevent discontinuities ('bumps') in the controller output.



On-Off Control

Each of the two output channels can be configured for On-Off control. This is a simple type of control often found in basic thermostats.

The control algorithm takes the form of a simple hysteretic relay.

For channel 1 (heat):

1. When PV > WSP, OP = 0%
2. When *PV* < (*WSP* – *Ch1OnOffHyst*), OP = 100%

For channel 2 (cool):

1. When *PV* > (*WSP* + *Ch2OnOffHyst*), OP = 100%
2. When *PV* < *WSP*, OP = 0%

This form of control will lead to oscillation about setpoint but it is by far the easier to tune. The hysteresis should be set according to the trade-off between oscillation amplitude and actuator switching frequency. The two hysteresis values can be gain scheduled.

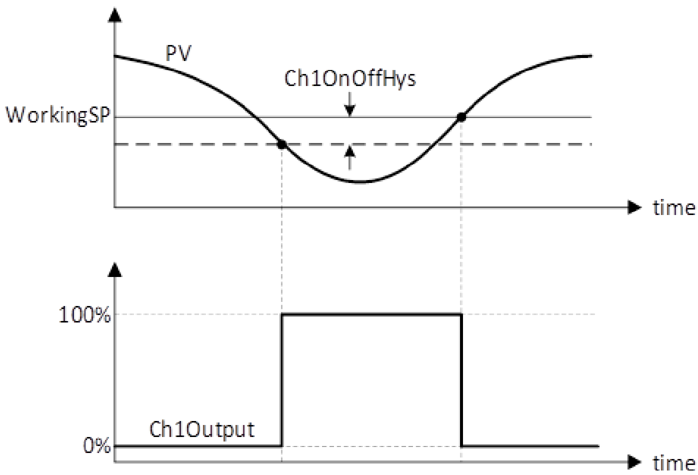


Figure 102 On Off algorithm for channel 1 output

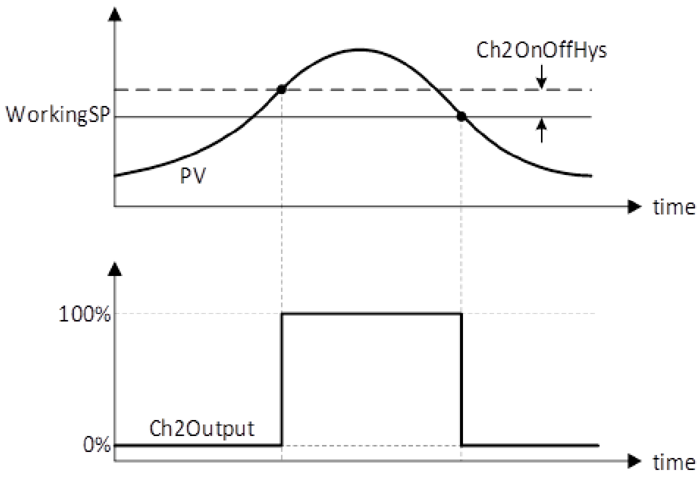


Figure 103 On Off algorithm for channel 2 output

Feedforward

A limitation of a PID control strategy is that it responds only to deviations between PV and SP. By the time a PID controller first starts to react to a process disturbance, it might be already too late and the disturbance is in progress; all that can be done is to try to minimize the extent of the disruption as much as possible. Feedforward control is often used to overcome this limitation. It uses a measurement of the disturbance variable itself and a priori knowledge of the process to predict the controller output that will exactly counter the disturbance before it has a chance to affect the PV.

SuperLoop incorporates a feedforward controller in addition to the normal feedback (PID) controller; it is capable of static or dynamic feedforward compensation. Broadly, there are three common uses for feedforward in these instruments and are described in turn below.

Feedforward on its own also has a major limitation. It is an open-loop strategy that relies entirely on the *a priori* knowledge of the process. Feedforward tuning deviation, uncertainty and process variation all help to prevent zero tracking deviation being achieved in practice.

Further, the feedforward controller can only respond to disturbances that are explicitly measured and whose effect is known.

To counter the relative disadvantages, the SuperLoop combines both types of control in an arrangement known as "Feedforward with Feedback Trim". The Feedforward controller provides the principal control output and the PID controller can trim this output appropriately to give zero tracking deviation.

The diagram below shows the feedforward with feedback trim structure.

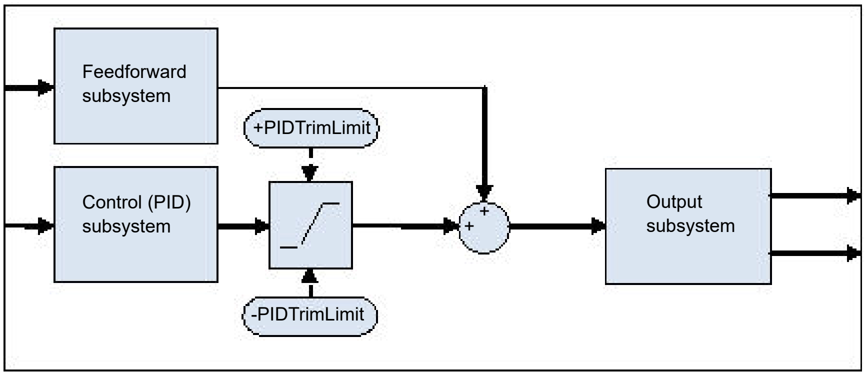


Figure 104 Feedforward with Feedback Trim

The block diagram for the feedforward generator structure is shown in Figure 105. It is capable of static and dynamic feedforward compensation using, as input, various sources: Remote measured disturbance variable DV, Secondary or Primary working setpoint, Secondary or Primary process variable.

Remote DV is used as feedforward input when the effect of a disturbance on the plant is known and therefore the feedforward static and dynamic parameters can be tuned to generate an output demand signal that compensate the disturbance effect. The static feedforward parameters **FFGain** and **FFOffset** can be found by characterizing the steady-state effect of the disturbance of the output demand via:

ΔOPss = FFGain * DV + FFOffset,

where the ΔOPss is the deviation of steady-state output demand due to DV.

Secondary or Primary working setpoint is used as feedforward input when the output demand for a certain target setpoint is known and therefore the feedforward static parameters can be tuned to generate an output demand equal to the steady-state value. The static feedforward parameters **FFGain** and **FFOffset** can be tuned by characterizing the steady-state characteristic of the plant via:

OPss = FFGain * SP + FFOffset

where OPss is the output demand when PV is stable at the setpoint SP.

In the two above cases, dynamic feedforward parameters (Lead-lag compensator time constants **sFFLeadTime** and **sFFLagTime**) can be tuned to further accelerate the response by adding an initial excess transient output, as shown in Figure 106. Finally, PID can trim the feedforward output to completely minimize the tracking deviation.

Secondary or Primary process variable can be used as feedforward input to implement a lead-lag compensator to improve the frequency response of the control system.

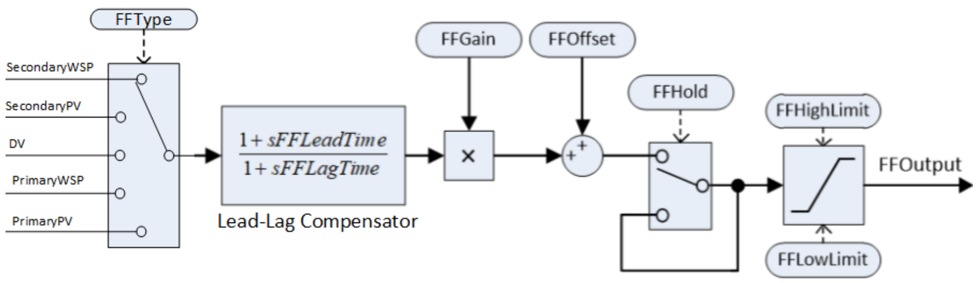


Figure 105 Feedforward Generator

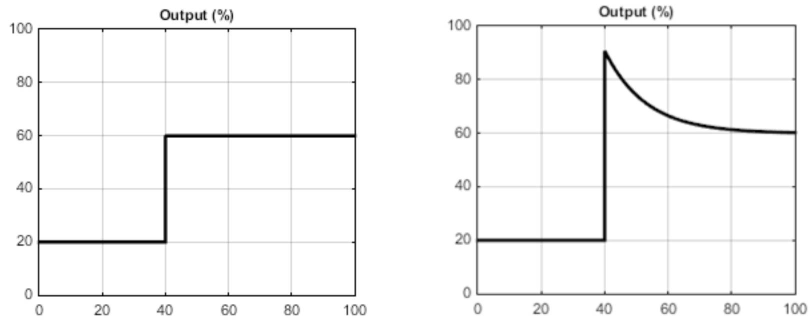


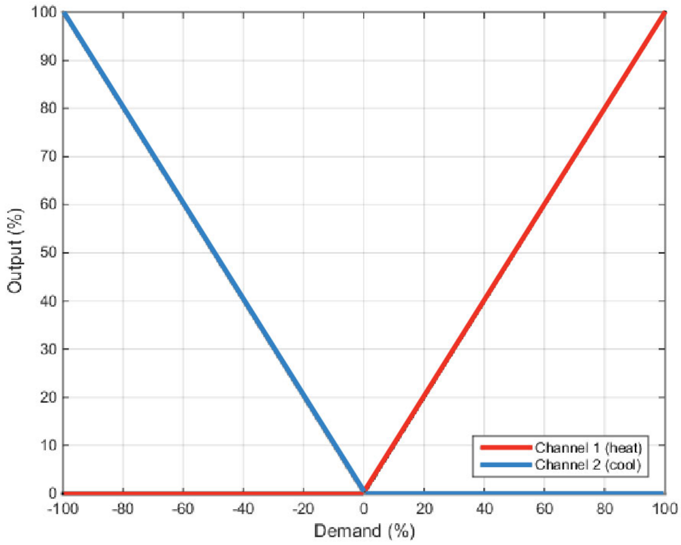
Figure 106 Example Feedforward Output Response to SP Change with Static and Dynamic Compensation

Split Range (heat/cool)

Inherent in the Loop is the concept of split-range for heat/cool.

Each SuperLoop can have two output channels. These two outputs operate in opposite directions. For example, consider a chamber with both a heater and a chiller. Both of these actuators are used to influence the temperature (the ‘process variable’, PV), but they operate in different directions: increasing heat output causes increasing PV, whereas increasing chiller output causes decreasing PV. Another example might be a gas carburizing furnace where the atmosphere is either enriched with methane (channel 1) or diluted with air (channel 2).

The way that the loop implements this is to allow the control output to extend over the span −100 to +100%. In this way, the range is split so that 0 to +100% is output on channel 1 (heat) and −100 to 0% is output on channel 2 (cool). The diagram below shows Split Range Outputs (Heat/Cool)



In addition, different actuator gains are handled by having a separate proportional band for each channel.

Cooling Algorithm

The method of cooling may vary from application to application.

For example, an extruder barrel may be cooled by forced air (from a fan), or by circulating water or oil around a jacket. The cooling effect will be different depending on the method. The cooling algorithm may be set to linear where the controller output changes linearly with the PID demand signal, or it may be set to water, oil or fan where the output changes non-linearly against the PID demand. The algorithm provides optimum performance for these methods of cooling.

Non-linear cooling

The loop provides a set of curves that can be applied to the cooling (ch2) output. These can be used to compensate for cooling non-linearities thereby making the process ‘look’ linear to the PID algorithm. Curves for *Oil, Fan* and *Water* cooling are provided.

Curves are always scaled to fit between 0 and the output low limit. Tuning the curve to the process is an important step in commissioning and can be achieved by adjusting the output low limit. The low limit should be set to the point at which the cooling effect is maximum, before it starts to drop off again.

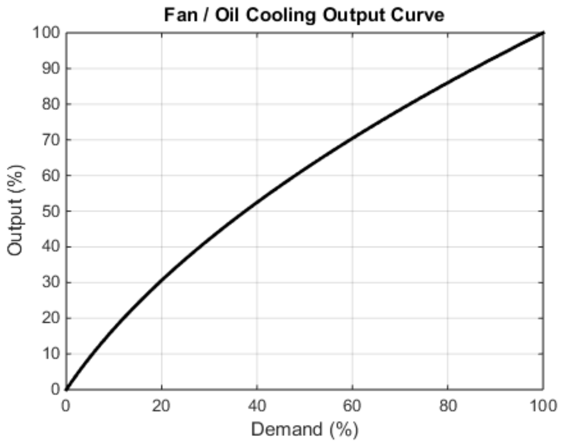
Be aware that any output rate limiting is applied to the output *before* non-linear cooling. Therefore, the actual controller output may change faster than any configured rate limit, but the power being delivered to the process will move at the correct rate, provided the curve has been correctly applied.

Air or Oil cooling

At low temperatures, the rate of heat transfer from one body to another can be considered linear and is proportional to the temperature difference between them. In other words, as the cooling media heats up, the rate of heat transfer slows down. So far, this is linear.

The non-linearity arises when a *flow* of cooling media is introduced. The higher the rate of flow (mass transfer), the less time a given 'unit' of media is in contact with the process and so the greater the average rate of heat transfer.

The air and oil characteristic is shown in the diagram below.

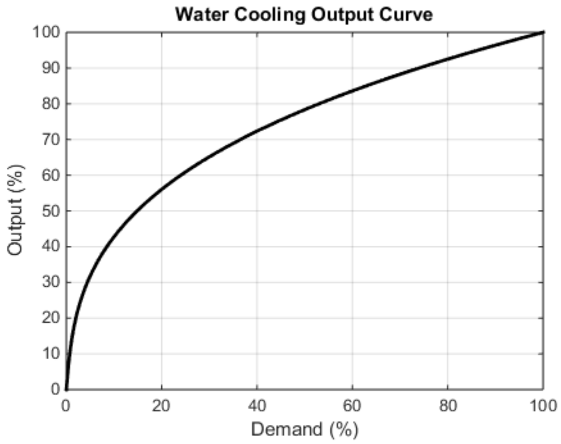


Evaporative Water cooling

Vaporizing water requires about five times as much energy as is required to raise its temperature from 0–100℃. This difference represents a large non-linearity, where at low cooling demands, the principal cooling effect is evaporative, but at higher cooling demands only the first few pulses of water flash off to steam.

To compound this, the mass transfer non-linearity described above for oil and air cooling is also true for water cooling.

Evaporative water cooling is often used in plastics extruder barrels and so this feature is ideal for that application. The evaporative water cooling characteristic is shown below.

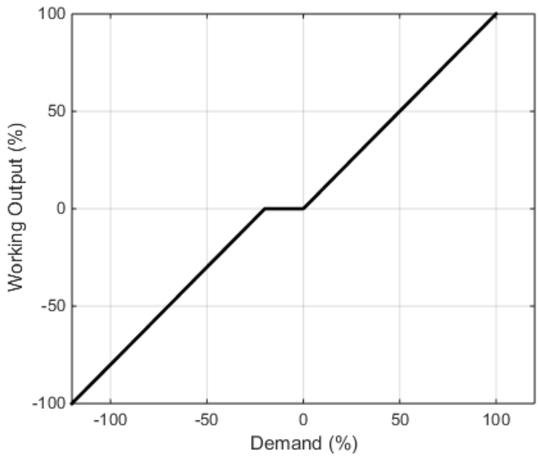


Channel 2 (heat/cool) Deadband

The channel 2 deadband introduces a gap between the point at which channel 1 switches off and the point at which channel 2 switches on, and vice versa. This is sometimes used to help prevent small and fleeting demands for cooling during normal process operation.

For a PID control channel, the deadband is specified in % output. For example, if the deadband is set to 10%, then the PID algorithm must demand −10% before ch2 will begin to switch on.

For an On/Off control channel, the deadband is specified in % of hysteresis. The diagram shows heat/cool with 20% deadband.



Bumpless Transfer

Where possible, the transfer to an Auto control mode from a non-Auto control mode will be ‘bumpless’. This means that the transition will go smoothly without large discontinuities.

Bumpless transfer relies on there being an integral term in the control algorithm to ‘balance out’ the step change. For this reason, it is sometimes called an ‘integral balance’.

The **IntBal** parameter allows the external application to request an integral balance. This is often useful if it is known that a step change in PV is to occur, for example a compensation factor has just changed in an oxygen probe calculation. The integral balance will help to prevent any proportional or derivative kick, instead allowing the output to be smoothly adjusted under integral action.

Note: A similar mechanism is available for Cascade Loop type, from Cascade control mode to non-Cascade control modes. For instance, we have **PrimaryIntBal** on top of **IntBal** in case of Cascade Loop type.

Sensor Break

‘Sensor Break’ is an instrument condition that occurs when the input sensor is broken or out of range. The Loop reacts to this condition by putting itself in Forced Manual mode (see above description). The type of transfer when entering Forced Manual, when the PV status is not good, can be selected using the **PVBadTransfer** parameter. The options are:

- Enter Forced Manual mode with the output set to the Fallback Value.
- Enter Forced Manual mode with the output held at the last good value (typically a value from about one second ago).

In Cascade Loop type, the "sensor break" condition of **PrimaryPV** can be configured via the parameter **PrimaryPVBadTransfer**. The parameter configures the type of transfer to in Forced Auto if, for instance, Primary PV goes bad (e.g. due to a sensor break). This is only followed if transitioning to Forced Auto from Cascade mode or PrimaryTune mode because of a bad status of at least one between **PrimaryPV**, **SecondaryRSP** or **SecondaryRSPTrim**.

- Transitioning from Auto or higher priority modes will be bumpless for the Secondary Local Setpoint.
- Transitioning due to the ForcedAuto input being asserted in modes with lower priority than Forced Auto will make Secondary Local Setpoint go to the Secondary Fallback Setpoint.

It is possible, with the **Config.ForcedModesRecovery** parameter to configure the loop recovery strategy on exit from Forced Manual mode. For example, when the PV recovers from a bad status.

In Cascade Loop Type it also configures the recovery strategy on exit from Forced Auto mode. E.g. when the Primary PV recovers from a bad status.

Start-up and Recovery

Proper start up is an important consideration and varies depending on the process. The Loop recovery strategy is followed under any of the following circumstances:

- Upon instrument start-up, after a power cycle, power outage event or power disruption.
- Upon exit from Instrument Configuration or Standby conditions.
- Upon exit from Forced Manual (F_MAN) mode to a lower priority mode (e.g. when the PV recovers from a bad status or an alarm condition goes away).

The strategy to follow is configured by the **StandbyModeRecoveryMode** parameter. The available options are:

- During Standby or Config mode, the Loop will assume Hold mode and the Loop output will hold to the last value. At Startup recovery or on exit from Instrument Standby or Config modes, the Loop will assume the last operating mode and initialize the output to its last value.
- Inhibit mode in Config and Standby, recovery from last mode. During Standby or Config mode, the Loop will assume Inhibit mode and the Loop output will step to Inhibit OP. At Startup recovery or on exit from Instrument Standby or Config modes, the loop will assume the last operating mode and initialize the output to Inhibit OP.
- Inhibit mode in Config and Standby, recovery in Manual. During Standby or Config mode, the Loop will assume Inhibit mode and the Loop output will step to Inhibit OP. At Startup recovery or on exit from Instrument Standby or Config modes, the loop will assume Manual mode and initialize the output to Inhibit OP.

Cascade Scaling

In Cascade loop mode, the Cascade Scaling block drives the Secondary PID setpoint. Cascade Scaling block executes the mapping from the Primary PID output to the Secondary setpoint. If Control Mode is switched to Auto or Manual, then the Secondary PID receives the Secondary local SP instead.

In the Cascade scaling block, the Secondary working setpoint can be limited via the Secondary Setpoint Limit parameters.

Note: Altering these limits does not affect the gain of the Cascade loop, therefore it does not require to repeat the tuning of Primary PID constants.

Depending on the cascade type the technique changes for scaling the Primary controller output into the Secondary setpoint, as it follows in the next sections.

Full-Scale Cascade Type

For the Full-scale cascade type, the following diagram represents the mapping from Primary PID output to the Secondary working setpoint. In Full-scale cascade type:

- The Secondary working setpoint is derived by mapping the Primary PID output range (0% to 100%) into the Secondary range defined by the Range limits.

Note: The Range limits must be established before tuning the Primary PID as they affect the gain of the cascade loop.

- An additional remote trim component **SecondaryRSPTrim** can be added to the setpoint produced by the Primary PID.

- The Full Scale Setpoint can be limited with upper and/or lower bounds which are relative to the Primary Working Setpoint, through the Limited Head Function – see Figure 108.

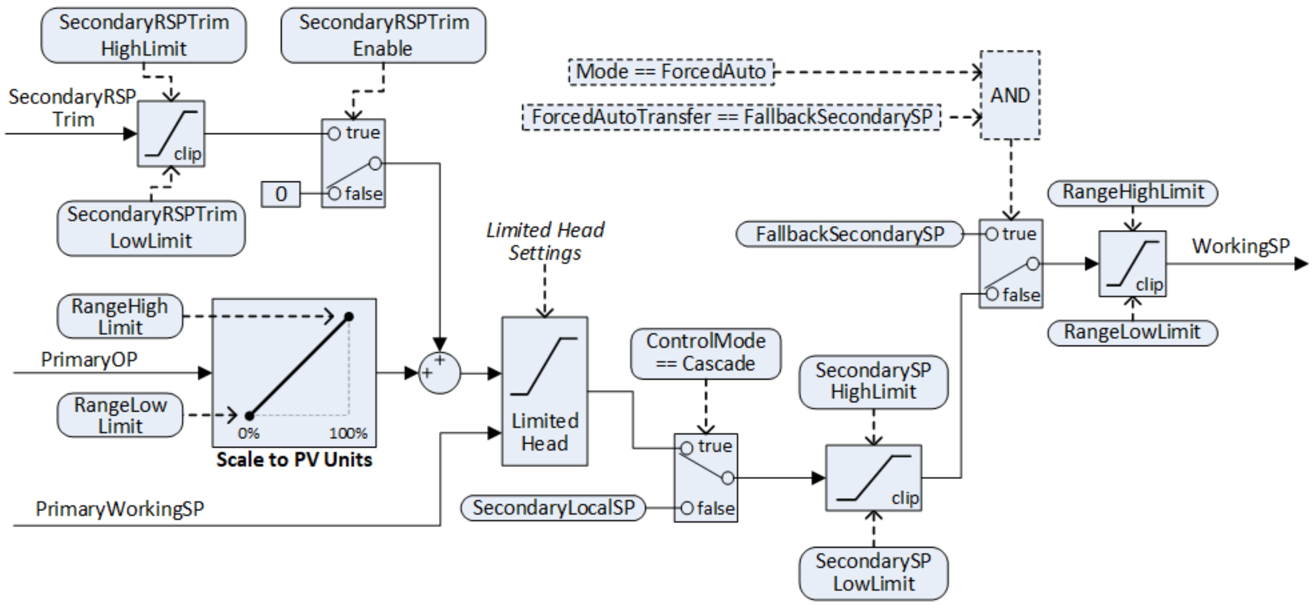


Figure 107 Cascade Scaling for Full Scale configuration (**CascadeType** = FullScale)

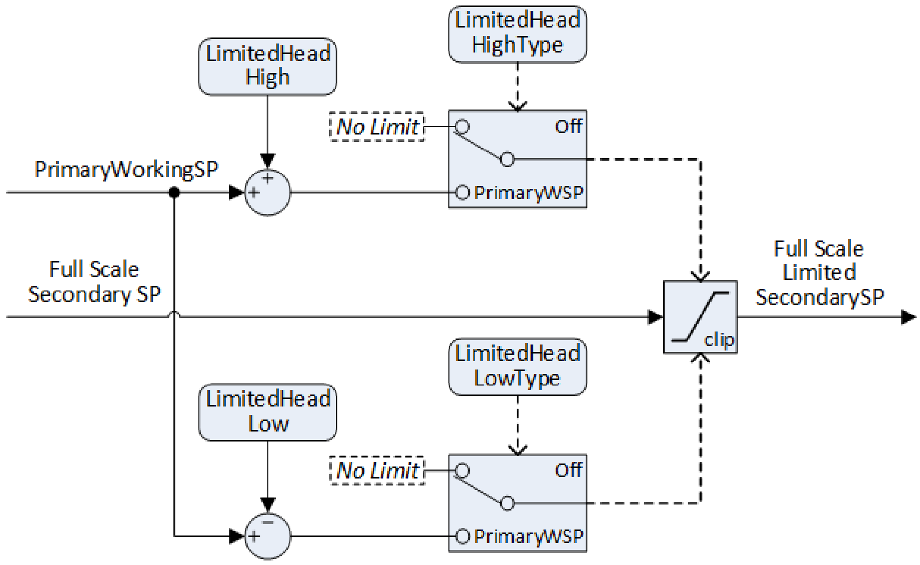


Figure 108 Limited Head function available for Full Scale configuration.

Trim Cascade Type

For the Trim cascade type the following diagram represents the mapping from Primary PID output to Secondary working setpoint. In Trim cascade type:

- The Secondary setpoint main component can be selected between Primary working SP, Primary PV and a remote secondary setpoint.
- The Primary PID trims the setpoint main component with its output that is mapped from its range (–100% to 100%) onto the cascade trim range.
- Trim limit parameters can be used to limit the magnitude of the trim component of the Secondary working setpoint.

Note: Altering these limits or the secondary range limits does not affect the gain of the Cascade loop, therefore it does not require to repeat the tuning of Primary PID constants. Trim ranges instead must be established before tuning the Primary PID.

When setting trim ranges and limits, it is important to remember that if the available range of trim values is too narrow, it may be impossible for the primary loop to generate a secondary setpoint which allows to reach the required primary setpoint.

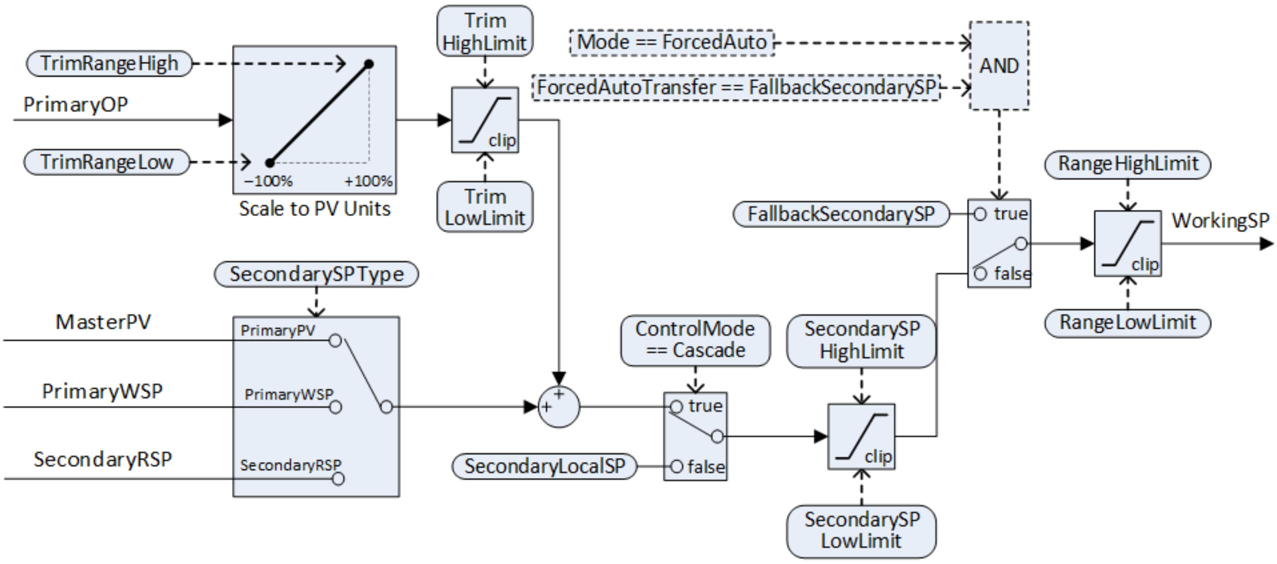


Figure 109 Cascade Scaling for Trim configuration (**CascadeType** = Trim).

Forced Auto Mode

The **PrimaryPVBadTransfer** parameter defines the behavior in Forced Auto. Mode transitions to Forced Auto automatically in Cascade control mode when the **PrimaryLoopBad** alarm is active, that is one between **PrimaryPV**, **SecondaryRSP** or **SecondaryRSPTrim** have bad status. The user can also make Mode transition to Forced Auto by asserting the ForcedAuto input flag. Possible Forced Auto transfer options are:

- **FallbackSecondarySP**, the secondary setpoint will be set to the **FallbackSecondarySP**.
- **HoldSecondarySP**, the secondary Working SP will be frozen at the last good value.
- **ForcedManualTransfer**, the strategy will follow the Forced manual transfer type defined by the **PVBadTransfer** parameter.

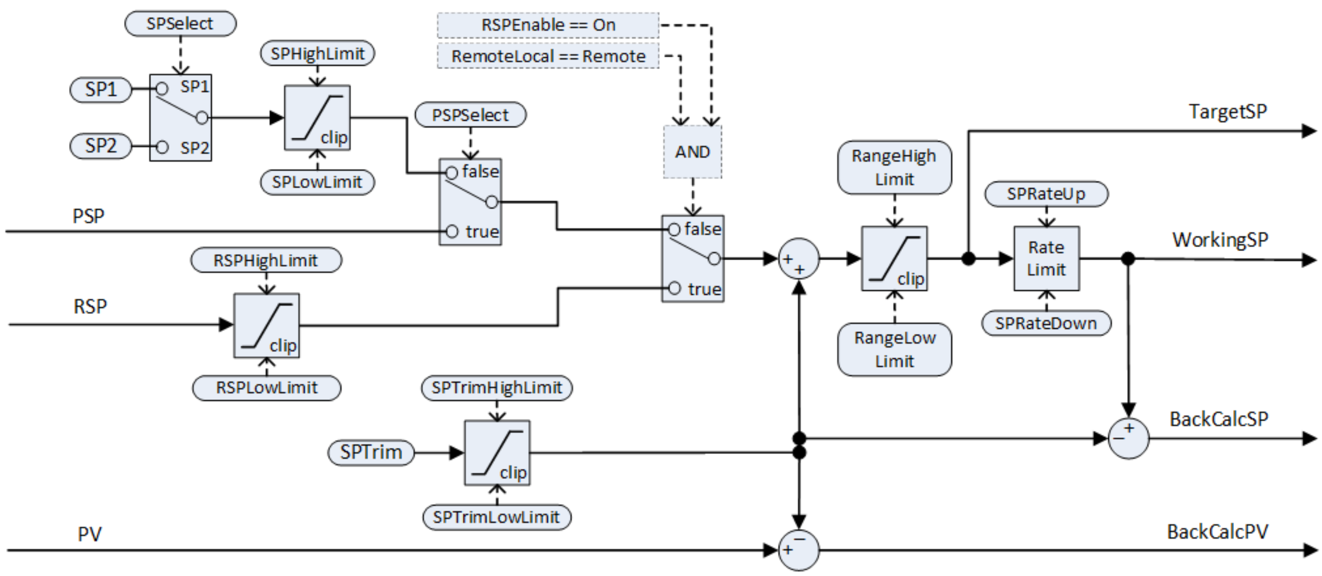
Setpoint Generation

The Setpoint generator produces the working setpoint for the process variable from a set of setpoint sources.

The diagrams below show Setpoint generator block in case of Single Loop type. The first shows the 'Remote Setpoint with local trim' configuration.

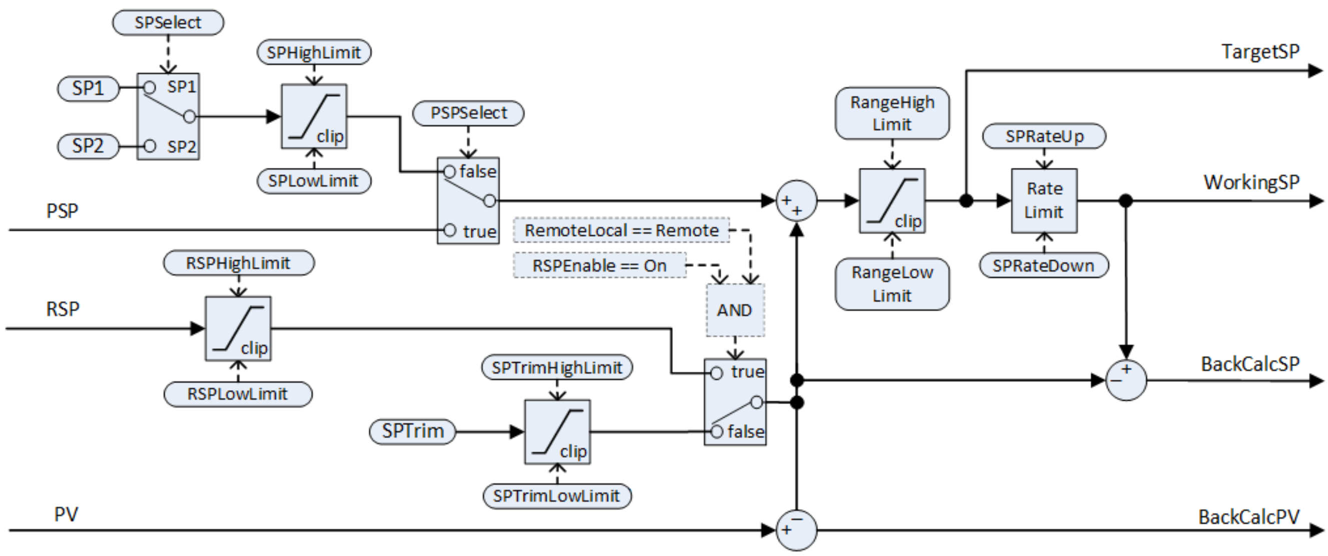
Note: In the case of Cascade Loop type, the Setpoint generator produces the Working setpoint for the Primary PID. In this case the Setpoint generator will maintain the same behavior but will drive Primary Target SP and Primary Working SP and will make use of Primary Range Limits and Primary SP Limits.

Setpoint Subsystem (Remote setpoint with Local Trim configuration)



The second diagram shows Setpoint subsystem in the ‘Local Setpoint with remote trim’ configuration.

Setpoint Subsystem (Local setpoint with Remote Trim configuration)



The setpoint subsystem resolves and generates the working setpoint for the control algorithms. The working setpoint can ultimately come from several different sources, programmer, local or remote, have local or remote trims applied, and be limited and rate limited.

Remote/Local Setpoint Source Selection

The **RemoteLocal** parameter selects between the remote or local setpoint source.

The **SPSource** parameter reports which source is currently active. The three values are:

- Local – the local setpoint source is active.
- Remote – the remote setpoint source is active.
- F_Local – the remote setpoint source has been selected but it cannot become active. The local setpoint source is active until the exceptional condition is resolved.

In order for the remote setpoint source to become active, the following conditions need to be fulfilled:

1. The **RemoteLocal** parameter has been set to ‘Remote’.
2. The RSP_En input is true.
3. The status of the RSP input is Good.

Note: The **RemoteLoc** parameter is enumerated as 0 = Remote and 1 = Local.

Local Setpoint Selection

There are three local setpoint sources: the two operator setpoints, SP1 and SP2; and the program setpoint, PSP. For selection parameters and priorities, refer to the above diagram.

Remote Setpoint

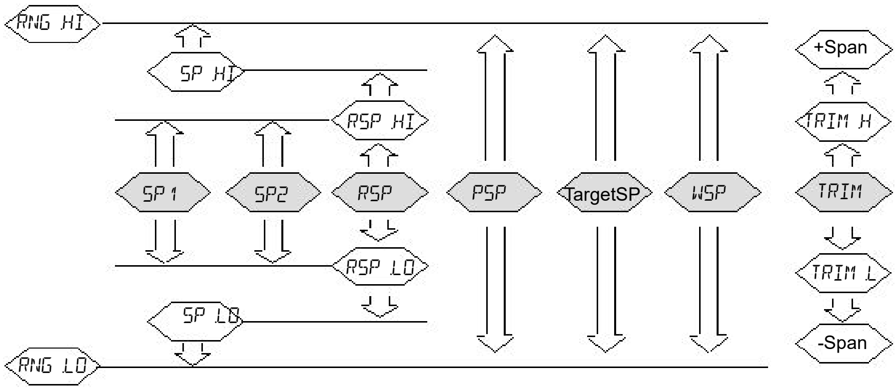
RSP is the remote setpoint source. It can be configured by the *RSPType* parameter in one of two ways:

- Remote setpoint (RSP) with a local trim (SPTrim).
  For example, in a continuous oven with several temperature zones, the primary controller can transmit its setpoint to each secondary’s RSP, and then a local trim can be applied in each secondary to achieve the desired temperature gradient through the oven.
- Local setpoint (SP1, SP2 or PSP) with a remote trim (RSP).
  For example, in a combustion air/fuel ratio application where the ratio setpoint is fixed, but a remote controller analyses the excess oxygen in the flue gases and is allowed to trim the ratio within a given band.

The remote setpoint is always limited by the **RSPHighLimit** and **RSPLowLimit** parameters.

Setpoint limits

The various setpoint parameters are subjected to limits according to the diagram below. Some of the limits themselves are also subjected to limits.



The *Span* is taken to be the value given by (*RangeHigh – RangeLow*).

Note: Whilst it is possible to set the RSP Limits outside the Range Limits, the RSP value will still be clipped to the Range Limits.

Setpoint Rate Limit

Rate limits can be applied to the final setpoint value. This can sometimes be useful to help prevent sudden step changes in controller output, and therefore help to prevent damage to the process or product.

Asymmetric rate limits are available. That is to say that the increasing rate limit can be set independently of the decreasing rate limit. This is often useful, for example, in a reactor application whereby a sudden increase in flow should be reduced so that an exothermic event does not overwhelm the cooling control loop. On the other hand, a sudden decrease in flow should be permitted.

The setpoint rate limits may be set in units per hour, per minute or per second, according to the **SPRateUnits** parameter.

Note: When transitioning into an automatic control mode from a non-automatic control mode such as manual, the WSP will be set equal to the PV whenever a rate limit is set. It will then move towards the target setpoint from there at the configured rate.

In addition, if the **SPRateServo** parameter is activated, the WSP will be set equal to the PV whenever the Target SP is changed and will then move towards the target from there. This only applies in Auto (including the transition to Auto) when SP1 or SP2 is active. It does not apply when using a remote or program setpoint.

Target SP

The Target SP is the setpoint value immediately prior to rate limiting (the Working SP is the value immediately after it). In many instruments it is possible to write to the Target SP directly. The effect of this is to trigger a back-calculation, which takes into account the trim value (either a local or remote trim), and then to write the back-calculated value to the selected setpoint source. This is so that the calculated Target SP on the next execution is equal to the entered value.

This usefully allows the target setpoint to be set to a desired value immediately, without having to manually make the calculations and determine which setpoint source is active.

Writing to the Target SP is not possible when a remote setpoint is active.

Tracking

There are three setpoint tracking modes available. They can each be turned on by enabling the appropriate parameter.

1. SP1/SP2 tracks PV
   Whilst the mode is MANUAL, whichever of SP1 or SP2 is active will track the PV (less the trim). This is so that the operating point is maintained whenever the mode is changed to Auto.
2. SP1/SP2 tracks PSP
   Whilst **PSPSelect** is activated, whichever of SP1 or SP2 is active will track the PSP. This is so that the operating point is maintained when the programmer is reset and **PSPSelect** goes false.
3. SP1/SP2/SPTrim tracks RSP
   When the RSP is active and acting as a Remote Setpoint, whichever of SP1 or SP2 is active will track RSP. If RSP is acting as a Remote Trim, then it is **SPTrim** that will track RSP. This is so that the operating point is maintained if the setpoint is switched to local.

Back-calculated SP and PV

Back-calculated versions of WSP and PV are provided as outputs. These are simply WSP/PV minus the active trim value. These outputs are given so that an external setpoint source (such as a setpoint programmer or a cascade primary) can track their output to them as necessary, thus helping to prevent bumps on mode changes and switchovers.

Setpoint Integral Balance

When the **SPIntBal** parameter is activated, the setpoint subsystem will issue an integral balance request to the PID algorithms whenever a step change in SP1 or SP2 occurs. This will cause any proportional or derivative kick to be suppressed and the PV will move smoothly to the new setpoint with the integral as the driving force and with minimum overshoot. The effect is the same as what is sometimes called ‘proportional and derivative on PV’ instead of deviation, but only applies to step changes in SP1 or SP2 and on transition to local setpoint from remote.

Output Subsystem

The diagram shows the block diagram of the Output subsystem.

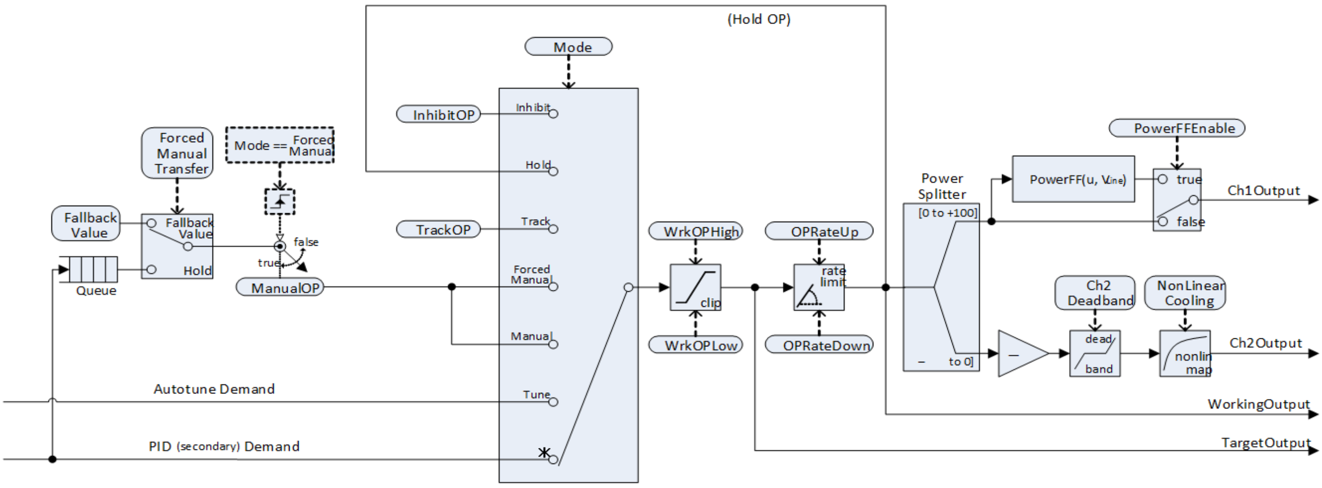


Figure 110 Output Subsystem

Output selection (including Manual Station)

The source of the output demand is resolved according to which controller mode is active. In Inhibit mode, the output demand is taken from **InhibitOP**. In Hold mode, the previous Working Output is held. In Track mode, the output demand is taken from **TrackOP**. In Manual and ForcedManual modes, the output is taken from **ManualOP**. In other modes, the output is taken from the Secondary PID output.

Output Limiting

The resolved demand is subject to position limiting. There are several different sources of position limits:

- The main limits, **OutputHighLimit** and **OutputLowLimit**.
- The active gain scheduled limits: **OutputHigh**(n) and **OutputLow**(n).
- The remote limits, **RemoteOPHigh** and **RemoteOPLow**.
- The tune limits (only during auto-tuning), **TuneOutputHigh** and **TuneOutputLow**.

The most restrictive limits always take priority. That is to say, the minimum of the upper limits and the maximum of the lower limits are used. These become the working output limits, **WrkOPHigh** and **WrkOPLow**.

The output limits are always applied in Auto modes. In non-Auto modes such as Manual, the **FallbackValue** may override a limit if that limit would help prevent the **FallbackValue** being achieved. For example, if the **OutputLowLimit** is 20% and the **FallbackValue** is 0%, then in Auto the working low limit will be 20%, while in Manual it will be 0%.

The remote output limits are only applied in Auto mode.

Rate limiting

The working output can be rate limited by setting the two parameters, **OPRateUp** and **OPRateDown**. They are always specified in % per second. Output rate limiting is only available for PID control channels and should be used only where necessary since it can significantly degrade process performance. As rate limiting, when configured, is applied also in modes such as Inhibit, Track, Forced Manual, the OP Rate Deactivate input can be used to deactivate it on demand.

Autotuning

The function block contains sophisticated autotuning algorithms that can tune the controller to the process. They work by performing experiments on the plant, by inducing perturbations, and observing and analysing the response. The autotuning sequence is described in detail below.

In commissioning a cascade loop:

- Autotune the Secondary PID first by selecting Secondary as Tune Type.
- Once the Secondary autotuning has successfully completed, then autotune the Primary PID

The above sequence must be respected because the Secondary loop is part of the process controlled by the Primary PID and therefore its tuning set must be established first.

The diagrams below show a simplified structure of the Eurotherm Autotuner for the Secondary and the Primary PID.

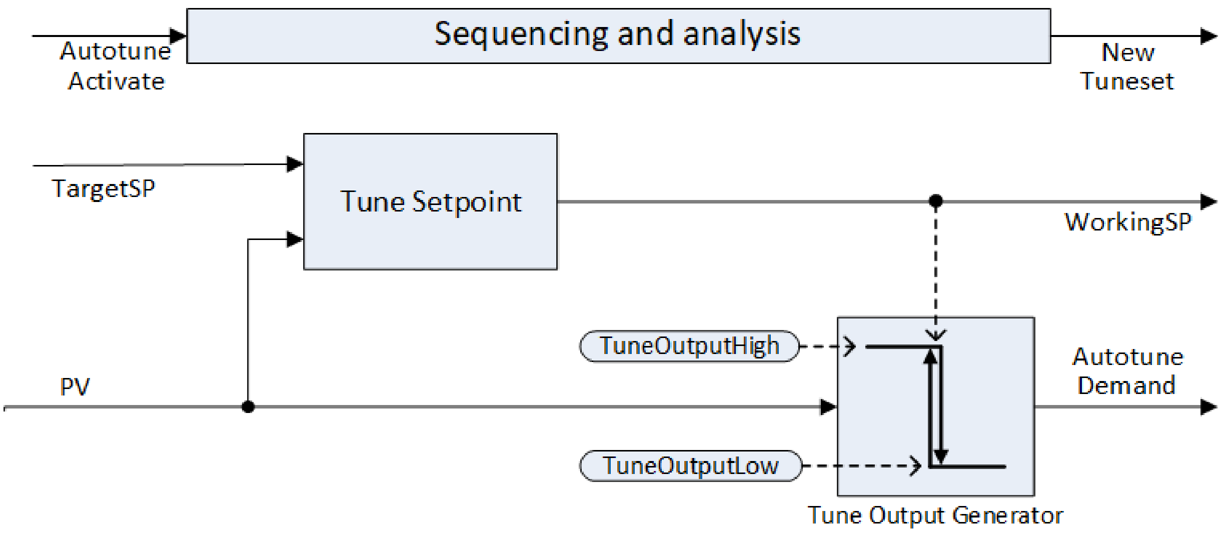


Figure 111 Autotune algorithm (**LoopType** = Single or **LoopType** = Cascade and **TuneType** = Secondary)

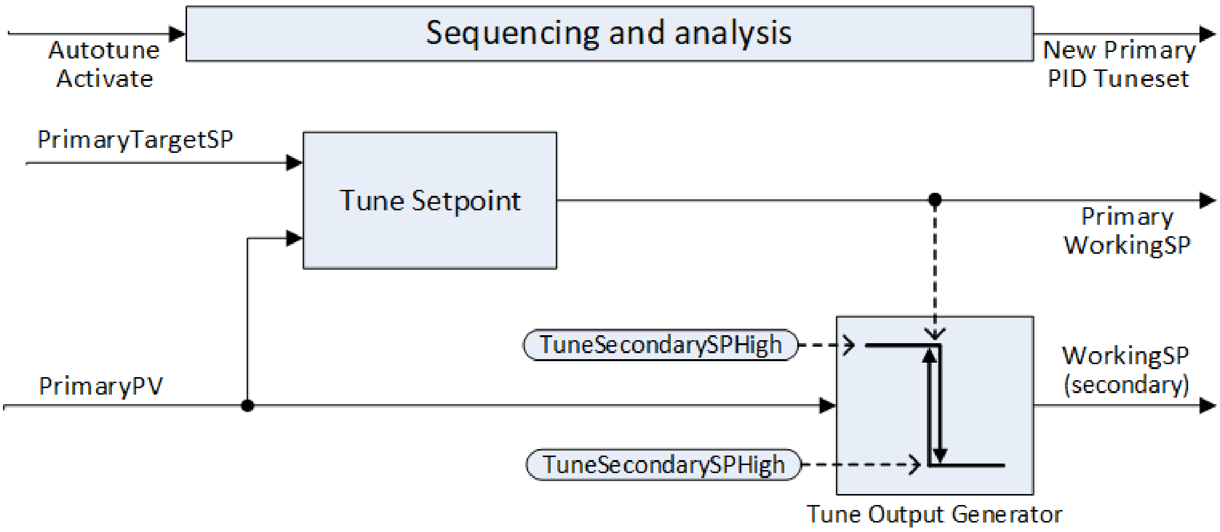
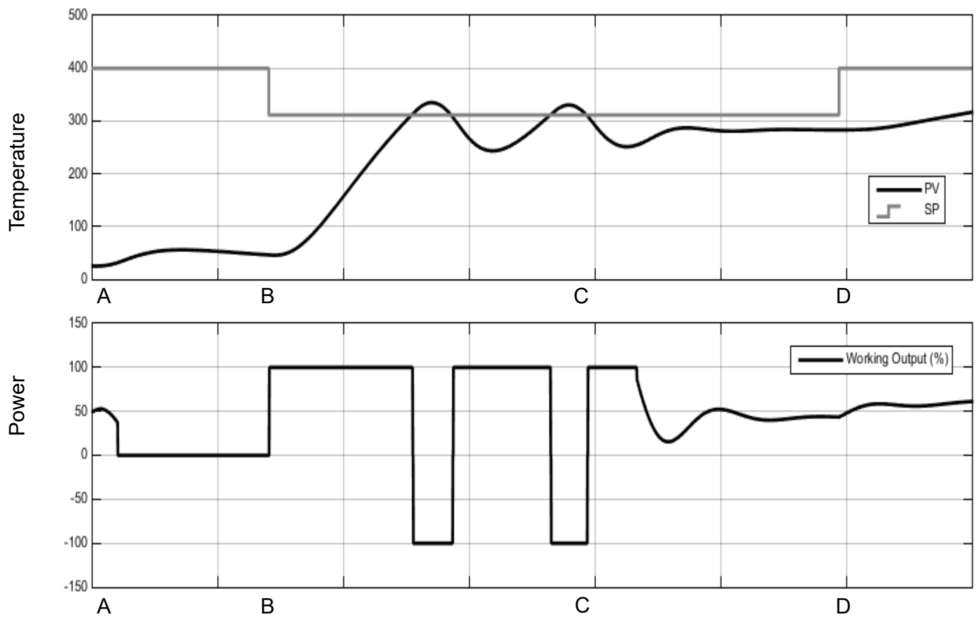


Figure 112 Autotune algorithm (**TuneType** = Primary and **LoopType** = Cascade)

The diagram shows an example of heat/cool Autotune with 'alternative' CH2 Tune Type.



The following are descriptions of the steps which are automatically executed by the autotune algorithm.

- Time A - Autotune Begins

  Setting the **AutotuneActivate** parameter to On and the controller mode to Auto will cause the autotune to begin.

  Before starting an autotune, you should turn off the PID terms that you do not want to use. For example, setting TD to Off will deactivate derivative action and the autotuner will therefore tune for a PI controller. If you do not want any integral, set TI to Off and the autotuner will tune for a PD controller.

  If the cutback thresholds, CBH and CBL, are set to Auto then the autotuner will not attempt to tune them.

  An autotune may be triggered at any time, but it will not begin while the following higher priority modes are active: Hold, Track, Forced Manual, Manual plus Forced Auto in case of Primary tune. Similarly, the tune will abort if one of the above higher priority mode is requested at any time during the tune, including for reasons such as PV failure.

Note: The PID tuning constants will be written to whichever gain set is active when tuning completes.

- Time A to B - Initial Delay

  This period always persists for precisely one minute.

  If the PV is already at the WSP then the working output will be frozen. Otherwise, the output is set to 0 and the process is allowed to drift while some initial measurements are made. The target setpoint may be changed during this initial delay, but not after it. You should set the target setpoint to the operating point at which you would like to tune. Care should be taken in setting the setpoint, to ensure that oscillations of the process will not damage the process or load. As the autotune experiment applies power demand equal to the Tune OP limits and induces PV oscillations, these may cause PV overrange for specific processes (e.g.thermal processes with high heat capacity and / or low heat losses): to avoid that it may be necessary to use a setpoint for tuning purposes that is below the normal operating point.

- Time B - Calculate Tune Setpoint

  Once the initial delay has elapsed, the tune setpoint is determined. It is calculated by:

  If PV = Target SP: Tune SP = Target SP

  If PV < Target SP: Tune SP = PV + 0.75(Target SP - PV)

  If PV > Target SP: Tune SP = PV - 0.75(PV - Target SP)

  Once determined, this tune setpoint will be used for the duration of the autotune and any changes to the target setpoint will be ignored until the autotune has completed. If you wish to change the tuning setpoint, you must abort and restart the autotune.

- Time B to C – PV Oscillation Experiment The autotuner will now drive the output between the **TuneOutputHigh** and **TuneOutputLow** generating PV oscillations to establish time constants of the process.

  If PV > SP: OP = TuneOutputLow

  If PV < SP: OP = TuneOutputHigh

  There is also a small amount of hysteresis, automatically applied, around the switching point – Tune SP – to prevent noise causing nuisance switching.

  The number of oscillations required before moving to the next stage depends on the controller configuration:

  – If either channel is configured for VPU, VPB or On/Off control, or if output rate limiting is activated, then the 'Fourier' autotune algorithm will run. This requires three cycles of oscillation.

  – If only PID is configured and there is no output rate limiting, then the 'PID' autotune algorithm will run. Only two cycles of oscillation are required.

  – There are certain circumstances, for example if the oscillation amplitude is very small, where the controller will automatically decide to use the Fourier algorithm.

  – There will be an additional half-cycle of oscillation at the beginning of this stage if the initial PV is above the SP.

- Time C to D - Relative Channel 2 Tuning Experiment

  This stage is only used for dual-channel heat/cool configurations. For heat-only or cool-only, it is skipped.

  The purpose of this stage is to determine the relative gain between channel 1 and channel 2. This is used to set the correct proportional bands. For example, in a heat/cool process, the heater and the chiller are typically not equally rated, e.g. the heater might be capable of putting much more energy into the process in a given period of time than the chiller is capable of removing.

  The type of experiment that is used can be selected with the **Ch2TuneType** parameter:

  – The Standard experiment is the default and gives good results for most processes. It will put the process into an additional cycle of oscillation but instead of applying minimum output, it will apply 0 output and allow the PV to drift. This option is not available if **TuneAlgo** is Fourier.

  – The Alternative experiment is recommended for processes that do not exhibit significant losses-for example, a very well-lagged tank or oven. It attempts to control the PV to the SP and collects data about the process input required to do so. The length of this stage is equivalent to between 1.5 and 2 oscillation cycles.

  – The KeepRatio option should only be selected when the relative gain of the two channels is well known. It causes this stage to be skipped and instead the existing proportional band ratio will be maintained. So, for example, if you know that the heating channel will deliver a maximum of 20kW and the cooling channel will deliver a maximum of -10kW, then setting the proportional bands such that the ratio Ch2PB/Ch1PB = 2 before autotuning will ensure the correct ratio is maintained.

- Time D - Analysis and Completion

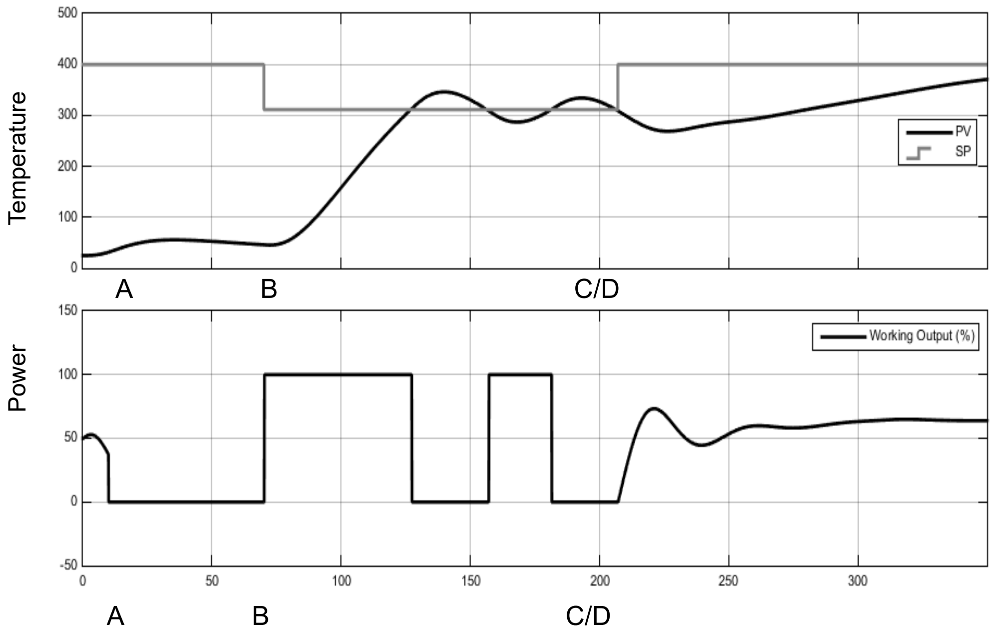
  The autotune experiments are now complete. Finally, an analysis will be performed on the collected data and the controller tuning constants will be chosen and written to whichever gain set is active. This analysis may take a number of seconds, typically less than 15, during which the output will be frozen. After the tune has completed, the working setpoint is released and can be modified in the usual manner. Authority over the output is returned bumplessly to the control algorithms.

Notes:

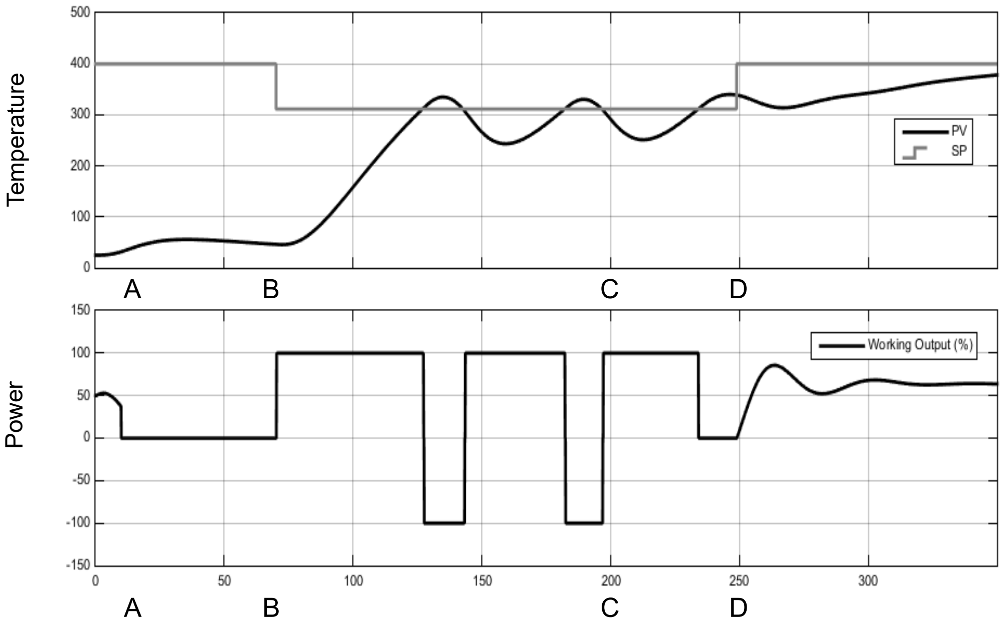
1. If any stage of the autotune sequence exceeds two hours in duration, the sequence will timeout and be aborted. The **StageTime** parameter counts up the time in each stage.
2. Channels configured for On/Off control cannot be autotuned but they will be exercised during the experiments if the opposite channel is not On/Off.
3. Carbon Potential loops, which have a setpoint in the range 0–2.0% (and other loops with small setpoint ranges), cannot be autotuned if the proportional band type is set to ‘Engineering Units’. For these loops, proportional band type should be set to ‘Percent’ and the **RangeHigh** and **RangeLow** set correctly. This allows autotune to work.

A number of further examples under different conditions are pictured below.

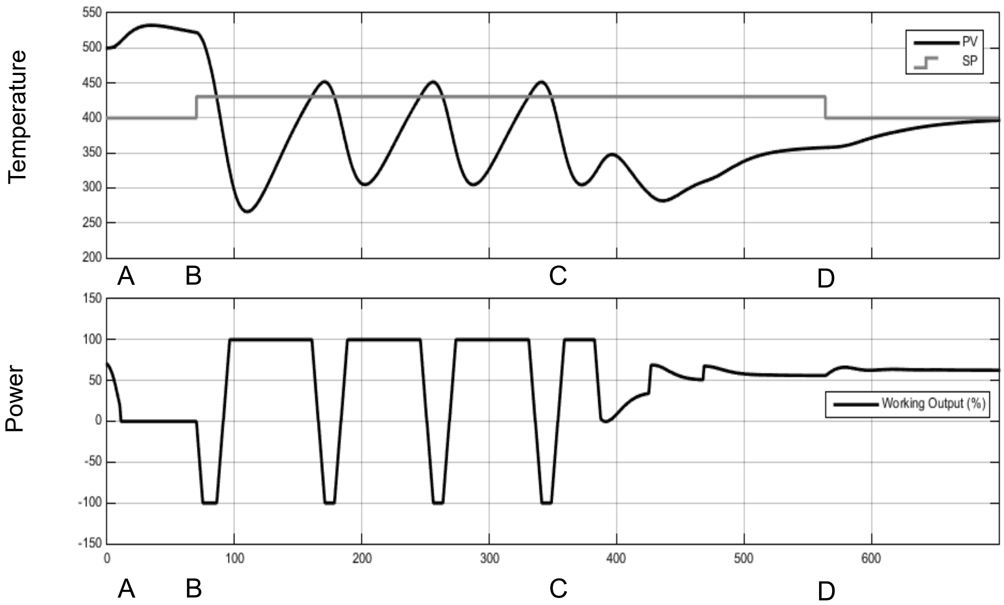
The first shows an example Heat-only Autotune.



The second example shows Heat/Cool Autotune with ‘Standard’ Ch2 tune type.



The third shows an example of Heat/Cool Autotune from above with output rate limiting.



Autotuning multiple zones

Autotune relies completely on the principle of cause and effect. It perturbs the process and then watches for what the effect will be. It is therefore essential that all external influences and disturbances are minimized as much as possible during an autotune.

When autotuning a process that has multiple interacting loops, for example a furnace with multiple temperature zones, each loop should be autotuned separately. They *should not* under any circumstances be autotuned at the same time, since the algorithms will not be able to ascertain what cause produced what effect. The procedure below should be followed:

1. Place all loops in manual and set the outputs to the approximate steady state value for the desired operating point. Allow the process to settle.
2. Activate autotune on a *single zone.* Allow the tune to complete.
3. After the zone has finished autotuning, allow it to settle out in auto and then place it back into manual.
4. Repeat steps 2 and 3 for each zone.

Parameters

Main Parameters

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Main			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
CascadeMode	Selects between Cascade and Secondary automatic modes, for Loop Type Cascade.	0	Cascade control mode selected This is the mode where both the Primary and Secondary controllers are operating and both the Primary PV and the (Secondary) PV are monitored. The Primary PID minimizes the difference between Primary PV and its setpoint by driving the (Secondary) working setpoint.		Oper
		1	Secondary control mode selected In this mode, only the Secondary controller is in automatic control and therefore the Primary PV will not be controlled to its setpoint but determined by the process. The operator can directly adjust the Secondary setpoint through the SecondaryLocalSP parameter. The Primary controller continues to monitor the Secondary loop, so that when the instrument is returned to Cascade mode it can resume control as smoothly as possible. In Secondary mode the Secondary Setpoint Limits and Ranges no longer apply and the Primary Process Variable may be driven over or under range, as the Primary controller operates in open loop.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Main			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
AutoManual	Selects between Auto and Manual operating modes	0	Auto mode selected In automatic control the instrument continuously monitors the process variable and compares it to the setpoint. It calculates an output which will try to minimize any difference. The setpoint may come from a local or remote source. Auto selection activates closed loop operation where the SuperLoop adjusts automatically the Working Output and the Channel Outputs to minimize the deviation between: PV and WorkingSP (Single Loop Type or Cascade Loop Type with "Secondary" selected as Cascade Mode) PrimaryPV and PrimaryWorkingSP (Cascade Loop Type with "Cascade" selected as Cascade Mode)		
		1	Manual mode selected In manual mode, the controller passes control of the output power to the operator. In Manual mode, the user sets the SuperLoop output using the ManualOP parameter. The controller continues to monitor the loop, so that when the instrument is returned to automatic mode it can resume control as smoothly as possible. In Manual mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop mode.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Main			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
RemoteLocal	Selects the Remote or Local setpoint source	0	Remote Setpoint Selects the remote setpoint source. For example, this mode is commonly used to implement a cascade topology with separate single PID loops or a multi-zone furnace with multiple Loops controlled by the same setpoint source. Although this parameter is used to select the remote setpoint, it may not become active. The RSPActivate input needs to be true and the RSP needs to have a good status before it will become active. If either of these conditions are not met, the loop will fallback to using the local setpoint.		
		1	Local Setpoint Selects the local setpoint source. In this case the loop uses one of its local setpoints (SP1/SP2) which is alterable via the front panel or over comms.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Main			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Mode	This reports the currently active operating mode. If multiple modes are selected at the same time then the mode with the highest priority will be activated.	0	Hold mode Priority 1: The controller output will be held at its current value. In Hold mode the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop.		
		1	Track mode Priority 2: The controller output will follow the track output parameter. The track output can either be a constant value or come from an external source (e.g. an analog input). In Track mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop.		
		2	Forced Manual mode Priority 3: This mode behaves in the same way as Manual but it indicates that Auto or Remote modes cannot currently be selected. This mode is selected if the LoopBad alarm is active (e.g. PV status is not good because of sensor break) and, optionally via the ForcedManual input flag, if a process alarm has triggered. When transferring to Forced Manual from Auto, Forced Auto or Cascade modes, the output will go to the Fallback Value (unless the hold action has been selected when the PV status is not good, in which case it will maintain the last good value). Transferring to Forced Manual from any other mode will be bumpless. In Forced Manual mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop.		
		3	Manual mode Priority 4: In manual mode, the controller passes authority over the output to the operator by making the output alterable via the ManualOP parameter. In Manual mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop mode.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Main			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Mode (Contd)	This reports the currently active operating mode. If multiple modes are selected at the same time then the mode with the highest priority will be activated.	4	Tune mode Priority 5: This mode indicates that the autotuner is running and has authority over the output. In Cascade Loop Type, this relates to the Secondary PID autotune.		
		5	Auto mode Priority 6 (lowest in Single Loop Type): In Auto mode, the automatic control algorithm has authority over the output. In automatic control the instrument continuously monitors the process variable and compares it to the setpoint. It calculates an output which will try to minimize any difference. The setpoint may come from a local or remote source. In case of cascade loop type, only the Secondary PID is in control and therefore the Primary PV will not be controlled to its setpoint but determined by the process.		
		6	Inhibit mode Priority 0 (highest): The controller output will step to Inhibit OP. In Inhibit mode the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop.		
		7	Forced Auto Priority 7: This mode, available only for Cascade Loop Type, behaves similarly to Auto, as the secondary PID has authority over the output, but it indicates that Cascade or Primary Tune cannot currently be selected. This mode is selected if the PrimaryBad alarm is active (e.g. PV status is not good because of sensor break) and, optionally via the ForcedAuto input flag, if a process alarm has triggered. The SP source for the secondary PID is defined by the ForcedAutoTransfer parameter and by default is the Fallback Secondary SP.		
		8	Primary Tune mode Priority 8: This mode, available only for Cascade Loop Type, indicates that the autotuner is running on the Primary PID and has authority over the Secondary PID target setpoint and indirectly over the working output.		

Blocks – SuperLoop.1 to SuperLoop.24		**Sub-block: Main**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Mode (Contd)	This reports the currently active operating mode. If multiple modes are selected at the same time then the mode with the highest priority will be activated.	9	Cascade Auto mode Priority 9 (lowest): In Cascade mode, available only for Cascade Loop Type, the automatic cascade loop algorithm has authority over the working output. This is the mode where both the Primary and Secondary PIDs are operating and both the Primary PV and the (Secondary) PV are monitored. The Primary PID minimizes the difference between Primary PV and its setpoint by driving the Secondary PID target setpoint. The primary setpoint may come from a local or remote source.		
SPSource	Indicates the currently active setpoint source	0	Forced Local Setpoint The remote setpoint has been selected but something is prohibiting it from becoming active. The loop has fallen back to using the local setpoint.		
		1	Remote setpoint The remote setpoint has been selected and is active.		
		2	Local setpoint The local setpoint has been selected and is active.		
PrimaryPV	Primary loop process variable	This is the process variable for the outer Primary loop of cascade control. This is typically wired from an analog input. The primary process variable is typically characterized by the slowest dynamic, such as the temperature of a furnace or the temperature of a workload in the furnace.			
PrimaryWorkingSP	Primary loop working setpoint	This is the working setpoint for the outer Primary loop. It may come from a number of different sources, such as internal SP or Remote SP. The working setpoint is read-only as it is generated by the Setpoint generator subsystem.			
PrimaryTargetSP	Primary loop target setpoint	The target setpoint is the primary loop setpoint for PrimaryPV before rate limiting. Writing to this parameter is possible when SP1 or SP2 is in use. Ultimately, writing to the PrimaryTargetSP will cause a new value of SP1 or SP2 to be calculated, taking into account any setpoint trims.			
PV	Loop process variable	This is the process value (PV) for the control loop. This is typically wired from an analog input. In case of Cascade Loop Type this is the Secondary loop process variable, typically associated with an actuator such as a heating element.			

Blocks – SuperLoop.1 to SuperLoop.24		**Sub-block: Main**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
TargetSP	Loop target setpoint	The target setpoint is the loop setpoint for PV before rate limiting. Writing to this parameter is possible when SP1 or SP2 is in use. Ultimately, writing to the TargetSP will cause a new value of SP1 or SP2 to be calculated, taking into account any setpoint trims.			
WorkingSP	Loop working setpoint	The working setpoint is the current value of the setpoint being used by the control loop (after rate limiting). The working setpoint is read-only as it is generated by the Setpoint generator subsystem. For Cascade Loop Type, this refers to the Secondary PID controller.			
WorkingOutput	Working Output (%)	This is the actual (%) output of the controller before it is split into the separate channel 1 and channel 2 outputs. Positive values indicate that channel 1 is active whilst negative values indicate that channel 2 is active.			
Inhibit	Used to select Inhibit mode. In this mode, the controller output will step to Inhibit OP. In Inhibit mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop. Inhibit has priority 0, the highest, and will override any other selected modes.	0	Off		
		1	On		
Hold	Used to select Hold mode. In this mode, the controller output will maintain its current value. In Hold mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop. Hold has priority 1, so only overridden by Inhibit.	0	Off		
		1	On		
Track	Used to select Track mode. In this mode, the controller output will follow the Track output value. The Track output may be a constant value or come from an external source (e.g. an analog input). In Track mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop. Track has priority 2, so only overridden by Inhibit and Hold.	0	Off		
		1	On		
ForcedManual	Used to select Forced Manual mode. This mode behaves in the same way as Manual but whilst active it indicates that Auto cannot currently be selected. When transferring to this mode from Auto, and this input is asserted, the output will step to the Fallback Value. This input can be wired to alarms or digital inputs and used during detected process anomalies. In Forced Manual mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop. Forced Manual has priority 3 and so overridden only by Inhibit, Hold and Track.	0	Off		
		1	On		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Main			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
ForcedAuto	Used to select Forced Auto mode. This mode behaves similarly to Auto, as the secondary PID has authority over the output, but it indicates that Cascade or Primary Tune cannot currently be selected. When transferring to this mode from Cascade mode, and this input is asserted, the Secondary Local Setpoint will step to the Fallback Secondary SP. This input can be wired to alarms or digital inputs and used during detected process anomalies. This mode has priority 7 and so overridden by user selection of Auto via Secondary Cascade Mode and any other higher priority mode.	0	Off		
		1	On		
PrimaryIntegralHold	If asserted, the integral component of the PID calculation will be frozen for the Primary PID controller.	0	No		
		1	Yes		
IntegralHold	If asserted, the integral component of the PID calculation will be frozen. In Cascade Loop Type, this acts on the Secondary PID only.	0	No		
		1	Yes		
PrimaryIntBal	On a rising-edge, the Primary PID algorithm will balance the integral such that I=OP-P-D. This can be used to minimize bumps in Secondary setpoint when it is known that, for example, an artificial step change in Primary PV will occur.	0	No		
		1	Yes		
IntBal	On a rising-edge, the PID algorithm will balance the integral such that I=OP-P-D. This can be used to minimize bumps in output when it is known that, for example, an artificial step change in PV will occur. In Cascade Loop Type, this acts on the Secondary PID only.	0	No		
		1	Yes		

Config Parameters

This list of parameters is used to configure the behavior of the SuperLoop and to activate its main features. The configuration as Single Loop or Cascade Loop is made via Loop Type in this list.

Blocks – SuperLoop.1 to SuperLoop.24		**Sub-block: Config**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
LoopType	The Eurotherm SuperLoop can be configured to operate in Single Loop or Cascade Loop mode through the LoopType parameter.	0	Single loop type: • One single Control Loop is available to generate the controller working output that minimizes the difference between the process variable PV and the working setpoint. • The available modes are, from the lowest to the highest priority: Auto, Tune, Manual, Forced Manual, Track, Hold and Inhibit. • Setpoint generator produces the working setpoint for the PV from a set of setpoint sources - e.g. local, remote, programmer setpoint. • For automatic fine-tune of the PID terms, the Eurotherm autotune algorithm can be used. • The Output conditioning block processes the target controller output by applying various algorithms and criteria and splits it into two channels - typically in temperature control applications heat and cool channels. It also manages manual, track and hold output modes. • Through the Feedforward generator an additional open-loop component can be added to the target output, which depends from a selectable disturbance variable.		
		1	Cascade loop type: • Two packaged and pre-wired Control Loops in cascade configuration automatically control two functionally and dynamically interdependent process variables via an output demand. • The primary process variable (PrimaryPV) is typically characterized by the slowest dynamic, such as the temperature of a furnace or the temperature of a workload in the furnace. • The secondary process variable (PV) is typically associated with an actuator such as a heating element. • The primary PID loop controls the primary PV to setpoint by driving the secondary loop. • The secondary loop controls the secondary PV to the setpoint cascaded from the primary PID by generating the controller working output. • With respect to Single Loop Type, the following modes are added, from lowest to highest priority: Forced Auto, Primary Tune and Cascade. • The Setpoint generator produces the setpoint for the Primary PV. • Both Primary PID terms and Secondary PID terms can be automatically tuned using the Eurotherm autotune algorithm. • Feedforward, Autotune and Output conditioning blocks operate as in single loop type.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Config			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
CascadeType	The Eurotherm SuperLoop, when in Cascade Loop Type, can be configured to operate as Full-scale or Trim cascade type.	0	Full scale cascade type In FullScale Cascade type the primary calculated output is scaled to become the main component of the secondary working setpoint. If the engineering units used in the primary and secondary loops are not the same, full-scale mode is usually adopted. It is simple to set up, because the secondary setpoint range is already defined by the secondary range limits, RangeHighLimit and RangeLowLimit. However, for applications where the two PVs have same units but an external source makes the steady-state deviation between secondary PV and primary PV not easily predictable, in setting up a Trim cascade configuration it might be challenging to establish the amount of SP trim that needs to be added to the main secondary SP component to reach the primary SP operating point. In these specific situations, e.g. in case of interactive multi-zone furnaces, Full-scale cascade type can be selected to make the primary loop drive the secondary SP within the whole secondary range.		
		1	Trim cascade type In Trim Cascade Type the primary output is scaled and then added to the primary setpoint, primary PV or a remote secondary SP in order to generate the working setpoint for the secondary controller. If the engineering units used in the primary and secondary loops are the same, for instance in heating applications, trim-scale mode is typically adopted.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Config			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ch1ControlType	Selects the Channel 1 control algorithm. Channel 1 and channel 2 operate in opposite directions. When both channels are configured, channel 1 is reverse acting and channel 2 is direct acting. For example, in a temperature control application, Ch1 is the heating channel and Ch2 is the cooling channel.	0	Channel is not used.		
		1	Hysteretic On/Off control algorithm The hysteretic on/off control algorithm operates as a simple thermostat, switching when above or below a threshold. Hysteresis is included to reduce excessive switching.		
		2	PID control algorithm The Eurotherm PID algorithm is based on an absolute (positional) algorithm in the ISA form.		
		3	Unbounded valve positioning PID control algorithm Unbounded VP is used to control a process where the final control element is a motorized valve. For example, a furnace with a gas burner. This control type uses a special velocitymode form of the Eurotherm PID algorithm.		
Ch2ControlType	Selects the Channel 2 control algorithm. Channel 1 and channel 2 operate in opposite directions. When both channels are configured, channel 1 is reverse acting and channel 2 is direct acting. For example, in a temperature control application, Ch1 is the heating channel and Ch2 is the cooling channel.	0	Channel is not used.		
		1	Hysteretic On/Off control algorithm The hysteretic on/off control algorithm operates as a simple thermostat, switching when above or below a threshold. Hysteresis is included to reduce excessive switching.		
		2	PID control algorithm The Eurotherm PID algorithm is based on an absolute (positional) algorithm in the ISA form.		
		3	Unbounded valve positioning PID control algorithm Unbounded VP is used to control a process where the final control element is a motorized valve. For example, a furnace with a gas burner. This control type uses a special velocitymode form of the Eurotherm PID algorithm.		
PrimaryControlAction	Selects the direction of Primary control. that is, reverse or direct acting.	0	Reverse acting Use this for systems where an increase in (Secondary) PV will cause a corresponding increase in Primary PV.		
		1	Direct acting Use this for systems where an increase in (Secondary) PV will cause a corresponding decrease in Primary PV.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Config			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryDerivativeType	This parameter configures whether the derivative term of the Primary PID responds to rate of change of Primary PV or rate of change of Primary deviation (that is, rate of change of the difference between PV and the setpoint). Derivative on PV is recommended by default, however derivative on deviation may sometimes be useful, for example to reduce overshoot at the end of a setpoint ramp. Care is needed on sensitive processes, as it will also cause derivative 'kick' (harsh changes in output) when the setpoint changes.	0	Derivative action on PV The derivative term responds only to the rate of change of the process variable.		
		1	Derivative action on deviation The derivative term will respond to the rate of change of the difference between PV and the setpoint.		
PrimaryPropBandUnits	This parameter configures the units used to specify the Primary PID proportional bands.	0	Engineering units Proportional bands set in engineering (PV) units. For example, degrees C.		
		1	Percent Proportional bands set in percent of loop span (RangeHighLimit minus RangeLowLimit).		
ControlAction	Selects the direction of control. i.e reverse or direct acting. This parameter is not available for dual-channel configurations, where channel 1 is always reverse acting and channel 2 is direct acting. For Cascade Loop Type, this refers to the Secondary PID controller.	0	Reverse acting Use this for systems where an increase in control output will cause a corresponding increase in PV (e.g. a heating process).		
		1	Direct acting Use this for systems where an increase in control output will cause a corresponding decrease in PV (e.g. a refrigeration process).		
DerivativeType	This parameter configures whether the derivative term of the PID responds to rate of change of PV or rate of change of deviation (that is, rate of change of the difference between PV and the setpoint). Derivative on PV is recommended by default, however derivative on deviation may sometimes be useful, for example to reduce overshoot at the end of a setpoint ramp. Care is needed on sensitive processes, as it will also cause derivative 'kick' (harsh changes in output) when the setpoint changes. For Cascade Loop Type, this relates to Secondary PID.	0	Derivative action on PV The derivative term responds only to the rate of change of the process variable.		
		1	Derivative action on deviation The derivative term will respond to the rate of change of the difference between PV and the setpoint.		
PropBandUnits	This parameter configures the units used to specify the PID proportional bands. For Cascade Loop Type, this refers to the Secondary PID controller.	0	Engineering units Proportional bands set in engineering (PV) units. For example, degrees C.		
		1	Percent Proportional bands set in percent of loop span (RangeHighLimit minus RangeLowLimit).		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Config			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
StandbyModeRecovery Mode	This parameter configures the behavior in the following circumstances: During and on exit from device Config mode or device Standby mode. On device start-up following a power cycle or power outage event.	0	Hold mode in Config and Standby, recovery from last mode During Standby or Config mode the Loop will assume Hold mode and the Loop output will hold to the last value. At Startup recovery or on exit from Instrument Standby or Config modes, the Loop will assume the last operating mode and initialize the output to its last value.		
		1	Inhibit mode in Config and Standby, recovery from last mode During Standby or Config mode the Loop will assume Inhibit mode and the Loop output will step to Inhibit OP. At Startup recovery or on exit from Instrument Standby or Config modes, the loop will assume the last operating mode and initialize the output to Inhibit OP.		
		2	Inhibit mode in Config and Standby, recovery in Manual During Standby or Config mode the Loop will assume Inhibit mode and the Loop output will step to Inhibit OP. At Startup recovery or on exit from Instrument Standby or Config modes, the loop will assume Manual mode and initialize the output to Inhibit OP.		
PrimaryPVBadTransfer	The parameter configures the type of transfer to in Forced Auto if, for instance, Primary PV goes bad (for example, due to a sensor break). This is only followed if transitioning to Forced Auto from Cascade mode or PrimaryTune mode because of a bad status of at least one between PrimaryPV, SecondaryRSP or SecondaryRSPTrim. Transitioning from Auto or higher priority modes will be bumpless for the Secondary Local Setpoint. Transitioning due to the ForcedAuto input being asserted in modes with lower priority than Forced Auto will make Secondary Local Setpoint go to the Secondary Fallback Setpoint.	0	Fallback Secondary SP The secondary setpoint will be set to the FallbackSecondarySP.		
		1	Hold Secondary SP The Secondary Working SP will be frozen at the last good value.		
		2	Forced Manual Transfer The strategy will follow the Forced Manual Transfer.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Config			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
ForcedModesRecovery	This parameter configures the loop recovery strategy on exit from Forced Manual mode. For example, when the PV recovers from a bad status. In Cascade Loop Type it also configures the recovery strategy on exit from Forced Auto mode. For example, when the Primary PV recovers from a bad status.	0	Recover from last operative mode On exit from Forced Manual or Forced Auto, the loop will assume the last operative mode.		
		1	Stay in Manual / Auto after Forced Manual / Auto On exit from Forced Manual, the loop will automatically transition to Manual mode. In Cascade Loop Type, on exit from Forced Auto, the loop will automatically transition to Auto mode.		
PVBadTransfer	This parameter configures the type of transfer to Forced Manual to perform if, for instance, PV goes bad (for example, due to a sensor break). This is only followed if transitioning to Forced Manual from Auto mode or Tune mode (or Cascade automatic modes for Cascade Loop type) because of a bad status of at least one between PV, DV or Remote Output Limits. Transitioning from Manual or higher priority modes will be bumpless. Transitioning due to the ForcedManual input being asserted in modes with lower priority than Forced Manual will go to the Fallback Value.	0	Fallback Output Value The FallbackValue will be applied to the output.		
		1	Hold The last good output will be applied. This will be an output value from approximately one second ago.		
ManualTransfer	This configures the type of transfer to perform when the mode is changed by the operator to Manual. This only applies when transitioning from Cascade Auto or Secondary Auto. Transfer from other modes will be bumpless.	0	Track (Bumpless) transfer The Manual Output will track the Working Output while the mode is not MANUAL. This aids in a bumpless transfer when the mode does go to MANUAL.		
		1	Step transfer The Manual Output will be set to the Manual Step Value while the mode is not MANUAL.		
		2	Last Value The Manual Output will remain at the last value used.		

Setpoint Parameters

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Setpoint			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
SPUnits	Units of the setpoint parameters of the Setpoint list.	0	No configured units		
		1	Absolute Temperature The parameter associated with this units definition is an absolute temperature and hence will adopt the global temperature units of the instrument. In addition, if the global units are changed the parameter, will be converted to the new units. for example degC to degF		
		2	Volts		
		3	milli Volts		
		4	Amps		
		5	milli Amps		
		6	pH		
		7	millimetres of mercury		
		8	pounds per square inch		
		9	Bar		
		10	milli Bar		
		11	Percent Relative Humidity		
		12	Percent		
		13	millimetres of water gauge		
		14	inches of water gauge		
		15			
		16	Ohms		
		17			
		18	percent of oxygen		
		19	parts per million		
		20	percent of carbon dioxide		
		21	percent carbon potential		
		22	percent per second		
		23			
		24	Relative temperature		
		25	Vacuum		
		26	Seconds		
		27	Minutes		
		28	Hours		
		29	Days		
		30	Megabytes		
		31	Per Minute		
		32	Milliseconds		
SPResolution	Resolution of the setpoint parameters of the Setpoint list.	0	No decimal places		
		1	One decimal place		
		2	Two decimal places		
		3	Three decimal places		
		4	Four decimal places		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Setpoint			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryRangeHighLimit	The Primary Range Limits provide a set of absolute upper and lower bounds for setpoints within the primary control loop. Any derived setpoints are ultimately clipped to be within the range limits. If the primary proportional band is configured as percentage of span, the span is derived from the Primary Range Limits.				
PrimaryRangeLowLimit					
PrimarySPHighLimit	Upper boundary of the Primary PID Setpoint.				
PrimarySPLowLimit	Lower boundary of the primary setpoint.				
RangeHighLimit	The Range Limits provide a set of absolute upper and lower bounds for setpoints within the control loop. Any derived setpoints are ultimately clipped to be within the Range Limits. If the proportional band is configured as percentage of span, the span is derived from the Range Limits. For Cascade Loop Type this refers to the Secondary PID.				
RangeLowLimit					
SPHighLimit	Upper boundary of the controller Setpoint. For Cascade Loop Type, this relates to Secondary controller.				
SPLowLimit	Lower boundary of the controller Setpoint. For Cascade Loop Type, this relates to Secondary controller.				
SPSelect	Selects between the loops local setpoints, SP1 and SP2.	0	Setpoint 1		
		1	Setpoint 2		
SP1	Setpoint 1 is the primary local setpoint of the controller.				
SP2	Setpoint 2 is the secondary local setpoint of the controller. It is often used as a standby setpoint.				
PSPSelect	This input selects the program setpoint (PSP). When asserted, it overrides the SP1/SP2 selection. It is typically wired to the setpoint programmer function block so that the loop will use the PSP when a program is in Run mode.	0	Off		
		1	On		
PSP	The program setpoint is an alternative local setpoint. The value is supplied by a setpoint programmer.				
RSPType	This parameter configures the remote setpoint topology.	0	Remote setpoint with local trim The remote setpoint (RSP) is used as a setpoint for the control algorithm. Optionally, a local trim can be applied.		
		1	Local setpoint with remote trim The local setpoint (SP1/SP2) is used as the setpoint for the control algorithm. The remote setpoint (RSP) acts as a remote trim on this local setpoint.		
RSPHighLimit	This sets an upper boundary for the RSP parameter. It applies whether the RSP is acting as an absolute setpoint or as a trim on a local setpoint.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Setpoint			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
RSPLowLimit	This sets a lower boundary for the RSP parameter. It applies whether the RSP is acting as an absolute setpoint or as a trim on a local setpoint.				
RSPActivate	This input is used to activate the remote setpoint (RSP). The remote setpoint cannot become active unless this input is activated. This is typically used in a cascade arrangement and allows the primary to signal to the secondary that it is providing a valid output. That is, the Primary PID controller's Loop.Diagnostics.PrimaryReady parameter is expected to be wired into here.	0	Off		
		1	On		
RSP	The remote setpoint (RSP) is typically used in a cascade control arrangement or in a multi-zone process, where a Primary PID controller is transmitting a setpoint to the secondary. For the remote setpoint to become active, the RSP status must be good, the RSPActivate input must be true and RemLocal must be set to Remote. The RSP can either be used as a setpoint itself (optionally with a local trim) or as a remote trim on a local setpoint.				
SPTrimHighLimit	The upper boundary of the local trim (SPTrim).				
SPTrimLowLimit	The lower boundary of the local trim (SPTrim).				
SPTrim	Trim is an offset added to the setpoint. The trim may be either positive or negative and the range of the trim is restricted by the Trim Limits. Setpoint trims may be used in a multi-zone process. A primary zone retransmits the setpoint to the other zones and a local trim may be applied to each zone to produce a profile along the length of the machine.				
SPRateUnits	This configures the units used to specify the Setpoint Rate Limits.	0	PV units per second.		
		1	PV units per minute.		
		2	PV units per hour.		
SPRateUp	Constrains the maximum rate at which the working setpoint can change in an increasing (upwards) direction. Setpoint Rate Limiting is often used to minimize rapid bumps in controller output that may damage equipment or product, or cause upset to downstream processes.	0	Off		
SPRateDown	Constrains the rate at which the working setpoint can change in a decreasing (downwards) direction. Setpoint rate limiting is often used to minimize rapid bumps in controller output that may damage equipment or product, or cause upset to downstream processes.	0	Off		
SPRateDeactivate	When true (set to 1), setpoint rate limiting is suspended.	0	No		
		1	Yes		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Setpoint			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
SPRateDone	When true (set to 1), this indicates that the setpoint is not currently being rate limited.	0	No		
		1	Yes		
SPRateServo	When the setpoint is being rate limited and servo-to-PV is activated, changing the target SP will cause the working SP to servo (step) to the current PV before ramping to the new target. This feature is only applied to SP1 and SP2 and not to the program or remote setpoints.	0	Off		
		1	On		
SPTracksPV	When activated in Single Loop Type, this option will cause the selected local setpoint (SP1/SP2) to track the PV whenever the controller is in Manual, Forced Manual or higher priority modes. In Cascade Loop Type instead, the selected local setpoint will track the Primary PV whenever the controller is in Auto, Forced Auto or higher priority modes. This then allows the process operating point to be maintained if the controller is later switched into Auto mode (for single loop type) or into Cascade mode (for cascade loop type).	0	Off		
		1	On		
SPTracksPSP	When activated, this option will cause the selected local setpoint (SP1/SP2) to track the program setpoint (PSP) while the program is running. This then allows the process operating point to be maintained when the program has completed and been reset.	0	Off		
		1	On		
SPTracksRSP	When activated, this option will cause the selected local setpoint (SP1/SP2) to track the remote setpoint (RSP) while the remote setpoint is active. If the RSP is acting as a remote trim on a local setpoint, then it is the local trim parameter (SPTrim) that will track RSP. This then allows the process operating point to be maintained if the mode is switched to AUTO.	0	Off		
		1	On		
SPIntBal	When activated, this causes the control algorithm to perform an integral balance whenever the target setpoint is changed. It does not apply when the mode is REMOTE. The effect of this option is to suppress proportional and derivative kicks whenever the setpoint changes, so that the output moves smoothly to its new value under integral action. This option is similar to having both proportional and derivative terms act on PV only, and not deviation.	0	Off		
		1	On		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Setpoint			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
BackCalcPV	This output is the back-calculated PV. It is the value of PV minus the setpoint trim. This is typically wired to the PV input of a setpoint programmer. Wiring this input, rather than the PV itself, allows the holdback feature to take account of any setpoint trim that may be applied and also allows setpoint programs to start smoothly with the working setpoint equal to the PV, if configured.				
BackCalcSP	This output is the back-calculated SP. It is the working setpoint minus the setpoint trim. This is typically wired to the servo input of a setpoint programmer, so that it can start smoothly without bumping the working setpoint, if configured.				

Cascade Scaling Parameters

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Cascade			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
SecondarySPType	This activates, in Trim Cascade Type, selection of the source used as the main 0component of the secondary loop setpoint, which is then trimmed by the primary PID.	0	Primary PV is used as the base for Trim mode calculation of the secondary setpoint. PrimarySP SecondarySPType is selected in applications where the response speed is the priority and the actuators can be driven at full power without causing any damage to the plant. The response is accelerated by passing directly the primary SP to the secondary PID, on top of which the primary PID adds its adjustment component.		
		1	Primary Working SP is used as the base for Trim mode calculation of the secondary setpoint. PrimaryPV SecondarySPType is selected in applications where the secondary process variable needs to change gradually to avoid damage to the plant, e.g. where thermal shock needs to be avoided. The actuator speed is controlled automatically by the dynamics of the plant itself, by deriving the main component of the secondary SP from the plant primary PV. The user can further constrain the primary PID trim component added to the secondary SP within the trim range: TrimRangeLow, TrimRangeHigh.		
		2	Secondary Remote SP is used as the base for Trim mode calculation of the secondary setpoint. SecondaryRSP is used as Secondary SP Type in special applications where the main component of the Secondary SP is wired from an external source - for example, an analog input PV. If the status of the SecondaryRSP is bad the cascade loop mode falls back from Cascade mode to Forced Auto mode.		
SecondaryRSPTrimActivate	This activates, in Full Cascade Type, the use of Remote Secondary Setpoint Trim. If activated, the Remote Secondary Setpoint Trim is added to the main component of the secondary loop setpoint and can be used to alter the behavior of the Full-scale cascade loop for special applications.	0	Off		
		1	On		
SecondaryRSPTrimHighLimit	Sets the upper boundary of the Remote Secondary Setpoint Trim.				

Blocks – SuperLoop.1 to SuperLoop.24		**Sub-block: Cascade**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
SecondaryRSPTrimLowLimit	Sets the lower boundary of the Remote Secondary Setpoint Trim.				
SecondaryRSPTrim	The Secondary remote trim setpoint parameter allows the behavior of the Full-scale type cascade loop to be altered for special applications. It can be activated using SecondaryRSPTrimActivate in Full-scale cascade type. If activated, it allows the value wired or written to be used as a trim of the main component of the Secondary SP controlled by the Primary PID. If SecondaryRSPTrim is activated and its status is bad, the cascade loop mode falls back from Cascade mode to Forced Auto mode.				
LimitedHeadHighType	Selection of the upper bound of the Limited Head function for Full Scale Setpoint. The High Limited Head function may be used to reduce overshoot when Primary PV and Secondary PV have same units (e.g. both temperatures).	0	Off Limited Head function not selected.		
		1	Primary Working Setpoint Limited Head function activated and based on Primary Working Setpoint. The Full Scale Setpoint is limited by an upper bound and/or by a lower bound, respectively for Limited Head High and Low Type. The upper bound is calculated as Primary Working Setpoint plus the Limited Head High, whereas the lower bound as Primary Working Setpoint minus Limited Head Low.		
LimitedHeadHigh	Adjustment parameter for the Limited Head upper bound of the Full Scale Secondary Setpoint. Lower values for Limited Head High can help in reducing the Primary PV overshoot, but a too low value can cause a sluggish response or even prevent from reaching the Primary setpoint. The Limited Head High lowest valid value is given by the gap between Primary SP and Secondary PV at steady state: for lower values than that, the Primary PV will not reach its setpoint. For values of the Limited Head High greater than the difference between Primary SP and the Secondary PV peak during the transient response, the strategy will not produce any change.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Cascade			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
LimitedHeadLowType	Selection of the lower bound of the Limited Head function for Full Scale Setpoint. The Low Limited Head function may be used to reduce undershoot when Primary PV and Secondary PV have same units (e.g. both temperatures).	0	Off Limited Head function not selected.		
		1	Primary Working Setpoint Limited Head function activated and based on Primary Working Setpoint. The Full Scale Setpoint is limited by an upper bound and/or by a lower bound, respectively for Limited Head High and Low Type. The upper bound is calculated as Primary Working Setpoint plus the Limited Head High, whereas the lower bound as Primary Working Setpoint minus Limited Head Low.		
LimitedHeadLow	Adjustment parameter for the Limited Head lower bound of the Full Scale Secondary Setpoint. Higher values for Limited Head Low can help in reducing the Primary PV undershoot, but a too high value can cause a sluggish response or even prevent from reaching the Primary setpoint. The Limited Head Low highest valid value is given by the gap between Primary SP and Secondary PV at steady state: for greater values than that, the Primary PV will not reach its setpoint. For values of the Limited Head Low lower than the difference between Primary SP and the Secondary PV peak during the transient response, the strategy will not produce any change.				
TrimRangeHigh	This defines, in Trim Cascade Type, the upper boundary of secondary loop setpoint trim to which upper boundary of Primary PID output is mapped. After this mapping, the secondary setpoint trim is further constrained within the secondary setpoint Trim Limits.				
TrimRangeLow	This defines, in Trim Cascade Type, the lower boundary of secondary loop setpoint trim to which the lower boundary of Primary PID output is mapped. After this mapping, the secondary setpoint trim is further constrained within the secondary setpoint Trim Limits.				
TrimHighLimit	Upper boundary used in Trim Cascade Type to constrain the secondary setpoint trim.				
TrimLowLimit	Lower boundary used in Trim Cascade Type to constrain the secondary setpoint trim.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Cascade			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
SecondaryRSP	The remote Secondary setpoint parameter allows the behavior of the Trim type cascade loop to be altered for special applications. As an alternative to using the primary setpoint (or in certain applications primary PV) in the calculation of the secondary setpoint, it allows the value wired or written to this input to the secondary loop to be used. This input is selected using RemoteSecondarySPActivate in Trim cascade type setting SecondarySPType to remote secondary SP. If the status of the SecondaryRSP is bad, the cascade loop mode falls back from Cascade mode to Forced Auto mode.				
SecondaryLocalSP	Secondary Local Setpoint used by the Secondary controller in (local) Auto mode.				
SecondaryLocalSPTracksPV	Whilst activated and in Manual, Forced Manual or higher priority modes, the secondary local setpoint will track the Secondary PV.	0	Off		
		1	On		
FallbackSecondarySP	This is the setpoint for the secondary loop when the primary sensor has gone into sensor break and the PV bad transfer for the primary is set to FallbackSecondarySP.				

Feedforward Parameters

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Feedforward			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
FFType	Selects the type of feedforward.	0	Feedforward is deactivated		
		1	The working setpoint is used as the input to the feedforward compensator.		
		2	The PV is used as the input to the feedforward compensator. This is sometimes used as an alternative to 'Delta-T' control.		
		3	The remote Disturbance Variable (DV) is used as the input to the feedforward compensator. This is usually a secondary process variable that can be used to head-off disturbances in the PV before they have a chance to occur.		
		4	The Primary working setpoint is used as the input to the feedforward compensator.		
		5	The Primary PV is used as the input to the feedforward compensator. This is sometimes used as an alternative to 'Delta-T' control.		
DV	The remote Disturbance Variable. This is typically a secondary measured process variable. This is usually a secondary process variable that can be used to compensate disturbances in the PV before they have a chance to occur.				
FFGain	The feedforward compensator gain. The feedforward input is multiplied by the gain.				
FFOffset	The feedforward compensator bias/offset. This value is added to the feedforward input after the gain.				
FFLeadTime	The feedforward compensator lead time constant can be used to 'speed-up' the feedforward action. **Set to 0 to deactivate the lead component.** Typically, the lead component is not used on its own without any lag. The lead and lag time constants allow dynamic compensation of the feedforward signal. The values are usually determined by characterizing the effect of the input on the process (for example, by a bump test). In the case of a Disturbance Variable, the values are chosen so that the disturbance and the correction 'arrive' at the process variable at the same instant, thereby minimizing any perturbation. Typically the lead time is set equal to the lag between the controller output and the PV, while the lag time is typically set equal to the lag between the DV and the PV.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Feedforward			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
FFLagTime	The feedforward compensator lag time constant can be used to slow down the feedforward action. **Set to 0 to deactivate the lag component.** The lead and lag time constants allow dynamic compensation of the feedforward signal. The values are usually determined by characterizing the effect of the input on the process (for example, by a bump test). In the case of a Disturbance Variable, the values are chosen so that the disturbance and the correction 'arrive' at the process variable at the same instant, thereby minimizing any perturbation. Typically the lead time is set equal to the lag between the controller output and the PV, while the lag time is typically set equal to the lag between the DV and the PV.				
FFHighLimit	The upper boundary of feedforward output. This boundary is applied to the feedforward output prior to it being added to the PID output.				
FFLowLimit	The lower boundary of feedforward output. This boundary is applied to the feedforward output prior to it being added to the PID output.				
FFHold	When true, the feedforward output will maintain its current value. This can be used to temporarily halt feedforward action.	0	No		
		1	Yes		
FFOutput	The feedforward output contribution.				
PIDTrimLimit	The PID Trim Limit constrains the effect of the PID output. Eurotherm's Feedforward implementation allows the Feedforward component to make the dominant contribution to the control output. The PID contribution can then be used as a trim on the Feedforward value. This arrangement is sometimes known as "Feedforward with Feedback Trim". This parameter defines symmetrical boundaries (expressed as a percentage of output) around the PID output, to constrain the magnitude of the PID contribution. To instead allow PID contribution to dominate, set a large value for this parameter (400.0).				

Autotune Parameters

Autotuning

This controller includes sophisticated autotuning algorithms that are able to determine appropriate values for the PID tuning constants (Ch1PB, Ch2PB, TI, TD, CBH, CBL). To do this, the algorithms perform experiments on the process by manipulating the controller output and analyzing the PV response.

Upon starting the autotune, there is a one minute delay while the loop is allowed to settle. During this time you may edit the loop setpoint. Once a minute has elapsed, no further changes in setpoint are allowed as they would interfere with the experiment.

Oscillations of the process value may damage the process being tuned. It is recommended to set the setpoint for tuning purposes below the normal running setpoint value.

The autotuner works by switching the output on and off to induce an oscillation in the process value.

From the information contained in this oscillation, it calculates the tuning parameter values.

If the process cannot tolerate +/-100% output being applied, then the output during tuning can be restricted by setting the Tuning Output Limits. However, the process value needs to oscillate to some degree for the tuner to be able to calculate values. Larger oscillations generally lead to a better signal-to-noise ratio and a better tune.

An autotune can be performed at any time, but normally it is performed only once during the initial commissioning of the process. However, if the process performance subsequently becomes unsatisfactory (because its characteristics have changed), you can re-tune again for the new conditions.

How To Tune

1. Set the setpoint to the value at which you will normally operate the process. If overshoot while tuning cannot be tolerated then enter a lower value than normal.
2. Activate the autotuner. The controller induces an oscillation in the process variable by first setting upper output boundary, and then lower output boundary. The first cycle is not complete until the process variable has reached the working setpoint.
3. After two or three cycles of oscillation the autotuner will move to the next stage of tuning. If the controller is configured with dual-channels (e.g. heat and cool), the autotuner will perform a further experiment. It will either put the PV into another cycle of oscillation or will attempt to control to the working setpoint.
4. The controller then calculates the tuning parameters.
5. The autotune is completed and the tuner switches itself off. Normal control action resumes. If you want ‘Proportional only’, ‘PD’, or ‘PI’ control, set the ‘TI’ or ‘TD’ parameters to off before activating the autotune. The tuner will leave them off and will not calculate a value for them.
6. If gain scheduling is activated, the autotuner will write the calculated parameters to the tuneset that is active when the tune completes.

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Autotune			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
TuneType	Selects which PID loop of the cascade loop is to be autotuned.	0	Tunes the secondary PID loop.		
		1	Tunes the primary PID loop.		
AutotuneActivate	Starts an Autotune. Aborts an autotune if set false (0) during tuning.	0	Off		
		1	On		
TuneSecondarySPHigh	This is the upper boundary value of the absolute setpoint which the primary autotuning can apply to the secondary loop. Other Setpoint Limits associated with the secondary loop may further restrict the actual value which is applied.				
TuneSecondarySPLow	This is the lower boundary value of the absolute setpoint which the primary autotuning can apply to the secondary loop. Other Setpoint Limits associated with the secondary loop may further restrict the actual value which is applied.				
TuneOutputHigh	Sets the upper output boundary that the autotuner will apply during the autotuning experiment.				
TuneOutputLow	Sets the lower output boundary that the autotuner will apply during the autotuning experiment.				
Ch2TuneType	Configures which experiment will be used to determine the relationship between channel 1 and channel 2 proportional bands.	0	Tunes the channel 2 proportional band using the standard relative channel 2 tuning algorithm.		
		1	Uses a model-based tuning algorithm that has been shown to give enhanced results with higher-order, low-loss plants. In particular, it performs well with heavily-lagged temperature processes.		
		2	This option can be used to stop the autotune trying to determine the channel 2 proportional band. Instead, it will maintain the existing ratio between the channel 1 and 2 proportional bands. In general, this option is not recommended unless there is a known reason to select it (for example, if the relative gain is already known and the tuner gives an incorrect value).		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Autotune			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
TuneAlgo	This parameter reports which autotuning algorithm is available for the current control configuration. The appropriate tuning algorithm is automatically determined.	0	Not available There is no autotuner available for the current control configuration.		
		1	Standard PID tune The standard autotuner based on a modified relay method. It requires two cycles to complete (not including the relative ch2 tune) This is used for PID-only configurations and where there is no output rate limiting configured.		
		2	Fourier autotune algorithm This algorithm uses the same modified relay method but uses a more complex analysis based on the work of Joseph Fourier. It requires three cycles to complete. If Channel 2 is configured then an additional tuning stage will be executed to determine the Channel 2 relative gain ratio. This algorithm is used for VP or mixed channel configurations and is also used whenever there is an Output Rate Limit set.		
TuneStatus	This reports the status of the autotuner.	0	Unavailable		Read-only
		1	Ready to run an autotune.		Read-only
		2	An autotune has been triggered but the loop mode is prohibiting it from starting. When the mode goes to AUTO, the tune will start.		Read-only
		3	The autotuner is running and currently has authority over the controller outputs.		Read-only
		4	The autotune successfully completed and has updated the tuneset parameters.		Read-only
		5	The last autotune was aborted.		Read-only
		6	One of the stages of the last autotune exceeded the two hours per stage boundary. This can occur if, for example, the output limits do not allow for the setpoint to be achieved.		Read-only
		7	A buffer overflow occurred while collecting process data. Contact Eurotherm support.		Read-only

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Autotune			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
TuneStage	This reports the stage of the current autotuning sequence.	0	Idle - not autotuning		Read-only
		1	The process is being monitored. This stage lasts one minute. The setpoint can be changed during this stage.		Read-only
		2	An initial oscillation is being established.		Read-only
		3	Highest output applied		Read-only
		4	Lowest output applied		Read-only
		5	Relative ch2 gain experiment running		Read-only
		6	PD control The autotuner is trying to control to setpoint and is examining the reponse.		Read-only
		7	Analysis The autotuner is calculating the new tuning parameters.		Read-only
StageTime	The time elapsed in the current autotune stage. This is reset each time the autotuner advances a stage. If this exceeds two hours, a timeout will occur.				Read-only

PrimaryPID (TuneSets) Parameters

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryGainScheduler	Primary Gain scheduling is provided so that processes which change their characteristics can be controlled. For example, in some temperature processes, the dynamical response may be very different at low temperatures from that at high temperatures. Primary Gain Scheduling typically uses one of the loop's parameters to select the active Primary PID set -- this parameter is known as the scheduling variable (SV). Multiple sets are available and for each a boundary is provided which defines the switching point. The internal scheduling variables (process variable, working setpoint, working output, deviation) used by this gain scheduler strategy are referred to the Primary Loop.	0	Gain scheduling is off		
		1	The active tuneset can be chosen manually setting ActiveSet.		
		2	The PID set is selected automatically using the PV (or PrimaryPV for cascade loop type) as scheduling variable.		
		3	The PID set is selected automatically using WorkingSP (or PrimaryWorkingSP for cascade loop type) as scheduling variable.		
		4	The PID set is selected automatically using WorkingOutput as scheduling variable.		
		5	The PID set is selected automatically using the deviation PV-WorkingSP (or PrimaryPVPrimaryWorkingSP for cascade loop type) as scheduling variable.		
		6	This option selects set 2 when the Setpoint source is remote, otherwise it selects set 1. This may be useful to effectively switch integral action off when the function block is used as a secondary in a cascade control strategy.		
		7	The PID set is selected automatically using the remote scheduling variable RemoteSV. If the status of Remote SV is bad, the first tuneset is selected.		
		8	This option selects set 2 when the controller is in Cascade control mode, otherwise it selects set 1. This may be useful to effectively switch integral action off for the secondary loop in Cascade mode and on in Secondary mode.		
		9	This options selects the same gain set number selected for the Primary PID gain scheduler. It is only available for Secondary gain scheduler.		
PrimaryNumSets	Number of activated tunesets for the Primary PID.				
PrimaryActiveSet	Currently selected Primary PID set.				
PrimaryRemoteSV	Remote input used to select the Primary PID set. The Schedule type needs to be set to REMOTE for this parameter to be available.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryBoundary	The Primary gain scheduler compares the scheduling variable against the specified boundary. If the scheduling variable is below the boundary then set 1 is active. If above the boundary then set 2 is active.				
PrimaryBoundary23	The Primary gain scheduler compares the scheduling variable against the specified boundary. If the scheduling variable is below the boundary then set 1 is active. If above the boundary then set 2 is active.				
PrimaryBoundaryHyst	This specifies the amount of hysteresis around the Primary gain scheduling boundary. This is used to avoid continuous switching as the scheduling variable passes through the boundary.				
PrimaryPropBand	The Primary proportional band. The Primary proportional band is the band within which the Primary PID controller output changes in a linear fashion between 0% to 100% (when considering the proportional term only). More generally, it determines the gain of the Primary PID controller. The smaller the proportional band, the more aggressively the Primary PID controller will respond to deviations of the Primary PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response.				
PrimaryIntegralTime	Integral helps to achieve zero steady-state control deviation. In a proportional-only controller, when the PV is exactly equal to the setpoint, the controller will deliver 0% output. For self-regulating processes this will lead to the PV settling at a point away from the setpoint. By activating integral action, the controller will monitor the deviation and add further output demands to remove the steady state deviations. Too small an integral time will cause the process to overshoot, while too large an integral time will slow the approach of the PV and cause sluggish response. The integral action may be deactivated by setting its value to Off (0). Integral times are specified in seconds.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryDerivativeTime	Derivative adds an anticipatory element to the controller. It can be used to increase the stability of the system, thereby allowing faster response to disturbances. The derivative acts on the loop's rate of change (either the rate of change of PV or rate of change of deviation, depending on configuration). The faster the rate of change, the more the derivative attempts to counteract it and the larger the derivative output component. Derivative is particularly effective in temperature processes. In some other applications, derivative may be the cause of instability. If the PV is subject to disturbances, then derivative can amplify those disturbances and cause excessive output changes, in these situations it is often better to deactivate the derivative and re-tune the loop. If set to Off (0), no derivative action will be applied. Derivative times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryCutbackHigh	Defines a High Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto).	0	Auto Three times the proportional band		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryCutbackLow	Defines a Low Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto).	0	Auto Three times the proportional band		
PrimaryManualReset	In controllers without integral action (also known as automatic reset), the manual reset parameter allows a constant addition to the output power to be set to remove any steady-state deviation. Effectively, it defines the output power when there is zero deviation. Manual Reset is specified in percentage output.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryPropBand2	The proportional band for Primary tuneset 2. The Primary proportional band is the band within which the Primary PID controller output changes in a linear fashion between 0% to 100% (when considering the proportional term only). More generally, it determines the gain of the Primary PID controller. The smaller the proportional band, the more aggressively the Primary PID controller will respond to deviations of the Primary PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response.				
PrimaryIntegralTime2	Integral time for Primary tuneset 2. Integral helps to achieve zero steady-state control deviation. In a proportional-only controller, when the PV is exactly equal to the setpoint, the controller will deliver 0% output. For self-regulating processes this will lead to the PV settling at a point away from the setpoint. By activating integral action, the controller will monitor the deviation and add further output demands to remove the steady state deviations. Too small an integral time will cause the process to overshoot, while too large an integral time will slow the approach of the PV and cause sluggish response. The integral action may be deactivated by setting its value to Off (0). Integral times are specified in seconds.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryDerivativeTime2	Derivative time for Primary tuneset 2. Derivative adds an anticipatory element to the controller. It can be used to increase the stability of the system, thereby allowing faster response to disturbances. The derivative acts on the loop's rate of change (either the rate of change of PV or rate of change of deviation, depending on configuration). The faster the rate of change, the more the derivative attempts to counteract it and the larger the derivative output component. Derivative is particularly effective in temperature processes. In some other applications, derivative may be the cause of instability. If the PV is subject to disturbances, then derivative can amplify those disturbances and cause excessive output changes, in these situations it is often better to deactivate the derivative and re-tune the loop. If set to Off (0), no derivative action will be applied. Derivative times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryCutbackHigh2	Defines a High Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto).	0	Auto Three times the proportional band		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryCutbackLow2	Defines a Low Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto).	0	Auto Three times the proportional band		
PrimaryManualReset2	Manual Reset for Primary tuneset 2. In controllers without integral action (also known as automatic reset), the manual reset parameter allows a constant addition to the output power to be set to remove any steady-state deviation. Effectively, it defines the output power when there is zero deviation. Manual Reset is specified in percentage output.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryPropBand3	The proportional band for Primary tuneset 3. The Primary proportional band is the band within which the Primary PID controller output changes in a linear fashion between 0% to 100% (when considering the proportional term only). More generally, it determines the gain of the Primary PID controller. The smaller the proportional band, the more aggressively the Primary PID controller will respond to deviations of the Primary PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response.				
PrimaryIntegralTime3	Integral time for Primary tuneset 3. Integral helps to achieve zero steady-state control deviation. In a proportional-only controller, when the PV is exactly equal to the setpoint, the controller will deliver 0% output. For self-regulating processes this will lead to the PV settling at a point away from the setpoint. By activating integral action, the controller will monitor the deviation and add further output demands to remove the steady state deviations. Too small an integral time will cause the process to overshoot, while too large an integral time will slow the approach of the PV and cause sluggish response. The integral action may be deactivated by setting its value to Off (0). Integral times are specified in seconds.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryDerivativeTime3	Derivative time for Primary tuneset 3. Derivative adds an anticipatory element to the controller. It can be used to increase the stability of the system, thereby allowing faster response to disturbances. The derivative acts on the loop's rate of change (either the rate of change of PV or rate of change of deviation, depending on configuration). The faster the rate of change, the more the derivative attempts to counteract it and the larger the derivative output component. Derivative is particularly effective in temperature processes. In some other applications, derivative may be the cause of instability. If the PV is subject to disturbances, then derivative can amplify those disturbances and cause excessive output changes, in these situations it is often better to deactivate the derivative and re-tune the loop. If set to Off (0), no derivative action will be applied. Derivative times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryCutbackHigh3	Defines a High Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto).	0	Auto Three times the proportional band		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryCutbackLow3	Defines a Low Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto).	0	Auto Three times the proportional band		
PrimaryManualReset3	Manual Reset for Primary tuneset 3. In controllers without integral action (also known as automatic reset), the manual reset parameter allows a constant addition to the output power to be set to remove any steady-state deviation. Effectively, it defines the output power when there is zero deviation. Manual Reset is specified in percentage output.	0	Off		

PID (TuneSets) Parameters

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
GainScheduler	Gain scheduling is provided so that processes which change their characteristics can be controlled. For example, in some temperature processes, the dynamical response may be very different at low temperatures from that at high temperatures. Gain Scheduling typically uses one of the loop's parameters to select the active PID set -- this parameter is known as the scheduling variable (SV). Multiple sets are available and for each a boundary is provided which defines the switching point. For Cascade Loop Type this refers to the Secondary PID.	0	Gain scheduling is off		
		1	The active tuneset can be chosen manually setting ActiveSet.		
		2	The PID set is selected automatically using the process variable PV (or PrimaryPV for cascade loop type) as scheduling variable.		
		3	The PID set is selected automatically using WorkingSP (or PrimaryWorkingSP for cascade loop type) as scheduling variable.		
		4	The PID set is selected automatically using WorkingOutput as scheduling variable.		
		5	The PID set is selected automatically using the deviation PV-WorkingSP (or PrimaryPVPrimaryWorkingSP for cascade loop type) as scheduling variable.		
		6	This option selects set 2 when the Setpoint source is remote, otherwise it selects set 1. This may be useful to effectively switch integral action off when the function block is used as a secondary in a cascade control strategy.		
		7	The PID set is selected automatically using the remote scheduling variable RemoteSV. If the status of Remote SV is bad, the first tuneset is selected.		
		8	This option selects set 2 when the controller is in Cascade control mode, otherwise it selects set 1. This may be useful to effectively switch integral action off for the secondary loop in Cascade mode and on in Secondary mode.		
		9	This options selects the same gain set number selected for the Primary PID gain scheduler. It is only available for Secondary gain scheduler.		
NumSets	Number of activated tunesets. For Cascade Loop Type this refers to the Secondary PID.				
ActiveSet	Currently selected PID set. For Cascade Loop Type this refers to the Secondary PID.				
RemoteSV	Remote input used to select the PID set. The Schedule type needs to be set to REMOTE for this parameter to be available. For Cascade Loop Type this refers to the Secondary PID.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Boundary	The gain scheduler compares the scheduling variable against the specified boundary. If the scheduling variable is below the boundary then set 1 is active. If above the boundary then set 2 is active. For Cascade Loop Type this refers to the Secondary PID.				
Boundary23	The gain scheduler compares the scheduling variable against the specified boundary. If the scheduling variable is below the boundary then set 2 is active. If above the boundary then set 3 is active. For Cascade Loop Type this refers to the Secondary PID.				
BoundaryHyst	This specifies the amount of hysteresis around the gain scheduling boundary. This is used to avoid continuous switching as the scheduling variable passes through the boundary. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
Ch1PropBand	The channel 1 proportional band. The channel 1 proportional band is the band within which the controller output changes in a linear fashion between 0% to 100% (when considering the proportional term only). More generally, it determines the gain of the controller. The smaller the proportional band, the more aggressively the controller will respond to deviations of the PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response. A proportional band is provided for each of the two channels so that the difference in process gain can be accounted for (for example, heating may be stronger than cooling, which necessitates a different proportional band). For Cascade Loop Type this refers to the Secondary PID.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ch2PropBand	The channel 2 proportional band. The channel 2 proportional band is the band within which the controller output changes in a linear fashion between -100% to 0% (when considering the proportional term only). More generally, it determines the gain of the controller. The smaller the proportional band, the more aggressively the controller will respond to deviations of PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response. A proportional band is provided for each of the two channels so that the difference in process gain can be accounted for (for example, heating may be stronger than cooling, which necessitates a different proportional band). For Cascade Loop Type this refers to the Secondary PID.				
IntegralTime	Integral helps to achieve zero steady-state control deviation. In a proportional-only controller, when the PV is exactly equal to the setpoint, the controller will deliver 0% output. For self-regulating processes this will lead to the PV settling at a point away from the setpoint. By activating integral action, the controller will monitor the deviation and add further output demands to remove the steady state deviations. Too small an integral time will cause the process to overshoot, while too large an integral time will slow the approach of the PV and cause sluggish response. The integral action may be deactivated by setting its value to Off (0). Integral times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		**Sub-block: PID**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
DerivativeTime	Derivative adds an anticipatory element to the controller. It can be used to increase the stability of the system, thereby allowing faster response to disturbances. The derivative acts on the loop's rate of change (either the rate of change of PV or rate of change of deviation, depending on configuration). The faster the rate of change, the more the derivative attempts to counteract it and the larger the derivative output component. Derivative is particularly effective in temperature processes. In some other applications, derivative may be the cause of instability. If the PV is subject to disturbances, then derivative can amplify those disturbances and cause excessive output changes, in these situations it is often better to deactivate the derivative and re-tune the loop. If set to Off (0), no derivative action will be applied. Derivative times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
CutbackHigh	Defines a High Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto). For Cascade Loop Type this refers to the Secondary PID.	0	Auto Three times the proportional band		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
CutbackLow	Defines a Low Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto). For Cascade Loop Type this refers to the Secondary PID.	0	Auto Three times the proportional band		
ManualReset	In controllers without integral action (also known as automatic reset), the manual reset parameter allows a constant addition to the output power to be set to remove any steady-state deviation. Effectively, it defines the output power when there is zero deviation. Manual Reset is specified in percentage output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
OutputHigh	This output boundary is applied when tuneset 1 is selected. These allow the Working Output Limits to be scheduled in the same way as the tuning parameters. The global Output Limits take precedence if they are more constraining than the scheduled output boundaries. Additionally, these scheduled boundaries do not prevent the Fallback Output Value from being achieved. For Cascade Loop Type this refers to the Secondary PID.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
OutputLow	This output boundary is applied when tuneset 1 is selected. These allow the Working Output Limits to be scheduled in the same way as the tuning parameters. The global Output Limits take precedence if they are more constraining than the scheduled output boundaries. Additionally, these scheduled boundaries do not prevent the Fallback Output Value from being achieved. For Cascade Loop Type this refers to the Secondary PID.				
Ch1OnOffHyst	This is set in the units of the PV. It defines the point below setpoint where the channel 1 output will activate. The output will deactivate when the PV is at setpoint. The hysteresis is used to minimize the chattering of the output at the control setpoint. If the hysteresis is set to 0 then any change in the PV when at setpoint will cause the output to switch. Typically, the hysteresis is set to a value which provides an acceptable life for the output contacts, but which does not cause unacceptable oscillations in the PV. If this performance is unacceptable, it is recommended that you try PID control with a time proportioning output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
Ch2OnOffHyst	This is set in the units of the PV. It defines the point above setpoint where the channel 2 output will activate. The output will deactivate when the PV is at setpoint. The hysteresis is used to minimize the chattering of the output at the control setpoint. If the hysteresis is set to 0 then any change in the PV when at setpoint will cause the output to switch. Typically, the hysteresis is set to a value which provides an acceptable life for the output contacts, but which does not cause unacceptable oscillations in the PV. If this performance is unacceptable, it is recommended that you try PID control with a time proportioning output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ch1PropBand2	The channel 1 proportional band for tuneset 2. The channel 1 proportional band is the band within which the controller output changes in a linear fashion between 0% to 100% (when considering the proportional term only). More generally, it determines the gain of the controller. The smaller the proportional band, the more aggressively the controller will respond to deviations of the PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response. A proportional band is provided for each of the two channels so that the difference in process gain can be accounted for (for example, heating may be stronger than cooling, which necessitates a different proportional band). For Cascade Loop Type this refers to the Secondary PID.				
Ch2PropBand2	The channel 2 proportional band for tuneset 2. The channel 2 proportional band is the band within which the controller output changes in a linear fashion between -100% to 0% (when considering the proportional term only). More generally, it determines the gain of the controller. The smaller the proportional band, the more aggressively the controller will respond to deviations of PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response. A proportional band is provided for each of the two channels so that the difference in process gain can be accounted for (for example, heating may be stronger than cooling, which necessitates a different proportional band). For Cascade Loop Type this refers to the Secondary PID.				
IntegralTime2	Integral time for tuneset 2. Integral helps to achieve zero steady-state control deviation. In a proportional-only controller, when the PV is exactly equal to the setpoint, the controller will deliver 0% output. For self-regulating processes this will lead to the PV settling at a point away from the setpoint. By activating integral action, the controller will monitor the deviation and add further output demands to remove the steady state deviations. Too small an integral time will cause the process to overshoot, while too large an integral time will slow the approach of the PV and cause sluggish response. The integral action may be deactivated by setting its value to Off (0). Integral times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
DerivativeTime2	Derivative time for tuneset 2. Derivative adds an anticipatory element to the controller. It can be used to increase the stability of the system, thereby allowing faster response to disturbances. The derivative acts on the loop's rate of change (either the rate of change of PV or rate of change of deviation, depending on configuration). The faster the rate of change, the more the derivative attempts to counteract it and the larger the derivative output component. Derivative is particularly effective in temperature processes. In some other applications, derivative may be the cause of instability. If the PV is subject to disturbances, then derivative can amplify those disturbances and cause excessive output changes, in these situations it is often better to deactivate the derivative and re-tune the loop. If set to Off (0), no derivative action will be applied. Derivative times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
CutbackHigh2	Defines a High Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto). For Cascade Loop Type this refers to the Secondary PID.	0	Auto Three times the proportional band		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
CutbackLow2	Defines a Low Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto). For Cascade Loop Type this refers to the Secondary PID.	0	Auto Three times the proportional band		
ManualReset2	Manual Reset for tuneset 2. In controllers without integral action (also known as automatic reset), the manual reset parameter allows a constant addition to the output power to be set to remove any steady-state deviation. Effectively, it defines the output power when there is zero deviation. Manual Reset is specified in percentage output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
OutputHigh2	This output boundary is applied when tuneset 2 is selected. These allow the Working Output Limits to be scheduled in the same way as the tuning parameters. The global Output Limits take precedence if they are more constraining than the scheduled output boundaries. Additionally, these scheduled boundaries do not prevent the Fallback Output Value from being achieved. For Cascade Loop Type this refers to the Secondary PID.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
OutputLow2	This output boundary is applied when tuneset 2 is selected. These allow the Working Output Limits to be scheduled in the same way as the tuning parameters. The global Output Limits take precedence if they are more constraining than the scheduled output boundaries. Additionally, these scheduled boundaries do not prevent the Fallback Output Value from being achieved. For Cascade Loop Type this refers to the Secondary PID.				
Ch1OnOffHyst2	This is set in the units of the PV. It defines the point below setpoint where the channel 1 output will activate. The output will deactivate when the PV is at setpoint. The hysteresis is used to minimize the chattering of the output at the control setpoint. If the hysteresis is set to 0 then any change in the PV when at setpoint will cause the output to switch. Typically, the hysteresis is set to a value which provides an acceptable life for the output contacts, but which does not cause unacceptable oscillations in the PV. If this performance is unacceptable, it is recommended that you try PID control with a time proportioning output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
Ch2OnOffHyst2	This is set in the units of the PV. It defines the point above setpoint where the channel 2 output will activate. The output will deactivate when the PV is at setpoint. The hysteresis is used to minimize the chattering of the output at the control setpoint. If the hysteresis is set to 0 then any change in the PV when at setpoint will cause the output to switch. Typically, the hysteresis is set to a value which provides an acceptable life for the output contacts, but which does not cause unacceptable oscillations in the PV. If this performance is unacceptable, it is recommended that you try PID control with a time proportioning output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ch1PropBand3	The channel 1 proportional band for tuneset 3. The channel 1 proportional band is the band within which the controller output changes in a linear fashion between 0% to 100% (when considering the proportional term only). More generally, it determines the gain of the controller. The smaller the proportional band, the more aggressively the controller will respond to deviations of the PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response. A proportional band is provided for each of the two channels so that the difference in process gain can be accounted for (for example, heating may be stronger than cooling, which necessitates a different proportional band). For Cascade Loop Type this refers to the Secondary PID.				
Ch2PropBand3	The channel 2 proportional band for tuneset 3. The channel 2 proportional band is the band within which the controller output changes in a linear fashion between -100% to 0% (when considering the proportional term only). More generally, it determines the gain of the controller. The smaller the proportional band, the more aggressively the controller will respond to deviations of PV from its setpoint. Too small a proportional band can cause oscillation, while too large a proportional band can cause sluggish response. A proportional band is provided for each of the two channels so that the difference in process gain can be accounted for (for example, heating may be stronger than cooling, which necessitates a different proportional band). For Cascade Loop Type this refers to the Secondary PID.				
IntegralTime3	Integral time for tuneset 3. Integral helps to achieve zero steady-state control deviation. In a proportional-only controller, when the PV is exactly equal to the setpoint, the controller will deliver 0% output. For self-regulating processes this will lead to the PV settling at a point away from the setpoint. By activating integral action, the controller will monitor the deviation and add further output demands to remove the steady state deviations. Too small an integral time will cause the process to overshoot, while too large an integral time will slow the approach of the PV and cause sluggish response. The integral action may be deactivated by setting its value to Off (0). Integral times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
DerivativeTime3	Derivative time for tuneset 3. Derivative adds an anticipatory element to the controller. It can be used to increase the stability of the system, thereby allowing faster response to disturbances. The derivative acts on the loop's rate of change (either the rate of change of PV or rate of change of deviation, depending on configuration). The faster the rate of change, the more the derivative attempts to counteract it and the larger the derivative output component. Derivative is particularly effective in temperature processes. In some other applications, derivative may be the cause of instability. If the PV is subject to disturbances, then derivative can amplify those disturbances and cause excessive output changes, in these situations it is often better to deactivate the derivative and re-tune the loop. If set to Off (0), no derivative action will be applied. Derivative times are specified in seconds. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
CutbackHigh3	Defines a High Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto). For Cascade Loop Type this refers to the Secondary PID.	0	Auto Three times the proportional band		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
CutbackLow3	Defines a Low Cutback threshold in the same units as the proportional band (either engineering units or percent of span, depending on configuration). Cutback is a system of approach control. The cutback high and low thresholds are used to tune the large-signal response of the system without affecting the small-signal performance. The normal PID parameters (PB, TI and TD) are typically tuned first for disturbance rejection. The cutback thresholds can then be used to independently tune the response to large setpoint changes. When the PV is above the setpoint and the deviation exceeds the cutback high threshold, the lower output boundary will be applied. Conversely, when the PV is below the setpoint and the deviation exceeds the cutback low threshold, the upper output boundary will be applied. This brings the PV rapidly towards the setpoint. Once the PV crosses the cutback threshold, the controller output starts to be 'cut back' in a manner designed to reduce overshoot. If you find that the PV is overshooting for large setpoint changes, try increasing the appropriate cutback threshold. Conversely, if you find that the output is reducing too early and causing a sluggish final approach, try decreasing the appropriate cutback threshold. The default is 0 (Auto). This sets the cutback thresholds to three times the proportional band. The autotuner will not attempt to tune the cutback parameters if they are set to 0 (Auto). For Cascade Loop Type this refers to the Secondary PID.	0	Auto Three times the proportional band		
ManualReset3	Manual Reset for tuneset 3. In controllers without integral action (also known as automatic reset), the manual reset parameter allows a constant addition to the output power to be set to remove any steady-state deviation. Effectively, it defines the output power when there is zero deviation. Manual Reset is specified in percentage output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
OutputHigh3	This output boundary is applied when tuneset 3 is selected. These allow the Working Output Limits to be scheduled in the same way as the tuning parameters. The global Output Limits take precedence if they are more constraining than the scheduled output boundaries. Additionally, these scheduled boundaries do not prevent the Fallback Output Value from being achieved. For Cascade Loop Type this refers to the Secondary PID.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: PID			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
OutputLow3	This output boundary is applied when tuneset 3 is selected. These allow the Working Output Limits to be scheduled in the same way as the tuning parameters. The global Output Limits take precedence if they are more constraining than the scheduled output boundaries. Additionally, these scheduled boundaries do not prevent the Fallback Output Value from being achieved. For Cascade Loop Type this refers to the Secondary PID.				
Ch1OnOffHyst3	This is set in the units of the PV. It defines the point below setpoint where the channel 1 output will activate. The output will deactivate when the PV is at setpoint. The hysteresis is used to minimize the chattering of the output at the control setpoint. If the hysteresis is set to 0 then any change in the PV when at setpoint will cause the output to switch. Typically, the hysteresis is set to a value which provides an acceptable life for the output contacts, but which does not cause unacceptable oscillations in the PV. If this performance is unacceptable, it is recommended that you try PID control with a time proportioning output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		
Ch2OnOffHyst3	This is set in the units of the PV. It defines the point above setpoint where the channel 2 output will activate. The output will deactivate when the PV is at setpoint. The hysteresis is used to minimize the chattering of the output at the control setpoint. If the hysteresis is set to 0 then any change in the PV when at setpoint will cause the output to switch. Typically, the hysteresis is set to a value which provides an acceptable life for the output contacts, but which does not cause unacceptable oscillations in the PV. If this performance is unacceptable, it is recommended that you try PID control with a time proportioning output. For Cascade Loop Type this refers to the Secondary PID.	0	Off		

Output Parameters

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Output			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
FallbackValue	The Fallback Output Value is used in the following circumstances: 1. If the Loop Bad alarm is active (for example, PV status goes bad because of sensor break), the loop will enter Forced Manual mode (ForcedManual) with either the fallback value or the last good output. This depends on the configured Loop Bad Transfer type. 2. If Forced Manual (ForcedManual) mode is activated by an external signal (for example a process alarm) then the fallback output value is applied.				
OutputHighLimit	The upper boundary of controller output. This parameter does not affect the Fallback Value being achieved in Manual mode.				
OutputLowLimit	The lower boundary of controller output. This parameter does not affect the Fallback Value being achieved in Manual mode.				
Ch1Output	Channel 1 (Heat) output. The Ch1 output is taken to be the positive output values (0 to +100). Typically this is wired to the control output (time proportioning or analog output).				
Ch2Output	Channel 2 (Cool) output. The Ch2 output is taken to be the negative output values (- 100 to 0). Typically this is wired to a control output (time proportioning or analog output).				
ManualOP	The Manual Output. This is used as the output when the loop is in Manual or ForcedManual modes. In Manual mode, the controller will still constrain the output to the Working Output Limits and the Output Rate Limits. In Manual mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop mode.				
TrackOP	This parameter value will be used as the output when the loop is in Track mode, unless its status is bad. In the case of bad status of this parameter, the Fallback OP will be used. In Track mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Output			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
InhibitOP	This parameter value will be used as the output when the loop is in Inhibit mode. In Inhibit mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop.				
OPRateUp	This constrains the rate at which the controller output can change in an increasing (upwards) direction. It is specified in percentage per second. Output Rate Limits can sometimes be useful in reducing rapid changes in output from damaging the process (for example, heater elements), however they can also significantly negatively affect the process performance. Generally, Setpoint Rate Limits are used to achieve the same purpose unless Output Rate Limits are deemed to be absolutely necessary.	0	Off		
OPRateDown	This constrains the rate at which the controller output can change a decreasing (downwards) direction. It is specified in percentage per second. Output Rate Limits can sometimes be useful in reducing rapid changes in output from damaging the process (for example, heater elements), however they can also significantly negatively affect the process performance. Generally, Setpoint Rate Limits are used to achieve the same purpose unless Output Rate Limits are deemed to be absolutely necessary.	0	Off		
OPRateDeactivate	When an Output Rate Limit has been configured, this input can be used as part of the strategy to temporarily deactivate rate limiting.	0	No		
		1	Yes		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Output			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PowerFFActivate	Power Feedforward is a feature which monitors the line voltage and compensates for fluctuations before they affect the process temperature. If a process were running at 25% power, the temperature close to set point and then the line voltage fell by 20% the heater power would drop by 36% because of the square law dependence of power on voltage. Sooner or later the temperature would fall. After a time, the thermocouple and controller would sense this fall and increase the on-time of the contactor just enough to bring the temperature back to setpoint. Meanwhile the material would be running a bit cooler than optimum which may cause some imperfection in the product. Power Feedforward reduces this effect by watching line voltage continuously and counteracting line voltage fluctuations by increasing or decreasing contactor duty cycle. This feature is only applicable to electrical heating processes where the heater is driven directly from the controller (not via a power controller). Deactivate it for other processes.	0	Off		
		1	On		
Ch2Deadband	The Ch1/Ch2 Deadband is a gap in percent between output 1 deactivating and output 2 activating and vice versa. For on/off control this is taken as a percentage of the hysteresis.	0	Off		
NonLinearCooling	A number of special non-linear cooling transformations can be applied to channel 2. These are used to compensate for the non-linear nature of cooling.	0	Off No non-linear cooling algorithm applied. Channel 2 will give a linear output.		
		1	Oil cooling Oil cooling exhibits a mass-transit non-linearity.		
		2	Water cooling This transformation compensates for both the mass-transit effect and also the strong non-linearity due to the latent heat of evaporation. When water cooling, the first few initial pulses tend to flash off into steam. This change in phase extracts much more energy from the process than merely heating the water.		
		3	Fan cooling Fan cooling also exhibits a mass-transit non-linearity.		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Output			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
ManualStepValue	If the Manual Transfer type has been configured as 'Step' then this value will be applied to the output on the transition from Auto to Manual. After the transition, whilst in Manual mode, the output can be altered using the ManualOP parameter. In Manual mode, the Setpoint Limits and Ranges no longer apply and the process may be driven over or under range, as the controller operates in open loop mode.				
Ch1TravelTime	The valve travel time in seconds for the channel 1 output. This parameter must be set if the Ch1 Control Type is set to VP. The valve travel time is the time taken for the valve to move from fully-closed to fully-open. This must be the measured time to move from endstop to endstop. It is not recommended to use what is specified in the valve data sheet as the endstop positions and the process can change this significantly.				
Ch2TravelTime	The valve travel time in seconds for the channel 2 output. This parameter must be set if the Ch2 Control Type is set to VP. The valve travel time is the time taken for the valve to move from fully-closed to fully-open. This must be the measured time to move from endstop to endstop. It is not recommended to use what is specified in the valve data sheet as the endstop positions and the process can change this significantly.				
RemoteOPHighLimit	Can be used to constrain the output of the loop from a remote source or calculation.				
RemoteOPLowLimit	Can be used to constrain the output of the loop from a remote source or calculation.				
RemoteOPLimsDeactivate	When asserted, Remote Output Limits are ignored.	0	No		
		1	Yes		

Diagnostics Parameters

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Diagnostics			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryLoopBad	Indicates that at least one has bad status between PrimaryPV, SecondaryRSP or SecondarySPTrim (if activated via SecondarySPTrimActivate).	0	Off		
		1	On		
LoopBad	Indicates that at least one of PV, DV, RemoteOPLowLimit, orRemoteOPHighLimit provided as input to the Loop has bad status.	0	Off		
		1	On		
PrimaryLoopBreakTime	The Primary loop break alarm attempts to detect loss of control in the Primary control loop by checking the Primary control output, the Primary process value and its rate of change. Loop break detection operates in all supported control algorithms. This is not to be confused with load failure and partial load failure.	0	Off		
PrimaryLoopBreakDeltaPV	If the Primary PID controller output is saturated, this is the smallest change in Primary PV that the system would expect to see in 2x Primary loop break times. If the Primary PID controller output is saturated and the Primary PV has not moved by this amount in 2x PrimaryLoopBreakTime then the Primary loop break alarm will be activated.				
PrimaryLoopBreak	Signals that a Primary loop break has been detected.	0	No		
		1	Yes		
LoopBreakTime	The loop break alarm attempts to detect loss of control in the control loop by checking the control output, the process value and its rate of change. Loop break detection operates in all supported control algorithms. For Cascade Loop Type, this relates to the Secondary PID controller. This is not to be confused with load failure and partial load failure.	0	Off		
LoopBreakDeltaPV	If the controller output is saturated, this is the smallest change in PV that the system would expect to see in 2x LoopBreakTime. If the controller output is saturated and the PV has not moved by this amount in 2x LoopBreakTime then the LoopBreak alarm will be activated.				
LoopBreak	Signals that a loop break has been detected. For Cascade Loop Type, this relates to the Secondary PID controller.	0	No		
		1	Yes		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Diagnostics			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimaryDeviation	This is the process deviation (sometimes called ‘error’) of the Primary PID controller. It is calculated as Primary PV minus Primary SP. Therefore, a positive deviation implies that the Primary PV is above Setpoint, while a negative deviation implies that the Primary PV is below Setpoint.				
PrimaryWorkingOutput	Primary PID controller output before the mapping performed through the Cascade scaling block.				
PrimaryProportionalOP	This is the output contribution from the proportional term of Primary controller.				
PrimaryIntegralOP	This is the output contribution from the proportional term of Primary controller.				
PrimaryDerivativeOP	This is the output contribution from the derivative term of Primary controller.				
Deviation	This is the process deviation (sometimes called ‘error’) of the controller. It is calculated as PV minus SP. Therefore, a positive deviation implies that the PV is above Setpoint, while a negative deviation implies that the PV is below Setpoint. For Cascade Loop Type, this relates to the Secondary PID controller.				
TargetOutput	The requested control output. This is the output taken before rate limiting.				
WrkOPHigh	This is the resolved upper output boundary that is currently in use. It is derived from the gain scheduled Output High Limit, the Remote High Limit and the global Output High Limit.				
WrkOPLow	This is the resolved lower output boundary that is currently in use. It is derived from the gain scheduled Output Low Limit, the Remote Low Limit and the global Output Low Limit.				
ProportionalOP	This is the output contribution from the proportional term. For Cascade Loop Type, this relates to the Secondary PID controller.				
IntegralOP	This is the output contribution from the integral term. For Cascade Loop Type, this relates to the Secondary PID controller.				
DerivativeOP	This is the output contribution from the derivative term. For Cascade Loop Type, this relates to the Secondary PID controller.				
LineVoltage	This is the line voltage measured by the instrument (in Volts). This is the value used for Power Feedforward if activated.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Diagnostics			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
PrimarySchedPB	The currently active proportional band of Primary PID controller.				
PrimarySchedTI	The currently active integral time of Primary PID controller.	0	Off		
PrimarySchedTD	The currently active derivative time of Primary PID controller.	0	Off		
PrimarySchedCBH	The currently active cutback high threshold of Primary PID controller.	0	Auto 3x the proportional band		
PrimarySchedCBL	The currently active cutback low threshold of Primary PID controller.	0	Auto 3x the proportional band		
PrimarySchedMR	The currently active manual reset value of Primary PID controller.	0	Off		
SchedCh1PB	The currently active channel 1 proportional band. For Cascade Loop Type, this relates to the Secondary PID controller.				
SchedCh2PB	The currently active channel 2 proportional band. For Cascade Loop Type, this relates to the Secondary PID controller.				
SchedTI	The currently active integral time. For Cascade Loop Type, this relates to the Secondary PID controller.	0	Off		
SchedTD	The currently active derivative time. For Cascade Loop Type, this relates to the Secondary PID controller.	0	Off		
SchedCBH	The currently active cutback high threshold. For Cascade Loop Type, this relates to the Secondary PID controller.	0	Auto 3x the proportional band		
SchedCBL	The currently active cutback low threshold. For Cascade Loop Type, this relates to the Secondary PID controller.	0	Auto 3x the proportional band		
SchedMR	Scheduled manual reset value.	0	Off		
PrimaryAtLimit	This flag will be asserted whenever the Primary PID controller output is saturated (has hit the upper or lower boundary).	0	No		
		1	Yes		
AtLimit	This flag will be asserted whenever the controller output is saturated (has reached one of Working Output High or Working Low Limits). This may be useful for a cascade strategy. For Cascade Loop Type, this relates to the Secondary PID controller.	0	No		
		1	Yes		
InInhibit	When asserted, indicates that Inhibit mode is active.	0	No		
		1	Yes		
InHold	When asserted, indicates that Hold mode is active.	0	No		
		1	Yes		
InTrack	When asserted, indicates that Track mode is active.	0	No		
		1	Yes		
InManual	When asserted, indicates that Manual is selected or ForcedManual mode is active.	0	No		
		1	Yes		

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Diagnostics			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
InTune	When asserted, indicates that Autotune is running. In Cascade loop type this refers to the Secondary PID autotune.	0	No		
		1	Yes		
InAuto	When asserted, indicates that Auto mode is selected or Forced Auto is active.	0	No		
		1	Yes		
InPrimaryTune	When asserted, indicates that Primary Autotune is running.	0	No		
		1	Yes		
InCascade	When asserted, indicates that Cascade Auto mode is selected.	0	No		
		1	Yes		
NotRemote	When true, this flag indicates that the controller is not ready to receive a remote setpoint. Typically this is wired back to the Track output value of an external Primary PID controller, such that the external Primary PID controller can track the SP of the controller when local setpoint is selected.	0	No		
		1	Yes		
PrimaryReady	When true, this flag indicates that the controller is able to run as a cascade primary. Typically this is wired to the RSPActivate input of a cascade secondary, such that the secondary can control to a local setpoint if the primary is taken out of Auto mode.	0	No		
		1	Yes		
AdditionalDiagnostics	When activated, additional parameters become available for commissioning purposes.	0	Off		
		1	On		
ActiveOvershootLimiting	This activates the Active Overshoot Limiting strategy for cascade control which uses the Active limit parameters to automatically limit the Secondary setpoint.	0	Active Overshoot Limiting for cascade control is deactivated.		
		1	Active Overshoot Limiting for cascade control is activated.		
ActiveLimitLow	Internal limit for the Secondary setpoint. This limit is calculated and applied by the internal Automatic Overshoot Limiting strategy for cascade control.				
ActiveLimitHigh	Internal limit for the Secondary setpoint. This limit is calculated and applied by the internal Automatic Overshoot Limiting strategy for cascade control.				
ActiveLimitOPDelta	Adjustment parameter for the active overshoot limiting strategy. Expressed in percent units. Increasing this parameter will make the active limits wider.				

Blocks – SuperLoop.1 to SuperLoop.24		Sub-block: Diagnostics			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
DiagnosticFlags	This parameter maps to several function block diagnostic flags. When equal to zero, it means that no condition was active since its last manual reset. It can be reset to zero by the operator when no condition is triggering any diagnostic flag. **Bit 0**: Not a Number (NaN) detected in the Secondary Loop section. If detected during automatic control, the block automatically transitions into Forced Manual mode. **Bit 1**: Not a Number (NaN) detected in the Primary Loop section. If detected during automatic cascade control, the block automatically transitions into Forced Auto mode. **Bit 2**: Not a Number (NaN) detected in the Setpoint Generator section. In case of Single Loop type and if detected during automatic control, the block automatically transitions into Forced Manual mode. In case of Cascade Loop type and if detected during automatic cascade control, the block automatically transitions into Forced Auto mode.				

Legacy Loop

For the Legacy loop, the Mini8 loop controller has up to 16 loops of control. Each Loop has two outputs, Channel 1 and Channel 2, each of which can be configured for PID or On/Off.

The control function block is divided into a number of sections the parameters of which are all listed under the Block 'Loop'.

The 'Loop' block contains sub-blocks for each section as shown diagrammatically below.

Loop Parameters – Main

Blocks – Loop.1 to Loop.16		**Sub-block: Main**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
AutoMan	To select Auto or Manual operation.	Auto	Automatic (closed loop) operation	Auto	Oper
		Man	Manual (output power adjusted by the user) operation		
PV	The process variable input value. This is typically wired from an analog input.	Range of the input source			Oper
Inhibit	Used to stop the loop controlling. If activated the loop will stop control and the output of the loop will be set to the 'safe' output value. On exit from Inhibit the transfer will be bumpless. This may be wired to an external source	No Yes	Inhibit de-activated Inhibit activated	No	Oper
TargetSP	The value of setpoint at which the control loop is aiming. It may come from a number of different sources, such as internal SP and remote SP.	Between setpoint limits			Oper
WorkingSP	The current value of the setpoint being used by the control loop. It may come from a number of different sources, such as internal SP and Remote SP. The working setpoint is always read-only as it is derived from other sources.	Between setpoint limits			Read Only
ActiveOut	The actual output of the loop before it is split into the channel 1 and channel 2 outputs.				Read Only
IntHold	Stops Integral action			No	Oper

Loop Set up

These parameters configure the type of control.

Block – Loop.1 to Loop.16		**Sub-block: Setup**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Ch1 ControlType	Selects the channel 1 control algorithm. You may select different algorithms for channels 1 and 2. In temperature control applications, Ch1 is usually the heating channel, Ch2 is the cooling channel.	Off OnOff PID	Channel turned off On/off control 3-term or PID control	PID	Conf
Ch2 ControlType	Control type for channel 2				

Block – Loop.1 to Loop.16		**Sub-block: Setup**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Control Action	Control Action	Rev	Reverse acting. The output increases when the PV is below SP. This is the recommended setting for heating control.	Rev	Conf
		Dir	Direct acting. The output increases when the PV is above SP. This is the recommended setting for cooling control		
PB Units	Proportional band units.	EngUnits	Engineering units e.g. C or F	Eng	Conf
		Percent	Per cent of loop span (range Hi - Range Lo)		
Derivative Type	Selects whether the derivative acts only on PV changes or on 'Control Deviation' (either PV or Setpoint changes).	PV	Only changes in PV cause changes to the derivative output.	PV	Conf
		Deviation	Changes to either PV or SP will cause a derivative output.		
The above two parameters appear if either Ch1 or Ch2 are configured for PID control					

On/Off Control

On/Off control simply turns heating power on when the PV is below setpoint and off when it is above setpoint. If cooling is used, cooling power is turned on when the PV is above setpoint and off when it is below. The outputs of such a controller will normally be connected to relays – hysteresis may be set as described in "Alarms" on page 117 to remove relay chatter or to provide a delay in the control output action.

Each of the two control channels can be configured for On-Off control.

This is a simple type of control often found in basic thermostats:

- Ch1Output switches to:
  - 100% when PV ≤ WorkingSP – Ch1OnOffHys
  - 0% when PV ≥ WorkingSP
- Ch2Output switches to:
  - 100% when PV ≥ WorkingSP + Ch2OnOffHys
  - 0% when PV ≤ WorkingSP

This form of control will lead to oscillation about setpoint but it is by far the easier to tune.

The hysteresis should be set according to the trade-off between oscillation amplitude and actuator switching frequency.

The two hysteresis values can be gain scheduled.

PID Control

Primary controller and Secondary controller are based on the Eurotherm PID control algorithm.

PID, also referred to as 'Three Term Control', is an algorithm which continuously adjusts the output, according to a set of rules, to compensate for changes in the process variable. It provides more stable control than On Off control but the parameters need to be set up to match the characteristics of the process under control.

The Eurotherm PID algorithm is based upon an ISA type algorithm in its positional (non-incremental) form. The ISA form is a gain-dependent parallel form where the proportional term (the proportional band) defines the gain of the overall controller. The ISA form is not to be confused with a gain-independent form where the three terms are completely independent.

PID Control

The PID output is the sum of the proportional, integral and derivative terms (P, I and D):

Output term	Depends on:	Tuning parameter
ProportionalOP	PV deviation from WorkingSP	Proportional band (Engineering units or percent)
IntegralOP	Duration of the PV deviation	Integral time (seconds)
DerivativeOP	Rate of change of PV (default) or PV deviation	Derivative time (seconds)

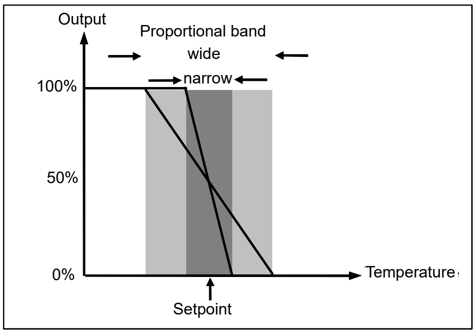
The PID tuning parameters can be:

- gain scheduled; activating one of the available GainScheduler strategies (Manual set, automatic set dependent on an internal or remote scheduling variable, etc.).
- autotuned; using the autotune algorithm.

Proportional Band

The proportional band, or gain, delivers an output which is proportional to the size of the deviation. It is the range over which the output power is continuously adjustable in a linear fashion from 0% to 100% (for a heat-only controller). Below the proportional band (PB) the output is full on (100%), above the proportional band the output is full off (0%) as shown in Figure 113.

The width of the proportional band determines the magnitude of the response to the deviation. If it too narrow (high gain) the system oscillates by being over responsive. If it is too wide (low gain) the control is sluggish. The ideal situation is when the proportional band is as narrow as possible without causing oscillation.



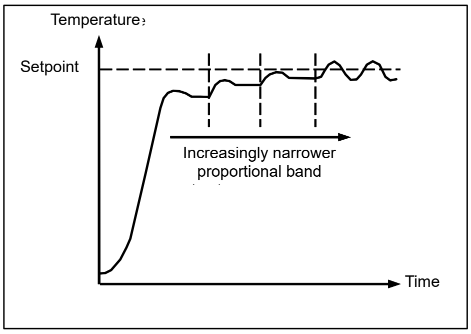


Figure 113 Proportional Action

Figure 113 also shows the effect of narrowing proportional band to the point of oscillation. A wide proportional band results in straight line control but with an appreciable initial deviation between setpoint and actual temperature. As the band is narrowed the temperature gets closer to setpoint until finally becoming unstable.

The proportional band may be set in engineering units or as a percentage of the controller range.

Integral Term

In a proportional-only controller, a deviation between setpoint and PV has to exist for the controller to deliver power. Integral is used to achieve zero steady state control.

The integral term slowly shifts the output level as a result of a deviation between setpoint and measured value. If the measured value is below setpoint, the integral action gradually increases the output in an attempt to correct the deviation. If it is above setpoint integral action gradually decreases the output or increases the cooling power to correct the deviation.

Figure 114 shows the result of introducing integral action.

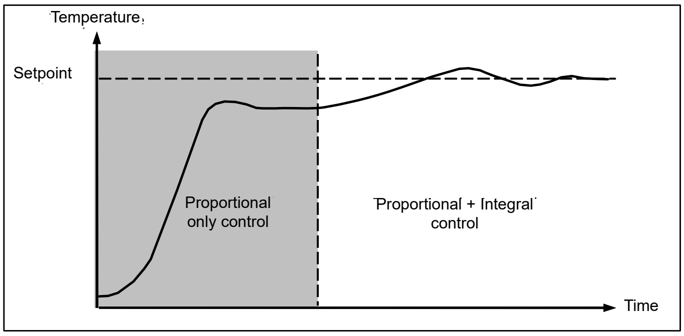


Figure 114 Proportional + Integral Control

The units for the integral term are measured in time (1 to 99999 seconds in Mini8 loop controllers). The longer the integral time constant, the more slowly the output is shifted and results in a sluggish response. Too small an integral time will cause the process to overshoot and even oscillate. The integral action may be deactivated by setting its value to Off.

Derivative Term

Derivative action, or rate, provides a sudden shift in output as a result of a rapid change in deviation, whether or not this is caused by PV alone (derivative on PV) or on SP changes as well (derivative on deviation selection). If the measured value falls quickly derivative provides a large change in output in an attempt to correct the perturbation before it goes too far. It is most beneficial in recovering from small perturbations.

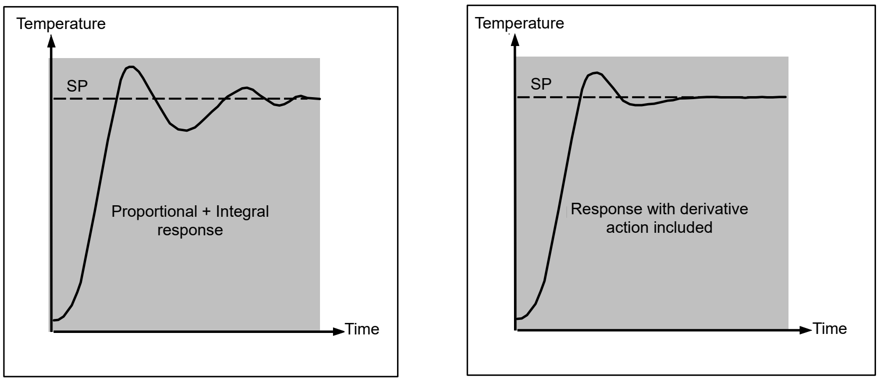


Figure 115 Proportional + Integral + Derivative Action

The derivative modifies the output to reduce the rate of change of deviationr. It reacts to changes in the PV by changing the output to remove the transient. Increasing the derivative time will reduce the settling time of the loop after a transient change.

Derivative is often mistakenly associated with overshoot inhibition rather than transient response. In fact, derivative should not be used to curb overshoot on start up since this will inevitably degrade the steady state performance of the system. Leave overshoot inhibition to the approach control parameters, High and Low Cutback, see "High and Low Cutback" on page 303.

Derivative is generally used to increase the stability of the loop, however, there are situations where derivative may be the cause of instability. For example, if the PV is noisy, then derivative can amplify that and cause excessive output changes, in these situations it is often better to deactivate the derivative and re-tune the loop.

If set to Off(0), no derivative action will be applied.

Derivative can be calculated on change of PV or change of deviation. If configured on deviation, then changes in the setpoint will be transmitted to the output. For applications such as furnace temperature control, it is common practice to select Derivative on PV to reduce thermal shock caused by a sudden change of output as a result of a change in setpoint.

High and Low Cutback

Cutback high ‘CBH’ and Cutback low ‘CBL’ are values that modify the amount of overshoot, or undershoot, that occurs during large step changes in PV (for example, under start-up conditions). They are independent of the PID terms which means that the PID terms can be set for optimal steady state response and the cutback parameters used to modify any overshoot which may be present.

Cutback involves moving the proportional band towards the cutback point nearest the measured value whenever the latter is outside the proportional band and the power is saturated (at 0 or 100% for a heat-only controller). The proportional band moves downscale to the lower cutback point and waits for the measured value to enter it. It then escorts the measured value with full PID control to the setpoint. In some cases it can cause a 'dip' in the measured value as it approaches setpoint as shown in Figure 116, but generally decreases the time to needed to bring the process into operation.

The action described above is reversed for falling temperature.

If cutback is set to Auto the cutback values are automatically configured to 3*PB.

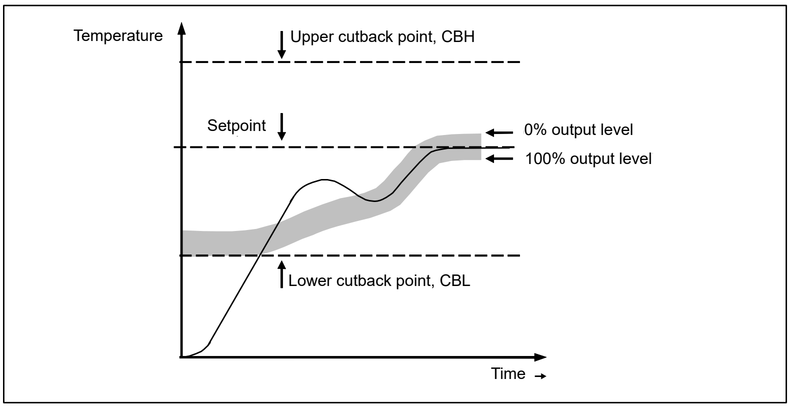


Figure 116 High and Low Cutback

Integral action and manual reset

In a full three-term controller (that is, a PID controller), the integral term automatically removes steady state deviations from the setpoint. If the controller is set as a PD controller, the integral term will be set to 'OFF'. Under these conditions the measured value may not settle precisely at setpoint. The Manual Reset parameter (MR) represents the value of the power output that will be delivered when the deviation is zero. Set this value manually to remove the steady state deviation.

Relative Cool Gain

The gain of channel 2 control output, relative to the channel 1 control output.

Relative Ch2 Gain compensates for the different quantities of energy needed to heat, as opposed to that needed to cool, a process. For example: water cooling applications might require a relative cool gain of 4 (cooling is four times faster than the heat-up process).

This parameter is set automatically when Autotune is used. A nominal setting of around 4 is often used.

Loop Break

The loop is considered to be broken if the PV does not respond to a change in the output in a given time. Since the time of response will vary from process to process the Loop Break Time (LBT – PID list) parameter allows a time to be set before a Loop Break Alarm (Lp Break - Diag list) is initiated.

The Loop Break Alarm attempts to detect a loss of restoring action in the control loop by checking the control output, the process value and its rate of change. This is not to be confused with 'Load Failure' and 'Partial Load Failure'. The loop break algorithm is purely software detection.

Occurrence of a loop break causes the Loop Break Alarm parameter to be set. It does not affect the control action unless it is wired (in software or hardware) to affect the control specifically.

It is assumed that, so long as the requested output power is within the output power limits of a control loop, the loop is operating in linear control and is therefore not in a loop break condition.

However, if the output becomes saturated then the loop is operating outside its linear control region.

Furthermore if the output remains saturated at the same output power for a significant duration, then this could indicate a break in the control loop. The source of this loop break is not important, but the resulting loss of control could be catastrophic.

Since the worst case time constant for a given load is usually known, a worst case time can be calculated over which the load should have responded with a minimum movement in temperature.

By performing this calculation the corresponding rate of approach towards setpoint can be used to determine if the loop can no longer control at the chosen setpoint. If the PV was drifting away from the setpoint or approaching the setpoint at a rate less than that calculated, the loop break condition would be met.

Loop Break and Autotune

If an autotune is performed the loop break time is automatically set to Ti*2 for a PI or PID loop or 12*Td for a PD loop.

For an On/Off controller, loop break detection is also based on loop break time with the PV threshold of 0.1*SPAN where SPAN = Range High – Range Low. Therefore, if the output is at limit and the PV has not moved by 0.1*SPAN in the loop break time a loop break will occur.

For all control configurations other than On/Off (i.e. where the Proportional Band is a valid parameter), if the output is in saturation and the PV has not moved by >0.5*Pb in the loop break time, a loop break condition is considered to have occurred.

If the loop break time is 0 (off) the loop break time is not set.

Cooling Algorithm

The method of cooling may vary from application to application.

For example, an extruder barrel may be cooled by forced air (from a fan), or by circulating water or oil around a jacket. The cooling effect will be different depending on the method. The cooling algorithm may be set to linear where the controller output changes linearly with the PID demand signal, or it may be set to water, oil or fan where the output changes non-linearly against the PID demand. The algorithm provides optimum performance for these methods of cooling.

Gain Scheduling

Gain scheduling is the automatic transfer of control between one set of PID values and another. It may be used in very non-linear systems where the control process exhibits large changes in response time or sensitivity, see diagram below. This may occur, for example, over a wide range of PV, or between heating and cooling where the rates of response may be significantly different. The number of sets depends on the non-linearity of the system. Each PID set is chosen to operate over a limited (approximately linear) range.

In the Mini8 loop controller, this is done at a preset strategy defined by the parameter 'Scheduler Type'. The choices are:

No.	Type	Description
0	Off	Just one fixed set of PID values
1	Set	The PID set can be selected manually or from a digital input
2	SP	The transfer between one set and the next depends on the value of the SP
3	PV	The transfer between one set and the next depends on the value of the PV
4	'Error'	The transfer between one set and the next depends on the value of the deviation ('control error')
5	OP	The transfer between one set and the next depends on the value of the OP demand
6	Rem Sched IP	The transfer between one set and the next depends on the value from a remote source for example, a digital input

The Mini8 loop controller has three sets of PID values for each loop – the maximum number, which you may wish to use, is set by 'Num Sets' parameter.

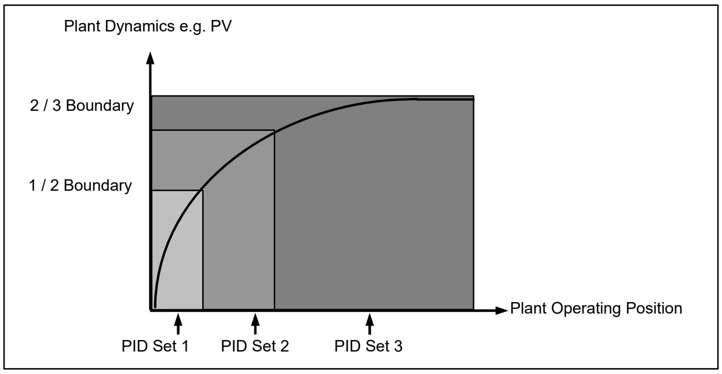


Figure 117 Gain Scheduling in a Non-Linear System

PID Parameters

Control loops must be specifically ordered – Order Code MINI8 – 4LP, 8LP or 16LP (4LPE, 8LPE, 16LPE, or 24LPE for Superloop). To activate a loop place one of the Loop function blocks on the graphical wiring page.

Block – Loop		**Sub-blocks: Loop1.PID to Loop16.PID**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
SchedulerType	To choose the type of gain scheduling	Off Set SP PV Error OP Rem	See above for explanation Parameters displayed will vary depending on type of scheduling selected.	Off	Oper
Num Sets	Selects the number of PID sets to present. Allows the lists to be reduced if the process does not require the full range of PID sets.	1 to 3		1	Oper
Scheduler RemoteInput	Scheduler Remote Input	1 to 3 (if SchedulerType is ‘Remote’)		1	Read Only
Active Set	Currently working set	Set1 Set2 Set3		Set1	Read Only except type ’Set’
Boundary 1-2	Sets the level at which PID set 1 changes to PID set 2	Range units		0	Oper
Boundary 2-3	Sets the level at which PID set 2 changes to PID set 3	Range units		0	Oper
ProportionalBand1, 2, 3	Proportional band Set1/Set2/Set3	0 to 99999 Eng units		300	Oper
IntegralTime 1, 2, 3	Integral term Set1/Set2/Set3			360s	Oper
DerivativeTime 1, 2, 3	Derivative term Set1/Set2/Set3			60s	Oper
RelCh2Gain 1, 2, 3	Relative cool gain Set1/Set2/Set3			1	Oper
CutbackHigh 1, 2, 3	Cutback high Set1/Set2/Set3			Auto	Oper
CutbackLow 1, 2, 3	Cutback low Set1/Set2/Set3			Auto	Oper
ManualReset 1, 2, 3	Manual reset Set1/Set2/Set3. This must be set to 0.0 when the integral term is set to a value			0.0	Oper
LoopBreakTime 1, 2, 3	Loop break time Set1/Set2/Set3			100	Oper
OutputHi 1, 2, 3	Output High Limit Set1/Set2/Set3			100	Oper
OutputLo 1, 2, 3	Output Low Limit Set1/Set2/Set3			-100	

Tuning

Tuning involves setting the following parameters.

Proportional Band ‘PB’, Integral Time ‘Ti’, Derivative Time ‘Td’, Cutback High ‘CBH’, Cutback Low ‘CBL’, and Relative Cool Gain ‘R2G’ (applicable to heat/cool systems only).

The controller is shipped with these parameters set to default values. In many cases the default values will give adequate straight line control, however, the response of the loop may not be ideal. Because the process characteristics are fixed by the design of the process it is necessary to adjust the control parameters in the controller to achieve optimal control. To determine the optimum values for any particular loop or process it is necessary to carry out a procedure called loop tuning. If significant changes are later made to the process which affect the way in which it responds it may be necessary to retune the loop.

Users have the choice of tuning the loop automatically or manually. Both procedures require the loop to oscillate and both are described in the following sections.

Loop Response

If we ignore the situation of loop oscillation, there are three categories of loop performance:

Under Damped In this situation the terms are set to reduce oscillation but do lead to an overshoot of the Process Value followed by decaying oscillation to finally settle at the Setpoint. This type of response can give a minimum time to Setpoint but overshoot may cause problems in certain situations and the loop may be sensitive to sudden changes in Process Value. This will result in further decaying oscillations before settling once again.

Critically Damped This represents an ideal situation where overshoot to small step changes does not occur and the process responds to changes in a controlled, non-oscillatory manner.

Over Damped In this situation the loop responds in a controlled but sluggish manner which will result in a loop performance which is non-ideal and unnecessarily slow.

The balancing of the P, I and D terms depends totally upon the nature of the process to be controlled.

In a plastics extruder, for example, a barrel zone will have a different response to a die, casting roll, drive loop, thickness control loop or pressure loop. To achieve optimal performance from an extrusion line, all loop tuning parameters must be set to their optimum values.

Gain scheduling is provided to allow specific PID settings to be applied at the different operating points of the process.

Initial Settings

In addition to the tuning parameters listed in "Tune Parameters" on page 311, there are a number of other parameters which can have an effect on the way in which the loop responds. Set these before either manual or automatic tuning is initiated. Parameters include, but are not limited to:

Setpoint Before starting a tune the loop conditions should be set as closely as practicable to the actual conditions which will be met in normal operation. For example, in a furnace or oven application a representative load should be included, an extruder should be running, and so on.

Heat/Cool Limits The minimum and maximum power delivered to the process may be limited by the parameters 'Output Lo' and 'Output Hi' both of which are found in the Loop OP list, see "Output Feature" on page 323. For a heat only controller the default values are 0 and 100%. For a heat/cool controller the defaults are -100 and 100%. Although it is expected that most processes will be designed to work between these limits there may be instances where it is desirable to limit the power delivered to the process. For example, if driving a 220V heater from a 240V source the heat limit may be set 80% so that the heater does not dissipate more than its maximum power.

Remote Output Limits ‘RemOPL’ and ‘RemOPHi’ (Loop OP List). If these parameters are used they should be set within the Heat/Cool Limits above.

Heat/Cool Deadband In controllers fitted with a second (cool) channel a parameter ‘Ch2 DeadBand’ is also available in the Loop OP block, see "Output Feature" on page 323, which sets the distance between the heat and cool proportional bands. The default value is 0% which means that heating will turn off at the same time as cooling turns on. The deadband may be set so that there is no possibility of the heat and cool channels being on together, particularly when cycling output stages are installed.

Minimum On Time If either or both of the output channels is fitted with a relay or logic output, the parameter ‘MinOnTime’ will appear in the relevant output block – "I/O" on page 89. This is the cycling time for a time proportioning output and should be set correctly before tuning is started.

Input Filter Time Constant The parameter ‘Filter Time Constant’ is found in the IO block "Thermocouple Input Parameters" on page 96.

Output Rate limit Output rate limit is active during tuning and may affect the tuning results. The parameter ‘Rate’ is found in the Loop OP block.

Other Considerations

- If a process includes adjacent interactive zones, each zone should be tuned independently.
- It is always better to start a tune when the PV and setpoint are far apart. This allows start up conditions to be measured and cutback values to be calculated more accurately.
- If the two loops are connected for cascade control, the inner loop may be tuned automatically but the outer should be tuned manually.
- In a programmer/controller tuning should only be attempted during dwell periods and not during ramp stages. If a programmer/controller is tuned automatically put the controller into Hold during each dwell period whilst autotune is active. It may be worth noting that tuning, carried out in dwell periods which are at different extremes of temperature may give different results owing to non-linearity of heating (or cooling). This may provide a convenient way to establish values for Gain Scheduling (see "Gain Scheduling" on page 306).

☺ Tip:

If an auto tune is initiated there are two further parameters which need to be set. These are ‘OutputHigh Limit’ and ‘OutputLow Limit’. These are found in the ‘Tune’ block, see also "Tune Parameters" on page 311.

Multi-zone applications

The tuning of one loop can be unduly influenced by the controlling effect of adjacent zone(s). Ideally the zone either side of the one being tuned should be turned OFF, or put in manual with the power level set to keep its temperature at about the usual operating level.

Automatic Tuning

Auto Tune automatically sets the following parameters:

Proportional Band '**PB**'	
Integral Time '**Ti**'	If '**Ti**' and/or '**Td**' is set to OFF, because you wish to use PI, PD or P only control, these terms will remain off after an autotune.
Derivative Time '**Td**'	
Cutback High '**CBH**'	If CBH and/or CBL is set to '**Auto'** these terms will remain at Auto after an autotune, i.e. 3*PB.
Cutback Low '**CBL**'	For autotune to set the cutback values, CBH and CBL must be set to a value (other than Auto) before autotune is started. Autotune will never return cutback values which are less than 1.6*PB.
Relative Cool Gain '**R2G**'	R2G is only calculated if the controller is configured as heat/cool. Following an autotune, '**R2G'** is always limited to between 0.1 and 10. If the calculated value is outside this limit a 'Tune Fail' alarm is given. In software releases up to and including 2.30, if the calculated value is outside this limit, R2G remains at its previous value but all other tuning parameters are changed.
Loop Break Time '**LBT**'	Following an autotune, '**LBT**' is set to 2*Ti (assuming the integral time is not set to OFF). If 'Ti' is set to OFF then 'LBT' is set to 12*Td.

Auto tune uses the 'one-shot' tuner which works by switching the output on and off to induce an oscillation in the process value. From the amplitude and period of the oscillation, it calculates the tuning parameter values. The autotune sequence for different conditions is described in "Autotune from Below SP – Heat/Cool" on page 313 to "Autotune at Setpoint – Heat/Cool" on page 315.

Tune Parameters

Block – Loop.Loop.1 to Loop.16		Sub-block: Tune			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
AutoTune Activate	To start self tuning	Off On	Stop Start	Stop	Oper
OutputHigh Limit	Set this to limit the maximum output power level which the controller will supply during the tuning process. If the high output power limit set in the output list is lower the autotune high limit will be clipped to this value.	Between Low Output and 100.0		100.0	Oper
OutputLow Limit	Set this to limit the minimum % output power level which the controller will supply during the tuning process. If the low output power limit set in the output list is higher the autotune low limit will be clipped to this value.	Between High Output and 0.0		0.0	Oper
State	Shows if self tuning is in progress	Off	Not running	Off	Read Only
		Ready			
		Running	In progress		
		Complete	Auto tune completed successfully		
		Timeout	Issue conditions, see "Unsuccessful Autotune Modes" on page 315.		
		TI_Limit			
		R2G_Limit			
Stage	Shows the progress of the self tuning	Reset		Reset	Read Only
		Settling	Displayed during the first minute		
		To SP	Heat (or cool) output on		
		Wait Min	Power output off		
		Wait Max	Power output on		
		Timeout	Issue conditions, see "Unsuccessful Autotune Modes" on page 315.		
		TI Limit			
		R2G Limit			
Stage Time	Time in the particular stage				Read Only

To Auto Tune a Loop - Initial Settings

Set parameters listed in "Initial Settings" on page 308.

‘Output High Limit’ and ‘Output Low Limit’ (‘OP’ List "Output Feature" on page 323) set the overall output limits. These limits apply at all times during tuning and during normal operation.

Set ‘OutputHigh Limit’ and ‘Output Low Limit’ (‘Tune’ list "Tune Parameters" on page 311). These parameters set the output power limits during Autotune.

☺ Tips:

The ‘tighter’ power limit will always apply. For example if ‘OutputHigh Limit’ (Tune List) is set to 80% and ‘Output High Limit’ (OP List) is set to 70% then the output power will be limited to 70%.

The measured value must oscillate to some degree for the tuner to be able to calculate values. The limits must be set to allow oscillation about the setpoint.

To Start Autotune

1. Select the loop to be tuned,
2. Set AutoTuneActivate to On.

A One-shot Tune can be performed at any time, but normally it is performed only once during the initial commissioning of the process. However, if the process under control subsequently becomes unstable (because its characteristics have changed), it may be necessary to tune again for the new conditions.

The autotune algorithm reacts in different ways depending on the initial conditions of the plant. The explanations given in this section are for the following conditions:

- Initial PV is below the setpoint and, therefore, approaches the setpoint from below for a heat/cool control loop.
- Initial PV is below the setpoint and, therefore, approaches the setpoint from below for a heat only control loop.
- Initial PV is at the same value as the setpoint. That is, within 0.3% of the range of the controller if ‘PB Units’ (Setup list) is set to ‘Percent’ or +1 engineering unit (1 in 1000) if the ‘PB Units’ is set to ‘Eng’. Range is defined as ‘Range Hi’ – ‘Range Lo’ for process inputs or the full temperature range defined for the relevant temperature input "Linearization Types and Ranges" on page 97.

☺ Tip:

If the PV is just outside the range stated above the autotune will attempt a tune from above or below SP.

Autotune and Sensor Break

When the controller is autotuning and sensor break occurs, the autotune will abort and the controller will output the sensor break output power ‘Sbrk OP’ set up in the OP List. Autotune must be re-started when the sensor break condition is no longer present.

Autotune and Inhibit

If the controller is in autotune when inhibit is asserted the tune goes to the OFF state (Stage = Reset). On inhibit being released the controller will re-start autotune.

Autotune and Gain Scheduling

When gain scheduling is activated and an autotune is performed, the calculated PID values will be written into the PID set that is active on completion of the tune. Therefore, the user may tune within the boundaries of a set and the values will be written into the appropriate PID set. When the schedule type is PV or OP and the boundaries between sets are close, the PID values may not be written to the correct set at the completion of the tune, since the range of the loop is not large. In this situation the scheduler (‘SchedulerType’) should be switched to ‘Set’ and the ‘Active Set’ chosen manually.

Autotune from Below SP – Heat/Cool

The point at which Automatic tuning is performed (Tune Control Point) is designed to operate just below the setpoint at which the process is normally expected to operate (Target Setpoint). This is so the process is not significantly overheated or overcooled. The Tune Control Point is calculated as follows:

Tune Control Point = Initial PV + 0.75(Target Setpoint – Initial PV).

The Initial PV is the PV measured at ‘B’ (after a 1 minute settling period)

Examples:

If Target Setpoint = 500°C and Initial PV = 20°C, then the Tune Control Point will be 380°C.

If Target Setpoint = 500°C and Initial PV = 400°C, then the Tune Control Point will be 475°C.

This is because the overshoot is likely to be less as the process temperature is already getting close to the target setpoint.

The sequence of operation for a tune from below setpoint for a heat/cool control loop is described below:

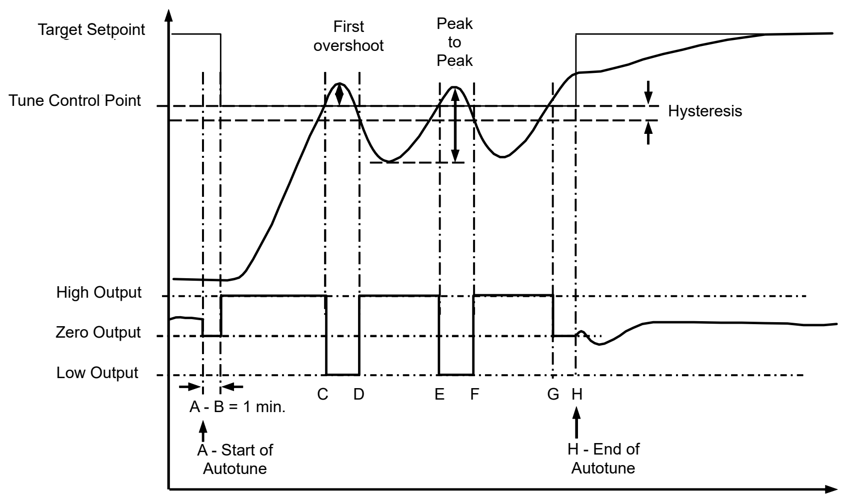


Figure 118 Autotune - Heat/Cool Process

Period	Action
A	Start of Autotune
A to B	Both heating and cooling power remains off for a period of 1 minute to allow the algorithm to establish steady state conditions.
B to D	First heat/cool cycle to establish first overshoot. ‘**CBL**’ is calculated on the basis of the size of this overshoot (assuming it is not set to Auto in the initial conditions).
B to F	Two cycles of oscillation are produced from which the peak to peak response and the true period of oscillation are measured. **PID** terms are calculated.
F to G	An extra heat stage is provided and all heating and cooling power is turned off at G allowing the plant to respond naturally. Measurements made during this period allow the relative cool gain ‘**R2G**’ to be calculated. ‘**CBH**’ is calculated from CBL*R2G.
H	Autotune is turned off at and the process is allowed to control at the target setpoint using the new control terms.

Autotune can also occur when the initial PV is above SP. The sequence is the same as tuning from below setpoint except that the sequence begins with full cooling applied at ‘B’ after the first one minute settling time.

Autotune From Below SP – Heat Only

The sequence of operation for a heat only loop is the same as that previously described for a heat/cool loop except that the sequence ends at ‘F’ since there is no need to calculate ‘R2G’.

At ‘F’ autotune is turned off and the process is allowed to control using the new control terms.

Relative cool gain, ‘R2G’, is set to 1.0 for heat-only processes.

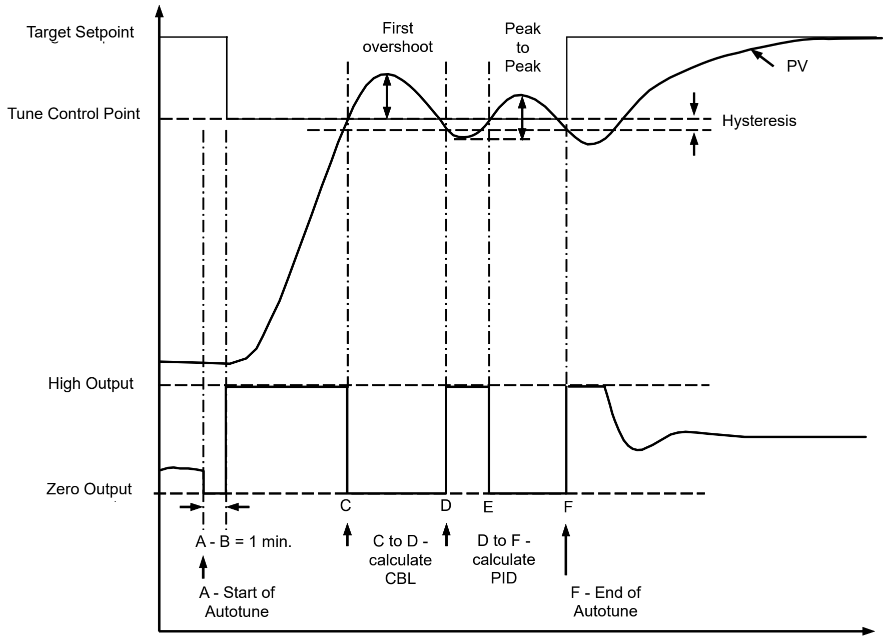


Figure 119 Autotune from below SP – Heat Only

For a tune from below setpoint ‘CBL’ is calculated on the basis of the size of the overshoot (assuming it was not set to Auto in the initial conditions). CBH is then set to the same value as CBL.

Note: As with the heat/cool case, Autotune can also occur when the initial PV is above SP. The sequence is the same as tuning from below setpoint except that the sequence starts with natural cooling applied at ‘B’ after the first one minute settling time.

In this case CBH is calculated – CBL is then set to the same value as CBH.

Autotune at Setpoint – Heat/Cool

It is sometimes necessary to tune at the actual setpoint being used. This is allowable in Mini8 loop controller and the sequence of operation is described below.

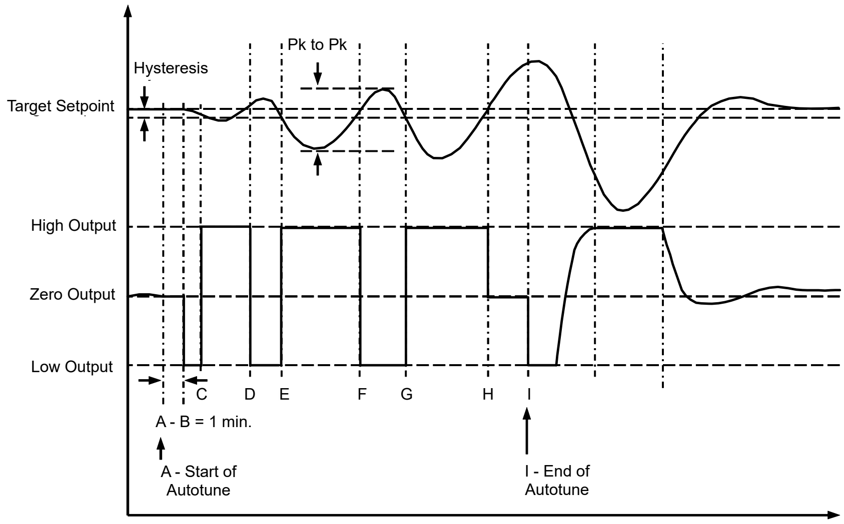


Figure 120 Autotune at Setpoint

Period	Action
A	Start of Autotune. A test is done at the **start of autotune** to establish the conditions for a tune at setpoint. The conditions are that the SP must remain within 0.3% of the range of the controller if '**PB Units**' (Setup list) is set to '**Percent**'. If '**PBUnits'** is set to '**Eng'** then the SP must remain within +1 engineering unit (1 in 1000). Range is defined as 'Range Hi' – 'Range Lo' for process inputs or the range defined in "Linearization Types and Ranges" on page 97 for temperature inputs.
A to B	The output is **frozen at the current value** for one minute and the conditions are continuously monitored during this period. If the conditions are met during this period autotune at setpoint is initiated at B. If at any time during this period the PV drifts outside the condition limits a tune at setpoint is abandoned. Tuning is then resumed as a tune from above or below setpoint depending on which way the PV has drifted. Since the loop is already at setpoint there is no need to calculate a Tune Control Setpoint – the loop is forced to oscillate around the Target Setpoint
C to G	Initiate oscillation - the process is forced to oscillate by switching the output between the output limits. From this the **period of oscillation** and the **peak to peak** response is measured. **PID** terms are calculated.
G to H	An extra heat stage is provided and all heating and cooling power is turned off at H allowing the plant to respond naturally. Measurements made during this period allow the relative cool gain '**R2G**' to be calculated.
I	Autotune is turned off and the process is allowed to control at the target setpoint using the new control terms.

For a tune at setpoint, autotune does not calculate cutback since there was no initial start up response to the application of heating or cooling. The exception is that the cutback values will never be returned less than 1.6*PB.

Unsuccessful Autotune Modes

The conditions for performing an autotune are monitored by the parameter 'State' (Tune block). If autotune is not successful, the following conditions are read by this parameter:

Timeout This will occur if any one stage is not completed within one hour. It could be due to the loop being open or not responding to the demands from the controller. Very heavily lagged systems may produce a timeout if the cooling rate is very slow.

TI Limit This will be displayed if Autotune calculates a value for the integral term greater than the maximum allowable integral setting i.e. 99999 seconds. This may indicate that the loop is not responding or that the tune is taking too long.

R2G Limit The calculated value of R2G is outside the range 0.1 and 10.0. In versions up to and including V2.3, R2G is set to 0.1 but all other PID parameters are updated.

R2G limit may occur if the gain difference between heating and cooling is too large. This could also occur if the controller is configured for heat/cool but the cooling medium is turned off or not working correctly. It could similarly occur if the cooling medium is on but heating is off or not working correctly.

Manual Tuning

If for any reason automatic tuning gives unsatisfactory results, you can tune the controller manually. There are a number of standard methods for manual tuning. The one described here is the Ziegler-Nichols method.

1. Adjust the setpoint to its normal running conditions (it is assumed this will be above the PV so that heat only is applied).
2. Set the Integral Time ‘Ti’ and the Derivative Time ‘Td’ to ‘OFF’.
3. Set High Cutback ‘CBH’ and Low Cutback ‘CBL’ to ‘Auto’.
4. Ignore the fact that the PV may not settle precisely at the setpoint.

If the PV is un-deviating, reduce the proportional band so that the PV just starts to oscillate. Allow enough time between each adjustment for the loop to stabilize. Record the proportional band value ‘PB’ and the period of oscillation ‘T’. If PV is already oscillating, measure the period of oscillation ‘T’, then increase the proportional band until it just stops oscillating. Record the value of the proportional band at this point.

Set the proportional band, integral time and derivative time parameter values according to the calculations given in the table below:

Type of control	Proportional band (PB)	Integral time (Ti) seconds	Derivative time (Td) seconds
Proportional only	2xPB	OFF	OFF
P + I control	2.2xPB	0.8xT	OFF
P + I + D control	1.7xPB	0.5xT	0.12xT

Manually Setting Relative Cool Gain

If the controller is fitted with a cool channel this should be activated before the PID values, calculated from the table above, are entered.

Observe the oscillation waveform and adjust R2G until a symmetrical waveform is observed.

Then enter the values from the table.

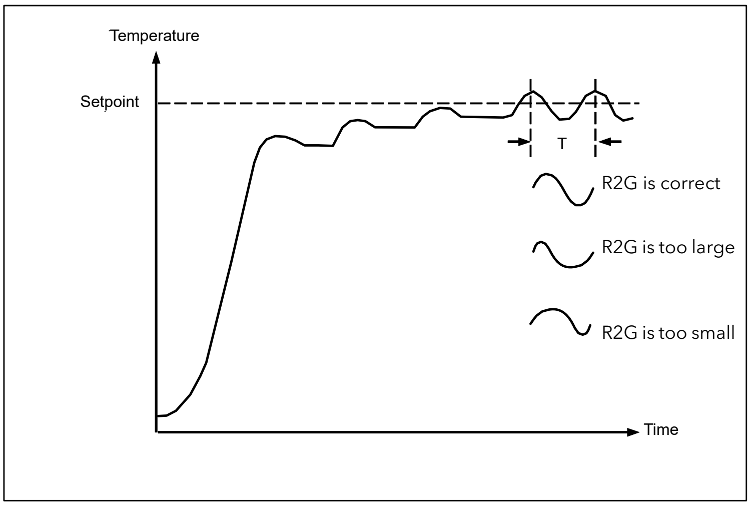


Figure 121 Setting Relative Cool Gain

Manually Setting the Cutback Values

Enter the PID terms calculated from the table in "Manual Tuning" on page 316 before setting cutback values.

The above procedure sets up the parameters for optimum steady state control. If unacceptable levels of overshoot or undershoot occur during start-up, or for large step changes in PV, then manually set the cutback parameters.

Proceed as follows:

1. Initially set the cutback values to one proportional bandwidth converted into display units. This can be calculated by taking the value in percentage that has been installed into the parameter ‘PB’ and entering it into the following formula:

   PB/100 * Span of controller = Cutback High and Cutback Low

   For example, if PB = 10% and the span of the controller is 0 -1200°C, then

   Cutback High and Low = 10/100 * 1200 = 120

2. If overshoot is observed following the correct settings of the PID terms increase the value of ‘CBL’ by the value of the overshoot in display units. If undershoot is observed increase the value of the parameter ‘CBH’ by the value of the undershoot in display units.

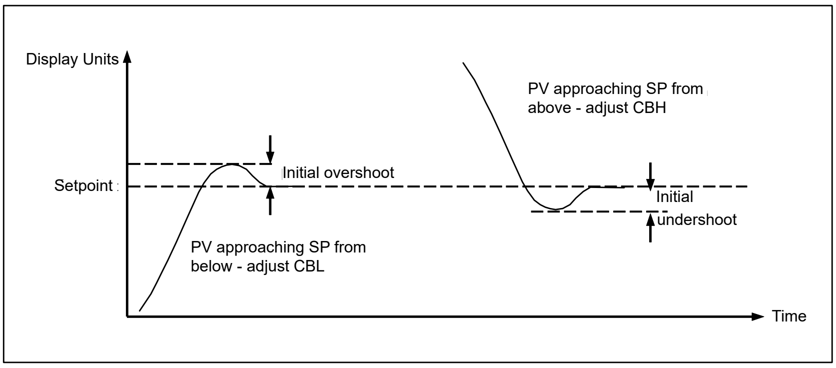


Figure 122 Manual Setting of Cutback

Setpoint Feature

For each of the 16 loops, the controller setpoint is the Working Setpoint that may come from a number of alternative sources. This is the value ultimately used to control the process variable in each loop.

The working setpoint may be derived from:

- SP1 or SP2, both of which are individually set, can be selected by an external signal or via the SPSelect parameter over communications.
- From an external (remote) analog source
- The output of a programmer function block and will, therefore, vary in accordance with the program in use.

The setpoint function block also provides the facility to limit the rate of change of the setpoint before it is applied to the control algorithm. It will also provide upper and lower limits. These are defined as setpoint limits for the local setpoints and instrument range high and low for other setpoint sources. All setpoints are ultimately subject to a limit of range hi and range lo.

User configurable methods for tracking are available, such that the transfer between setpoints and between operational modes will not cause a bump in the setpoint.

Setpoint Feature

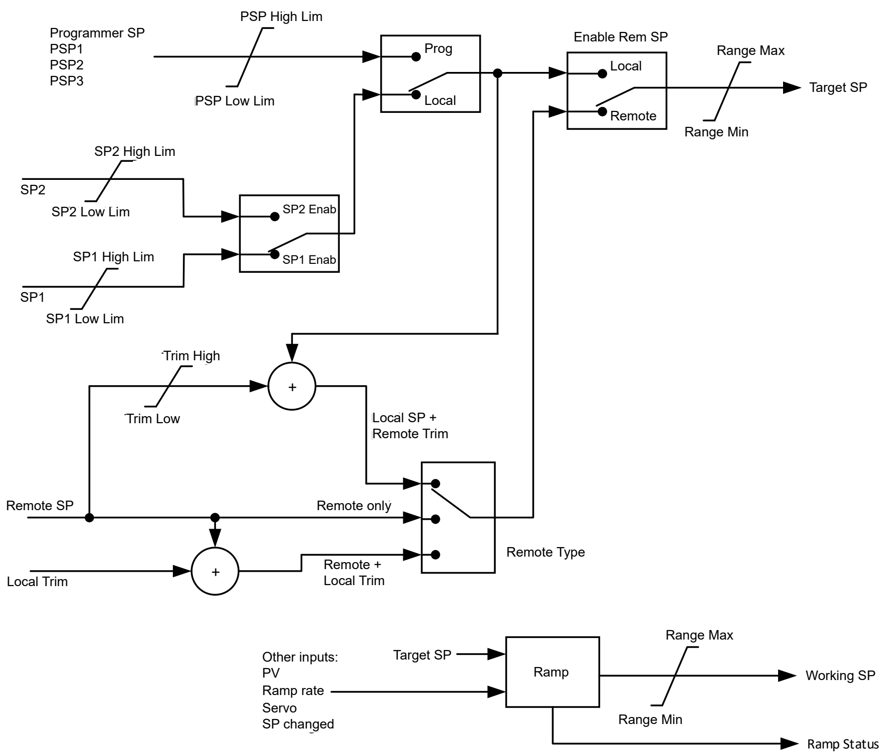


Figure 123 Setpoint Function Block

SP Tracking

When setpoint tracking is activated and the local setpoint is selected, the local setpoint is copied to ‘TrackSP’. Tracking now dictates that the alternate SP follows or tracks this value. When the alternate setpoint is selected it initially takes on the tracked value so that no bump takes place. The new setpoint is then adopted gradually. A similar action takes place when returning to the local setpoint.

Manual Tracking

When the controller is operating in manual mode the currently selected SP tracks the PV. When the controller resumes automatic control there will be no step change in the resolved SP.

Rate Limit

Rate limit will control the rate of change of setpoint. It is activated by the ‘Rate’ parameter. If this is set to Off then any change made to the setpoint will be effective immediately. If it is set to a value then any change in the setpoint will be effected at the value set in units per minute. Rate limit also acts on SP2 and when switching between SP1 and SP2.

When rate limit is active the ‘RateDone’ parameter will display ‘No’. When the setpoint has been reached this parameter will change to ‘Yes’.

When ‘Rate’ is set to a value (other than Off) an additional parameter ‘Rate Deactivate’ is displayed which allows the setpoint rate limit to be turned off and on without the need to adjust the ‘Rate’ parameter between Off and a value.

Setpoint Parameters

Block – Loop.1 to Loop.16		Sub-Block: SP			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Range High	The Range limits provide a set of absolute maximums and minimums for setpoints within the control loop. Any derived setpoints are ultimately clipped to be within the Range limits. If the Proportional Band is configured as % of Span, the span is derived from the Range limits.	Full range of the input type			Conf
Range Low					Conf
SP Select	Select local or alternate setpoint	SP1 SP2	Setpoint 1 Setpoint 2	SP1	Oper
SP1	Primary setpoint for the controller	Between SP high and SP low limits			Oper
SP2	Setpoint 2 is the secondary setpoint of the controller. It is often used as a standby setpoint.				Oper
SP HighLimit	Maximum limit allowed for the local setpoints	Between Range Hi and Range Lo			Oper
SP LowLimit	Minimum limit allowed for the local setpoints				Oper
Alt SP Select	To allow the alternative setpoint to be used. This may be wired to a source such as the programmer Run input.	No Yes	Alternative setpoint disallowed Alternative setpoint allowed		Oper
Alt SP	This may be wired to an alternative source such as the programmer or remote setpoint				Oper
Rate	Limits the maximum rate at which the working setpoint can change. The rate limit may be used to protect the load from thermal shock which may be caused by large step changes in setpoint.	Off or 0.1 to 9999.9 engineering units per minute		Off	Oper
RateDone	Flag which indicates when the setpoint is changing or completed	No Yes	Setpoint changing Complete		Read Only
Rate Deactivate	Setpoint rate deactivate	No Yes	Activated Deactivated		Oper
ServoToPV	Servo to PV Activate When Rate is set to any value other than Off and Servo to PV is d, changing the active SP will cause the working SP to servo to the current PV before ramping to the new target SP.	No Yes	Deactivated Activated	No	Conf Read Only in L3
SP Trim	Trim is an offset added to the setpoint. The trim may be either positive or negative, the range of the trim may be restricted by the trim limits. Setpoint trims may be used in a retransmission system. A primary zone may retransmit the setpoint to the other zones, a local trim may be applied to each zone to produce a profile along the length of the machine.	Between SP Trim Hi and SP Trim Lo			Oper
SPTrim HighLimit	Setpoint trim high limit				Oper
SPTrim LowLimit	Setpoint trim low limit				Oper
ManualTrack	To activate manual tracking. When the loop is switched from Manual to Auto, the Setpoint is set to the current PV. This is useful if the load is started in Manual Mode, then later switched to Auto to maintain the operating point.	Off On	Manual tracking deactivated Manual tracking activated		Read Only

Block – Loop.1 to Loop.16		Sub-Block: SP			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
SP Track	Setpoint tracking facilitates bumpless transfer in setpoint when switching between a local and an alternate setpoint such as the programmer. This s the tracking interface provided by TrackPV and TrackVal, which is used by the programmer and other setpoint providers external to the control loop.	Off On	Setpoint tracking deactivated Setpoint tracking activated		Conf
Track PV	The programmer tracks the PV when it is servoing or tracking.				Read Only
Track SP	Manual Tracking Value. The SP to track for manual tracking.				Read Only
SPIntBal	SP Integral Balance This is also known as debump in some instances. It forces the integral to be balanced upon changes in target setpoint.	Off On		Off	L3 Read Only Alterable in Conf

Setpoint Limits

The setpoint generator provides limits for each of the setpoint sources as well as an overall set of limits for the loop. These are summarized in the diagram below.

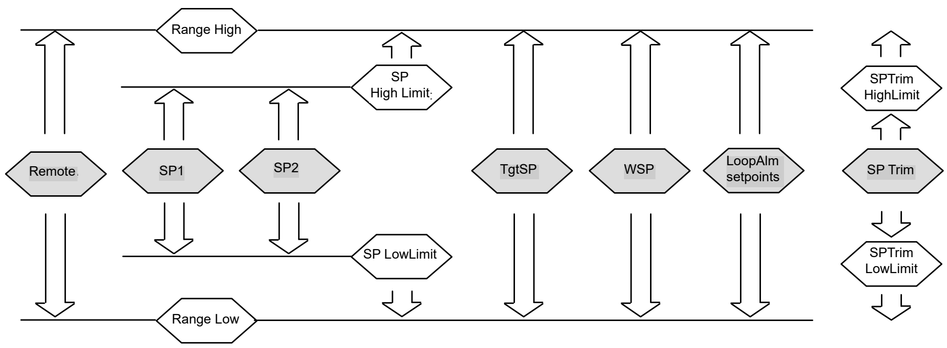


Figure 124 Setpoint Limits

☺ Tip:

'Range High' and 'Range Low' provide the range information for the control loop. They are used in control calculations to generate proportional bands. Span = Range High – Range Low.

Setpoint Rate Limit

Allows the rate of change of setpoint to be controlled. This removes step changes in the setpoint. It is a simple symmetrical rate limiter and is applied to the working setpoint which includes setpoint trim. It is determined by the 'Rate' parameter. If this is set to Off then any change made to the setpoint will be effective immediately. If it is set to a value then any change in the setpoint will be effected at the value set in units per minute. Rate limit applies to SP1, SP2 and Remote SP.

When rate limit is active the 'RateDone' flag will display 'No'. When the setpoint has been reached this parameter will change to 'Yes'. This flag will be cleared if the target setpoint subsequently changes.

When ‘Rate’ is set to a value (other than Off) an additional parameter ‘Rate Deactivate’ is displayed which allows the setpoint rate limit to be turned off and on without the need to adjust the ‘Rate’ parameter between Off and a value.

If the PV is in sensor break, the rate limit is suspended and the working setpoint takes the value of 0. On sensor break being released the working setpoint goes from 0 to the selected setpoint value at the rate limit.

Setpoint Tracking

The setpoint used by the controller may be derived from a number of sources. For example:

- Local setpoints SP1 and SP2. These may be selected using the parameter ‘SP Select’ in the SP block, through digital communications or by configuring a digital input which selects either SP1 or SP2. This might be used, for example, to switch between normal running conditions and standby conditions. If Rate Limit is switched off the new setpoint value is adopted immediately when the switch is changed.
- A programmer generating a setpoint which varies over time, see "Setpoint Programmer" on page 384. When the programmer is running the ‘Track SP’ and ‘Track PV’ parameters update continuously so that the programmer can perform its own servo (see also "Servo" on page 393). This is sometimes referred to as ‘Program Tracking’.
- From a Remote analog source. The source could be an external analog input into an analog input module wired to the ‘Alt SP’ parameter or a User Value wired to the ‘Alt SP’ parameter. The remote setpoint is used when the parameter ‘Alt SP Select’ is set to ‘Yes’.

Setpoint tracking (sometimes referred to as Remote Tracking) helps ensure that the Local setpoint adopts the Remote setpoint value when switching from Local to Remote to maintain bumpless transfer from Remote to Local. Bumpless transfer does not take place when changing from Local to Remote.

Note: If Rate Limit is applied the setpoint will change at the rate set when changing from Local to Remote.

Manual Tracking

When the controller is operating in manual mode the currently selected SP (SP1 or SP2) tracks the PV. When the controller resumes automatic control there will be no step change in the resolved SP. Manual tracking does not apply to the remote setpoint or programmer setpoint.

Output Feature

The output function block allows you to set up output conditions from the control block, such as output limits, hysteresis, output feedforward, behavior in sensor break, and so on.

Block – Loop.1 to Loop.16		**Sub-block: OP**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Output High Limit	Maximum output power delivered by channels 1 and 2. By reducing the high power limit, it is possible to reduce the rate of change of the process, take care however, as reducing the power limit will reduce the controllers ability to react to disturbance.	Between Output Lo and 100.0%		100.0	Oper
Output Low Limit	Minimum (or maximum negative) output power delivered by channels 1 and 2.	Between Output Hi and -100.0%		-100.0	
Ch1 Out	Channel 1 (Heat) output. The Ch1 output is the positive power values (0 to Output Hi) used by the heat output. Typically this is wired to the control output (time proportioning or DC output).	Between Output Hi and Output Lo			R/O
Ch2 Out	The Ch2 output is negative portion of the control output (0 – Output Lo) for heat/cool applications. It is inverted to be a positive number so that it can be wired into one of the outputs (time proportioning or DC outputs).	Between Output Hi and Output Lo			R/O
Ch2 DeadBand	Ch1/Ch2 Deadband is a gap in percent between output 1 going off and output 2 coming on and vice versa. For on/off control this is taken as a percentage of the hysteresis.	Off to 100.0%		Off	Oper
Rate	Limits the rate at which the output from the PID can change in % change per minute. Output rate limit is useful in stopping rapid changes in output from damaging the process or the heater elements.	Off to 9999.9 percent per minute		Off	Oper
Rate Deactivate	Output rate deactivate	No Yes	Activated Deactivated		Oper
Ch1 OnOff Hysteresis	Channel hysteresis only shown when channel 1 is configured as OnOff. Hysteresis sets the difference between output on and output off to reduce (relay) chatter.	0.0 to 200.0		10.0	Oper
Ch2 OnOff Hysteresis		0.0 to 200.0		10.0	Oper
SensorBreak Mode	Defines the action taken if the Process Variable is bad, i.e. a sensor break has occurred. This can be configured as hold, in which case the output of the loop is held at its last good value. Alternatively the output can switch to a 'safe' output power defined at configuration.	Safe Hold	To select the level set by 'Safe OP' To hold the current output level at the point when sensor break occurs	Safe	Oper
Safe OP Val	Sets the output level to be adopted when loop is inhibited.	Between Output Hi and Output Lo		0	Oper
SbrkOp	Sets the output level to be adopted when in sensor break condition.	Between Output Hi and Output Lo		0	Oper

Block – Loop.1 to Loop.16		**Sub-block: OP**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Manual Mode	Selects the mode of manual operation.	Track	In auto the manual output tracks the control output such that a change to manual mode will not result in a bump in the output.		Oper
		Step	On transition to manual the output will be the manual op value as last set by the operator.		
ManualOutVal	The output when the loop is in manual. Note: In manual mode the controller will still limit the maximum power to the power limits, however, it is recommended that the instrument is not left unattended at a high power setting. It is important that the over range alarms are configured to protect your process. *We recommend that all processes are fitted with an independent over range “policeman”.*	Between Output Hi and Output Lo			Read Only
ForcedOP	Forced manual output value. When ‘Man Mode’ = ‘Step’ the manual output does not track and on transition to manual the target output will step from its current value to the ‘ForcedOP’ value.	-100.0 to 100.0		0.0	Oper
Cool Type	Selects the type of cooling channel characterization to be used. Can be configured as water, oil or fan cooling.	Linear Oil Water Fan	These are set to match the type of cooling medium applicable to the process		Conf
FeedForward Type	Feedforward type The following four parameters appear if FF Type ≠ None	None	No signal fed forward	None	Conf
		Remote	A remote signal fed forward		
		SP	Setpoint fed forward		
		PV	PV fed forward		
FeedForward Gain	Defines the gain of the feedforward value, the feed forward value is multiplied by the gain.				Conf
FeedForward Offset	Defines the offset of the feedforward value this is added to the scaled feedforward.				Oper
FeedForward Trim Limit	Feedforward trim limits the effect of the PID output. Defines symmetrical limits around the PID output, such that this value is applied to the feedforward signal as a trim.				Oper
FF_Rem	Remote Feedforward signal. Allows an another signal to be used as Feedforward.	This is not affected by FeedForward Gain or Offset			Read Only
FeedForward Val	The calculated Feedforward Value.				Read Only
TrackOutVal	Value for the loop output to track when OP Track is Activated.				
Track Activate	When activated, the output of the loop will follow the track output value. The loop will bumplessly return to control when tracking is turned off.	Off On	Deactivated Activated		Oper
RemOPL	Remote output low limit. Can be used to limit the output of the loop from a remote source or calculation. This must always be within the main limits.	-100.0 to 100.0			Oper
RemOPH	Remote output high limit	-100.0 to 100.0			Oper

Output Limits

The diagram shows where output limits are applied.

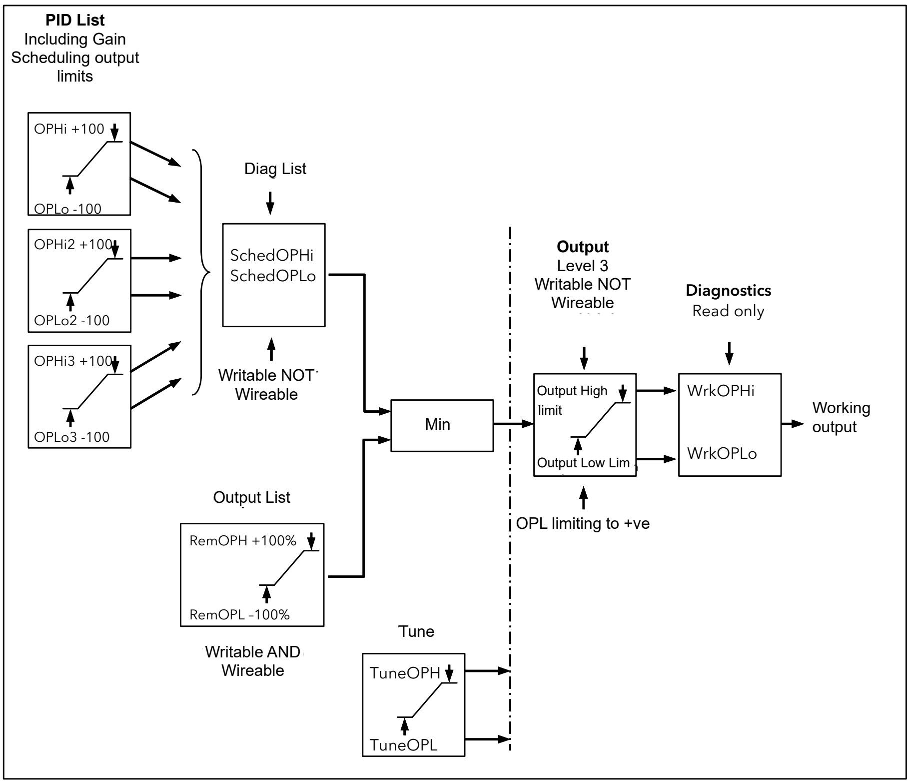


Figure 125 Output Limits

- Individual output limits may be set in the PID list for each set of PID parameters when gain scheduling is used.
- The parameters 'SchedOPHi' and 'SchedOPHLo', found the Diagnostics block, may be set to values which override the gain scheduling output values.
- A limit may also be applied from an external source. These are 'RemOPH' and 'RemOPLo' (Remote output high and low) found in the Output block. These parameters are wireable. For example they may be wired to an analog input module so that a limit may applied through some external strategy. If these parameters are not wired +100% limit is applied every time the instrument is powered up.
- The tightest set (between Remote and PID) is connected to the output where an overall limit is applied using parameters 'Output High Limit' and 'Output Low Limit' settable in Oper Level.
- 'Wrk OPHi' and 'Wrk OPHLo' found in the Diagnostics block are read-only parameters showing the overall working output limits.

The tune limits are a separate part of the algorithm and are applied to the output during the tuning process. The overall limits 'Output High Limit' and 'Output Low Limit' always have priority.

Output Rate Limit

The output rate limiter is a simple rate of change limiter which will stop the control algorithm demanding step changes in output power. It may be set in percent per minute.

The rate limit is performed by determining the direction in which the output is changing, and then incrementing or decrementing the Working Output (‘ActiveOut’ in the Main block) until ‘ActiveOut’ = the required output.

The amount by which to increment or decrement will be calculated based on the sampling rate of the algorithm (i.e. 110ms) and the rate limit that has been set. If the change in output is less than the rate limit increment the change will take effect immediately.

The rate limit direction and increment will be calculated on every execution of the rate limit. Therefore, if the rate limit is changed during execution, the new rate of change will take immediate effect. If the output is changed whilst rate limiting is taking place, the new value will take immediate effect on the direction of the rate limit and in determining whether the rate limit has completed.

The rate limiter is self-correcting such that if the increment is small and is lost in the floating point resolution, the increment will be accumulated until it takes effect.

The output rate limit will remain active even if the loop is in manual mode.

Sensor Break Mode

Sensor break is detected by the measurement system and a flag is passed to the control block which indicates sensor break. On the loop being informed that a sensor break has occurred it may be configured using ‘SensorBreak Mode’ to respond in one of two ways. The output may go to a pre-set level or remain at its current value.

The pre-set value is defined by the parameter ‘SbrkOP’. If rate limit is not configured the output will step to this value otherwise it will ramp to this value at the rate limit.

If configured as ‘Hold’ the output of the loop will stay at its last good value. If Output Rate Limit (Rate) has been configured a small step may be seen as the working output will limit to the 2 second old value.

On exit from sensor break the transfer is bumpless – the power output will ramp from its pre-set value to the control value.

Forced Output

This feature allows the user to specify what the output of the loop should do when moving from automatic control to manual control. The default is that the output power will be maintained and is then editable by the user. If forced manual is activated, two modes of operation can be configured. The forced manual step setting means the user can set a manual output power value and on transition to manual the output will be forced to that value. If ‘Track Activate’ is activated the output steps to the forced manual output and then subsequent edits to the output power are tracked back into the manual output value.

The parameters associated with this feature are ‘ForcedOP’ and ‘ManualMode’ = ‘Step’.

Feedforward

A limitation of a PID control strategy is that it responds only to deviations between PV and SP. By the time a PID controller first starts to react to a process disturbance, it might be already too late and the disturbance is in progress; all that can be done is to try to minimize the extent of the disruption as much as possible. Feedforward control is often used to overcome this limitation. It uses a measurement of the disturbance variable itself and a priori knowledge of the process to predict the controller output that will exactly counter the disturbance before it has a chance to affect the PV.

Feedforward is a value, which is scaled and added to the PID output, before any limiting. It can be used for the implementation of cascade loops or constant head control. Feedforward is implemented such that the PID output is limited to trim limits and acts as a trim on a FeedForward Value. The FeedForward Val is derived either from the PV or setpoint by scaling the PV or SP by the 'FeedForward Gain' and 'FeedForward Offset'. Alternatively, a remote value may be used for the FeedForward Val, this is not subject to any scaling. The resultant FeedForward Val is added to the limited PID OP and becomes the PID output as far as the output algorithm is concerned. The feedback value then generated must then have the FF contribution removed before being used again by the PID algorithm. The diagram below shows how feedforward is implemented.

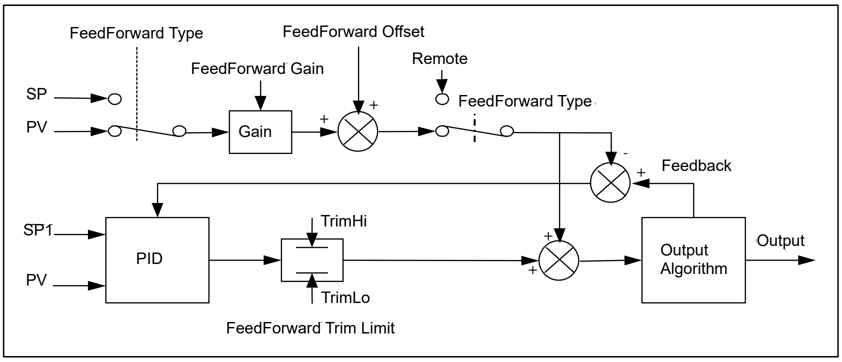


Figure 126 Implementation of Feedforward

Feedforward on its own also has a major limitation. It is an open-loop strategy that relies entirely on the a priori knowledge of the process. Feedforward tuning deviation, uncertainty and process variation all help to prevent zero tracking deviation being achieved in practice.

Further, the feedforward controller can only respond to disturbances that are explicitly measured and whose effect is known.

To counter the relative disadvantages, the SuperLoop combines both types of control in an arrangement known as "Feedforward with Feedback Trim". The Feedforward controller gives the principal control output and the PID controller can trim this output appropriately to give zero tracking deviation.

Remote DV is used as feedforward input when the effect of a disturbance on the plant is known and therefore the feedforward static and dynamic parameters can be tuned to generate an output demand signal that compensate the disturbance effect. The static feedforward parameters FFGain and FFOffset can be found by characterizing the steady-state effect of the disturbance of the output demand via ΔOPss = FFGain * DV + FFOffset, where the ΔOPss is the deviation of steady-state output demand due to DV.

Secondary or Primary working setpoint is used as feedforward input when the output demand for a certain target setpoint is known and therefore the feedforward static parameters can be tuned to generate an output demand equal to the steady-state value. The static feedforward parameters FFGain and FFOffset can be tuned by characterizing the steady-state characteristic of the plant via OPss = FFGain * SP + FFOffset, where OPss is the output demand when PV is stable at the setpoint SP.

In the two above cases, dynamic feedforward parameters (Lead-lag compensator time constants sFFLeadTime and sFFLagTime) can be tuned to further accelerate the response by adding an initial excess transient output. Finally, PID can trim the feedforward output to completly minimize the tracking deviation.

Secondary or Primary process variable can be used as feedforward input to implement a lead-lag compensator to improve the frequency response of the control system.

Effect of Control Action, Hysteresis and Deadband

For temperature control ‘Loop.1.Control Action’ will be set to ‘Reverse’. For a PID controller this means that the heater power decreases as the PV increases. For an on/off controller output 1 (usually heat) will be on (100%) when PV is below the setpoint and output 2 (usually cool) will be on when PV is above the setpoint.

Hysteresis applies to on/off control only. It defines the difference in temperature between the output switching off and switching back on again. The examples below show the effect in a heat/cool controller.

Deadband (Ch2 DeadB) can operate on both on/off control or PID control where it has the effect of widening the period when no heating or cooling is applied. However, in PID control its effect is modified by both the integral and derivative terms. Deadband might be used in PID control, for example, where actuators take time to complete their cycle, to stop heating and cooling being applied at the same time. Deadband is likely to be used, therefore, in on/off control only. The second example below adds a deadband of 20 to the above example.

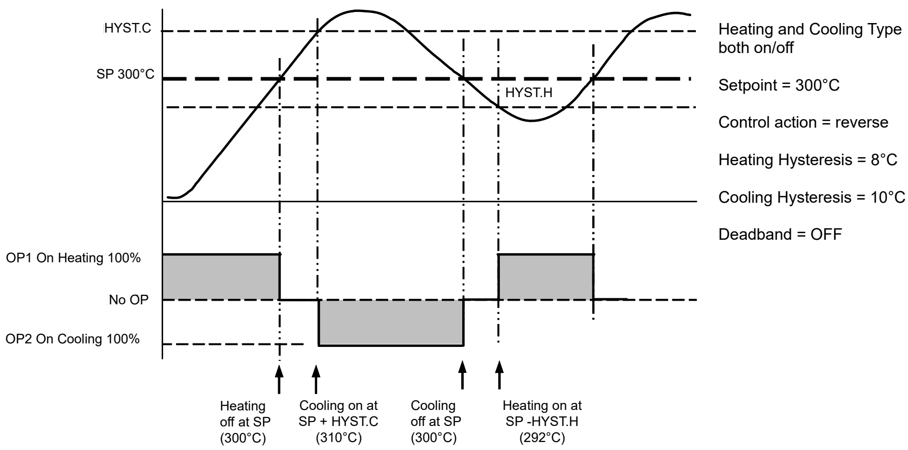


Figure 127 Deadband OFF

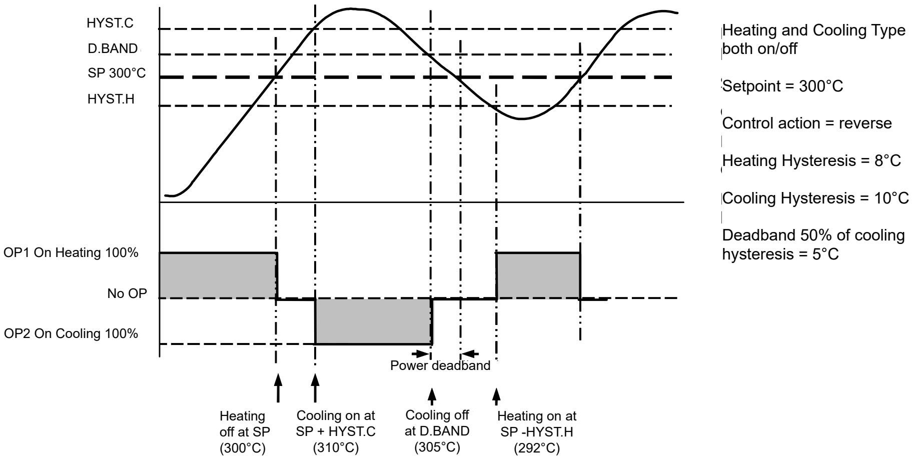
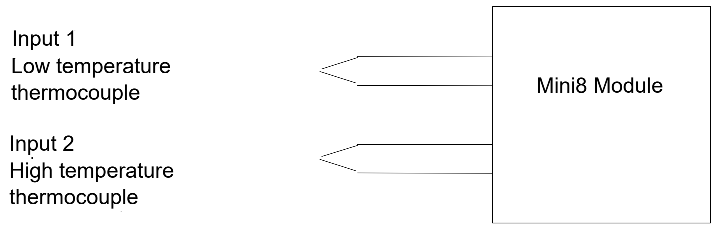


Figure 128 Deadband ON set at 50% of Cooling. Hysteresis = 5°C

Switch Over

This facility is commonly used in temperature applications which operate over a wide temperature range. A thermocouple may be used to control at lower temperatures and a pyrometer then controls at very high temperatures. Alternatively two thermocouples of different types may be used.

The diagram below shows a process heating over time with boundaries which define the switching points between the two devices. The higher boundary (2 to 3) is normally set towards the top end of the thermocouple range and this is determined by the 'Switch High' parameter. The lower boundary (1 to 2) is set towards the lower end of the pyrometer (or second thermocouple) range using the parameter 'Switch Low'. The controller calculates a smooth transition between the two devices.



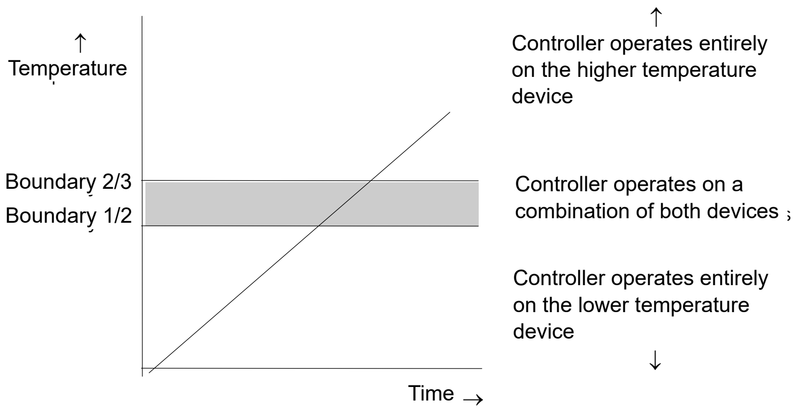


Figure 129 Thermocouple to Pyrometer Switching

Example: To Set the Switch Over Levels

Set Access to configuration level

1. Open the 'SwitchOver' block.
2. Set 'SwitchHigh' to a value which is suitable for the pyrometer (or high temperature thermocouple) to take over the control of the process.
3. Set 'SwitchLow' to a value which is suitable for the low temperature thermocouple to control the process.

Switch Over Parameters

Block – SwitchOver		Sub-block: .1			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
InHigh	Sets the high limit for the switch over block. It is the highest reading from input 2 since it is the high range input sensor.	Input range			Oper
InLow	Sets the low limit for the switch over block. It is the lowest reading from input 1 since it is the low range input sensor.				Oper
Switch High	Defines the high boundary of the switchover region.	Between Input Hi and Input Lo			Oper
Switch Low	Defines the low boundary of the switchover region.				Oper
In1	The first input value. This must be the low range sensor.	These will normally be wired to the thermocouple/pyrometer input sources via the PV Input or Analog Input Module. The range will be the range of the input chosen.			Read Only if wired
In2	The second input value. This must be the high range sensor.				Read Only if wired
Fallback Value	In the event of a bad status, the output may be configured to adopt the fallback value. This allows the strategy to dictate a 'safe' output in the event of an issue being detected.	Between Input Hi and Input Lo		0.0	Oper
Fallback Type	Fall back type	ClipBad (0) ClipGood (1) FallBad (2) FallGood (3) UpScaleBad (4) DownScaleBad (5)		ClipBad	Conf
SelectIn	Indicates which input is currently selected	Input1 (0)	Input 1 has been selected		Read Only
		Input2 (1)	Input 2 has been selected		
		Both (2)	Both inputs are used to calculate the output		
BadMode	The action taken if the selected input is BAD	UseGood (0)	Assumes the value of a good input If the currently selected input is BAD the output will assume the value of the other input if it is GOOD	UseGood	Conf
		ShowBad (1)	If selected input is BAD the output is BAD		
Out	Output produced from the two-input measurements				Read Only
Status	Status of the switchover block	Good (0) ChannelOff (1) OverRange (2) UnderRange (3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7) HWExceeded (8) NoData (8)			Read Only

Transducer Scaling

The Mini8 loop controller includes two transducer calibration function blocks. These are software function blocks that provide a method of offsetting the calibration of the input when compared to a known input source. Transducer scaling is often performed as a routine operation on a machine to take out system deviations. For this reason it can be carried out in operator mode.

Transducer scaling can be applied to any TC8/ET8 input set up as a linear PV input. They can be wired to the transducer scaling inputs.

Three types of calibration are explained in this chapter:

- Auto-tare
- Load Cell Calibration
- Comparison Calibration

Auto-Tare Calibration

The auto-tare function is used, for example, when it is required to weigh the contents of a container but not the container itself.

The procedure is to place the empty container on the weighbridge and 'zero' the controller. Since it is likely that following containers may have different tare weights the auto-tare feature is always available.

Further parameters are available which are used to pre-configure the tare measurement or for interrogation purposes. Tare calibration may be carried out no matter what type of transducer is in use.

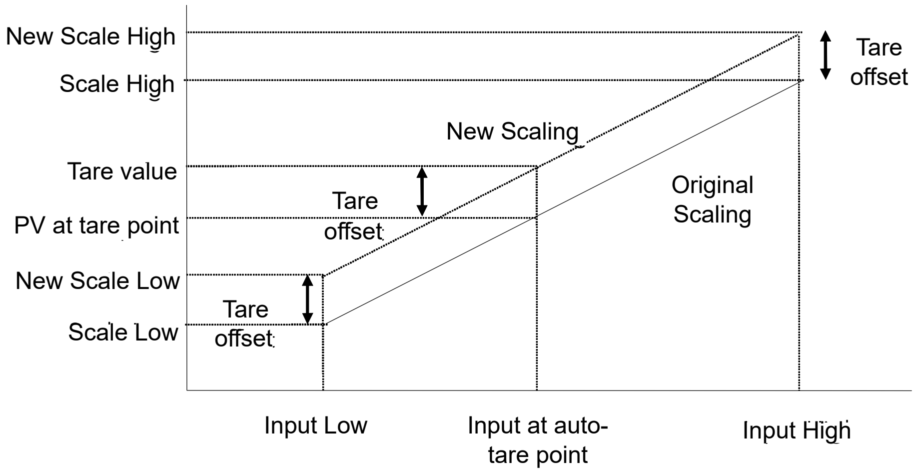


Figure 130 Effect of Auto Tare

Load Cell

A load cell provides a mV analog output which may be connected to a linear TC8/ET8 input. When no load is placed on the cell the output is normally zero. However, in practice there may be a residual output and this can be calibrated out in the controller. The high end is calibrated by placing a reference weight on the load cell and performing a high-end calibration in the controller.

Comparison Calibration

Comparison calibration is used to calibrate the controller against a second reference instrument.

The load is removed (or taken to a minimum) from the reference device. The controller low-end calibration is done using the 'Cal Enable' parameter and entering the reading from the reference instrument.

Add a weight and when the reading has stabilized, select the 'Cal Hi Enable' parameter then enter the new reading from the reference instrument.

Transducer Scaling Parameters

Block – Txdr		**Sub-blocks: . or .2**			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Cal Type	Used to select the type of transducer calibration to perform See descriptions at the beginning of this section.	Off (0)	Transducer type unconfigured	Off	Conf
		Shunt ()	Shunt calibration		
		LoadCell (2)	Load Cell		
		Compare (3)	Comparison		
Cal Enable	To make the transducer ready for calibration. Must be set to Yes to allow calibration to be done at L. This includes Tare Cal.	No (0) Yes ()	Not ready Ready	No	Conf
Range Max	The maximum permissible range of the scaling block	Range min to 99999		000	Conf
Range Min	The minimum permissible range of the scaling block	-9999 to Range max		0	Conf
Start Tare	Begin tare calibration	No Yes	Start tare calibration	No	Oper if 'Cal Enable' = 'Yes'
Start Cal	Starts the Calibration process. Note: for Load Cell and Comparison calibration 'Start Cal' starts the first calibration point.	No Yes	Start calibration	No	Oper if 'Cal Enable' = 'Yes'
Start HighCal	For Load Cell and Comparison calibration the 'Start High Cal' must be used to start the second calibration point.	No Yes	Start high calibration	No	Oper if 'Cal Enable' = 'Yes'
Clear Cal	Clears the current calibration constants. This returns the calibration to unity gain	No Yes	To delete previous calibration values	No	Conf
Tare Value	Enter the tare value of the container				Conf
InHigh	Sets the scaling input high point				Oper
InLow	Sets the scaling input low point				Oper
Scale High	Sets the scaling output high point. Usually the same as the 'Input Lo'				Oper
Scale Low	Sets the scaling output low point. Usually 80% of 'Input Hi'				Oper
Cal Band	The calibration algorithms use the threshold to determine if the value has settled. When switching in the shunt resistor, the algorithm waits for the value to settle to within the threshold before starting the high calibration point.				Conf
CalAdjust	The adjust is used in the Comparison Calibration method.	When edited, the Adjust parameter can be set to the desired value. On confirm, the new adjust value is used to set the scaling constants			Oper

Block – Txdr		Sub-blocks: . or .2			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
ShuntOut	Indicates when the internal shunt calibration resistor is switched in. Only appears if ‘Cal Type’ = ‘Shunt’	Off On	Resistor not switched in Resistor switched in		Oper
Cal Active	Indicates calibration taking place	Off On	Inactive Active		Read Only
InVal	The input value to be scaled.	-9999.9 to 9999.9			Oper
OutVal	The Input Value is scaled by the block to produce the Output Value				Oper
Cal Status	Indicates the progress of calibration	CalOff (0) Calibrating () Passed (2) ‘Failed’ (3)	No calibration in progress Calibration in progress Calibration Passed Calibration Unsuccessful		L Read Only
Status	The status of the output accounting for sensor break signals passed to the block and the state of the scaling.	Good (0) ChannelOff () OverRange (2) UnderRange (3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7) HWExceeded (8) NoData (9)			Conf

Parameter Notes

Enable Cal This may be wired to a digital input for an external switch. If not wired, then the value may be changed.

When enabled the transducer parameters may be altered as described in the previous sections. When the parameter has been turned On it will remain on until turned off manually even if the controller is power cycled.

Start Tare This may be wired to a digital input for an external switch. If not wired, then the value may be changed.

Start Cal This may be wired to a digital input for an external switch. If not wired, then the value may be changed.

It starts the calibration procedure for:
Shunt Calibration
The low point for Load Cell Calibration
The low point for Comparison Calibration

Start Hi Cal This may be wired to a digital input for an external switch. If not wired, then the value may be changed.

It starts:
The high point for Load Cell Calibration
The high point for Comparison Calibration

Clear Cal This may be wired to a digital input for an external switch. If not wired, then the value may be changed.

When enabled the input will reset to default values. A new calibration will overwrite the previous calibration values if Clear Cal is not enabled between calibrations.

Tare Calibration

The Mini8 loop controller has an auto-tare function that is used, for example, when it is required to weigh the contents of a container but not the container itself.

The procedure is to place the empty container on the weighbridge and ‘zero’ the controller. The procedure is as follows:

1. Place container on weighbridge.
2. Go to Txdr. (or 2) Folder.
3. Transducer calibration Type must be ‘Load Cell’.
4. CalEnable must be set to ‘Yes’.
5. Set StartTare to ‘Yes’.
6. The controller automatically calibrates to the tare weight which is measured by the transducer and stores this value.
7. During this measurement Cal Status will show progress. If the calibration is unsuccessful it is probably an ‘out-of-range’ problem.

Load Cell

A load cell output must be within the range 0 to 77mV to go into a TC8/ET8 input. Use a shunt for mA inputs, mV can possibly go direct, Volt inputs must use a potential divider. To calibrate a load cell:

1. Remove all load from the transducer to establish a zero reference.
2. Go to Txdr. (or 2) Folder.
3. Transducer calibration Type must be ‘Load Cell’.
4. CalEnable must be set to ‘Yes’.
5. Set Start Cal to ‘Yes’
6. The controller will calibrate the low point.
7. Set StartHighCal to ‘Yes’
8. The controller will calibrate the high point.

Cal Status advises progress and result.

Comparison Calibration

Comparison calibration is used to calibrate the input against a second reference instrument. Typically this might be a local display on the weighing device itself. To calibrate against a known reference source:

1. Add a load at the low end of the scale range.
2. Go to Txdr. (or Txdr.2) Folder.
3. Transducer calibration Type must be ‘Comparision’.
4. CalEnable must be set to ‘Yes’.
5. Enter the reading from the reference instrument into ‘Cal Adjust’.
6. Add a load at the high end of the scale.
7. Set StartHighCal to ‘yes’
8. The controller will calibrate the high point.

Cal Status advises progress and result.

User Values

User values are registers provided for use in calculations. They may be used as constants in equations or temporary storage in extended calculations. Up to 32 User Values are available. They are arranged in four groups of eight. Each User Value can then be set up in the ‘UserVal’ folder.

User Value Parameters

Block – UsrVal		Sub-blocks: .1 to .40		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Units	Units assigned to the User Value	None (0) C_F_K_Temp (1) V (2) mV (3) A (4) mA (5) pH (6) mmHg (7) psi (8) Bar (9) mBar (10) PercentRH (11) Percent (12) mmWG (13) inWG (14) inWW (15) Ohms (16) PSIG (17) PercentO2 (18) PPM (19) PercentCO2 (20) PercentCarb (21) PercentPerSec (22) RelTemp (24) Vacuum (25) Secs (26) Mins (27) Hours (28) Days (29) Mb (30) Mb (31) ms (32)		Conf
Resolution	Resolution of the User Value	X (0) X.X (1) X.XX (2) X.XXX (3) X.XXXX (4)		Conf
High Limit	The high limit may be set for each user value so that the value cannot be set to an out-of-bounds value.			Oper
Low Limit	The low limit of the user value may be set so that the user value cannot be edited to an illegal value. This is important if the user value is to be used as a setpoint.			Oper
Val	To set the value within the range limits	See Note		Oper

Block – UsrVal		Sub-blocks: .1 to .40			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Status	Can be used to force a good or bad status onto a user value. This is useful for testing status inheritance and fallback strategies.	Good (0) ChannelOff (1) OverRange (2) UnderRange (3) HardwareStatusInvalid (4) Ranging (5) Overflow (6) Bad (7) HWExceeded (8) NoData (9)	See Note		Oper

Note: If ‘Val’ is wired into but ‘Status’ is not, then, instead of being used to force the Status, it will indicate the status of the value as inherited from the wired connection to ‘Val’.

Calibration

In this chapter, calibration refers to calibration of the inputs of the TC4/TC8/ET8 modules and the RT4 module. Calibration is accessed using the ‘Cal State’ parameter that is only available in configuration level. Since the controller is calibrated during manufacture to traceable standards for every input range, it is not necessary to calibrate the controller when changing ranges.

However, it is recognized that, for operational reasons, it may be a requirement to check or re-calibrate the controller. This new calibration is saved as a User Calibration. It is always possible to revert to the factory calibration if necessary.

☺ Tip:

Consider using the ‘Offset’ parameter for User Cal (e.g. Mod.1.Offset). This can be set to correct any measured difference between the Mini8 loop controller given PV and a calibration value obtained from another source. This is useful where the process setpoint remains at about the same value during use.

Alternatively, if the setpoint range is wide use the two point calibration with the ‘LoPoint’, ‘LoOffset’, and ‘HiPoint’, ‘HiOffset’ parameters.

TC4/TC8 User Calibration

Set Up

No pre-calibration warm-up is required.

As calibration is a single-point on eight channels, quick enough (a few minutes) to avoid self-heating effects, there are no special environmental, mounting position or ventilation requirements for calibration.

Calibration should be performed at a reasonable ambient temperature (15°C to 35°C, 59°F to 95°F). Calibration outside these limits will compromise the expected working accuracy.

Every channel of every TC8/ET8 card must be individually connected to the calibrator source using thick copper wire (so the sensor-break voltage drop in the wires and source impedance is minimal).

The voltage source, monitor DVM and the target Mini8 loop controller should be at the same temperature (to remove added series EMF due to thermocouple effects).

Calibration of Mini8 loop controller requires the use of iTools.

The Mini8 loop controller must be in Configuration Mode.

Zero Calibration

No “zero” calibration point is required for TC4 or TC8 input channels.

Voltage Calibration

The iTools view below is shown for Module 1.

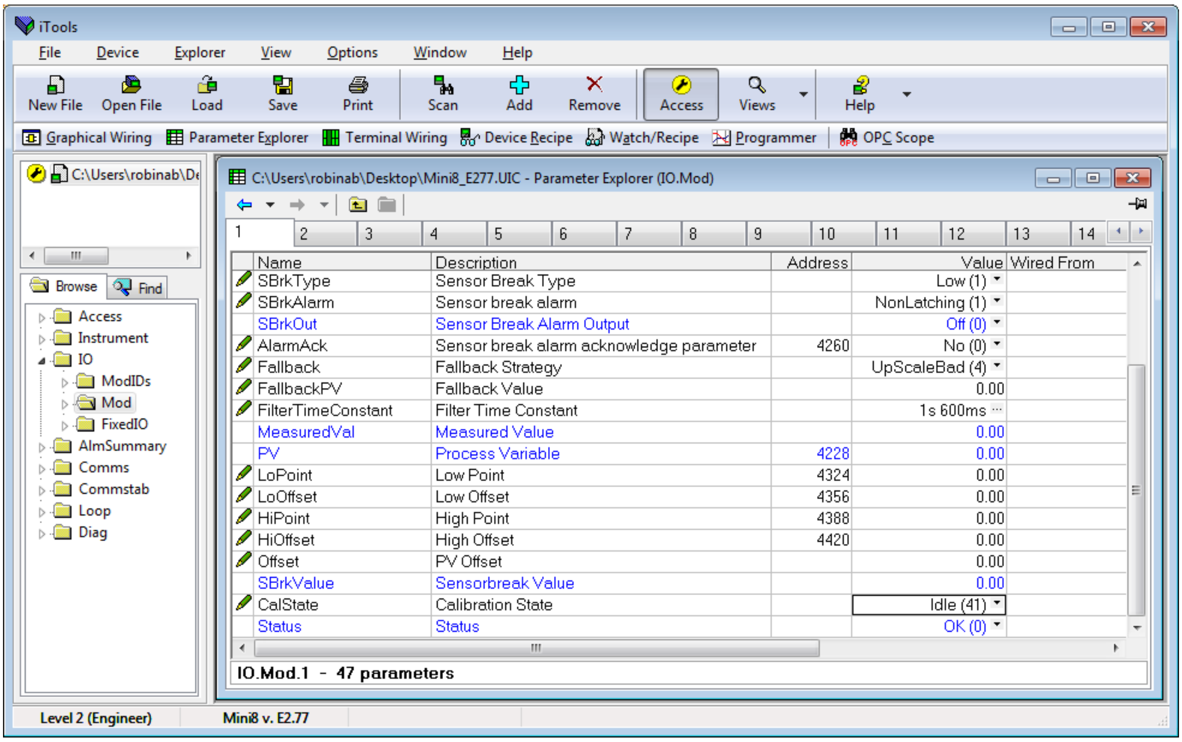


Figure 131 Voltage Calibration - Module 1

1. Set the Calibrator voltage source to an accurate 50.000mV.
2. Connect the 50mV to channel 1.
3. Set ‘CalState’ to ‘HiCal’ and then select ‘Confirm’.
4. When complete set ‘CalState’ to ‘SaveUser’.
5. Exit configuration mode.

CJC Calibration

No CJC calibration required; the sampled values are ratiometric, providing uncalibrated uncertainty of ±1°C.

Sensor-Break Limit Check

Apply a 900Ω resistor to each channel in turn, set ‘Sensor Break Type’ to ‘Low’, and filter to OFF. Verify the SBrkValue is greater than 24.0 and less than 61.0.

ET8 User Calibration

The ET8 requires four calibration phases:

- Hi_50mV Calibration
- Lo_50mV Calibration
- Hi_1V Calibration
- Lo_0V Calibration

Hi_50mV Calibration

Proceed as follows:

1. Set the Calibrator voltage source to an accurate 50.00mV.
2. For each ET8 channel, set the IOType to Thermocouple(11), apply the 50mV reference to each channel in turn.
3. Set the CalState parameter to Hi_50mV (123). The following sequence of CalState enumerations should occur:
   – Confirm? - select: Go (201)
   – Busy(212) - wait about 10 seconds until:
   – Passed(220) - select: Accept (221)
   – Idle(121)

Lo_50mV Calibration

Proceed as follows:

1. For each ET8 channel, IOType should remain set to Thermocouple(11), apply a short-circuit to each channel.
2. Set the CalState parameter to Lo_50mV (122). The following sequence of CalState enumerations should occur:
   – Confirm? - select: Go (201)
   – Busy(212) - wait about 10 seconds
   – Passed - select: Accept (221)
   – Idle(121)

When all eight channels have been successfully calibrated then save the coefficients to EEPROM by performing a “Save User” command: change the “CalState” parameter of Channel 1 (for the card) to SaveUser(125).

Hi_1V Calibration

Proceed as follows:

1. Set the Calibrator voltage source to an accurate 1.00V.
2. For each ET8 channel, set the IOType to ET8Cal(18), apply this 1V reference to each channel in turn.
3. Set the CalState parameter to Hi_1V (13). The following sequence of CalState enumerations should occur:
   - Confirm? - select: Go (201)
   - Busy(212) - wait about 10 seconds
   - Passed - select: Accept (221)
   - Idle(121)

Lo_0V Calibration

Proceed as follows:

1. For each ET8 channel, the IOType should remain set to ET8Cal(18), apply a short-circuit to each channel.
2. Set the CalState parameter to Lo_0V (12). The following sequence of CalState enumerations should occur:
   - Confirm? - select: Go (201)
   - Busy(212) - wait about 10 seconds
   - Passed - select: Accept (221)
   - Idle(121)
3. Channel “Status” should change from “not calibrated” to “OK”.

When all calibration phases have been successfully calibrated then save the coefficients to EEPROM by performing a “Save User” command: change the “CalState” parameter of Channel 1 (for the card) to SaveUser(125).

Note: To return to normal working, set the IOType parameter to Thermocouple(11) or mV(13) for each channel.

To Return to TC4/TC8/ET8 Factory Calibration

To clear the User calibration and restore the calibration from the factory:

1. Put Mini8 loop controller into Configuration Mode.
2. Set the ‘Calibration State’ to ‘LoadFact’.
3. Return Instrument to Operating Mode.

RT4 User Calibration

Set Up

No pre-calibration warm-up is required.

There are no special environmental, mounting position or ventilation requirements for calibration.

Calibration should be performed at a reasonable ambient temperature (15°C to 35°C -59°F to 95°F). Calibration outside these limits will compromise the expected working accuracy.

Each channel of the RT4 card must be individually connected to the calibrated resistance box using the four-wire connection.

The Mini8 loop controller must be in Configuration Mode.

Calibration

1. Set the Resistance Range to Low or High as required.
2. Wire the resistance box to channel 1 using the four-wire connection.
3. Set the resistance box to 150.0Ω ±0.02% for Low Resistance calibration or 1500Ω ±0.02% for High Resistance calibration.
4. Set ‘CalState’ to’ LoCal’ and then select ‘Confirm’ followed by ‘Go’.
5. The instrument will show ‘Busy’ followed by ‘Passed’, assuming the calibration is successful, or ‘Failed’ if not. If ‘Failed’, check that the correct calibration resistance has been selected.
6. When complete set ‘CalState’ to’ SaveUser’.
7. Set the Resistance box to 400.0Ω ±0.02% for Low Resistance calibration or 4000Ω ±0.02% for High Resistance calibration.
8. Set ‘CalState’ to ‘HiCal’ and then select ‘Confirm’ followed by ‘Confirm’ followed by ‘Go’.
9. The instrument will show ‘Busy’ followed by ‘Passed’, assuming the calibration is successful, or ‘Failed’ if not. If ‘Failed’ check that the correct calibration resistance has been selected.
10. When complete set ‘CalState’ to’ SaveUser’. This will enable the new calibration data to be used following a power down of the instrument. If the data is not saved it will be lost at power down.
11. Exit configuration mode.

To Return to RT4 Factory Calibration

To clear the User calibration and restore the calibration from the factory for RTDs it is necessary to set the Resistance Range to the one in use – Low or High.

For Pt100

1. Put Mini8 loop controller into Configuration Mode.
2. For Low Resistance select ‘Resistance Type’ = ‘Low’. This selects the previously saved (SaveUser) calibration data for Pt100.
3. Set the ‘Calibration State’ to ‘LoadFact’.
4. After a few seconds the ‘CalSate’ parameter returns to ‘Idle’. The Factory calibration data is now restored, overwriting the previously stored User Calibration.
5. Return Instrument to Operating Mode.

For Pt1000

1. Put Mini8 loop controller into Configuration Mode.
2. For High Resistance select ‘Resistance Type’ = ‘High’. This selects the previously saved (SaveUser) calibration data for Pt1000.
3. Set the ‘Calibration State’ to ‘LoadFact’.
4. After a few seconds the ‘CalSate’ parameter returns to ‘Idle’. The Factory calibration data is now restored, overwriting the previously stored User Calibration.
5. Return Instrument to Operating Mode.

Calibration Parameters

Block - IO		Sub-blocks: Mod.1 to Mod.32			
Name	**Parameter Description**	**Value**		**Default**	**Access Level**
Cal State	Calibration state of the input	Idle	Normal operation	Idle	Conf
		Hi-50mV	High input calibration for mV ranges		
		Load Fact	Restore factory calibration values		
		Save User	Save the new calibration values		
		Confirm	To start the calibration procedure when one of the above has been selected		
		Go	Starting the automatic calibration procedure		
		Busy	Calibration in progress		
		Passed	Calibration successful		
		'Failed'	Calibration unsuccessful		
Status	PV Status The current status of the PV.	0	Normal operation		Read Only
		1	Initial startup mode		
		2	Input in sensor break		
		3	PV outside operating limits		
		4	Saturated input		
		5	Uncalibrated channel		
		6	No Module		

The above list shows the values of CalState, which appear during a normal calibration procedure. The full list of possible values follows – the number is the enumeration for the parameter.

1: Idle

2: Low calibration point for Volts range

3: High calibration point for Volts range

4: Calibration restored to factory default values

5: User calibration stored

6: Factory calibration stored

11: Idle

12: Low calibration point for HZ input

13: High calibration point for the HZ input

14: Calibration restored to factory default values

15: User calibration stored

16: Factory calibration stored

20: Calibration point for factory rough calibration

21: Idle

22: Low calibration point for the mV range

23: Hi calibration point for the mV range

24: Calibration restored to factory default values

25: User calibration stored

26: Factory calibration stored

30: Calibration point for factory rough calibration

31: Idle

32: Low calibration point for the mV range

33: High calibration point for the mV range

34: Calibration restored to factory default values

35: User calibration stored

36: Factory calibration stored

41: Idle

42: Low calibration point for RTD calibration (150Ω for Low Resistance range, 1500Ω for High Resistance range)

43: High calibration point for RTD calibration (400Ω for Low Resistance range, 4000Ω for High Resistance range)

44: Calibration restored to factory default values

45: User calibration stored

46: Factory calibration stored

51: Idle

52: CJC calibration used in conjunction with Term Temp parameter

54: Calibration restored to factory default values

55: User calibration stored

56: Factory calibration stored

200: Confirmation of request to calibrate

201: Used to start the calibration procedure

202: Used to abort the calibration procedure

210: Calibration point for factory rough calibration

212: Indication that calibration is in progress

213: Used to abort the calibration procedure

220: Indication that calibration completed successfully

221: Calibration accepted but not stored

222: Used to abort the calibration procedure

223: Indication that calibration was unsuccessful

Config Lock

Introduction

Config Lock is available as an orderable option and is protected by Feature Security.

Config Lock allows users to help prevent unauthorized viewing, reverse engineering or cloning of controller configurations. This includes application specific internal (soft) wiring, limited access to certain Configuration level and Operator level parameters via comms (by iTools or a third party comms package).

When Config Lock is enabled, users are prevented from accessing soft wiring from any source, and it is not possible to Load or Save the configuration of the instrument via iTools or by using the Save/Restore facility.

Altering configuration and/or operator parameters via Comms may also be restricted when Config Lock is implemented.

Once the security function has been set up for a particular application it may be cloned into every other identical application without further configuration.

Using Config Lock

When Config Lock is supplied, four Config Lock parameters are displayed in the 'Instrument - Security' list in iTools.

- **ConfigLockPassword**
  This password is selected by the OEM. Any alpha/numeric text can be used and the field is editable whilst the Config Lock Status is 'Unlocked'. A minimum of eight characters should be used. It is not possible to clone the Config Lock Security Password. (Highlight the complete row before entering).

- **ConfigLockEntry**
  Enter the Config Lock password to activate and deactivate Config Lock. The controller must be in configuration level to enter this password. When the correct password is entered the **ConfigLockStatus** will toggle between 'Locked' and 'Unlocked'. (Highlight the complete row before entering). Three login attempts are allowed before lockout which is followed by a 90 minute password lockout period.

- **ConfigLockStatus**
  Read only showing 'Locked' or 'Unlocked'.
  - If Unlocked two lists are available which allow an OEM to restrict which parameters are alterable when the controller is in Operator and Configuration Access levels.
  - Parameters added in the **ConfigLockConfigList** WILL be available to the operator when the controller is in Configuration level. Parameters not added in this list will not be available to the operator.
  - Parameters added to the **ConfigLockOperList** will NOT be available to the operator when the controller is in Operator access level.
  - If the **ConfigLockStatus** is 'Locked' these two lists are not shown. The controller configuration is prevented from being cloned and the internal wiring cannot be accessed via comms.

- **ConfigLockParameterLists**
  This parameter is only writeable when the **ConfigLock Status** is ‘Unlocked’.
  - When ‘Off’, Operator type parameters are alterable in Operator access level and Config parameters are alterable in Configuration access level (all within other limitations such as high and low limits).
  - When ‘On’, parameters added to the **ConfigLockConfigList** WILL be available to the operator when the controller is in configuration level. Parameters not added in this list will not be available to the operator. Parameters added to the **ConfigLockOperList** will NOT be available to the operator when the controller is in Operator access level.
  - The table at the end of this section shows an example for just two parameters ‘Alarm 1 Type’ (configuration type parameter) and ‘Alarm 1 Threshold’ (operator type parameter).

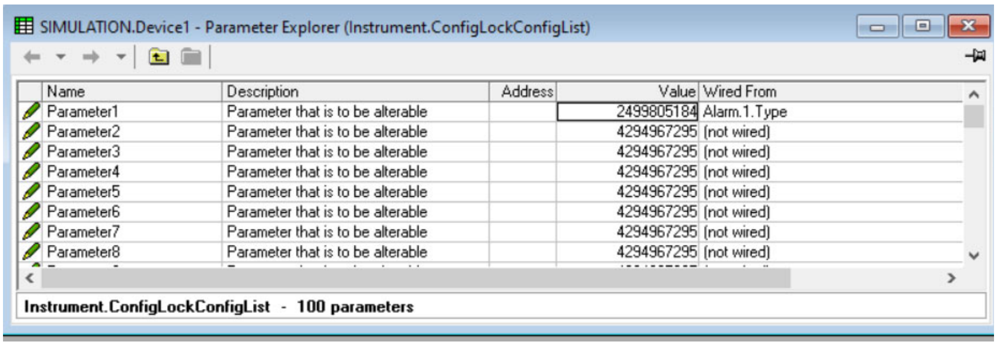
When entering or exiting Config Lock, allow a few seconds for iTools to synchronize.

Config Lock Configuration List

The **ConfigLockConfigList** allows the OEM to choose up to 100 configuration parameters which are to remain Read/Write while in Configuration level and Config Lock is enabled. In addition to these the following parameters are always writeable in configuration mode:

Config Lock Password Entry, Comms Configuration password, Controller Coldstart parameter.

The required parameters may be by dragged and dropped from a browser list (on the left hand side) into the Wired From cell in the **ConfigLockConfigList**. Alternatively, double click into the 'WiredFrom' cell and select the parameter from the pop-up list. These parameters are those chosen by the OEM which are to remain alterable when Config Lock is enabled and the controller is in Configuration access level.

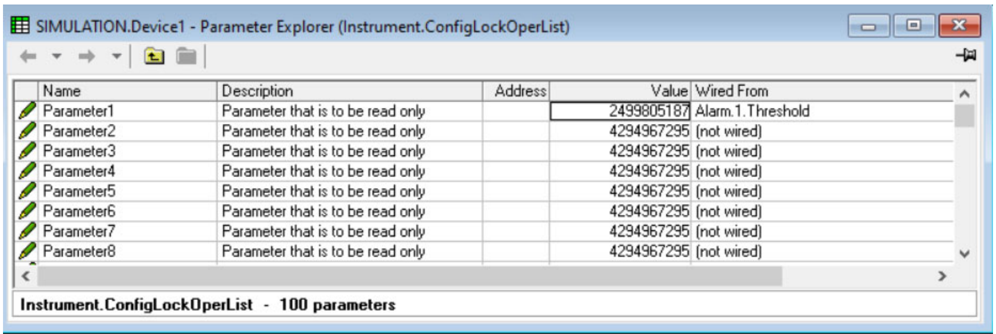


The view shows the first eight parameters of which Parameter 1 has been populated with a configuration parameter (Alarm 1 Type). Examples of configuration parameters include Alarm Types, Input Types, Range Hi/Lo, Modules Expected, etc.

When the Config Lock Status is Locked, this list is not shown.

Config Lock Operator List

The **ConfigLockOperatorList** operates in the same way as the **ConfigLockConfigList** except the parameters selected are those which are available in Operator access level. Examples are programmer mode, alarm setting parameters, etc. The example below shows 'Alarm 1 Threshold' which is to be read only in Operator access level.



The example shows the first 8 of 100 parameters of which the first has been selected as 'Alarm 1 Threshold'. This parameter is to be read only when Config Lock is enabled and the controller is in Operator access level.

When the **ConfigLockStatus** is Locked, this list is not shown.

Effect of the ‘Config Lock ParamList’ Parameter

The table below shows the availability of the two ‘Alarm 1’ parameters set up in the previous pages when the **ConfigLockParamList** parameter is turned On or Off.

‘Alarm 2’ is used as an example of all parameters which have not been included in Config Lock.

’ConfigLockParamLists’	Parameter	Controller in Configuration Access		Controller in Operator Access	
		Alterable	Not alterable	Alterable	Not alterable
On	A1 Type	✓			✓
	A2 Type		✓		✓
	A1 Threshold		✓		✓
	A2 Threshold	✓		✓	
Off	A1 Type	✓			✓
	A2 Type	✓			✓
	A1 Threshold	✓		✓	
	A2 Threshold	✓		✓	

The iTools views shown in the next page show how this example is presented in the iTools browser:

'ConfigLockParamLists' On

The iTools views shown below show the alterability of the alarm parameters used in the previous examples. Alarm 1 has been set up in Config Lock. Alarm 2 is used as an example of parameters not set up in Config Lock.

Text in black shows parameters are alterable. Text in blue is not alterable.

Controller in Configuration Mode

'Alarm 1 Type' is alterable
'Alarm 1 Threshold' is not alterable

Name	Description	ddress	Value
Type	Alarm type	536	AbsHi (1)
Status	Alarm status	2113	Off (0)
Input	Input to be evaluated	2114	47.50
Threshold	Threshold	13	999.70
Hysteresis	Hysteresis	47	2.30

'Alarm 2 Type' is not alterable
'Alarm 2 Threshold' is alterable

Name	Description	ddress	Value
Type	Alarm type	537	AbsLo (2)
Status	Alarm status	2137	Off (0)
Input	Input to be evaluated	2138	47.49
Threshold	Threshold	14	-10.00
Hysteresis	Hysteresis	68	1.00

Controller in Operator Mode

'Alarm 1Type' is not alterable
'Alarm 1 Threshold' is not alterable

Name	Description	ddress	Value
Type	Alarm type	536	AbsHi (1)
Status	Alarm status	2113	Off (0)
Input	Input to be evaluated	2114	47.48
Threshold	Threshold	13	999.70
Hysteresis	Hysteresis	47	2.30

'Alarm 2 Type' is not alterable
'Alarm 2 Threshold' is alterable

Name	Description	ddress	Value
Type	Alarm type	537	AbsLo (2)
Status	Alarm status	2137	Off (0)
Input	Input to be evaluated	2138	47.45
Threshold	Threshold	14	-10.00
Hysteresis	Hysteresis	68	1.00

'ConfigLockParaLists' Off

Controller in Configuration Mode

'Alarm 1 Type' is alterable
'Alarm 1 Threshold' is alterable

Name	Description	ddress	Value
Type	Alarm type	536	AbsHi (1)
Status	Alarm status	2113	Off (0)
Input	Input to be evaluated	2114	47.46
Threshold	Threshold	13	999.70

'Alarm 2 Type' is alterable
'Alarm 2 Threshold' is alterable

Name	Description	ddress	Value
Type	Alarm type	537	AbsLo (2)
Status	Alarm status	2137	Off (0)
Input	Input to be evaluated	2138	47.47
Threshold	Threshold	14	-10.00

Controller in Operator Mode

'Alarm 1 Type' is not alterable
'Alarm 1 Threshold' is alterable

Name	Description	ddress	Value
Type	Alarm type	536	AbsHi (1)
Status	Alarm status	2113	Off (0)
Input	Input to be evaluated	2114	47.56
Threshold	Threshold	13	999.70

'Alarm 2 Type' is not alterable
'Alarm 2 Threshold' is alterable

Name	Description	ddress	Value
Type	Alarm type	537	AbsLo (2)
Status	Alarm status	2137	Off (0)
Input	Input to be evaluated	2138	47.50
Threshold	Threshold	14	-10.00

Notes:

1. Parameters are alterable within other set limits.
2. The availability applies to access through comms.

Modbus SCADA Table

These parameters are single register Modbus values for use with Third Party Modbus masters (clients) in SCADA packages or PLCs. Scaling of the parameters has to be configured – the Modbus master (client) scaling has to match the Mini8 loop controller parameter resolution so that the decimal point is in the correct position.

If a parameter has no address the CommsTab feature can be used to map the parameter to a Modbus address, however, it should be noted that the address field will not be updated.

Comms Table

The tables that follow do not include every parameter in the Mini8 loop controller. The Comms Table is used to make most parameters available at any SCADA address.

Folder – Commstab		**Sub-folders: .1 to .250**		
Name	**Parameter Description**	**Value**	**Default**	**Access Level**
Destination	Modbus Destination	Not Used 0 to 15819	Not used	Conf
Source	Source Parameter	Taken from source parameter		Conf
Native	Native Data Format	0 Integer 1 Native (i.e. Float or Long)	Integer	Conf
ReadOnly	Read Only Read/Write only if source is R/W	0 Read/Write 1 Read Only	Read/Write	Conf
Minutes	Minutes Units in which time is scaled	0 Seconds 1 Minutes	Seconds	Conf

Entering a value in the Source parameter may be done in two ways:

- Drag the required parameter into the Source.
- Right-click the Source parameter, select Edit Wire and browse to the required parameter.

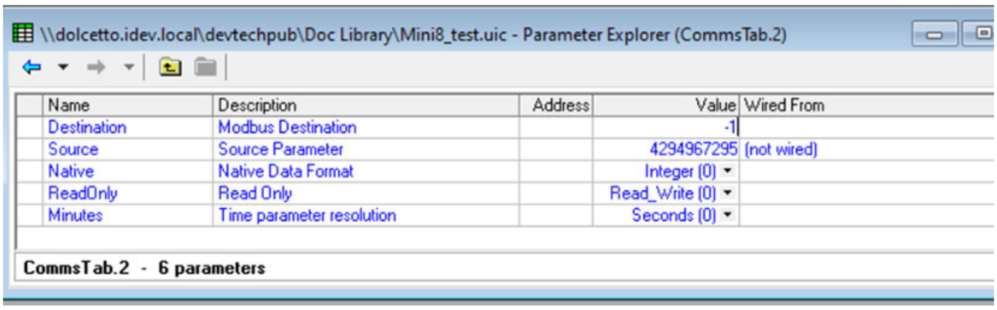


Figure 132 Parameter Explorer

There are 250 comms table entries available.

SCADA Table

The parameters are available in scaled integer format, accessed via their associated Modbus address.

Wherever possible use an OPC client with the iTools OPCserver as the server. In this arrangement the parameters are all referenced by name and the values are floating point so the decimal point for all parameters is inherited.

Refer to the auto-generated SCADA help file in iTools for a list of the parameters. This file is accessed via ‘Device Help’ option.

Modbus Function Codes

Mini8 loop controller supports the following function codes:

3, 4	Multiple parameter read
6	Single parameter write
7	Status read
8	Loop back
16	Multiple parameter write

Function codes 103 and 106 are special codes used by iTools. These should not be used.

Mini8 loop controller does not support function code 23.

DeviceNet Parameter Tables

IO Re-Mapping Object

DeviceNet comes pre-configured with the key parameters of eight PID loops and alarms (60 input parameters process variables, alarm status, and so on and 60 output parameters – setpoints, and so on). Loops 9-16 are not included in the DeviceNet tables as there are insufficient attributes for the DeviceNet parameters.

The Mini8 loop controller DeviceNet communication is supplied with a default input assembly table (80 bytes) and output assembly table (48 bytes). The parameters included are listed below.

Note: To modify these tables see the next section.

The default Input assembly table:

Input Parameter	Offset	Value (Attr ID)
PV – Loop 1	0	0
Working SP – Loop 1	2	1
Working Output – Loop 1	4	2
PV – Loop 2	6	14 (0EH)
Working SP – Loop 2	8	15 (0FH)
Working Output – Loop 2	10	16 (10H)
PV – Loop 3	12	28 (1CH)
Working SP – Loop 3	14	29 (1DH)
Working Output – Loop 3	16	30 (1EH)
PV – Loop 4	18	42 (2AH)
Working SP – Loop 4	20	43 (2BH)
Working Output – Loop 4	22	44 (2CH)
PV – Loop 5	24	56 (38H)
Working SP – Loop 5	26	57 (39H)
Working Output – Loop 5	28	58 (3AH)
PV – Loop 6	30	70 (46H)
Working SP – Loop 6	32	71 (47H)
Working Output – Loop 6	34	72 (48H)
PV – Loop 7	36	84 (54H)
Working SP – Loop 7	38	85 (55H)
Working Output – Loop 7	40	86 (56H)
PV – Loop 8	42	98 (62H)
Working SP – Loop 8	44	99 (63H)
Working Output – Loop 8	46	100 (64H)
Analogue Alarm Status 1	48	144 (90H)
Analogue Alarm Status 2	50	145 (91H)
Analogue Alarm Status 3	52	146 (92H)
Analogue Alarm Status 4	54	147 (93H)
Sensor Break Alarm Status 1	56	148 (94H)
Sensor Break Alarm Status 2	58	149 (95H)
Sensor Break Alarm Status 3	60	150 (96H)
Sensor Break Alarm Status 4	62	151 (97H)
CT Alarm Status 1	64	152 (98H)
CT Alarm Status 2	66	153 (99H)
CT Alarm Status 3	68	154 (9AH)
CT Alarm Status 4	70	155 (9BH)

Input Parameter	Offset	Value (Attr ID)
New Alarm Output	72	156 (9CH)
Any Alarm Output	74	157 (9DH)
New CT Alarm Output	76	158 (9EH)
Program Status	78	184 (B8H)
TOTAL LENGTH	**80**	

The default output assembly table.

Output Parameter	Offset	Value
Target SP – Loop 1	0	3
Auto/Manual – Loop 1	2	7
Manual Output – Loop 1	4	4
Target SP – Loop 2	6	17 (11H)
Auto/Manual – Loop 2	8	21 (15H)
Manual Output – Loop 2	10	18 (12H)
Target SP – Loop 3	12	31 (1FH)
Auto/Manual – Loop 3	14	35 (23H)
Manual Output – Loop 3	16	32 (20H)
Target SP – Loop 4	18	45 (2DH)
Auto/Manual – Loop 4	20	49 (31H)
Manual Output – Loop 4	22	46 (2EH)
Target SP – Loop 5	24	59 (3BH)
Auto/Manual – Loop 5	26	63 (3FH)
Manual Output – Loop 5	28	60 (3CH)
Target SP – Loop 6	30	73 (49H)
Auto/Manual – Loop 6	32	77 (4DH)
Manual Output – Loop 6	34	74 (4AH)
Target SP – Loop 7	36	87 (57H)
Auto/Manual – Loop 7	38	91 (5BH)
Manual Output – Loop 7	40	88 (58H)
Target SP – Loop 8	42	101 (65H)
Auto/Manual – Loop 8	44	105 (69H)
Manual Output – Loop 8	46	102 (66H)
TOTAL LENGTH	**48**	

Application Variables Object

This is the list of parameters available to be included in the input and output tables.

Parameter	Attribute ID
Process Variable – Loop 1	0
Working Setpoint – Loop 1	1
Working Output – Loop 1	2
Target Setpoint – Loop 1	3
Manual Output – Loop 1	4
Setpoint 1 – Loop 1	5
Setpoint 2 – Loop 1	6
Auto/Manual Mode – Loop 1	7
Proportional Band – Loop 1 working Set	8
Integral Time – Loop 1 working Set	9
Derivative Time – Loop 1 working Set	10
Cutback Low – Loop 1 working Set	11
Cutback High – Loop 1 working Set	12
Relative Cooling Gain – Loop 1 working Set	13
Process Variable – Loop 2	14
Working Setpoint – Loop 2	15
Working Output – Loop 2	16
Target Setpoint – Loop 2	17
Manual Output – Loop 2	18
Setpoint 1 – Loop 2	19
Setpoint 2 – Loop 2	20
Auto/Manual Mode – Loop 2	21
Proportional Band – Loop 2 working Set	22
Integral Time – Loop 2 working Set	23
Derivative Time – Loop 2 working Set	24
Cutback Low – Loop 2 working Set	25
Cutback High – Loop 2 working Set	26
Relative Cooling Gain – Loop 2 working Set	27
Process Variable – Loop 3	28
Working Setpoint – Loop 3	29
Working Output – Loop 3	30
Target Setpoint – Loop 3	31
Manual Output – Loop 3	32
Setpoint 1 – Loop 3	33
Setpoint 2 – Loop 3	34
Auto/Manual Mode – Loop 3	35
Proportional Band – Loop 3 working Set	36
Integral Time – Loop 3 working Set	37
Derivative Time – Loop 3 working Set	38
Cutback Low – Loop 3 working Set	39
Cutback High – Loop 3 working Set	40
Relative Cooling Gain – Loop 3 working Set	41
Process Variable – Loop 4	42
Working Setpoint – Loop 4	43
Working Output – Loop 4	44
Target Setpoint – Loop 4	45
Manual Output – Loop 4	46
Setpoint 1 – Loop 4	47
Setpoint 2 – Loop 4	48

Parameter	Attribute ID
Auto/Manual Mode – Loop 4	49
Proportional Band – Loop 4 working Set	50
Integral Time – Loop 4 working Set	51
Derivative Time – Loop 4 working Set	52
Cutback Low – Loop 4 working Set	53
Cutback High – Loop 4 working Set	54
Relative Cooling Gain – Loop 4 working Set	55
Process Variable – Loop 5	56
Working Setpoint – Loop 5	57
Working Output – Loop 5	58
Target Setpoint – Loop 5	59
Manual Output – Loop 5	60
Setpoint 1 – Loop 5	61
Setpoint 2 – Loop 5	62
Auto/Manual Mode – Loop 5	63
Proportional Band – Loop 5 working Set	64
Integral Time – Loop 5 working Set	65
Derivative Time – Loop 5 working Set	66
Cutback Low – Loop 5 working Set	67
Cutback High – Loop 5 working Set	68
Relative Cooling Gain – Loop 5 working Set	69
Process Variable – Loop 6	70
Working Setpoint – Loop 6	71
Working Output – Loop 6	72
Target Setpoint – Loop 6	73
Manual Output – Loop 6	74
Setpoint 1 – Loop 6	75
Setpoint 2 – Loop 6	76
Auto/Manual Mode – Loop 6	77
Proportional Band – Loop 6 working Set	78
Integral Time – Loop 6 working Set	79
Derivative Time – Loop 6 working Set	80
Cutback Low – Loop 6 working Set	81
Cutback High – Loop 6 working Set	82
Relative Cooling Gain – Loop 6 working Set	83
Process Variable – Loop 7	84
Working Setpoint – Loop 7	85
Working Output – Loop 7	86
Target Setpoint – Loop 7	87
Manual Output – Loop 7	88
Setpoint 1 – Loop 7	89
Setpoint 2 – Loop 7	90
Auto/Manual Mode – Loop 7	91
Proportional Band – Loop 7 working Set	92
Integral Time – Loop 7 working Set	93
Derivative Time – Loop 7 working Set	94
Cutback Low – Loop 7 working Set	95
Cutback High – Loop 7 working Set	96
Relative Cooling Gain – Loop 7 working Set	97
Process Variable – Loop 8	98
Working Setpoint – Loop 8	99
Working Output – Loop 8	100

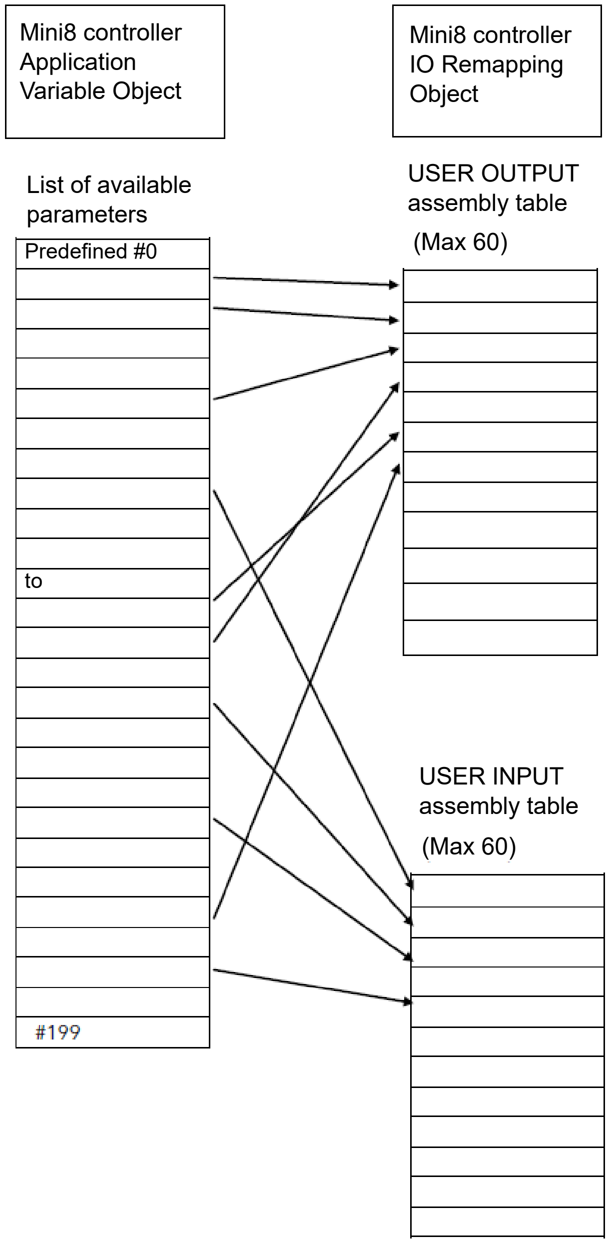
Parameter	Attribute ID
Target Setpoint – Loop 8	101
Manual Output – Loop 8	102
Setpoint 1 – Loop 8	103
Setpoint 2 – Loop 8	104
Auto/Manual Mode – Loop 8	105
Proportional Band – Loop 8 working Set	106
Integral Time – Loop 8 working Set	107
Derivative Time – Loop 8 working Set	108
Cutback Low – Loop 8 working Set	109
Cutback High – Loop 8 working Set	110
Relative Cooling Gain – Loop 8 working Set	111
Module PV – Channel 1	112
Module PV – Channel 2	113
Module PV – Channel 3	114
Module PV – Channel 4	115
Module PV – Channel 5	116
Module PV – Channel 6	117
Module PV – Channel 7	118
Module PV – Channel 8	119
Module PV – Channel 9	120
Module PV – Channel 10	121
Module PV – Channel 11	122
Module PV – Channel 12	123
Module PV – Channel 13	124
Module PV – Channel 14	125
Module PV – Channel 15	126
Module PV – Channel 16	127
Module PV – Channel 17	128
Module PV – Channel 18	129
Module PV – Channel 19	130
Module PV – Channel 20	131
Module PV – Channel 21	132
Module PV – Channel 22	133
Module PV – Channel 23	134
Module PV – Channel 24	135
Module PV – Channel 25	136
Module PV – Channel 26	137
Module PV – Channel 27	138
Module PV – Channel 28	139
Module PV – Channel 29	140
Module PV – Channel 30	141
Module PV – Channel 31	142
Module PV – Channel 32	143
Analogue Alarm Status 1	144
Analogue Alarm Status 2	145
Analogue Alarm Status 3	146
Analogue Alarm Status 4	147
Sensor Break Alarm Status 1	148
Sensor Break Alarm Status 2	149
Sensor Break Alarm Status 3	150
Sensor Break Alarm Status 4	151
CT Alarm Status 1	152

Parameter	Attribute ID
CT Alarm Status 2	153
CT Alarm Status 3	154
CT Alarm Status 4	155
New Alarm Output	156
Any Alarm Output	157
New CT Alarm Output	158
Reset New Alarm	159
Reset New CT Alarm	160
CT Load Current 1	161
CT Load Current 2	162
CT Load Current 3	163
CT Load Current 4	164
CT Load Current 5	165
CT Load Current 6	166
CT Load Current 7	167
CT Load Current 8	168
CT Load Status 1	169
CT Load Status 2	170
CT Load Status 3	171
CT Load Status 4	172
CT Load Status 5	173
CT Load Status 6	174
CT Load Status 7	175
CT Load Status 8	176
PSU Relay 1 Output	177
PSU Relay 2 Output	178
PSU Digital Input 1	179
PSU Digital Input 2	180
Program Run	181
Program Hold	182
Program Reset	183
Program Status	184
Current Program	185
Program Time Left	186
Segment Time Left	187
User Value 1	188
User Value 2	189
User Value 3	190
User Value 4	191
User Value 5	192
User Value 6	193
User Value 7	194
User Value 8	195
User Value 9	196
User Value 10	197
User Value 11	198
User Value 12	199

Table Modification

Make a list of parameters required in the input and output tables to suit the application. If the parameter is listed in the predefined list then use the attribute number of that parameter.

To set up the controller so that the required parameters are available on the network requires setting up the INPUT and OUTPUT data assembly tables with the IDs from the Application Variable Object.



Technical Specification

The I/O electrical specifications are quoted as factory calibrated worst-case; for life, over full ambient temperature range and supply voltage. Any "typical" figures quoted are the expected values at 25°C ambient and 24Vdc supply.

The nominal update of all inputs and function blocks is every 110ms. However, in complex applications the Mini8 loop controller will automatically extend this time in multiples of 110ms.

This instrument conforms with the essential protection requirements of the EMC Directive 2014/35/EU. It satisfies the general requirements of the industrial environment defined in EN 61326.

Environmental Sustainability

UKCA/EU RoHS directive	UKCA/EU RoHS Declaration
Mercury free	Yes
RoHS exemption information	Yes
China RoHS regulation	China RoHS declaration
Environmental disclosure	Product environmental profile
Circularity profile	End of Life Information

Note: Refer to the Mini8 Loop Controller Product Information page on the Eurotherm website (www.eurotherm.com) for details.

Environmental Specification

Power supply voltage	17.8Vdc minimum to 28.8Vdc maximum
Supply ripple	2Vp-p maximum
Power consumption	15W maximum
Maximum applied voltage any terminal	42Vpeak
Operating temperature	0 to 55°C (32°F to 131°F)
Storage temperature	-10°C to +70°C (14°F to 158°F)
Relative humidity	5% to 95% RH non-condensing
Altitude	<2000m (<6561.68ft)
Approvals	CE, UKCA UL, cUL
Safety	Meets EN61010-1: 2019 and UL 61010-1: 2012 Installation Category II Pollution Degree 2.
EMC	EN61326:2013 Emissions: Class A - Heavy Industrial Immunity: Industrial
Protection	IP20 The Mini8 loop controller must be mounted in a protective enclosure
RoHS compliance	UKCA/EU RoHS REACH WEEE China RoHS

Network Communications Support

Modbus RTU: EIA-485, 2 x RJ45, user select switch for 3-wire or 5-wire	Baud rates: 4800bps, 9600bps, 19200bps
DeviceNet: CAN, 5-pin standard "open connector" with screw terminals	Baud rates: 125kbps, 250kbps, 500kbps
Ethernet: Standard Ethernet RJ45 connector	Baud rate: 10Base-T
EtherCAT	
Isolation between RJ45 connector and system	1500Vac
Modbus, DeviceNet, and Ethernet are mutually exclusive options; refer to the Mini8 controller order code.	

Configuration Communications Support

Modbus RTU: 3-wire EIA-232, through RJ11 configuration port	Baud rates: 4800, 9600, 19200
All versions of Mini8 loop controller support one configuration port. The configuration port can be used simultaneously with the network link.	

Fixed I/O Resources

The PSU card supports two independent and isolated relay contacts	
Relay output types	On/Off (C/O contacts, "On" closing the N/O pair)
Contact current	<1A (resistive loads)
Terminal voltage	<42Vpk
Contact material	Gold
Snubbers	Snubber networks are NOT fitted
Contact isolation	42Vpeak maximum
The PSU card supports two independent and isolated logic inputs	
Input types	Logic (24Vdc)
Input logic 0 (off)	-28.8V to +5Vdc
Input logic 1 (on)	+10.8V to +28.8Vdc
Input current	2.5mA (approx) at 10.8V; 10mA maximum at 28.8V supply
Detectable pulse width	110ms minimum
Isolation to system	42Vpeak maximum

TC8/ET8 8-Channel and TC4 4-Channel TC Input Card

The TC8/ET8 supports eight independently programmable and electrically isolated channels, catering for all standard and custom thermocouple types. The TC4 supports four channels to the same specification.	
Channel types	TC, mV Input Range: -77mV to +77mV
Resolution	20 bit (ΣΔ converter), 1.6µV with 1.6s filter time
Temperature coefficient	< ±50ppm (0.005%) of reading/ °C (TC4/TC8) <±1µV/C ±25ppm/C of measurement, from 25°C ambient (ET8)
Cold junction range	-10°C to +70°C (14°F to 158°F)
CJ rejection	> 30:1 (TC4/TC8) 100:1 (ET8)
CJ accuracy	±1°C (TC4/TC8) ±0.25°C (ET8)
Linearisation types	C, J, K, L, R, B, N, T, S, LINEAR mV, custom
Total accuracy	±1°C ±0.1% of reading (using internal CJC) (TC4/TC8) ±0.25°C ±0.05% of reading at 25°C ambient (ET8)
Channel PV filter	0.0 seconds (off) to 999.9 seconds, 1st order low-pass
Sensor break: AC detector	Off, Low or High resistance trip levels
Input resistance	>100MΩ
Input leakage current	<±100nA (1nA typical).
Common mode rejection	>120dB, 47 - 63Hz
Series mode rejection	>60dB, 47 - 63Hz
Isolation channel-channel	42V peak maximum
Isolation to system	42V peak maximum

DO8 8-Channel Digital Output Card

The DO8 supports eight independently programmable channels, the output switches requiring external power supply. Each channel is current and temperature protected, foldback limiting occurring at about 100mA. The supply line is protected to limit total card current to 200mA. The eight channels are isolated from the system (but not from each other). To maintain isolation it is essential to use an independent and isolated PSU.	
Channel types	On/Off, Time proportioned
Channel supply (Vcs)	15Vdc to 30Vdc
Logic 1 voltage output	> (Vcs - 3V) (not in power limiting)
Logic 0 voltage output	< 1.2Vdc no-load, 0.9V typical
Logic 1 current output	100mA maximum (not in power limiting)
Minimum pulse time	20ms
Channel power limiting	Current limiting capable of driving short-circuit load
Terminal supply protection	Card supply is protected by 200mA self-healing fuse
Isolation (channel-channel)	N/A (channels share common connections)
Isolation to system	42V peak maximum

RL8 8-Channel Relay Output Card

The RL8 supports eight independently programmable channels. This module may only be fitted in slot 2 or 3, giving a maximum of 16 relays in a Mini8 loop controller. The Mini8 loop controller chassis must be grounded using the protective earth ground stud.		
Channel types	On/Off, Time proportioned	
Maximum contact voltage	264Vac	
Maximum contact current	2A ac	
Contact snubber	Fitted on module	
Minimum contact wetting	5Vdc, 10mA	
Minimum pulse time	220ms	
Isolation (channel-channel)	264V	230V nominal
Isolation to system	264V	

CT3 3-Channel Current-Transformer Input Card

Requires DO8 card to be fitted to allow the controller to be configured. The CT3 supports three independent channels designed for heater current monitoring. A scan block allows periodic test of nominated outputs to detect load changes due to issues with the heater.	
Channel types	A (current)
Factory set accuracy	better than ±2% of range
Current input range	0mA to 50mA RMS, 50/60Hz nominal
Transformer ratio	10/0.05 to 1000/0.05
Input load burden	1W
Isolation	None (provided by CT)

Load Failure Detection

Requires CT3 module	
Maximum number of loads	16 Time proportioned outputs
Maximum loads per CT	Six loads per CT input
Alarms	1 in 8 ‘Partial load failure’, Over current, SSR short-circuit, SSR open circuit
Commissioning	Automatic or manual
Measurement interval	1 sec - 60 sec

DI8 8-Channel Digital Input Card

The DI8 supports eight independent input channels.	
Input types	Logic (24Vdc)
Input logic 0 (off)	-28.8V to +5Vdc
Input logic 1 (on)	+10.8V to +28.8Vdc
Input current	2.5mA (approx) at 10.8V; 10mA maximum at 28.8V supply
Detectable pulse width	110ms minimum
Isolation channel-channel	42V peak maximum
Isolation to system	42V peak maximum

RT4 Resistance Thermometer Input Card

The RT4 supports four independently programmable and electrically isolated resistance input channels. Each channel may be connected as 2-wire, 3-wire or 4-wire and either Low or High resistance range.		
Channel types	Low resistance/Pt100	High resistance/Pt1000
Input range	0 to 420Ω, -242.02°C to +850°C (-404°F to +1562°F) for Pt100	0 to 4200Ω, -242.02°C to +850°C (-404°F to +1562°F) for Pt1000
Calibration accuracy	±0.1Ω ±0.1% of reading, 22Ω to 420Ω ±0.3°C ±0.1% of reading, -200°C to +850°C	±0.6Ω ±0.1% of reading, 220Ω to 4200Ω ±0.2°C ±0.1% of reading, -200°C to +850°C
Resolution	0.008Ω, 0.02°C	0.6Ω, 0.15°C
Measurement noise	0.016Ω, 0.04°C peak to peak, 1.6s channel filter 0.06Ω, 0.15°C peak to peak, no filter	0.2Ω, 0.05°C peak to peak, 1.6s channel filter 0.6Ω, 0.15°C peak to peak, no filter
Linearity accuracy	±0.02Ω, ±0.05°C	±0.2Ω, ±0.05°C
Temp coefficient	±0.002% of Ω reading per deg C ambient change relative to normal ambient 25°C	±0.002% of Ω reading per deg C ambient change relative to normal ambient 25°C
Lead resistance	22Ω max in each leg. Total resistance including leads is restricted to the 420Ω maximum limit. 3-wire connection assumed matched leads.	22Ω maximum in each leg. Total resistance including leads is restricted to the 4200Ω maximum limit. For the 3-wire connection it is assumed that the leads are matched.
Maximum bulb current	300μA	300μA
Isolation channel-channel	42V peak maximum	42Vpeak maximum
Isolation to system	42V peak maximum	42Vpeak maximum

AO8 8-Channel and AO4 4-Channel 4-20mA Output Card

The AO8 supports eight independently programmable and electrically isolated mA output channels for 4-20mA current-loop applications. The AO4 supports four channels to the same specification. The AO4 and AO8 modules may only be fitted in slot 4.	
Channel types	mA (current) Output
Output range	0-20mA, 360Ω load maximum
Setting accuracy	±0.5% of reading
Resolution	1 part in 10000 (1uA typical)
Isolation channel-channel	42V peak maximum
Isolation to system	42Vpeak maximum

Recipes

Recipes are a software orderable option	
Number of recipes	5
Tags	40 tags in total

Toolkit Blocks

User wires	Orderable options of 30, 60 120, 250 or 360. 360 User wires provide access to the Enhanced Toolkit blocks	
User values	32 real values 40 enhanced	
2-Input maths	24 blocks 32 enhanced	Add, subtract, multiply, divide, absolute difference, maximum, minimum, hot swap, sample and hold, power, square root, log, ln, exponential, switch
2-Input logic	24 blocks 40 enhanced	AND, OR, XOR, latch, equal, not equal, greater than, less than, greater than or equal to, less than or equal to
8-Input logic	4 blocks	AND, OR, XOR
8-Input multiple operator	4 blocks	Maximum, minimum, average. Input/Outputs to allow cascading of blocks
8-Input multiplexer	4 blocks 8 enhanced	Eight sets of eight values selected by input parameter
BCD input	2 blocks	Two decades (eight inputs giving 0 to 99).
Input monitor	2 blocks	Maximum, minimum, time above threshold
32 point linearization	2 blocks 8 enhanced	32-point linearization fit
Polynomial fit	2 blocks	Characterization by poly fit table
Switchover	1 block	Smooth transition between two input values
Timer blocks	8 blocks	OnPulse, OnDelay, OneShot, MinOn Time
Counter blocks	2 blocks	Up or down, directional flag
Totaliser blocks	2 blocks	Alarm at threshold value
Transducer scaling	2 blocks	Transducer auto-tare, calibration & comparison calibration
packbit	4 blocks 8 enhanced	Packs 16 individual bits into a 16 bit integer
unpackbit	4 blocks 8 enhanced	Unpacks a 16 bit integer into 16 individual bits

PID Control Loop Blocks (Superloop or Legacy Loop)

Number of loops	0, 4, 8 or 16 Loops (order options). 24 for Superloop
Control modes	On/Off, single PID, dual channel OP
Control outputs	Analog 4-20mA, time proportioned logic
Cooling algorithms	Linear, water, fan, or oil
Tuning	Three sets PID, one-shot auto-tune.
Auto manual control	Bumpless transfer or forced manual output available
Setpoint rate limit	Ramp in units per sec, per minute or per hour.
Output rate limit	Ramp in % change per second
Other features	Feedforward, input track, sensor break OP, loop break alarm, remote SP, two internal loop setpoints

Process Alarms

Number of alarms	64 alarms (configurable as analog, digital, or sensor break
Alarm types	Absolute high, absolute low, deviation high, deviation low, deviation band, sensor break, logic high, logic low, rising edge, falling edge, edge, falling rate of change, rising rate of change
Alarm modes	Latching or non-latching, blocking, time delay

Parameter Index

Parameter	Folder	Section	Parameter	Folder	Section
Ack	Analogue alarms	Alarm Parameters	CalEnable	Transducer scaling	Transducer Scaling Parameters
Ack	Digital alarms	Alarm Parameters	CalState	IO - Thermocouple input	Thermocouple Input Parameters
ActiveSet	Loop PID	PID Parameters	CalState	IO - PRT input	RT Input Parameters
ActiveLimitHigh	SuperLoop - Diagnostics	Diagnostics Parameters	CalState	Calibration	Calibration Parameters
ActiveLimitLow	SuperLoop - Diagnostics	Diagnostics Parameters	CalStatus	Transducer scaling	Transducer Scaling Parameters
ActiveLimitOPDelta	SuperLoop - Diagnostics	Diagnostics Parameters	CalType	Transducer scaling	Transducer Scaling Parameters
ActiveOut	Loop - main	Loop Parameters – Main	CalAdjust	Transducer scaling	Transducer Scaling Parameters
ActiveOvershootLimiting	SuperLoop - Diagnostics	Diagnostics Parameters	CalibrateCT1	IO - Current monitor	Parameter Configuration
ActiveSet	SuperLoop - PID	PID (TuneSets) Parameters	CalibrateCT2	IO - Current monitor	Parameter Configuration
Address	Comms - CC (config)	Configuration Communications Parameters (Main)	CalibrateCT3	IO - Current monitor	Parameter Configuration
Address	Comms - Modbus	Modbus Parameters	CascadeMode	SuperLoop - Main	Main Parameters
Address	Comms - Devicenet	DeviceNet Parameters	CascadeType	SuperLoop - Config	Config Parameters
Address	Comms - Ethernet	Ethernet Parameters	CascIn	Multi operators	Multiple Input Operator Block Parameters
AdditionalDiagnostics	SuperLoop - Diagnostics	Diagnostics Parameters	CascNumIn	Multi operators	Multiple Input Operator Block Parameters
AdvSeg	Programmer - Setup	Introduction to Setpoint Programmer	Ch1ControlType	Loop set up	Loop Set up
AlarmSP	Totaliser	Totalizer Parameters	Ch1ControlType	SuperLoop - Config	Config Parameters
AlarmAck	IO - Thermocouple input	Thermocouple Input Parameters	Ch1OnOffHyst	SuperLoop - PID	PID (TuneSets) Parameters
AlarmAck	IO - PRT input	RT Input Parameters	Ch1OnOffHyst2	SuperLoop - PID	PID (TuneSets) Parameters
AlarmDays	Input monitor	Input Monitor Parameters	Ch1OnOffHyst3	SuperLoop - PID	PID (TuneSets) Parameters
			Ch1OnOffHysteresis	Output function block	Output Feature
			Ch1Out	Output function block	Output Feature
			Ch1Output	SuperLoop - Output	Output Parameters
			Ch1PropBand	SuperLoop - PID	PID (TuneSets) Parameters
AlarmOut	Totaliser	Totalizer Parameters	Ch1PropBand2	SuperLoop - PID	PID (TuneSets) Parameters
AlarmTime	Input monitor	Input Monitor Parameters	Ch1PropBand3	SuperLoop - PID	PID (TuneSets) Parameters
AnAlarmStatus1	Alarm summary	Alarm Summary	Ch1TravelTime	SuperLoop - Output	Output Parameters
AnAlarmStatus2	Alarm summary	Alarm Summary	Ch2ControlType	Loop set up	Loop Set up
AnAlarmStatus3	Alarm summary	Alarm Summary	Ch2ControlType	SuperLoop - Config	Config Parameters
AnAlarmStatus4	Alarm summary	Alarm Summary	Ch2DeadBand	Output function block	Output Feature
AnyAlarm	Alarm summary	Alarm Summary	Ch2Deadband	SuperLoop - Output	Output Parameters
AtLimit	SuperLoop - Diagnostics	Diagnostics Parameters	Ch2Gain	Load	Load Parameters
Attenuation	Load	Load Parameters	Ch2OnOffHyst	SuperLoop - PID	PID (TuneSets) Parameters
AutoMan	Loop - main	Loop Parameters – Main	Ch2OnOffHyst2	SuperLoop - PID	PID (TuneSets) Parameters
AutoManual	SuperLoop - Main	Main Parameters	Ch2OnOffHyst3	SuperLoop - PID	PID (TuneSets) Parameters
AutoTuneEnable	Loop tune	Tune Parameters	Ch2OnOffHysteresis	Output function block	Output Feature
AutotuneActivate	SuperLoop - Autotune	Autotune Parameters	Ch2Out	Output function block	Output Feature
AverageOut	Multi operators	Multiple Input Operator Block Parameters	Ch2Output	SuperLoop - Output	Output Parameters
BackCalcPV	SuperLoop - Setpoint	Setpoint Parameters	Ch2PropBand	SuperLoop - PID	PID (TuneSets) Parameters
BackCalcSP	SuperLoop - Setpoint	Setpoint Parameters	Ch2PropBand2	SuperLoop - PID	PID (TuneSets) Parameters
Baud	Comms - CC (config)	Configuration Communications Parameters (Main)	Ch2PropBand3	SuperLoop - PID	PID (TuneSets) Parameters
Baud	Comms - Modbus	Modbus Parameters	Ch2TravelTime	SuperLoop - Output	Output Parameters
Baud	Comms - Devicenet	DeviceNet Parameters	Ch2TuneType	SuperLoop - Autotune	Autotune Parameters
BCDValue	BCD Input	BCD Parameters	ControlAction	SuperLoop - Config	Config Parameters
			CJCTemp	IO - Thermocouple input	Thermocouple Input Parameters
Block	Analogue alarms	Alarm Parameters	CJCType	IO - Thermocouple input	Thermocouple Input Parameters
Block	Digital alarms	Alarm Parameters	ClearCal	Transducer scaling	Transducer Scaling Parameters
Boundary	SuperLoop - PID	PID (TuneSets) Parameters	ClearLog	Instrument - Diagnostics	Instrument / Diagnostics
Boundary1-2	Loop PID	PID Parameters	ClearOverflow	Counter	Counter Parameters
Boundary2-3	Loop PID	PID Parameters	ClearStats	Instrument - Diagnostics	Instrument / Diagnostics
Boundary23	SuperLoop - PID	PID (TuneSets) Parameters	ClearLog	Alarm log	Instrument / Diagnostics
BoundaryHyst	SuperLoop - PID	PID (TuneSets) Parameters	ClearMemory	Access	Access Folder
BroadcastAddress	Comms - Modbus	Modbus Parameters	Clock	Counter	Counter Parameters
BroadcastEnabled	Comms - Modbus	Modbus Parameters	CntrlOverrun	Instrument - Diagnostics	Instrument / Diagnostics
BroadcastValue	Comms - Modbus	Modbus Parameters	Commission	IO - Current monitor	Parameter Configuration
CalActive	Transducer scaling	Transducer Scaling Parameters	CommissionStatus	IO - Current monitor	Parameter Configuration
CalBand	Transducer scaling	Transducer Scaling Parameters	CommsStack	Instrument - Diagnostics	Instrument / Diagnostics

Parameter	Folder	Section	Parameter	Folder	Section
DisplayHigh	IO - Thermocouple input	Thermocouple Input Parameters	FFType	SuperLoop - Feedforward	Feedforward Parameters
FilterTimeConstant	IO - Thermocouple input	Thermocouple Input Parameters	Inhibit	IO - Current monitor	Parameter Configuration
FilterTimeConstant	IO - PRT input	RT Input Parameters	Inhibit	Analogue alarms	Alarm Parameters
ForcedAuto	SuperLoop - Main	Main Parameters	Inhibit	Digital alarms	Alarm Parameters
ForcedManual	SuperLoop - Main	Main Parameters	Inhibit	Loop - main	Loop Parameters – Main
ForcedModesRecovery	SuperLoop - Config	Config Parameters	Inhibit	SuperLoop - Main	Main Parameters
ForcedOP	Output function block	Output Feature	InhibitOP	SuperLoop - Output	Output Parameters
Gain	Load	Load Parameters	InHigh	Switch over	Switch Over Parameters
GainScheduler	SuperLoop - PID	PID (TuneSets) Parameters	InHigh	Transducer scaling	Transducer Scaling Parameters
GlobalAck	Alarm summary	Alarm Summary	InHighLimit	Input linearisation	Input Linearization Parameters
High Limit	User values	User Value Parameters	InHighScale	Polynomial	Polynomial
HighLimit	Maths operators	Math Operator Parameters	InHold	SuperLoop - Diagnostics	Diagnostics Parameters
HighLimit	Mux8 operators	Multiple Input Operator Parameters	InInhibit	SuperLoop - Diagnostics	Diagnostics Parameters
HiOffset	IO - Thermocouple input	Thermocouple Input Parameters	InInvert	Input operators	Eight-Input Logic Operators
HiOffset	IO - PRT input	RT Input Parameters	InLow	Switch over	Switch Over Parameters
HiPoint	IO - Thermocouple input	Thermocouple Input Parameters	InLow	Transducer scaling	Transducer Scaling Parameters
HiPoint	IO - PRT input	RT Input Parameters	InLowLimit	Input linearisation	Input Linearization Parameters
Hold	Totaliser	Totalizer Parameters	InLowScale	Polynomial	Polynomial
Hold	SuperLoop - Main	Main Parameters	InManual	SuperLoop - Diagnostics	Diagnostics Parameters
			InPrimaryTune	SuperLoop - Diagnostics	Diagnostics Parameters
Hysteresis	Analogue alarms	Alarm Parameters	InputStatus	Multi operators	Multiple Input Operator Block Parameters
Ident	IO - Logic input	IO / FixedIO / D	InstType	Instrument - InstInfo	Instrument / Info
Ident	IO - Logic output	Logic Out Parameters	InStatus	Input monitor	Input Monitor Parameters
Ident	IO - Relay output	Relay Parameters	IntBal	SuperLoop - Main	Main Parameters
Ident	IO - Thermocouple input	Thermocouple Input Parameters	IntegralHold	SuperLoop - Main	Main Parameters
Ident	IO - PRT input	RT Input Parameters	IntegralOP	SuperLoop - Diagnostics	Diagnostics Parameters
Ident	IO - Analogue output	Analog Output	IntegralTime	SuperLoop - PID	PID (TuneSets) Parameters
Ident	IO - Fixed IO	Fixed IO	IntegralTime 1, 2, 3	Loop PID	PID Parameters
Ident	Comms - CC (config)	Configuration Communications Parameters (Main)	IntegralTime2	SuperLoop - PID	PID (TuneSets) Parameters
Ident	Comms - Modbus	Modbus Parameters	IntegralTime3	SuperLoop - PID	PID (TuneSets) Parameters
Ident	Comms - Devicenet	DeviceNet Parameters	Interval	IO - Current monitor	Parameter Configuration
Ident	Comms - EtherNet	Ethernet Parameters	IntHold	Loop - main	Loop Parameters – Main
IdleStack	Instrument - Diagnostics	Instrument / Diagnostics	InTrack	SuperLoop - Diagnostics	Diagnostics Parameters
In	Analogue alarms	Alarm Parameters	InTune	SuperLoop - Diagnostics	Diagnostics Parameters
In	Digital alarms	Alarm Parameters	InVal	Transducer scaling	Transducer Scaling Parameters
In	Timer	Timer Parameters	Invert	IO - Logic input	IO / FixedIO / D
In	Totaliser	Totalizer Parameters	Invert	IO - Logic output	Logic Out Parameters
In	Input monitor	Input Monitor Parameters	Invert	IO - Relay output	Relay Parameters
In	Input linearisation	Input Linearization Parameters	Invert	IO - Fixed IO	Fixed IO
In	Polynomial	Polynomial	Invert	Logic operators	Logic Operator Parameters
In1	BCD Input	BCD Parameters	IOType	IO - Thermocouple input	Thermocouple Input Parameters
In1	Logic operators	Logic Operator Parameters	IOType	IO - PRT input	RT Input Parameters
In1	Maths operators	Math Operator Parameters	IOType	IO - Analogue output	Analog Output
In1	Switch over	Switch Over Parameters	IOType	IO - Fixed IO	Fixed IO
In1 to In8	Multi operators	Multiple Input Operator Block Parameters	IOType	IO - Logic input	IO / FixedIO / D
In1 to In8	Mux8 operators	Multiple Input Operator Parameters	IOType	IO - Logic output	Logic Out Parameters
In1 to In8	Input operators	Eight-Input Logic Operators	IOType	IO - Relay output	Relay Parameters
In1 to In14	Input linearisation	Input Linearization Parameters	IPAddress 1	Comms - Ethernet	Ethernet Parameters
In1Mul	Maths operators	Math Operator Parameters	IPAddress 2	Comms - Ethernet	Ethernet Parameters
In2	BCD Input	BCD Parameters	IPAddress 3	Comms - Ethernet	Ethernet Parameters
In2	Logic operators	Logic Operator Parameters	IPAddress 4	Comms - Ethernet	Ethernet Parameters
In2	Maths operators	Math Operator Parameters			
In2	Switch over	Switch Over Parameters	Latch	Analogue alarms	Alarm Parameters
In2Mul	Maths operators	Math Operator Parameters	Latch	Digital alarms	Alarm Parameters
In3	BCD Input	BCD Parameters			
In4	BCD Input	BCD Parameters			
In5	BCD Input	BCD Parameters			
In6	BCD Input	BCD Parameters			
In7	BCD Input	BCD Parameters	LimitedHeadHigh	SuperLoop - Cascade	Cascade Scaling Parameters
In8	BCD Input	BCD Parameters	LimitedHeadHighType	SuperLoop - Cascade	Cascade Scaling Parameters

Parameter	Folder	Section
InAuto	SuperLoop - Diagnostics	Diagnostics Parameters
InCascade	SuperLoop - Diagnostics	Diagnostics Parameters
LineVoltage	SuperLoop - Cascade	Cascade Scaling Parameters
LinType	IO - Thermocouple input	Thermocouple Input Parameters
LinType	IO - PRT input	RT Input Parameters
LinType	Polynomial	Polynomial
LoOffset	IO - Thermocouple input	Thermocouple Input Parameters
LoOffset	IO - PRT input	RT Input Parameters
LoopBad	SuperLoop - Diagnostics	Diagnostics Parameters
LoopBreak	SuperLoop - Diagnostics	Diagnostics Parameters
LoopBreakDeltaPV	SuperLoop - Diagnostics	Diagnostics Parameters
LoopBreakTime	SuperLoop - Diagnostics	Diagnostics Parameters
LoopBreakTime 1, 2, 3	Loop PID	Setpoint Limits
LoopOutCh1	Load	Load Parameters
LoopType	SuperLoop - Config	Config Parameters
LowLimit	User values	User Value Parameters
LowLimit	Maths operators	Math Operator Parameters
LowLimit	Mux8 operators	Multiple Input Operator Parameters
MAC1	Comms - Ethernet	Ethernet Parameters
MAC2	Comms - Ethernet	Ethernet Parameters
MAC3	Comms - Ethernet	Ethernet Parameters
MAC4	Comms - Ethernet	Ethernet Parameters
MAC5	Comms - Ethernet	Ethernet Parameters
MAC6	Comms - Ethernet	Ethernet Parameters
ManualMode	Output function block	Output Feature
ManualOP	SuperLoop - Output	Output Parameters
ManualOutVal	Output function block	Output Feature
ManualReset	SuperLoop - PID	PID (TuneSets) Parameters
ManualReset 1, 2, 3	Loop PID	PID Parameters
ManualReset2	SuperLoop - PID	PID (TuneSets) Parameters
ManualReset3	SuperLoop - PID	PID (TuneSets) Parameters
ManualStepValue	SuperLoop - Output	Output Parameters
ManualTrack	Setpoint	Setpoint Limits
ManualTransfer	SuperLoop - Config	Config Parameters
Max	Input monitor	Input Monitor Parameters
MaxConTick	Instrument - Diagnostics	Instrument / Diagnostics
MaxOut	Multi operators	Multiple Input Operator Block Parameters
MaxLeakPh1	IO - Current monitor	Parameter Configuration
MaxLeakPh2	IO - Current monitor	Parameter Configuration
MaxLeakPh3	IO - Current monitor	Parameter Configuration
MeasuredVal	IO - Logic input	Logic In Parameters
MeasuredVal	IO - Thermocouple input	Thermocouple Input Parameters
MeasuredVal	IO - PRT input	RT Input Parameters
MeasuredVal	IO - Fixed IO	IO / FixedIO
MeasuredVal	IO - Logic output	Logic Out Parameters
MeasuredVal	IO - Relay output	Relay Parameters
Min	Input monitor	Input Monitor Parameters
MinOut	Multi operators	Multiple Input Operator Block Parameters
MinCPUFree	Instrument - Diagnostics	Instrument / Diagnostics
MinOnTime	IO - Logic output	Logic Out Parameters

Parameter	Folder	Section
LimitedHeadLow	SuperLoop - Cascade	Cascade Scaling Parameters
LimitedHeadLowType	SuperLoop - Cascade	Cascade Scaling Parameters
Mode	SuperLoop - Main	Main Parameters
Module1	IO - ModIDs	IO/ ModIDs
Module2	IO - ModIDs	IO/ ModIDs
Module3	IO - ModIDs	IO/ ModIDs
Module4	IO - ModIDs	IO/ ModIDs
Native	Comms - SCADA Table	Comms Table
NewAlarm	Alarm summary	Alarm Summary
NewCTAlarm	Alarm summary	Alarm Summary
Noise	Load	Load Parameters
NonLinearCooling	SuperLoop - Output	Output Parameters
NotRemote	SuperLoop - Diagnostics	Diagnostics Parameters
NumIn	Input operators	Eight-Input Logic Operators
NumIn	Multi operators	Multiple Input Operator Block Parameters
NumSets	Loop PID	PID Parameters
NumSets	SuperLoop - PID	PID (TuneSets) Parameters
NumValidIn	Multi operators	Multiple Input Operator Block Parameters
Offset	IO - Thermocouple input	Thermocouple Input Parameters
Offset	IO - PRT input	RT Input Parameters
Offset	Load	Load Parameters
Oper	Logic operators	Logic Operator Parameters
Oper	Input operators	Eight-Input Logic Operators
Oper	Maths operators	Math Operator Parameters
OPRateDeactivate	SuperLoop - Output	Output Parameters
OPRateDown	SuperLoop - Output	Output Parameters
OPRateUp	SuperLoop - Output	Output Parameters
Out	Analogue alarms	Alarm Parameters
Out	Digital alarms	Alarm Parameters
Out	Timer	Timer Parameters
Out	Input monitor	Input Monitor Parameters
Out	Logic operators	Logic Operator Parameters
Out	Input operators	Eight-Input Logic Operators
Out	Maths operators	Math Operator Parameters
Out	Mux8 operators	Multiple Input Operator Block Parameters
Out	Input linearisation	Input Linearization Parameters
Out	Polynomial	Polynomial
Out	Switch over	Switch Over Parameters
OutInvert	Input operators	Eight-Input Logic Operators
Out1 to Out14	Input linearisation	Input Linearization Parameters
OutHiLimit	Multi operators	Multiple Input Operator Block Parameters
OutHighLimit	Input linearisation	Input Linearization Parameters
OutHighScale	Polynomial	Polynomial
OutLoLimit	Multi operators	Multiple Input Operator Block Parameters
OutLowLimit	Input linearisation	Input Linearization Parameters
OutLowScale	Polynomial	Polynomial
OutputHigh	SuperLoop - PID	PID (TuneSets) Parameters
OutputHigh2	SuperLoop - PID	PID (TuneSets) Parameters
OutputHigh3	SuperLoop - PID	PID (TuneSets) Parameters
OutputHighLimit	Output function block	Output Feature
OutputHighLimit	SuperLoop - Output	Output Parameters
OutputLow	SuperLoop - PID	PID (TuneSets) Parameters
OutputLow2	SuperLoop - PID	PID (TuneSets) Parameters
OutputLow3	SuperLoop - PID	PID (TuneSets) Parameters
OutputLowLimit	Output function block	Output Feature
OutputHi 1, 2, 3	Loop PID	PID Parameters
OutputHigh Limit	Loop tune	Tune Parameters
OutputLo 1, 2, 3	Loop PID	PID Parameters
OutputLowLimit	Loop tune	Tune Parameters

Parameter	Folder	Section	Parameter	Folder	Section
PrimaryRemoteSV	SuperLoop - Primary PID	PrimaryPID (TuneSets) Parameters	RelHumid	Humidity	Humidity Parameters
PrimarySchedCBH	SuperLoop - Diagnostics	Diagnostics Parameters	RemOPH	Output function block	Output Feature
PrimarySchedCBL	SuperLoop - Diagnostics	Diagnostics Parameters	RemOPL	Output function block	Output Feature
RemoteInput	Loop PID	PID Parameters	SecondaryRSPTrimActivate	SuperLoop - Cascade	Cascade Scaling Parameters
RemoteLocal	SuperLoop - Main	Main Parameters	SecondaryRSPTrimHighLimit	SuperLoop - Cascade	Cascade Scaling Parameters
RemoteOPHighLimit	SuperLoop - Output	Output Parameters	SecondaryRSPTrimLowLimit	SuperLoop - Cascade	Cascade Scaling Parameters
RemoteOPLimsDeactivate	SuperLoop - Output	Output Parameters	SecondarySPType	SuperLoop - Cascade	Cascade Scaling Parameters
RemoteOPLowLimit	SuperLoop - Output	Output Parameters	SegmentsLeft	Instrument - Diagnostics	Instrument / Diagnostics
RemoteSV	SuperLoop - PID	PID (TuneSets) Parameters	Select	Mux8 operators	Multiple Input Operator Parameters
Reset	Counter	Counter Parameters	SelectIn	Switch over	Switch Over Parameters
Reset	Totaliser	Totalizer Parameters	SensorBreakMode	Output function block	Output Feature
Reset	Input monitor	Input Monitor Parameters	SerialNo	Instrument - InstInfo	Instrument / Info
Resolution	IO - Thermocouple input	Thermocouple Input Parameters	ServoToPV	Setpoint	Setpoint Parameters
Resolution	IO - PRT input	RT Input Parameters	Source	Comms - SCADA Table	Comms Table
Resolution	IO - Analogue output	Analog Output	SP1	Setpoint	Setpoint Limits
Resolution	Totaliser	Totalizer Parameters	SP1	SuperLoop - Setpoint	Setpoint Parameters
Resolution	Humidity	Humidity Parameters	SP2	Setpoint	Setpoint Limits
Resolution	Maths operators	Math Operator Parameters	SP2	SuperLoop - Setpoint	Setpoint Parameters
Resolution	Multi operators	Multiple Input Operator Block Parameters	SPHighLimit	Setpoint	Setpoint Limits
Resolution	Input linearisation	Input Linearization Parameters	SPHighLimit	SuperLoop - Setpoint	Setpoint Parameters
Resolution	Polynomial	Polynomial	SPIntBal	Setpoint	Setpoint Limits
Resolution	Load	Load Parameters	SPIntBal	SuperLoop - Setpoint	Setpoint Parameters
Resolution	User values	User Value Parameters	SPLowLimit	Setpoint	Setpoint Limits
RippleCarry	Counter	Counter Parameters	SPLowLimit	SuperLoop - Setpoint	Setpoint Parameters
RSP	SuperLoop - Setpoint	Setpoint Parameters	SPRateDeactivate	SuperLoop - Setpoint	Setpoint Parameters
RSPActivate	SuperLoop - Setpoint	Setpoint Parameters	SPRateDone	SuperLoop - Setpoint	Setpoint Parameters
RSPHighLimit	SuperLoop - Setpoint	Setpoint Parameters	SPRateDown	SuperLoop - Setpoint	Setpoint Parameters
RSPLowLimit	SuperLoop - Setpoint	Setpoint Parameters	SPRateServo	SuperLoop - Setpoint	Setpoint Parameters
RSPType	SuperLoop - Setpoint	Setpoint Parameters	SPRateUnits	SuperLoop - Setpoint	Setpoint Parameters
RstNewAlarm	Alarm summary	Alarm Summary	SPRateUp	SuperLoop - Setpoint	Setpoint Parameters
RstNewCTAlarm	Alarm summary	Alarm Summary	SPResolution	SuperLoop - Setpoint	Setpoint Parameters
Run	Totaliser	Totalizer Parameters	SPSelect	Setpoint	Setpoint Limits
SafeOPVal	Output function block	Output Feature	SPSelect	SuperLoop - Setpoint	Setpoint Parameters
Sbrk	Humidity	Humidity Parameters	SPSource	SuperLoop - Main	Main Parameters
SBrkAlarm	IO - Thermocouple input	Thermocouple Input Parameters	SPTrack	Setpoint	Setpoint Limits
SBrkAlarm	IO - PRT input	RT Input Parameters	SPTracksPSP	SuperLoop - Setpoint	Setpoint Parameters
SBrkAlarmStatus1	Alarm summary	Alarm Summary	SPTracksPV	SuperLoop - Setpoint	Setpoint Parameters
SBrkAlarmStatus2	Alarm summary	Alarm Summary	SPTracksRSP	SuperLoop - Setpoint	Setpoint Parameters
SBrkAlarmStatus3	Alarm summary	Alarm Summary	SPTrim	Setpoint	Setpoint Limits
SBrkAlarmStatus4	Alarm summary	Alarm Summary	SPTrim	SuperLoop - Setpoint	Setpoint Parameters
SbrkOp	Output function block	Output Feature	SPTrimHighLimit	Setpoint	Setpoint Limits
SbrkOutput	IO - Thermocouple input	Thermocouple Input Parameters	SPTrimHighLimit	SuperLoop - Setpoint	Setpoint Parameters
SbrkOutput	IO - PRT input	RT Input Parameters	SPTrimLowLimit	Setpoint	Setpoint Limits
SBrkType	IO - Thermocouple input	Thermocouple Input Parameters	SPTrimLowLimit	SuperLoop - Setpoint	Setpoint Parameters
SBrkType	IO - PRT input	RT Input Parameters	SPUnits	SuperLoop - Setpoint	Setpoint Parameters
SBrkValue	IO - PRT input	RT Input Parameters	Stage	Loop tune	Tune Parameters
SBrkValue	IO - Thermocouple input	Thermocouple Input Parameters	StageTime	Loop tune	Tune Parameters
SbyAct	IO - Logic output	Logic Out Parameters	StageTime	SuperLoop - Autotune	Autotune Parameters
SbyAct	IO - Relay output	Relay Parameters	Standby	Access	Access Folder
SbyAct	IO - Fixed IO	Fixed IO	StandbyModeRecoveryMode	SuperLoop - Config	Config Parameters
ScaleLow	Transducer scaling	Transducer Scaling Parameters	StartCal	Transducer scaling	Transducer Scaling Parameters
SchedCBH	SuperLoop - Diagnostics	Diagnostics Parameters	StartHighCal	Transducer scaling	Transducer Scaling Parameters
SchedCBL	SuperLoop - Diagnostics	Diagnostics Parameters	StartTare	Transducer scaling	Transducer Scaling Parameters
SchedCh1PB	SuperLoop - Diagnostics	Diagnostics Parameters	State	Loop tune	Tune Parameters
SchedCh2PB	SuperLoop - Diagnostics	Diagnostics Parameters	Status	IO - Thermocouple input	Thermocouple Input Parameters
SchedMR	SuperLoop - Diagnostics	Diagnostics Parameters	Status	IO - PRT input	RT Input Parameters
SchedTI	SuperLoop - Diagnostics	Diagnostics Parameters	Status	IO - Analogue output	Analog Output
SchedTD	SuperLoop - Diagnostics	Diagnostics Parameters	Status	Comms - Devicenet	DeviceNet Parameters
Scheduler	Loop PID	PID Parameters	Status	Logic operators	Logic Operator Parameters
SchedulerType	Loop PID	PID Parameters	Status	Maths operators	Math Operator Parameters
SecondaryLocalSP	SuperLoop - Cascade	Cascade Scaling Parameters	Status	Mux8 operators	Multiple Input Operator Parameters

Eurotherm Ltd
Faraday Close, Worthing
West Sussex, BN13 3PL
Phone: +44 (0) 1903 268500

www.eurotherm.com

HA033635 Issue 2

Contact your local sales representative

Publised July 2023

WATLOW®
Powered by Possibility