ICC402 Register Supplement

Table of Contents

Foreword 0

Part I Introduction 7

1 Register Types ... 7
- 32-bit Fixed Point (1 to 1023) ... 7
- 32-bit Floating Point (1025 to 2047) ... 8
- 24-bit Fixed Point (2049 to 4095) ... 8
- 16-bit Fixed Point (4097 to 8192) ... 8
- 8-bit Fixed Point (8193 to 16384) ... 8
- Text Registers (16385 to 20479) ... 8
- Macro Code Registers (32769 to 65536) ... 8

2 Memory Types ... 8

3 Communication Formats ... 9
- ASCII Mode ... 9
- Modbus Mode ... 11
- Character Frame Formats ... 12
- Command Response Time ... 13

4 ASCII Mode Format ... 14
- ASCII Read/Write Examples ... 16
- Multiple Write ... 16

5 Macro Compiling & Uploading ... 17

Part II Register List 19

1 ASCII Text Registers ... 19
- Register 16385, 16387, 16389 ... 23
- ASCII Characters for 14-Segment Display ... 23
- Print String - Register 16543 ... 24

2 Analog Output ... 26
- Analog Output 1 ... 28
- Analog Output 2 ... 28
- Analog Output 3 ... 28
- Analog Output 4 ... 29
- Analog Output 5 ... 29
- Analog Output 6 ... 29
- Analog Output 7 ... 30
- Analog Output 8 ... 30
- Analog Output Data Source Selection - Register 4379 to 4386 ... 30

3 Auxiliary Analog Inputs ... 30

4 Channel 1 ... 32
- CH1 Setup Registers ... 33

5 Channel 2 ... 33
- CH2 Setup Registers ... 35

6 Channel 3 ... 35
- CH3 Setup Registers ... 36

7 Channel 4 ... 37

CH4 Setup Registers ... 37
8 Channel 5 ... 38
CH5 Setup Registers ... 39
9 Channel 6 ... 39
CH6 Setup Registers ... 41
10 Channel 7 ... 41
CH7 Setup Registers ... 43
11 Channel 8 ... 43
CH8 Setup Registers ... 45
12 Channel 9 ... 45
CH9 Setup Registers ... 47
13 Channel 10 ... 47
CH10 Setup Registers ... 49
14 Channel 11 ... 49
CH11 Setup Registers ... 51
15 Clock ... 51
16 Codes ... 52
Calibration ... 53
Code Blanking ... 54
Code 1 ... 55
Select Data Source ... 56
Display Format Mode ... 57
Code 2 ... 58
Code 3 ... 58
Code 4 ... 59
Code 5 ... 60
Code 6 ... 61
Code 7 ... 62
Code 8 ... 63
Code 9 ... 63
17 Data Logging ... 64
Data Logging Concepts ... 67
Read Only Registers ... 69
Maximum Number Of Log Samples - Register 487 ... 72
Log Write Pointer - Register 489 ... 73
Log Read Pointer - Register 491 ... 73
Numeric Log Sample Values - Register 493 to 523 ... 73
Log Register Source - Registers 4417 to 4432 ... 74
Number Of Log Sample Reads - Register 4462 ... 74
Read Log Sample Data - Register 8443 to 8450 ... 74
Read Single Log Data at Log Read Pointer - Register 16553 ... 75
Read Log Data at Log Read Pointer - Register 16555 ... 77
External Data Logger ... 78
18 Digital ... 79
Control Inputs ... 87
Digital Read Only ... 88
I/O Module Type - Register 8430 ... 97
19 Edit Mode ... 98
Non-volatile Write Flag - Register 8222 ... 98
Edit State - Register 8223 ... 99

20 Display 101
Display Data Source Selection - Register 4371 to 4373 105
Peak/Valley Data Source Selection - Register 4374 to 4376 105
Brightness/Contrast - Register 8206 105
Display Options For Current Display - Register 8366 106
Alphanumeric Character for Current Display - Register 8392 106
Display Format 106
Octal Format 107
Text 108
21 Linearization 109
Linearization Table 1 109
Linearization Table 2 113
Linearization Table 3 116
Linearization Table 4 119
22 PID 122
PID 1 123
PID 2 125
PID 3 127
PID 4 129
PID 5 131
PID 6 133
PID Auto/Manual/Off Mode 134
Setpoint Filter 135
Setpoint Weighting Factor - P 136
Setpoint Weighting Factor - D 137
PID Output Saturation 138
PID Control Registers - Register 1057 to 1079 139
23 Digital Pins 140
24 Result 141
Result Setup Registers 141
25 Serial Port 142
Serial Port Settings - Registers 8207 - 8209 143
Serial Address - Registers 8211 - 8213 144
Start / End of String Characters - Registers 8226 to 8228 144
String Length - Registers 8230 to 8232 145
Serial Receive Count- Registers 8454 - 8456 145
Serial Transmit Count - Registers 8465 - 8467 145
Port 1 146
Serial Buffer Port 1 - Registers 12289 to 12544 147
ModBus Master 147
Port 2 152
Serial Buffer Port 2 - Registers 12545 to 12800 153
Port 3 154
Serial Buffer Port 3 - Registers 12801 - 13056 155
26 Setpoint 155
Setpoint Control Registers 156
Setpoint 3-digit Graphic 157
Setpoint Latch Mask - Register 4100 157
Relay Energize Functions 158
Relay De-energize Mask - Register 4101 160
Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160
Reset Destination - Register 4229 to 4240 161

Setpoint Data Source Selection - Register 4401 to 4406 ... **161**
Setpoint Tracking - Register 8261 to 8272 ... **161**
Delay Type - Register 8277 to 8288 ... **162**
Hysteresis Type ... **163**
Setpoint Trigger Functions - Register 8293 to 8304 ... **163**
Setpoint Status Flags - Register 4097 ... **164**
Setpoint Trigger Flags - Register 4098 ... **165**
Setpoint Blanking ... **165**
Setpoint 1 ... **166**
SP1 Setup ... 167
Setpoint 2 ... **168**
SP2 Setup ... 169
Setpoint 3 ... **170**
SP3 Setup ... 171
Setpoint 4 ... **172**
SP4 Setup ... 173
Setpoint 5 ... **174**
SP5 Setup ... 175
Setpoint 6 ... **176**
SP6 Setup ... 177
Setpoint 7 ... **178**
SP7 Setup ... 179
Setpoint 8 ... **180**
SP8 Setup ... 181
Setpoint 9 ... **182**
SP9 Setup ... 183
Setpoint 10 ... **184**
SP10 Setup ... 185
Setpoint 11 ... **186**
SP11 Setup ... 187
Setpoint 12 ... **188**
SP12 Setup ... 189

27 Smart Module ... **189**

Smart Module Setup ... **191**
Smart Input Expander ... **192**
Smart Module 2 Setup ... **193**
Smart Module 3 Setup ... **195**

28 Status Registers ... **197**

View Mode Blanking ... **201**
Error Status ... **202**
Status Switches ... **206**
Register 239 - Alarm Status ... **207**
Alarm Status 16 bit ... 207
Alarm Status Read ... 209
Alarm Status Write ... 211

29 Switches ... **215**

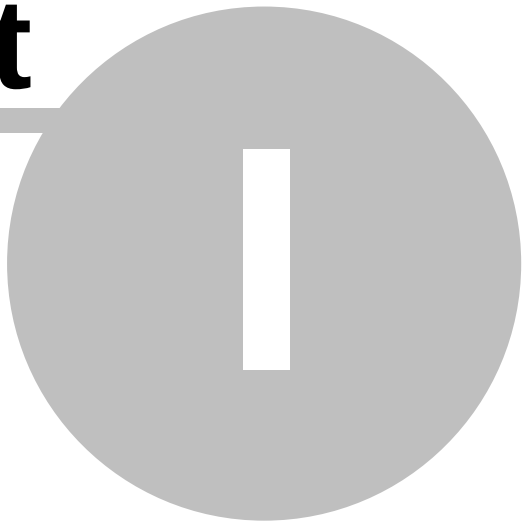
30 Timers ... **216**

31 Total ... **217**

Total 1 ... **219**
Total 2 ... **220**
Total 3 ... **221**
Total 4 ... **222**

Total 5 ..... **223**
Total 6 ..... **224**
K Factor **224**
Input Rate Value - Registers 4305 to 4310 ..... **224**
Totalizer Data Source Selection - Register 4391 to 4396 ..... **224**
Totalizer Time Period and Rollover - Register 8345 to 8350 ..... **225**
32 User **225**
Auxiliary **225**
Setup (Auxiliary) 227
Memory **227**
16-bit User Memory - Registers 5121 to 6144 230
8-bit User Memories - Registers 10241 to 12288 231
Text Memory **231**
Startup Text 232
Variables **232**
Bit Flags 233
Floating Point 233
Integers 234
Text Variables 235
33 Miscellaneous Registers **235**
Index **238**

Part



1 Introduction

This Introduction shows how the different register types are used and arranged for the ICC402 controller.

See Register Types 7

See also
Communication Formats 9

ASCII Mode Format 14

Macro Compiling & Uploading 17

Register List 19

1.1 Register Types

The controller uses 8, 16, 24, and 32-bit signed, unsigned, and floating point registers. There are two types of register used in the controller.

Configuration Register
A configuration register stores signal constants that change only when they are reprogrammed. For example, registers 1099 and 359 store channel 1 input signal scale and offset settings.

Working Register
A working register stores signal data that changes regularly due to variations in the input signal, as well as the processes carried out by the controller's functions on the input signal. For example, register 9 stores the processed data for the input signal after it has been processed through the channel 1 functions programmed into the controller.

See also
32-bit Fixed Point (1 to 1023) 7

32-bit Floating Point (1025 to 2047) 8

24-bit Fixed Point (2049 to 4095) 8

16-bit Fixed Point (4097 to 8192) 8

8-bit Fixed Point (8193 to 16384) 8

Text Registers (16385 to 20479) 8

Macro Code Registers (32769 to 65536) 8

1.1.1 32-bit Fixed Point (1 to 1023)

Register addresses 1 to 1023 are used for 32-bit fixed point addresses. To accommodate for Modbus usage of 32-point registers, only odd register addresses are used, providing a maximum of 511 registers.

1.1.2 32-bit Floating Point (1025 to 2047)

Register addresses 1025 to 2047 are used for 32-bit floating point addresses. All registers in this range are single precision floating point numbers that conform to the IEEE-754 standard format. To accommodate for Modbus usage of 32-point registers, only odd register addresses are used, providing a maximum of 511 registers.

1.1.3 24-bit Fixed Point (2049 to 4095)

Register addresses 2049 to 4095 are used for 24-bit fixed point addresses. To accommodate for Modbus usage of 24-point registers, only odd register addresses are used, providing a maximum of 1023 registers.

1.1.4 16-bit Fixed Point (4097 to 8192)

Register addresses 4097 to 8192 are used for 16-bit fixed point addresses. Both odd and even addresses in this range are used, providing a maximum of 4096 registers.

1.1.5 8-bit Fixed Point (8193 to 16384)

Register addresses 8193 to 16384 are used for 8-bit fixed point addresses. Both odd and even addresses in this range are used, providing a maximum of 8192 registers.

1.1.6 Text Registers (16385 to 20479)

Register addresses 16385 to 20479 are used for accessing text strings. Only odd addresses in this range are used, providing a maximum of 2047 text strings. Registers 16385 to 16525 are arranged so that they relate to registers numbers 1 to 141 with an offset of 16384 added to them.

See also
ASCII Text Registers 19

1.1.7 Macro Code Registers (32769 to 65536)

Register addresses 32769 to 65536 are 16-bit unsigned registers used for macro code storage. Both odd and even addresses in this range are used, providing a maximum of 32767 registers.

1.2 Memory Types

ICC402 series controllers use different types of memory to store register information. In some cases the data is stored in RAM only and is lost at power down (i.e. volatile memory). In other cases the data must be retained at power down so it must be saved in non volatile memory as well. There are also some restrictions on the way some memory types can be used so that their endurance specifications are not exceeded.

The table below shows the different memory types available in the ICC402 series controllers and the memory characteristics and restrictions which may apply.

Memory Type	Memory Characteristics
RAM	Random Access Memory. This memory is fast to access and is generally used for most working variables. It is volatile memory and it contents are not saved after a power down. Generally this memory is set to zero when the controller is turned on.
EEPROM	Electrically Erasable Programmable Read-Only Memory. This memory is slower to access and usually has a write time of between 5 - 10mS. It also has a limitation of 1x10^6 write cycles which must not be exceeded. There is no limit on the number of read cycles. EEPROM memory is non volatile and it's contents are retained even with no power applied. The controller uses this memory type for non volatile storage of data which is not accessed continuously by the operating system but is needed from time to time.
RAM/EEPROM	This memory type is made up of a combination of the two memory types shown above (i.e. RAM and EEPROM). It is probably the most common memory type used by the controller as it allows fast access and also non volatile storage. When writing to this type of memory from the macro, the RAM value is always updated and the EEPROM value is only updated if the |NON_VOLATILE_WRITE 98 flags is set just prior to the write instruction. This allows the macro to continuously write to a register without exceeding the maximum write cycle limit. When writing to this register via the serial port, both the RAM and EEPROM are updated so care must be taken not to exceed the maximum number of write cycles.
RAM/FLASH	This type of memory is similar to RAM/EEPROM in that it allows fast access and non volatile storage but it uses FLASH memory for the non volatile storage instead of EEPROM. FLASH memory is similar to EEPROM but is usually programmed in larger blocks of memory. This type of memory is used by the controller to store variables which are changing continuously and also need non volatile storage. A write to one of these registers from the macro or the serial port only changes the RAM value. This means that there are no limitations on how many times the register is written. When the power is removed from the controller it senses this and quickly copies the contents of these registers into FLASH memory. When power is restored, the contents of the FLASH memory are copied back into the RAM registers.
RAM/NVRAM	This type of memory uses RAM for fast access and non volatile RAM for data storage. The non volatile RAM is a real time clock device which uses a small battery to retain the contents of the memory during power down. The controller uses this type of memory to store time information.
SDcard	If the external data logging module is connected then some of the registers associated with data logging are also stored on the SD memory card. This card is similar to the EEPROM in that it is slower to access.
RAMinputModule	Smart input modules have an on board microprocessor which contains registers in RAM that can be accessed directly from the standard register list.
EEPROMinputModule	Some input modules carry an onboard EEPROM memory chip which holds calibration and setup data. This memory has the same features and restrictions as the EEPROM listed above. Some calibration registers in this memory are write protected by a lock key. Write access is disabled until the correct key value (170) has been written to key lock register (8439). This protects calibration data from inadvertent writes.

1.3 Communication Formats

See ASCII Mode 9

Modbus Mode 11

Character Frame Formats 12

Command Response Time 13

1.3.1 ASCII Mode

The ASCII mode is a simple communication protocol using the standard ASCII character set. This mode provides external communication between the controller and a PC allowing remote programming to be carried out. It was designed specifically so that it could be used with standard

terminal emulation software allowing the user to communicate with the controller without the need for specialised software. Because of this fact it does not include any error checking or CRC bytes and is intended for configuration of the controller over short distances.

ICC402 Series controllers use a serial communication channel to transfer data from the controller to another device. With serial communications, data is sent one bit at a time over a single communications line. The voltage is switched between a high and a low level at a predetermined transmission speed (baud rate) using ASCII encoding. Each ASCII character is transmitted individually as a byte of information (eight bits) with a variable idle period between characters. The idle period is the time between the receiving device receiving the stop bit of the last byte sent and the start bit of the next byte. The receiving device (for example a PC) reads the voltage levels at the same interval and then translates the switched levels back to an ASCII character. The voltage levels depend on the interface standard being used.

The following table lists the voltage level conventions used for RS-232 and RS-485. The voltage levels listed are at the receiver.

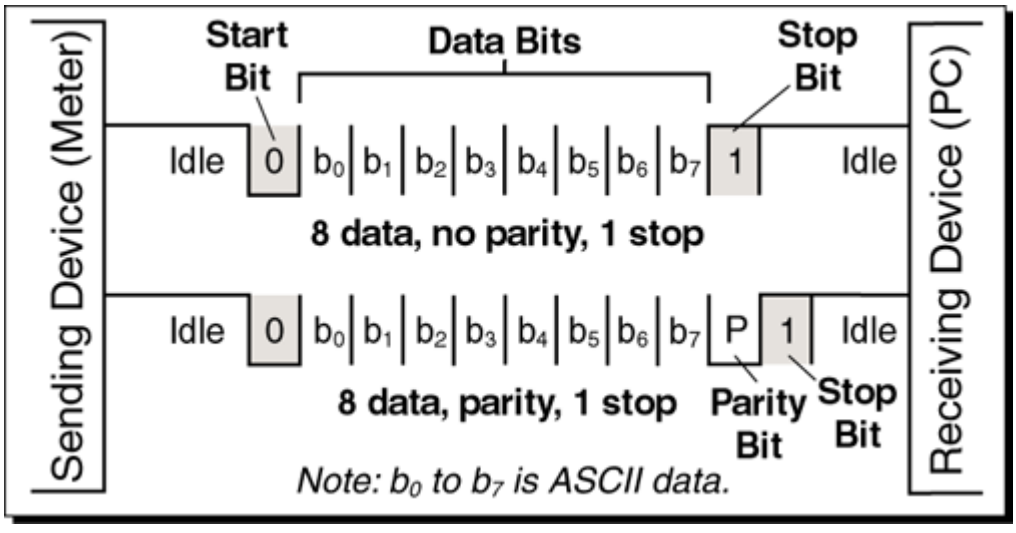
Interface Voltage Level Conventions

Logic	Interface State	RS-232	RS-485
1	Mark (idle)	TXD, RXD: -3 V to -15 V	a+b < -200 mV
0	Space (active)	TXD, RXD: +3 V to +15 V	a-b > +200 mV

Each ASCII character is **framed** with:

- A start bit.
- An optional error detection parity bit.
- And one or more ending stop bits.

For communication to take place, the data format and baud rate (transmission speed) must match that of the other equipment in the communication circuit. The following diagram shows the character frame formats used by the controller.



Character Frame Formats Diagram

See also
Modbus Mode 11

Character Frame Formats 12

Command Response Time 13

1.3.2 Modbus Mode

The Modbus mode uses the Modbus communication protocol to provide external communication between a ICC402 Series controller and a process device for monitoring, control, and automation purposes.

ICC402 Series controllers use Modbus RTU (Remote Terminal Unit) communication. This is an 8-bit binary transmission mode. The main advantage of this mode is that its greater character density allows better data throughput than ASCII for the same baud rate. Each message must be transmitted in a continuous stream.

ICC402 Series controllers can be configured as a Modbus slave device or a Modbus master. In the Modbus slave mode, the controller acts as a slave to a Modbus master (PC or PLC). Data transfers are based on registers and can only be initiated by the Modbus master. The Modbus master must be configured to accept this type of data. Once this is done, seamless communication between the Modbus master and Modbus slave can be initiated.

In Modbus master mode the controller initiates all communications to other Modbus slaves on the bus. On ICC402 Series controllers, the Modbus master mode must be used in conjunction with the MODBUS_MASTER_MACRO which defines which slave devices are accessed. (see ModBus Master 147). In Modbus master mode, ICC402 Series controllers can only access Modbus Holding registers (in the Modbus 40000 address range) and Input registers (in the Modbus 30000 address range) in external Modbus devices. Coils (20000) are not currently supported.

Modbus Command Summary
All of the registers currently incorporated in ICC402 Series controllers are accessed as "Holding Registers". Although strictly speaking this means that all of the registers are read/write registers, there are some exceptions to this rule. However the majority of these registers are read/write registers. There are no Discrete input registers, Coils or Input registers available in the ICC402.

The following Modbus function codes are supported by ICC402 Series controllers in slave mode;

Function Code	Description
3	Read holding registers
6	Write single holding register
16	Write multiple holding registers

The following Modbus function codes are supported by ICC402 Series controllers in master mode;

Function Code	Description
3	Read holding registers
4	Read input registers
6	Write single holding register
16	Write multiple holding registers

Addressing Convention

All registers numbers contained in this document refer to the original Modbus convention for addressing where register 1 is addressed as 0x0000 in the data packet.

For example, the register number for the Channel 1 processed data register is shown in this document as 9. In Modbus terms this is referred to as 40009. However the actual or direct address contained in the Modbus data packet would be 0x0008 (i.e. 1 count less).

Data Orientation
ICC402 Series controllers contain a combination of 8 bit, 16 bit, 24 bit, 32 bit integer, 32 bit floating point registers. The original Modbus protocol only allows for 16bit data registers so to access larger registers, multiple 16 bit registers are accessed. You will notice that all 24 and 32 bit register numbers in the ICC402 are odd addresses only so that they are spaced 2 register addresses apart from each other. This allows block reads of 32 bit registers to be carried out while still maintaining the correct register addresses.

In ICC402 Series controllers the data for 24 and 32 bit registers is transmitted LSW (Least Significant Word) first followed by the MSW (Most Significant Word). In Modbus master mode the user can specify the MB_SWAPPED option to access slave devices which use the alternate format. (See ModBus Master 147)

For example;
If register 40009 points to a 32 bit long which contains the value 12345678 (0xBC614E hex) then

1st pair of 8 bit bytes transmitted = 0x61 0x4E
2nd pair of 8 bit bytes transmitted = 0x00 0xBC

If the internal register is a 32 bit floating point number then the 1st two 8 bit values transmitted are the least significant 16 bits of the mantissa, while the next two 8 bit values transmitted give the sign, 8 bits of exponent and the most significant 7 bits of the mantissa.

For example;
If register pair 41025 points to a 32 bit float which contains the value –12.5 (0xC1480000 hex) then

1st pair of 8 bit bytes transmitted = 0x00 0x00
2nd pair of 8 bit bytes transmitted = 0XC1 0x48

8 bit Registers
In cases where the internal register is only an 8 bit value, the MSB will be set to zero (if the register is an 8 bit unsigned value) or to the sign (if the register is an 8 bit signed value).

ICC402 Series controllers also contain text string registers which can be up to 64 characters long. For text string registers each character is sent in the same order as it appears in the original text string (i.e. 2 characters per 16bit word).

Data packet Size
ICC402 Series controllers can transmit and receive Modbus data packets up to 255 bytes in length.

See also
Character Frame Formats 12

1.3.3 Character Frame Formats

Start Bit and Data Bits
Data transmission always begins with the start bit. The start bit signals the receiving device to prepare to receive data. One bit period later, the least significant bit of the ASCII encoded character

is transmitted, followed by the remaining data bits. The receiving device then reads each bit position as they are transmitted and, since the sending and receiving devices operate at the same transmission speed (baud rate), the data is read without timing errors.

Parity Bit
To prevent errors in communication, the sum of data bits in each character (byte) must be the same: either an odd amount or an even amount. The parity bit is used to maintain this similarity for all characters throughout the transmission. It is necessary for the parity protocol of the sending and receiving devices to be set before transmission. There are three options for the parity bit, it can be set to either:

- None – there is no parity.
- Odd – the sum of bits in each byte is odd.
- Even – the sum of bits in each byte is even.

After the start and data bits of the byte have been sent, the parity bit is sent. The transmitter sets the parity bit to 1 or 0 making the sum of the bits of the first character odd or even, depending on the parity protocol set for the sending and receiving devices.
As each subsequent character in the transmission is sent, the transmitter sets the parity bit to a 1 or a 0 so that the protocol of each character is the same as the first character: odd or even.

The parity bit is used by the receiver to detect errors that may occur to an odd number of bits in the transmission. However, a single parity bit cannot detect errors that may occur to an even number of bits. Given this limitation, the parity bit is often ignored by the receiving device. You set the parity bit of incoming data and also set the parity bit of outgoing data to odd, even, or none (mark parity).

Stop Bit
The stop bit is the last character to be transmitted. The stop bit provides a single bit period pause to allow the receiver to prepare to re-synchronize to the start of a new transmission (start bit of next byte). The receiver then continuously looks for the occurrence of the start bit.

See also
Command Response Time 13

1.3.4 Command Response Time

The controller uses half-duplex operation to send and receive data. This means that it can only send or receive data at any given time. It cannot do both simultaneously. The controller ignores commands while transmitting data, using RXD as a busy signal.

When the controller receives commands and data, after the first command string has been received, timing restrictions are imposed on subsequent commands. This allows enough time for the controller to process the command and prepare for the next command.

See the Timing Diagram below. At the start of the time interval **t1**, the sending device (PC) prints or writes the string to the serial port, initiating a transmission. During **t1** the command characters are under transmission and at the end of this period the controller receives the command terminating character. The time duration of time interval **t1** depends on the number of characters and baud rate of the channel:

```
t1 = (10 * # of characters) / baud rate
```

At the start of time interval **t2**, the controller starts to interpret the command, and when complete performs the command function.

After receiving a valid command string, the controller always indicates to the sending device when it

is ready to accept a new command. After a read command, the controller responds with the requested data followed by a carriage return (øDH) and a line feed (øAH) character. After receiving a write command, the controller executes the write command and then responds with a carriage return/line feed.

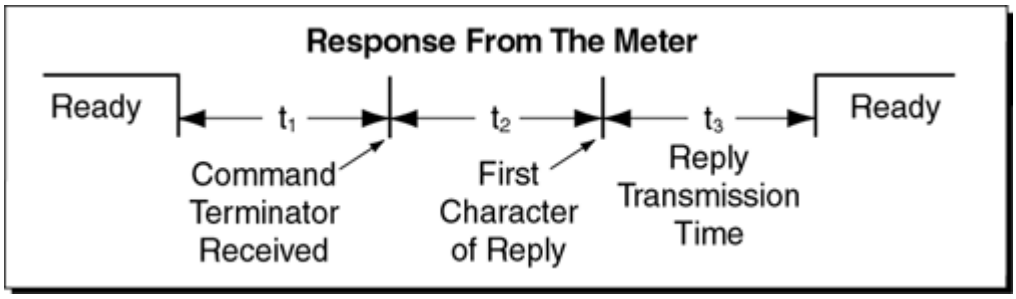
The sending device should wait for the carriage return/line feed characters before sending the next command to the controller.

If the controller is to reply with data, time interval **t2** is controlled by using the command terminating character: **$** or *****. The **$** terminating character results in a response time window of 50 milliseconds minimum and 100 milliseconds maximum. This allows enough time to release the sending driver on the RS-485 bus. Terminating the command line with the ***** symbol, results in a response time window (**t2**) of 2 milliseconds minimum and 50 milliseconds maximum. The faster response time of this terminating character requires that sending drivers release within 2 milliseconds after the terminating character is received.

At the start of time interval **t3**, the controller responds with the first character of the reply. As with **t1**, the time duration of **t3** depends on the number of characters and baud rate of the channel:

```
t3 = (10 * # of characters) / baud rate
```

At the end of **t3** the controller is ready to receive the next command.
The maximum throughput of the controller is limited to the sum of the times: **t1**, **t2**, **t3**.

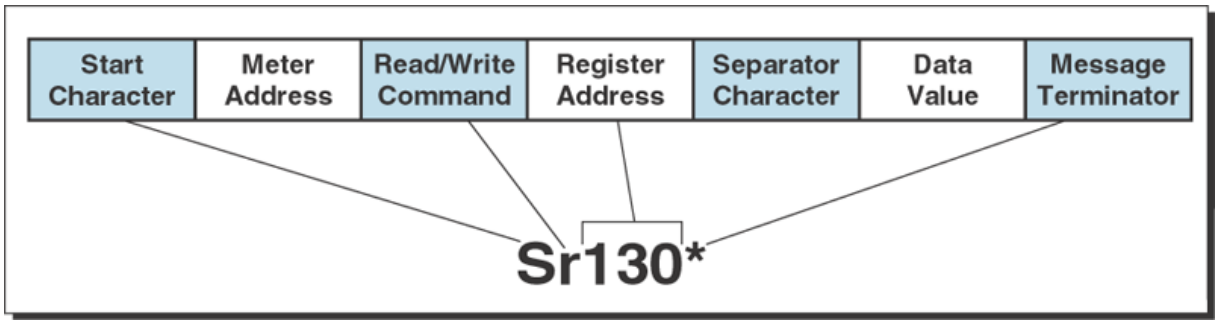


Timing Diagram

See also
ASCII Mode Format 14

1.4 ASCII Mode Format

Command String Construction
When sending commands to the controller using a Terminal emulation program, a string containing at least one command character must be constructed. A command string consists of the following characters and must be constructed in the order shown:



Command String Construction Diagram

Start Character
Use **S** or **s** for the start character of a command string. This must be the first character in the string.

Controller Address
Use an ASCII number from **0** to **255** for the controller address. If the character following the **start** character is not an ASCII number, then address **0** is assumed. All controllers respond to address **0**.

Read / Write Command
The next character must be an ASCII **R** or **r** for read, an ASCII **U** or **u** for an unformatted read, or an ASCII **W** or **w** for write. Any other character aborts the operation.

Register Address
The register address for the read/write operation is specified next. It can be an ASCII number from **1** to **65535** or, for special text registers, an ASCII letter from **A** to **Z** which is not case sensitive. If the address character is omitted in a read command, the controller always responds with the data value currently on the display. The register address must be specified for a write command.

Separator Character
After the register address in a write command, the next character must be something other than an ASCII number. This is used to separate the register address from the data value. It can be a **space** or a **,** (comma), or any other character except a **$** (dollar) or an ***** (asterisk).

Data Value
After the separator character, the data value is sent. It must be an ASCII number in the range of **-32766** to **32766**.

Message Terminator
The last character in the message is the message terminator and this must be either a **$** (dollar) or an ***** (asterisk).

If the **$** is used as a terminator, a minimum delay of 50 milliseconds is inserted before a reply is sent.

If the ***** is used as a terminator, a minimum delay of 2 milliseconds is inserted before a reply is sent.

NOTE: The **$** and the ***** characters must not appear anywhere else in the message string.

Controller Response
After the controller has completed a read or write instruction it responds by sending a carriage return/line feed (CR/LF) back to the host. If the instruction was a read command, the CR/LF follows the last character in the ASCII string. If it was a write command, a CR/LF is the only response sent back to the host. The host must wait for this before sending any further commands to the controller.

Unformatted Read
In the ASCII mode data is normally read as formatted data which includes decimal point and any text characters that may be selected to show display units. However it is also possible to read unformatted data (i.e. no decimal point and no text characters) by using a "**U**" or "**u**" in the read command instead of an "**R**" or "**r**". The following command sequence would be used to read unformatted data in channel 4 from controller address 3.

S3U15*

NOTE: There is no unformatted write command. When writing to fixed point registers, any decimal point and text characters are ignored.

See also
ASCII Read/Write Examples 16

Multiple Write 16

ASCII Text Registers 19

1.4.1 ASCII Read/Write Examples

Examples	Description
SR$	Read **display** value, 50 milliseconds delay, all controllers respond.
s15r$	Read **display** value, 50 milliseconds delay, controller address 15 responds.
SR57*	Read **Peak** value, 2 milliseconds delay, all controllers respond.
Sr8194*	Read **Code 1** setting, 2 milliseconds delay, all controllers respond.
s2w1 -10000$	Write **-10000** to the display register of controller address 2, 50 milliseconds delay.
SW16393 Chan_1$	Write ASCII text string **Chan_1** to channel 1 text register, 50 milliseconds.
s10w8206,7*	Change brightness to **7** on controller address 8206, 2 milliseconds delay.

See also
Multiple Write 16

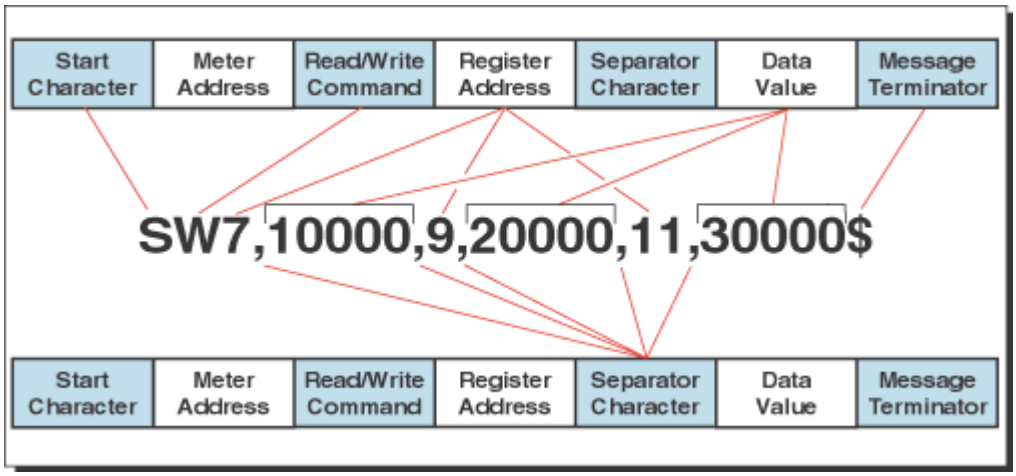
ASCII Text Registers 19

1.4.2 Multiple Write

This feature allows multiple registers to be written in a single ASCII command string. It is similar to a normal write command with the following differences:

- After the first data value, a separator character is inserted instead of the message terminator. Then the next register address is specified, followed by another separator character and the next data value. This procedure is repeated for each new register. The message terminator is added after the last data value in the string.

- Any number of registers can be written in the above manner provided the total length of the command string does not exceed 73 ASCII characters, including spaces and message terminator.



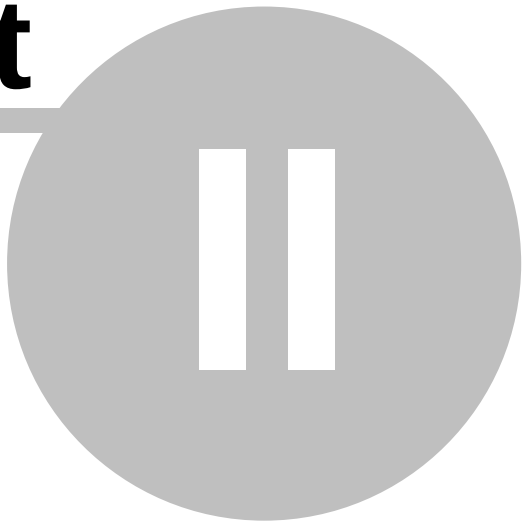
Two examples of the multiple write command

See also
ASCII Text Registers 19

1.5 Macro Compiling & Uploading

A macro is a set of commands that run automatically when the controller is powered up. Texmate NZ has a growing library of macros to suit a wide range of customer applications. Macros can be installed in the controller at the factory during initial programming or by the customer at some later date. Macros are written by Texmate NZ or the customer using either the TDS or the Configuration Utility program, both available for free download at www.texmate.co.nz.

Part



2 Register List

The registers described in the topics of this Help are available for controller configuration and macro programming purposes. Each register is identified in four ways, by:

Name
This is the name of the register and relates to its function.

Description
This describes the function of each register.

Symbol Type
Under Symbol Type, the following abbreviations identify the register type:

B_	The symbol B_ is followed by a number from 0 up to 31 and describes the register bit number.
F_32	The symbol F_32 identifies the register as a floating point 32-bit register (IEEE-754).
_R	The symbol _R identifies the register as a read only register and may be attached to another symbol. For example, B_0_R identifies this as bit 0 read only.
S_	The symbol S_ is followed by either 16, 24, or 32, identifying the register as a 16, 24, or 32-bit signed integer.
U_	The symbol U_ is followed by either 8, 16, or 32 identifying the register as an 8, 16, or 32-bit unsigned integer.
O_	The symbol O_ is followed by either 8, 16, or 32 identifying the register as an 8, 16, or 32-bit unsigned integer which is displayed in an octal format.
_W	The symbol _W identifies the register as a write only register.
L_	The symbol _L identifies the register as a text string that contains printable ASCII characters from 0x20 - 0x7a.

Register Number
This is the number that identifies the register in the controller.

See Also
Memory Types 8

2.1 ASCII Text Registers

The ICC402 Series controller incorporates a number of text registers for storage of ASCII text strings. These strings vary in length from 8 to 62 characters depending on the intended function of the text register. The USER_TEXT and TEXT_VARIABLE registers are intended for macro use to store user text data.

Text registers can be accessed in the ASCII serial mode via the Texmate NZ Configuration Utility or from the Macro. Access to text registers via Modbus mode is not currently supported.

Name	Description	Symbol Type	Register Number	Memory Type
DISPLAY_STRING	Write Display Text to primary display (for use with serial port in ASCII mode)	L_100	16385	RAM 8
DISPLAY_STRING2	Write Display Text to second display (for use with serial port in ASCII mode)	L_100	16387	RAM 8
DISPLAY_STRING3	Write Display Text to third display (for use with serial port in ASCII mode)	L_100	16389	RAM 8
RESULT_TEXT	Text display for Result	L_8	16391	EEPROM 8
CHANNEL1_TEXT	Text display for Channel 1	L_8	16393	EEPROM 8
CHANNEL2_TEXT	Text display for Channel 2	L_8	16395	EEPROM 8
CHANNEL3_TEXT	Text display for Channel 3	L_8	16397	EEPROM 8
CHANNEL4_TEXT	Text display for Channel 4	L_8	16399	EEPROM 8
CHANNEL5_TEXT	Text display for Channel 5	L_8	16401	EEPROM 8
CHANNEL6_TEXT	Text display for Channel 6	L_8	16403	EEPROM 8
CHANNEL7_TEXT	Text display for Channel 7	L_8	16405	EEPROM 8

PEAK1_TEXT	Text display for Peak	L_8	16441	EEPROM 8
VALLEY1_TEXT	Text display for Valley	L_8	16443	EEPROM 8
PEAK2_TEXT	Text display for Peak 2	L_8	16445	EEPROM 8
VALLEY2_TEXT	Text display for Valley 2	L_8	16447	EEPROM 8
PEAK3_TEXT	Text display for Peak 3	L_8	16449	EEPROM 8
VALLEY3_TEXT	Text display for Valley 3	L_8	16451	EEPROM 8
AUX1_TEXT	Text display for Auxiliary 1	L_8	16463	EEPROM 8
AUX2_TEXT	Text display for Auxiliary 2	L_8	16465	EEPROM 8
AUX3_TEXT	Text display for Auxiliary 3	L_8	16467	EEPROM 8
AUX4_TEXT	Text display for Auxiliary 4	L_8	16469	EEPROM 8
AUX5_TEXT	Text display for Auxiliary 5	L_8	16471	EEPROM 8
AUX6_TEXT	Text display for Auxiliary 6	L_8	16473	EEPROM 8
AUX7_TEXT	Text display for Auxiliary 7	L_8	16475	EEPROM 8
AUX8_TEXT	Text display for Auxiliary 8	L_8	16477	EEPROM 8
AUX9_TEXT	Text display for Auxiliary 9	L_8	16479	EEPROM 8
AUX10_TEXT	Text display for Auxiliary 10	L_8	16481	EEPROM 8
AUX11_TEXT	Text display for Auxiliary 11	L_8	16483	EEPROM 8
AUX12_TEXT	Text display for Auxiliary 12	L_8	16485	EEPROM 8
AUX13_TEXT	Text display for Auxiliary 13	L_8	16487	EEPROM 8
AUX14_TEXT	Text display for Auxiliary 14	L_8	16489	EEPROM 8
AUX15_TEXT	Text display for Auxiliary 15	L_8	16491	EEPROM 8
AUX16_TEXT	Text display for Auxiliary 16	L_8	16493	EEPROM 8

SETPOINT1_TEXT	Text display for Setpoint 1	L_8	16495	EEPROM 8
SETPOINT2_TEXT	Text display for Setpoint 2	L_8	16497	EEPROM 8
SETPOINT3_TEXT	Text display for Setpoint 3	L_8	16499	EEPROM 8
SETPOINT4_TEXT	Text display for Setpoint 4	L_8	16501	EEPROM 8
SETPOINT5_TEXT	Text display for Setpoint 5	L_8	16503	EEPROM 8
SETPOINT6_TEXT	Text display for Setpoint 6	L_8	16505	EEPROM 8
SETPOINT7_TEXT	Text display for Setpoint 7	L_8	16507	EEPROM 8
SETPOINT8_TEXT	Text display for Setpoint 8	L_8	16509	EEPROM 8
SETPOINT9_TEXT	Text display for Setpoint 9	L_8	16511	EEPROM 8
SETPOINT10_TEXT	Text display for Setpoint 10	L_8	16513	EEPROM 8
SETPOINT11_TEXT	Text display for Setpoint 11	L_8	16515	EEPROM 8
SETPOINT12_TEXT	Text display for Setpoint 12	L_8	16517	EEPROM 8
OVER_TEXT	Text display for over range	L_8	16539	EEPROM 8
UNDER_TEXT	Text display for under range	L_8	16541	EEPROM 8
PRINT_STRING	Print String	L_62	16543	EEPROM 8
SINGLE_LOG	This register reads or write a single data log sample if data logging is enabled.	L_8	16553	EEPROM 8
STARTUP_TEXT_LINE 1 232	Non-volatile 16 character text string for user defined startup text on line 1 (1602 LCD display only).	L_16	16545	EEPROM 8
STARTUP_TEXT_LINE 2 232	Non-volatile 16 character text string for user defined startup text on line 2 (1602 LCD display).	L_16	16547	EEPROM 8
METER_TYPE		L_8_R	16565	FLASH 8
USER_TEXT 1 to USER_TEXT64	Are all non-volatile 30 character text strings for user defined text storage, using only odd number register addresses from 16567 to 16693.	L_30	16567 to 16693	EEPROM 8
TEXT_VARIABLE1	30 character text string variable in RAM.	L_30	16897	RAM 8
TEXT_VARIABLE2	30 character text string variable in RAM.	L_30	16899	RAM 8
TEXT_VARIABLE3	30 character text string variable in RAM.	L_30	16901	RAM 8
TEXT_VARIABLE4	30 character text string variable in RAM.	L_30	16903	RAM 8
TEXT_VARIABLE5	30 character text string variable in RAM.	L_30	16905	RAM 8
TEXT_VARIABLE6	30 character text string variable in RAM.	L_30	16907	RAM 8
TEXT_VARIABLE7	30 character text string variable in RAM.	L_30	16909	RAM 8

See also
Register 16385 23

ASCII Characters for 14-segment 23

Print String - Register 16543 24

2.1.1 Register 16385, 16387, 16389

Registers 16385 to 16389 are used to write a text string directly to the controller display via the serial port in ASCII mode. Text strings are only displayed while the controller is in it's normal operating mode, known as the **operational display**. Text strings are ignored when the controller is in any edit or view mode.

A scrolling text string of up to 100 characters long can be sent to the display. The string is scrolled through once and then the display returns to the operational display. The special ~ (tilde) character is used to insert an instantaneous register value into the text string. See note on register 16543 24 (print string) for more information on this feature.

To send text to the primary display, the following commands can be used:

This Text String	Displays This
SW16385	This text string scrolls across the display_____ *
SW16385	Setpoint 1 = ~6 Volts $

Register 16385 is available on all models of ICC402 Series controllers. Registers 16387 and 16389 are also available on controller versions that have multiple displays (such as the DI-602, DI-802, DI-503, etc.) A write to register 16387 scrolls a text message on the second line of the display and a write to register 16389 scrolls a message on the third line of the display.

A read of registers 16385 to 16389 results in the text **DISP** being displayed.

2.1.2 ASCII Characters for 14-Segment Display

The following characters can be selected for the last digit by selecting and entering the appropriate register control value.

Character	Register Control Value (decimal)	Character	Register Control Value (decimal)
Space	32	@	64
!	33	A	65
"	34	B	66
#	35	C	67
$	36	D	68
%	37	E	69
&	38	F	70
'	39	G	71
(	40	H	72
)	41	I	73
*	42	J	74
+	43	K	75
,	44	L	76
-	45	M	77
.	46	N	78
/	47	O	79
0	48	P	80
1	49	Q	81
2	50	R	82
3	51	S	83
4	52	T	84
5	53	U	85
6	54	V	86
7	55	W	87
8	56	X	88
9	57	Y	89
:	58 (displays as decimal point)	Z	90
;	59	[	91 (displays same as **C**)
<	60	\	92
=	61	]	93
>	62	^	94
?	63	_	95

2.1.3 Print String - Register 16543

When setup in the print mode, the controller can print data from any register directly to a serial printer, or to a PC where it can be imported into a spreadsheet.

Register 16543 is a special register that allows you to specify the text and data stored in specific registers to be printed out when a print command is issued by the controller while in the print mode. Through the serial port, register 16543 can be either written to or read from using a terminal program

on a PC.

Writing To Register 16543
Writing to register 16543 tells the controller to print the data stored in one or more of the controller's registers when the print command is issued. To get the controller to print, the printer must be connected to the controller through the serial port and the controller must be programmed to run in the **print mode**. The data to be printed depends on how the controller has been programmed.

For example, to display a flow rate and total. The total length of a write string can be up to 62 ASCII characters long. See Printing Restrictions.

Reading From Register 16543
Reading from register 16543 allows you to check your settings prior to removing the PC from the serial port and connecting to a printer. Register 16543 can be read in the normal manner: SR16543$.

Example of Writing To Register 16543
The following example shows a write to register 16543 with the controller setup to display flow rate and total flow of channel 1.

swx Rate = ~1 (add carriage return and line feed)
Total = ~37$

The above write to register 16543 means the following:

sw16543: Start writing to register 16543.
Rate =: Tells the controller to print the word Rate =.
~1: Tells the controller to print the current flow rate (display data), held in register 1, after the word Rate =.
Total =: Tells the controller to print the word Total =.
~37: Tells the controller to print the current total flow (stored data), held in register 37, after the word Flow =.

The printer would then print, for example, the following:

Rate = 2000
Total = 25000

This means that the current flow rate is 2000 and the total flow at this point is 25000.

Example of Reading From Register 16543
Having written the above example to the controller, to check the contents of register 16543 using the terminal program through the PC, type the following:

sr16543$

The following is shown on the PC screen:
Rate = ~1
Total = ~37$

Printing Restrictions
When printing, any alphanumeric ASCII character can be used within the following restrictions:

- The **$** and ***** characters are reserved for the terminating character at the end of the string and cannot be used as part of the text string.

- The total string length must be no greater than 62 bytes long. This includes spaces, tabs, carriage returns, line feeds, and the terminating character. There must be a separator space between the register address 16543 and the start of the string. **Note, this separator space does not have to be included in text string length calculations**.

- Any number following a **~** (tilde) character is interpreted as a register address. During a printout the register's current value is printed out in this position.

- The ASCII character \ is treated as a special character in the print string. When a \ is encountered, a ***** is printed in its place (***** is reserved as a terminating character and normally can not appear anywhere in the text string). This allows the print output of one controller to be connected to another controller that is operating in the ASCII mode.

  For example, if the print string reads:

  swx sw3 ~11\ sw4 ~13\ sw6 ~1\$

  The printer prints the following:

  sw3 (current register value)***** **sw4** (current register value)*****
  sw6 (current register value)*****

Up to seven different registers can be specified in one text string, provided that the total string length is no greater than 62 bytes long and the total length of the resulting printout is less than 100 bytes long (including time stamp if selected).

For example, the following tab delimited output could be specified to input display data, processed result, processed channel 1, processed channel 2, peak, valley, and total, directly into a spreadsheet:

swx ~1(tab)~7(tab)~9(tab)~11(tab)~57(tab)~59(tab)~37$

When calculating the length of the printout, an allowance of 7 bytes for each register address should be used, plus any extra text or separating characters such as tabs or spaces.

NOTE: As a new line is usually represented by a carriage return and a line feed, 2 bytes should be added for each new line in text string length calculations.

2.2 Analog Output

Analog output registers contain the calibration and scaling information and scaled output data for each individual analog output.

The available analog output signals can be used for the following applications:

- To drive remote process instruments.
- As an isolated 4 to 20 mA or 0 to 10 V DC signal for further processing via a PLC.
- As a 4 to 20 mA PID output for process control applications (e.g. temperature control).
- As a manual loader output to manually control the operation of actuated plant equipment such as valves, dampers, hydraulic and pneumatic cylinders and slides.

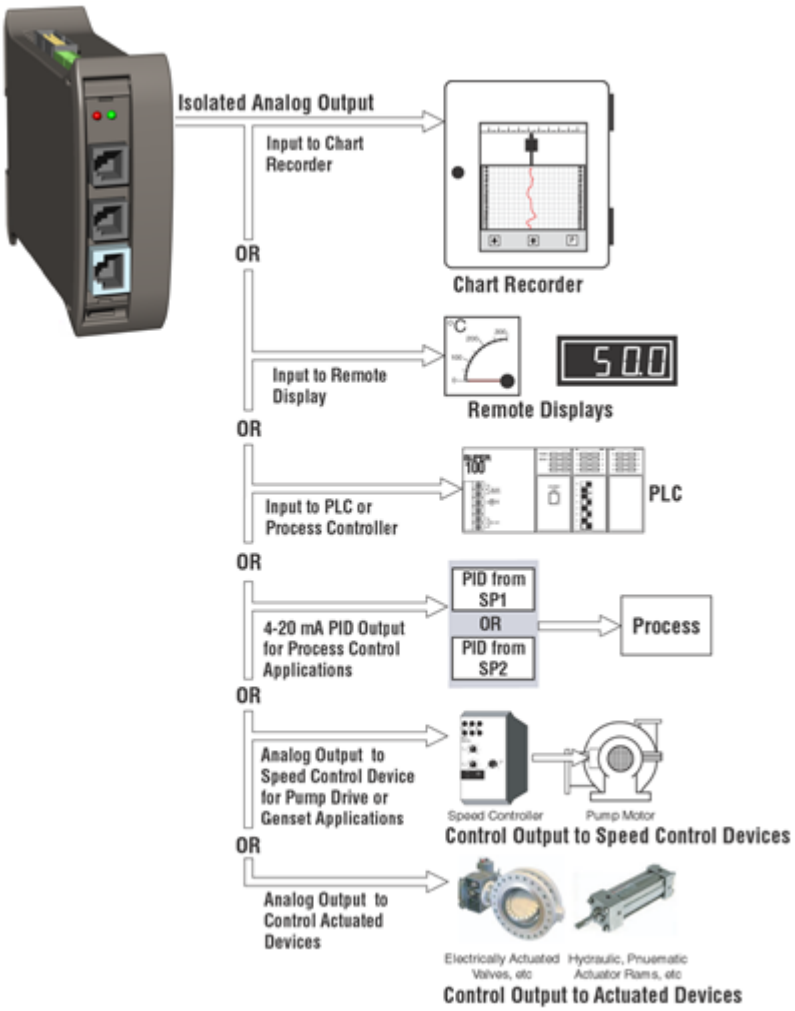
NOTE: Analog output 5 to 8 are not available at present.

The analog output can be programmed over:

- The full scale range of the controller and the selected input module.
- Any part within the full scale range of the input channel.
- Any part within the full scale range of the linearized input signal.
- The proportional band of the selected PID register.

The source of data for the analog output can be selected from any processed controller input signal.

The span range of the analog output can be as small as 100 counts between the low and high analog output signal.



Once calibrated, the span range of the analog output can be easily changed (rescaled) without having to recalibrate the output. The low and high analog signal output values (mA or volts) follow the new span range.

See also

Analog Output 1 28

Analog Output 2 28

Analog Output 3 28

Analog Output 4 29

Analog Output 5 29

Analog Output 6 29

Analog Output 7 30

Analog Output 8 30

2.2.1 Analog Output 1

Name	Description	Symbol Type	Register Number	Memory Type
D2A_AOP1_CAL_HIGH	16-bit register. Holds the **high** end calibration value for analog output 1.	S_16	4169	RAM/EEPROM 8
D2A_AOP1_CAL_LOW	16-bit register. Holds the **low** end calibration value for analog output 1.	S_16	4157	RAM/EEPROM 8
D2A_AOP1_CAL_FULL_SCALE	32-bit register. Holds the **full scale** value for analog output 1.	S_32	405	RAM/EEPROM 8
D2A_AOP1_ZERO	32-bit register. Holds the **zero scale** value for analog output 1.	S_32	381	RAM/EEPROM 8
DATA_SOURCE_ANALOG1 30	8-bit register. Holds the register number of the **data source** for analog output 1.	U_8	4379 30	RAM/EEPROM 8
Read Only				
ANALOG_OUTPUT1	Read Only 16-bit register. Holds the **scaled output data** for analog output 1.	S_16	4145	RAM 8

2.2.2 Analog Output 2

Name	Description	Symbol Type	Register Number	Memory type
D2A_AOP2_CAL_HIGH	16-bit register. Holds the **high** end calibration value for analog output 2.	S_16	4170	RAM/EEPROM 8
D2A_AOP2_CAL_LOW	16-bit register. Holds the **low** end calibration value for analog output 2	S_16	4158	RAM/EEPROM 8
D2A_AOP2_CAL_FULL_SCALE	32-bit register. Holds the **full scale** value for analog output 2.	S_32	407	RAM/EEPROM 8
D2A_AOP2_ZERO	32-bit register. Holds the **zero scale** value for analog output 2.	S_32	383	RAM/EEPROM 8
DATA_SOURCE_ANALOG2 30	8-bit register. Holds the register number of the **data source** for analog output 2.	U_8	4380 30	RAM/EEPROM 8
Read Only				
ANALOG_OUTPUT2	Read Only 16-bit register. Holds the **scaled output data** for analog output 2.	S_16	4146	RAM 8

2.2.3 Analog Output 3

Name	Description	Symbol Type	Register Number	Memory Type
D2A_AOP3_CAL_HIGH	16-bit register. Holds the **high** end calibration value for analog output 3.	S_16	4171	RAM/EEPROM 8
D2A_AOP3_CAL_LOW	16-bit register. Holds the **low** end calibration value for analog output 3.	S_16	4159	RAM/EEPROM 8
D2A_AOP3_CAL_FULL_SCALE	32-bit register. Holds the **full scale** value for analog output 3.	S_32	409	RAM/EEPROM 8
D2A_AOP3_ZERO	32-bit register. Holds the **zero scale** value for analog output 3.	S_32	385	RAM/EEPROM 8
DATA_SOURCE_ANALOG3 30	8-bit register. Holds the register number of the **data source** for analog output 3.	U_8	4381 30	RAM/EEPROM 8
Read Only				

2.2.4 Analog Output 4

Name	Description	Symbol Type	Register Number	Memory Type
D2A_AOP4_CAL_HIGH	16-bit register. Holds the **high** end calibration value for analog output 4.	S_16	4172	RAM/EEPROM 8
D2A_AOP4_CAL_LOW	16-bit register. Holds the **low** end calibration value for analog output 4.	S_16	4160	RAM/EEPROM 8
D2A_AOP4_CAL_FULL_SCALE	32-bit register. Holds the **full scale** value for analog output 4.	S_32	411	RAM/EEPROM 8
D2A_AOP4_ZERO	32-bit register. Holds the **zero scale** value for analog output 4.	S_32	387	RAM/EEPROM 8
DATA_SOURCE_ANALOG4 30	8-bit register. Holds the register number of the **data source** for analog output 4.	U_8	4382 30	RAM/EEPROM 8
Read Only				
ANALOG_OUTPUT4	Read Only 16-bit register. Holds the **scaled output data** for analog output 4.	S_16	4148	RAM 8

2.2.5 Analog Output 5

Name	Description	Symbol Type	Register Number	Memory Type
D2A_AOP5_CAL_HIGH	16-bit register. Holds the **high** end calibration value for analog output 5.	S_16	4173	RAM/EEPROM 8
D2A_AOP5_CAL_LOW	16-bit register. Holds the **low** end calibration value for analog output 5.	S_16	4161	RAM/EEPROM 8
D2A_AOP5_CAL_FULL_SCALE	32-bit register. Holds the **full scale** value for analog output 5.	S_32	413	RAM/EEPROM 8
D2A_AOP5_ZERO	32-bit register. Holds the **zero scale** value for analog output 5.	S_32	389	RAM/EEPROM 8
DATA_SOURCE_ANALOG5 30	8-bit register. Holds the register number of the **data source** for analog output 5.	U_8	4383 30	RAM/EEPROM 8
Read Only				
ANALOG_OUTPUT5	Read Only 16-bit register. Holds the **scaled output data** for analog output 5.	S_16	4149	RAM 8

2.2.6 Analog Output 6

Name	Description	Symbol Type	Register Number	Memory Type
D2A_AOP6_CAL_HIGH	16-bit register. Holds the **high** end calibration value for analog output 6.	S_16	4174	RAM/EEPROM 8
D2A_AOP6_CAL_LOW	16-bit register. Holds the **low** end calibration value for analog output 6.	S_16	4162	RAM/EEPROM 8
D2A_AOP6_CAL_FULL_SCALE	32-bit register. Holds the **full scale** value for analog output 6.	S_32	415	RAM/EEPROM 8
D2A_AOP6_ZERO	32-bit register. Holds the **zero scale** value for analog output 6.	S_32	391	RAM/EEPROM 8
DATA_SOURCE_ANALOG6 30	8-bit register. Holds the register number of the **data source** for analog output 6.	U_8	4384 30	RAM/EEPROM 8
Read Only				

2.2.7 Analog Output 7

Name	Description	Symbol Type	Register Number	Memory Type
D2A_AOP7_CAL_HIGH	16-bit register. Holds the **high** end calibration value for analog output 7.	S_16	4175	RAM/EEPROM 8
D2A_AOP7_CAL_LOW	16-bit register. Holds the **low** end calibration value for analog output 7.	S_16	4163	RAM/EEPROM 8
D2A_AOP7_CAL_FULL_SCALE	32-bit register. Holds the **full scale** value for analog output 7.	S_32	417	RAM/EEPROM 8
D2A_AOP7_ZERO	32-bit register. Holds the **zero scale** value for analog output 7.	S_32	393	RAM/EEPROM 8
DATA_SOURCE_ANALOG7 30	8-bit register. Holds the register number of the **data source** for analog output 7.	U_8	4385 30	RAM/EEPROM 8
Read Only				
ANALOG_OUTPUT7	Read Only 16-bit register. Holds the **scaled output data** for analog output 7.	S_16	4151	RAM 8

2.2.8 Analog Output 8

Name	Description	Symbol Type	Register Number	Memory Type
D2A_AOP8_CAL_HIGH	16-bit register. Holds the **high** end calibration value for analog output 8.	S_16	4176	RAM/EEPROM 8
D2A_AOP8_CAL_LOW	16-bit register. Holds the **low** end calibration value for analog output 8.	S_16	4164	RAM/EEPROM 8
D2A_AOP8_CAL_FULL_SCALE	32-bit register. Holds the **full scale** value for analog output 8.	S_32	4019	RAM/EEPROM 8
D2A_AOP8_ZERO	32-bit register. Holds the **zero scale** value for analog output 8.	S_32	395	RAM/EEPROM 8
DATA_SOURCE_ANALOG8 30	8-bit register. Holds the register number of the **data source** for analog output 8.	U_8	4386 30	RAM/EEPROM 8
Read Only				
ANALOG_OUTPUT8	Read Only 16-bit register. Holds the **scaled output data** for analog output 8.	S_16	4152	RAM 8

2.2.9 Analog Output Data Source Selection - Register 4379 to 4386

Registers 4379 to 4386 are 16-bit registers that specify the data source for the analog output channels. The number they contain is the ASCII/Modbus register number for the required data source.

NOTE: Only registers that hold integer values can be used as a data source for the display. Floating point and text registers can not be used.

2.3 Auxiliary Analog Inputs

ICC402 series controllers include up to 4 low resolution analog inputs which can be used with selected input modules. Each channel can be configured to run in either 10 bit or 14 bit resolution. When running in 14 bit resolution the effective sampling frequency is greater than 50hz with all 4 channels active. If all 4 auxiliary analog input channels are set to 10 bit resolution mode then the

effective sampling rate is 100hz.

Depending on the input module being used, the input channel can be configured for single ended mode or differential mode. In single ended mode all measurements are referenced to ground and output will always be a positive number. In differential mode the channel can be used to measure a positive or negative input. Only auxiliary analog input channels 1 & 3 can be placed in differential mode. If channel 1 is operating in differential mode then channel 2 becomes inoperative. If channel 3 is operating in differential mode then channel 4 becomes inoperative.

NOTE: If some of the auxiliary analog input channels are set to operate in 14 bit mode and some in 10 bit mode the sampling rate of those channels operating in 10 bit mode will increase to over 16khz. However this will only be realised when accessing the channel via the serial port or macro.

The following table shows the various registers associated with these analog inputs.

Name	Description	Symbol Type	Register Number	Memory Type
AUX_ANALOG1	16-bit signed register. Holds the input value for auxiliary analog input 1.	S_16	4463	RAM 8
AUX_ANALOG2	16-bit signed register. Holds the input value for auxiliary analog input 2.	S_16	4464	RAM 8
AUX_ANALOG3	16-bit signed register. Holds the input value for auxiliary analog input 3.	S_16	4465	RAM 8
AUX_ANALOG4	16-bit signed register. Holds the input value for auxiliary analog input 4.	S_16	4466	RAM 8
AUX_ANALOG1_SETUP	8-bit register. Holds the currently programmed setup for auxiliary analog input 1 (Note, the display is in octal).	O_8 107	8509	RAM/EEPROM 8
AUX_ANALOG2_SETUP	8-bit register. Holds the currently programmed setup for auxiliary analog input 1 (Note, the display is in octal).	O_8 107	8510	RAM/EEPROM 8
AUX_ANALOG3_SETUP	8-bit register. Holds the currently programmed setup for auxiliary analog input 1 (Note, the display is in octal).	O_8 107	8511	RAM/EEPROM 8
AUX_ANALOG4_SETUP	8-bit register. Holds the currently programmed setup for auxiliary analog input 1 (Note, the display is in octal).	O_8 107	8512	RAM/EEPROM 8

Auxiliary Analog Setup Registers – Registers 8509 to 8512
Registers 8509 to 8512 are 8-bit registers used to control the functionality of auxiliary analog inputs 1 - 4. When reading or writing to these registers via the serial port, the data is treated in octal format 107. The function of each octal digit in the auxiliary analog setup registers is shown in the below.

1st Digit (Most Significant) - Input Type

0 = Single ended mode
1 = Differential mode (channels 1 & 3 only - see note above)
2 = Reserved for future development
3 = Reserved for future development

2nd Digit - Resolution

0 = 10 bit
1 = 14 bit (over sampled)
2 = Reserved for future development
3 = Reserved for future development
4 = Reserved for future development

5 = Reserved for future development
6 = Reserved for future development
7 = Reserved for future development

3rd Digit (Least Significant) - Gain

0 = Disabled
1 = Gain of 0.5
2 = Gain of 1
3 = Gain of 2
4 = Gain of 4
5 = Gain of 8
6 = Gain of 16
7 = Reserved for future development

2.4 Channel 1

Channel 1 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH1	32-bit register that holds the processed data for CH1.	S_32	9	RAM/FLASH 8
CH1_PROCESSED	32-bit register that holds the processed data for CH1.	S_32	259	RAM/FLASH 8
CH1_RAW	32-bit register that holds the raw data for CH1, prior to scaling and post processing.	S_32	283	RAM 8
CH1_SCALED	32-bit register that holds the scaled data for CH1, prior to post processing.	S_32	307	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH1_PROCESSED, CH1_RAW, or CH1_SCALED registers modifies only that register. A write to the CH1 register automatically updates CH1_PROCESSED, CH1_RAW, and CH1_SCALED. The controller takes into account scale and offset and post processing and calculates the CH1_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH1 register is identical to a read of CH1_PROCESSED register.

See also

CH1 Setup Registers 33

2.4.1 CH1 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH1	16-bit register. Auto zero correction offset for CH1.	S_16	4134	RAM/FLASH 8
AVERAGING_SAMPLES_CH1	8-bit register. Averaging samples for CH1 (range 0 to 256 samples).	U_8	8310	RAM/EEPROM 8
AVERAGING_WINDOW_CH1	16-bit register. Averaging window for CH1 (range 0 to 65535 counts).	U_16	4294	RAM/EEPROM 8
AZ_APERTURE_BAND_CH1	16-bit register. Auto zero aperture band for CH1 (range 0 to 65535 counts).	U_16	4336	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH1	8-bit register. Auto zero capture band for CH1 (range 0 to 254 counts, 255 = manual zero).	U_8	8322	RAM/EEPROM 8
AZ_MOTION_BAND_CH1	8-bit register. Auto zero motion band for CH1 (0 to 255 counts / second).	U_8	8334	RAM/EEPROM 8
CH1_PRESCALER	16-bit register. Sets the prescale value in CH1 counter mode (range 1 - 32767).	U_16	4117	RAM/EEPROM 8
CH1_PRESCALER_COUNT	16-bit register. Gives the current prescaler count value for CH1 in counter mode.	S_16	4121	RAM/FLASH 8
CHANNEL1_TEXT	Text display for CH1.	L_8	16393	EEROM 8
DISPLAY_FORMAT_CH1 106	8-bit register. Controls the display format settings for CH1 (displayed in octal 107 format).	O_8 107	8368 106	RAM/EEPROM 8
OFFSET_CH1	32-bit register. Holds the calibration offset for CH1.	S_32	359	RAM/EEPROM 8
SCALE_FACTOR_CH1	32-bit floating point register. Holds the calibration scale factor for CH1.	F_32	1099	RAM/EEPROM 8
TEXT_CHARACTER_CH1 108	8-bit register. Holds the ASCII value for the last digit text character for CH1 (0 = no character).	U_8	8394 108	RAM/EEPROM 8

See also
Channel 1 32

2.5 Channel 2

Channel 2 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH2	32-bit register that holds the processed data for CH2.	S_32	11	RAM/FLASH 8
CH2_PROCESSED	32-bit register that holds the processed data for CH2.	S_32	261	RAM/FLASH 8
CH2_RAW	32-bit register that holds the raw data for CH2, prior to scaling and post processing.	S_32	285	RAM 8
CH2_SCALED	32-bit register that holds the scaled data for CH2, prior to post processing.	S_32	309	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH2_PROCESSED, CH2_RAW, or CH2_SCALED registers modifies only that register. A write to the CH2 register automatically updates CH2_PROCESSED, CH2_RAW, and CH2_SCALED. The controller takes into account scale and offset and post processing and calculates the CH2_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH2 register is identical to a read of CH2_PROCESSED register.

See also
CH2 Setup Registers 35

2.5.1 CH2 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH2	16-bit register. Auto zero correction offset for CH2.	S_16	4135	RAM/FLASH 8
AVERAGING_SAMPLES_CH2	8-bit register. Averaging samples for CH2 (range 0 to 255 samples).	U_8	8311	RAM/EEPROM 8
AVERAGING_WINDOW_CH2	16-bit register. Averaging window for CH2 (range 0 to 65535 counts).	U_16	4295	RAM/EEPROM 8
AZ_APERTURE_BAND_CH2	16-bit register. Auto zero aperture band for CH2 (range 0 to 65535 counts).	U_16	4337	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH2	8-bit register. Auto zero capture band for CH2 (range 0 to 254 counts, 255 = manual zero).	U_8	8323	RAM/EEPROM 8
AZ_MOTION_BAND_CH2	8-bit register. Auto zero motion band for CH2 (0 to 255 counts / second).	U_8	8335	RAM/EEPROM 8
CH2_PRESCALER	16-bit register. Sets the prescale value in CH2 counter mode (range 1 to 32767).	U_16	4118	RAM/EEPROM 8
CH2_PRESCALER_COUNT	16-bit register. Gives the current prescaler count value for CH2 in counter mode.	S_16	4122	RAM/FLASH 8
CHANNEL2_TEXT	Text display for CH2.	L_8	16395	EEPROM 8
DISPLAY_FORMAT_CH2 106	8-bit register. Controls the display format settings for CH2 (displayed in octal 107 format).	O_8 107	8369 106	RAM/EEPROM 8
OFFSET_CH2	32-bit register. Holds the calibration offset for CH2.	S_32	361	RAM/EEPROM 8
SCALE_FACTOR_CH2	32-bit floating point register. Holds the calibration scale factor for CH2.	F_32	1101	RAM/EEPROM 8
TEXT_CHARACTER_CH2 108	8-bit register. Holds the ASCII value for the last digit text character for CH2 (0 = no character).	U_8	8395 108	RAM/EEPROM 8

See also
Channel 2 33

2.6 Channel 3

Channel 3 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH3	32-bit register that holds the processed data for CH3.	S_32	33	RAM/FLASH 8
CH3_PROCESSED	32-bit register that holds the processed data for CH3.	S_32	263	RAM/FLASH 8
CH3_RAW	32-bit register that holds the raw data for CH3, prior to scaling and post processing.	S_32	287	RAM 8
CH3_SCALED	32-bit register that holds the scaled data for CH3, prior to post processing.	S_32	311	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH3_PROCESSED, CH3_RAW, or CH3_SCALED registers modifies only that register. A write to the CH3 register automatically updates CH3_PROCESSED, CH3_RAW, and CH3_SCALED. The controller takes into account scale and offset and post processing and calculates the CH3_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH3 register is identical to a read of CH3_PROCESSED register.

See also
CH3 Setup Registers 36

2.6.1 CH3 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH3	16-bit register. Auto zero correction offset for CH3.	S_16	4136	RAM/FLASH 8
AVERAGING_SAMPLES_CH3	8-bit register. Averaging samples for CH3 (range 0 to 255 samples).	U_8	8312	RAM/EEPROM 8
AVERAGING_WINDOW_CH3	16-bit register. Averaging window for CH3 (range 0 to 65535 counts).	U_16	4296	RAM/EEPROM 8
AZ_APERTURE_BAND_CH3	16-bit register. Auto zero aperture band for CH3 (range 0 to 65535 counts).	U_16	4338	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH3	8-bit register. Auto zero capture band for CH3 (range 0 to 254 counts, 255 = manual zero).	U_8	8324	RAM/EEPROM 8
AZ_MOTION_BAND_CH3	8-bit register. Auto zero motion band for CH3 (0 to 255 counts / second).	U_8	8336	RAM/EEPROM 8
CHANNEL3_TEXT	Text display for CH3.	L_8	16397	EEPROM 8
DISPLAY_FORMAT_CH3 108	8-bit register. Controls the display format settings for CH3 (displayed in octal 107 format).	O_8 107	8370 108	RAM/EEPROM 8
OFFSET_CH3	32-bit register. Holds the calibration offset for CH3.	S_32	363	RAM/EEPROM 8
SCALE_FACTOR_CH3	32-bit floating point register. Holds the calibration scale factor for CH3.	F_32	1103	RAM/EEPROM 8
TEXT_CHARACTER_CH3 108	8-bit register. Holds the ASCII value for the last digit text character for CH3 (0 = no character).	U_8	8396 108	RAM/EEPROM 8

See also
Channel 3 35

2.7 Channel 4

Channel 4 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH4	32-bit register that holds the processed data for CH4.	S_32	15	RAM/FLASH 8
CH4_PROCESSED	32-bit register that holds the processed data for CH4.	S_32	265	RAM/FLASH 8
CH4_RAW	32-bit register that holds the raw data for CH4, prior to scaling and post processing.	S_32	289	RAM 8
CH4_SCALED	32-bit register that holds the scaled data for CH4, prior to post processing.	S_32	313	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH4_PROCESSED, CH4_RAW, or CH4_SCALED registers modifies only that register. A write to the CH4 register automatically updates CH4_PROCESSED, CH4_RAW, and CH4_SCALED. The controller takes into account scale and offset and post processing and calculates the CH4_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH4 register is identical to a read of CH4_PROCESSED register.

See also
CH4 Setup Registers 37

2.7.1 CH4 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH4	16-bit register. Auto zero correction offset for CH4.	S_16	4137	RAM/FLASH 8
AVERAGING_SAMPLES_CH4	8-bit register. Averaging samples for CH4 (range 0 to 255 samples).	U_8	8313	RAM/EEPROM 8

AVERAGING_WINDOW_CH4	16-bit register. Averaging window for CH4 (range 0 to 65535 counts).	U_16	4297	RAM/EEPROM 8
AZ_APERTURE_BAND_CH4	16-bit register. Auto zero aperture band for CH4 (range 0 to 65535 counts).	U_16	4339	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH4	8-bit register. Auto zero capture band for CH4 (range 0 to 254 counts, 255 = manual zero).	U_8	8325	RAM/EEPROM 8
AZ_MOTION_BAND_CH4	8-bit register. Auto zero motion band for CH4 (0 to 255 counts / second).	U_8	8337	RAM/EEPROM 8
CHANNEL4_TEXT	Text display for CH4.	L_8	16399	EEPROM 8
DISPLAY_FORMAT_CH4 106	8-bit register. Controls the display format settings for CH4 (displayed in octal 107 format).	O_8 107	8371 106	RAM/EEPROM 8
OFFSET_CH4	32-bit register. Holds the calibration offset for CH4.	S_32	365	RAM/EEPROM 8
SCALE_FACTOR_CH4	32-bit floating point register. Holds the calibration scale factor for CH4.	F_32	1105	RAM/EEPROM 8
TEXT_CHARACTER_CH4 108	8-bit register. Holds the ASCII value for the last digit text character for CH4 (0 = no character).	U_8	8397 108	RAM/EEPROM 8

See also
Channel 4 37

2.8 Channel 5

Channel 5 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH5	32-bit register that holds the processed data for CH5.	S_32	17	RAM/FLASH 8
CH5_PROCESSED	32-bit register that holds the processed data for CH5.	S_32	267	RAM/FLASH 8
CH5_RAW	32-bit register that holds the raw data for CH5, prior to scaling and post processing.	S_32	291	RAM 8
CH5_SCALED	32-bit register that holds the scaled data for CH5, prior to post processing.	S_32	315	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH5_PROCESSED, CH5_RAW, or CH5_SCALED registers modifies only that register.

A write to the CH5 register automatically updates CH5_PROCESSED, CH5_RAW, and CH5_SCALED. The controller takes into account scale and offset and post processing and calculates the CH5_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH5 register is identical to a read of CH5_PROCESSED register.

See also
CH5 Setup Registers 39

2.8.1 CH5 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH5	16-bit register. Auto zero correction offset for CH5.	S_16	4138	RAM/FLASH 8
AVERAGING_SAMPLES_CH5	8-bit register. Averaging samples for CH5 (range 0 to 255 samples).	U_8	8314	RAM/EEPROM 8
AVERAGING_WINDOW_CH5	16-bit register. Averaging window for CH5 (range 0 to 65535 counts).	U_16	4298	RAM/EEPROM 8
AZ_APERTURE_BAND_CH5	16-bit register. Auto zero aperture band for CH5 (range 0 to 65535 counts).	U_16	4340	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH5	8-bit register. Auto zero capture band for CH5 (range 0 to 254 counts, 255 = manual zero).	U_8	8326	RAM/EEPROM 8
AZ_MOTION_BAND_CH5	8-bit register. Auto zero motion band for CH5 (0 to 255 counts / second).	U_8	8338	RAM/EEPROM 8
CHANNEL5_TEXT	Text display for CH5.	L_8	16401	EEPROM 8
DATA_SOURCE_CH5	16-bit register selecting the source register for channel 5 (range 0 to 16384).	U_16	4455	RAM/EEPROM 8
DISPLAY_FORMAT_CH5 106	8-bit register. Controls the display format settings for CH5 (displayed in octal 107 format).	O_8 107	8372 106	RAM/EEPROM 8
OFFSET_CH5	32-bit register. Holds the calibration offset for CH5.	S_32	367	RAM/EEPROM 8
SCALE_FACTOR_CH5	32-bit floating point register. Holds the calibration scale factor for CH5.	F_32	1107	RAM/EEPROM 8
TEXT_CHARACTER_CH5 108	8-bit register. Holds the ASCII value for the last digit text character for CH5 (0 = no character).	U_8	8398 108	RAM/EEPROM 8

See also
Channel 5 38

2.9 Channel 6

Channel 6 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.

- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH6	32-bit register that holds the processed data for CH6.	S_32	19	RAM/FLASH 8
CH6_PROCESSED	32-bit register that holds the processed data for CH6.	S_32	269	RAM/FLASH 8
CH6_RAW	32-bit register that holds the raw data for CH6, prior to scaling and post processing.	S_32	293	RAM 8
CH6_SCALED	32-bit register that holds the scaled data for CH6, prior to post processing.	S_32	317	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH6_PROCESSED, CH6_RAW, or CH6_SCALED registers modifies only that register. A write to the CH6 register automatically updates CH6_PROCESSED, CH6_RAW, and CH6_SCALED. The controller takes into account scale and offset and post processing and calculates the CH6_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH6 register is identical to a read of CH6_PROCESSED register.

See also
CH6 Setup Registers 41

2.9.1 CH6 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH6	16-bit register. Auto zero correction offset for CH6.	S_16	4139	RAM/FLASH 8
AVERAGING_SAMPLES_CH6	8-bit register. Averaging samples for CH6 (range 0 to 255 samples).	U_8	8315	RAM/EEPROM 8
AVERAGING_WINDOW_CH6	16-bit register. Averaging window for CH6 (range 0 to 65535 counts).	U_16	4299	RAM/EEPROM 8
AZ_APERTURE_BAND_CH6	16-bit register. Auto zero aperture band for CH6 (range 0 to 65535 counts).	U_16	4341	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH6	8-bit register. Auto zero capture band for CH6 (range 0 to 254 counts, 255 = manual zero).	U_8	8327	RAM/EEPROM 8
AZ_MOTION_BAND_CH6	8-bit register. Auto zero motion band for CH6 (0 to 255 counts / second).	U_8	8339	RAM/EEPROM 8
CHANNEL6_TEXT	Text display for CH6.	L_8	16403	EEPROM 8
DATA_SOURCE_CH6	16-bit register selecting the source register for channel 6 (range 0 to 16384).	U_16	4456	RAM/EEPROM 8
DISPLAY_FORMAT_CH6 106	8-bit register. Controls the display format settings for CH6 (displayed in octal 107 format).	O_8 107	8373 106	RAM/EEPROM 8
OFFSET_CH6	32-bit register. Holds the calibration offset for CH6.	S_32	369	RAM/EEPROM 8
SCALE_FACTOR_CH6	32-bit floating point register. Holds the calibration scale factor for CH6.	F_32	1109	RAM/EEPROM 8
TEXT_CHARACTER_CH6 108	8-bit register. Holds the ASCII value for the last digit text character for CH6 (0 = no character).	U_8	8399 108	RAM/EEPROM 8

See also
Channel 6 39

2.10 Channel 7

Channel 7 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH7	32-bit register that holds the processed data for CH7.	S_32	21	RAM/FLASH 8
CH7_PROCESSED	32-bit register that holds the processed data for CH7.	S_32	271	RAM/FLASH 8
CH7_RAW	32-bit register that holds the raw data for CH7, prior to scaling and post processing.	S_32	295	RAM 8
CH7_SCALED	32-bit register that holds the scaled data for CH7, prior to post processing.	S_32	319	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH7_PROCESSED, CH7_RAW, or CH7_SCALED registers modifies only that register. A write to the CH7 register automatically updates CH7_PROCESSED, CH7_RAW, and CH7_SCALED. The controller takes into account scale and offset and post processing and calculates the CH7_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH7 register is identical to a read of CH7_PROCESSED register.

See also
CH7 Setup Registers 43

2.10.1 CH7 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH7	16-bit register. Auto zero correction offset for CH7.	S_16	4140	RAM/FLASH 8
AVERAGING_SAMPLES_CH7	8-bit register. Averaging samples for CH7 (range 0 to 255 samples).	U_8	8316	RAM/EEPROM 8
AVERAGING_WINDOW_CH7	16-bit register. Averaging window for CH7 (range 0 to 65535 counts).	U_16	4300	RAM/EEPROM 8
AZ_APERTURE_BAND_CH7	16-bit register. Auto zero aperture band for CH7 (range 0 to 65535 counts).	U_16	4342	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH7	8-bit register. Auto zero capture band for CH7 (range 0 to 254 counts, 255 = manual zero).	U_8	8328	RAM/EEPROM 8
AZ_MOTION_BAND_CH7	8-bit register. Auto zero motion band for CH7 (0 to 255 counts / second).	U_8	8340	RAM/EEPROM 8
CHANNEL7_TEXT	Text display for CH7.	L_8	16405	EEPROM 8
DATA_SOURCE_CH7	16-bit register selecting the source register for channel 7 (range 0 to 16384).	U_16	4457	RAM/EEPROM 8
DISPLAY_FORMAT_CH7 106	8-bit register. Controls the display format settings for CH7 (displayed in octal 107 format).	O_8 107	8374 106	RAM/EEPROM 8
OFFSET_CH7	32-bit register. Holds the calibration offset for CH7.	S_32	371	RAM/EEPROM 8
SCALE_FACTOR_CH7	32-bit floating point register. Holds the calibration scale factor for CH7.	F_32	1111	RAM/EEPROM 8
TEXT_CHARACTER_CH7 108	8-bit register. Holds the ASCII value for the last digit text character for CH7 (0 = no character).	U_8	8400 108	RAM/EEPROM 8

See also
Channel 7 41

2.11 Channel 8

Channel 8 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH8	32-bit register that holds the processed data for CH8.	S_32	23	RAM/FLASH 8
CH8_PROCESSED	32-bit register that holds the processed data for CH8.	S_32	273	RAM/FLASH 8
CH8_RAW	32-bit register that holds the raw data for CH8, prior to scaling and post processing.	S_32	297	RAM 8
CH8_SCALED	32-bit register that holds the scaled data for CH8, prior to post processing.	S_32	321	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH8_PROCESSED, CH8_RAW, or CH8_SCALED registers modifies only that register. A write to the CH8 register automatically updates CH8_PROCESSED, CH8_RAW, and CH8_SCALED. The controller takes into account scale and offset and post processing and calculates the CH8_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH8 register is identical to a read of CH8_PROCESSED register.

See also
CH8 Setup Registers 45

2.11.1 CH8 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH8	16-bit register. Auto zero correction offset for CH8.	S_16	4141	RAM/FLASH 8
AVERAGING_SAMPLES_CH8	8-bit register. Averaging samples for CH8 (range 0 to 255 samples).	U_8	8317	RAM/EEPROM 8
AVERAGING_WINDOW_CH8	16-bit register. Averaging window for CH8 (range 0 to 65535 counts).	U_16	4301	RAM/EEPROM 8
AZ_APERTURE_BAND_CH8	16-bit register. Auto zero aperture band for CH8 (range 0 to 65535 counts).	U_16	4343	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH8	8-bit register. Auto zero capture band for CH8 (range 0 to 254 counts, 255 = manual zero).	U_8	8329	RAM/EEPROM 8
AZ_MOTION_BAND_CH8	8-bit register. Auto zero motion band for CH8 (0 to 255 counts / second).	U_8	8341	RAM/EEPROM 8
CHANNEL8_TEXT	Text display for CH8.	L_8	16407	EEPROM 8
DATA_SOURCE_CH8	16-bit register selecting the source register for channel 8 (range 0 to 16384).	U_16	4458	RAM/EEPROM 8
DISPLAY_FORMAT_CH8 106	8-bit register. Controls the display format settings for CH8 (displayed in octal 107 format).	O_8 107	8515 106	RAM/EEPROM 8
OFFSET_CH8	32-bit register. Holds the calibration offset for CH8.	S_32	373	RAM/EEPROM 8
SCALE_FACTOR_CH8	32-bit floating point register. Holds the calibration scale factor for CH8.	F_32	1113	RAM/EEPROM 8
TEXT_CHARACTER_CH8 108	8-bit register. Holds the ASCII value for the last digit text character for CH8 (0 = no character).	U_8	8519 108	RAM/EEPROM 8

See also
Channel 8 43

2.12 Channel 9

Channel 9 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH9	32-bit register that holds the processed data for CH9.	S_32	25	RAM/FLASH 8
CH9_PROCESSED	32-bit register that holds the processed data for CH9.	S_32	275	RAM/FLASH 8
CH9_RAW	32-bit register that holds the raw data for CH9, prior to scaling and post processing.	S_32	299	RAM 8
CH9_SCALED	32-bit register that holds the scaled data for CH9, prior to post processing.	S_32	323	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH9_PROCESSED, CH9_RAW, or CH9_SCALED registers modifies only that register. A write to the CH9 register automatically updates CH9_PROCESSED, CH9_RAW, and CH9_SCALED. The controller takes into account scale and offset and post processing and calculates the CH9_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH9 register is identical to a read of CH9_PROCESSED register.

See also
CH9 Setup Registers 47

2.12.1 CH9 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH9	16-bit register. Auto zero correction offset for CH9.	S_16	4142	RAM/FLASH 8
AVERAGING_SAMPLES_CH9	8-bit register. Averaging samples for CH9 (range 0 to 255 samples).	U_8	8318	RAM/EEPROM 8
AVERAGING_WINDOW_CH9	16-bit register. Averaging window for CH9 (range 0 to 65535 counts).	U_16	4302	RAM/EEPROM 8
AZ_APERTURE_BAND_CH9	16-bit register. Auto zero aperture band for CH9 (range 0 to 65535 counts).	U_16	4344	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH9	8-bit register. Auto zero capture band for CH9 (range 0 to 254 counts, 255 = manual zero).	U_8	8330	RAM/EEPROM 8
AZ_MOTION_BAND_CH9	8-bit register. Auto zero motion band for CH9 (0 to 255 counts / second).	U_8	8342	RAM/EEPROM 8
CHANNEL9_TEXT	Text display for CH9.	L_8	16409	EEPROM 8
DATA_SOURCE_CH9	16-bit register selecting the source register for channel 9 (range 0 to 16384).	U_16	4459	RAM/EEPROM 8
DISPLAY_FORMAT_CH9 106	8-bit register. Controls the display format settings for CH9 (displayed in octal 107 format).	O_8 107	8516 106	RAM/EEPROM 8
OFFSET_CH9	32-bit register. Holds the calibration offset for CH9.	S_32	375	RAM/EEPROM 8
SCALE_FACTOR_CH9	32-bit floating point register. Holds the calibration scale factor for CH9.	F_32	1115	RAM/EEPROM 8
TEXT_CHARACTER_CH9 108	8-bit register. Holds the ASCII value for the last digit text character for CH9 (0 = no character).	U_8	8520 108	RAM/EEPROM 8

See also
Channel 9 45

2.13 Channel 10

Channel 10 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH10	32-bit register that holds the processed data for CH10.	S_32	27	RAM/FLASH 8
CH10_PROCESSED	32-bit register that holds the processed data for CH10.	S_32	277	RAM/FLASH 8
CH10_RAW	32-bit register that holds the raw data for CH10, prior to scaling and post processing.	S_32	301	RAM 8
CH10_SCALED	32-bit register that holds the scaled data for CH10, prior to post processing.	S_32	325	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH10_PROCESSED, CH10_RAW, or CH10_SCALED registers modifies only that register. A write to the CH10 register automatically updates CH10_PROCESSED, CH10_RAW, and CH10_SCALED. The controller takes into account scale and offset and post processing and calculates the CH10_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH10 register is identical to a read of CH10_PROCESSED register.

See also
CH10 Setup Registers 49

2.13.1 CH10 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH10	16-bit register. Auto zero correction offset for CH10.	S_16	4143	RAM/FLASH 8
AVERAGING_SAMPLES_CH10	8-bit register. Averaging samples for CH10 (range 0 to 255 samples).	U_8	8319	RAM/EEPROM 8
AVERAGING_WINDOW_CH10	16-bit register. Averaging window for CH10 (range 0 to 65535 counts).	U_16	4303	RAM/EEPROM 8
AZ_APERTURE_BAND_CH10	16-bit register. Auto zero aperture band for CH10 (range 0 to 65535 counts).	U_16	4345	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH10	8-bit register. Auto zero capture band for CH10 (range 0 to 254 counts, 255 = manual zero).	U_8	8331	RAM/EEPROM 8
AZ_MOTION_BAND_CH10	8-bit register. Auto zero motion band for CH10 (0 to 255 counts / second).	U_8	8343	RAM/EEPROM 8
CHANNEL10_TEXT	Text display for CH10.	L_8	16411	EEPROM 8
DATA_SOURCE_CH10	16-bit register selecting the source register for channel 10 (range 0 to 16384).	U_16	4460	RAM/EEPROM 8
DISPLAY_FORMAT_CH10 106	8-bit register. Controls the display format settings for CH10 (displayed in octal 107 format).	O_8 107	8517 106	RAM/EEPROM 8
OFFSET_CH10	32-bit register. Holds the calibration offset for CH10.	S_32	377	RAM/EEPROM 8
SCALE_FACTOR_CH10	32-bit floating point register. Holds the calibration scale factor for CH10.	F_32	1117	RAM/EEPROM 8
TEXT_CHARACTER_CH10 108	8-bit register. Holds the ASCII value for the last digit text character for CH10 (0 = no character).	U_8	8521 108	RAM/EEPROM 8

See also
Channel 10 47

2.14 Channel 11

Channel 11 registers can be selected as the data source for:

- The primary display.
- The second display, if installed.
- The third display, if installed.
- Trigger for setpoints SP1 to SP6.
- Setpoint reset destination.
  The reset destination mode allows you to select a register to be reset using the contents of another register triggered by a setpoint.

Name	Description	Symbol Type	Register Number	Memory Type
CH11	32-bit register that holds the processed data for CH11.	S_32	29	RAM/FLASH 8
CH11_PROCESSED	32-bit register that holds the processed data for CH11.	S_32	279	RAM/FLASH 8
CH11_RAW	32-bit register that holds the raw data for CH11, prior to scaling and post processing.	S_32	303	RAM 8
CH11_SCALED	32-bit register that holds the scaled data for CH11, prior to post processing.	S_32	327	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If the channel is disabled or in a counter mode, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, from the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being overwritten by the operating system in the controller.

A write to the CH11_PROCESSED, CH11_RAW, or CH11_SCALED registers modifies only that register. A write to the CH11 register automatically updates CH11_PROCESSED, CH11_RAW, and CH11_SCALED. The controller takes into account scale and offset and post processing and calculates the CH11_RAW value. (This is useful to reset the count in counter applications that use a scale and offset value other than 1 and 0).

NOTE: 32-point linearization is not supported by the above feature.

A read of CH11 register is identical to a read of CH11_PROCESSED register.

See also
CH11 Setup Registers 51

2.14.1 CH11 Setup Registers

Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_CH11	16-bit register. Auto zero correction offset for CH11.	S_16	4144	RAM/FLASH 8
AVERAGING_SAMPLES_CH11	8-bit register. Averaging samples for CH11 (range 0 to 255 samples).	U_8	8320	RAM/EEPROM 8
AVERAGING_WINDOW_CH11	16-bit register. Averaging window for CH11 (range 0 to 65535 counts).	U_16	4304	RAM/EEPROM 8
AZ_APERTURE_BAND_CH11	16-bit register. Auto zero aperture band for CH11 (range 0 to 65535 counts).	U_16	4346	RAM/EEPROM 8
AZ_CAPTURE_BAND_CH11	8-bit register. Auto zero capture band for CH11 (range 0 to 254 counts, 255 = manual zero).	U_8	8332	RAM/EEPROM 8
AZ_MOTION_BAND_CH11	8-bit register. Auto zero motion band for CH11 (0 to 255 counts / second).	U_8	8344	RAM/EEPROM 8
CHANNEL11_TEXT	Text display for CH11.	L_8	16413	EEPROM 8
DATA_SOURCE_CH11	16-bit register selecting the source register for channel 11 (range 0 to 16384).	U_16	4461	RAM/EEPROM 8
DISPLAY_FORMAT_CH11 106	8-bit register. Controls the display format settings for CH11 (displayed in octal 107 format).	O_8 107	8518 106	RAM/EEPROM 8
OFFSET_CH11	32-bit register. Holds the calibration offset for CH11.	S_32	379	RAM/EEPROM 8
SCALE_FACTOR_CH11	32-bit floating point register. Holds the calibration scale factor for CH11.	F_32	1119	RAM/EEPROM 8
TEXT_CHARACTER_CH11 108	8-bit register. Holds the ASCII value for the last digit text character for CH11 (0 = no character).	U_8	8522 108	RAM/EEPROM 8

See also
Channel 11 49

2.15 Clock

The following registers are used to hold time and date information from the real-time clock. These read/write registers are continuously updated by the operating system of the controller. If the real-time clock option is installed in the controller, then these registers are maintained even during power down. If the real-time clock option is not installed in the controller then these registers are still updated by the controller, but all values are lost when the power is removed from the controller.

Name	Description	Symbol Type	Register Number	Memory Type
DATE	8-bit register. Holds the real-time clock **date** (range 1 to 31).	U_8	8242	RAM/NVRAM 8
DAYS	8-bit register. Holds the real-time clock **days of the week** (Sunday = 1, Saturday = 7).	U_8	8241	RAM/NVRAM 8
HOURS	8-bit register. Holds the real-time clock **hours** count (range 0 to 23).	U_8	8240	RAM/NVRAM 8
MINUTES	8-bit register. Holds the real-time clock **minutes** count (range 0 to 59).	U_8	8239	RAM/NVRAM 8
HOURS_MINUTES	16-bit read only register. Holds the real-time clock count in minutes for **hours : minutes** (range 0 to 1439 (00:00 to 23:59)).	U_16_R	4438	RAM 8
HRS_MIN_SEC	32-bit read only register. Holds the real-time clock count in seconds for **hours : minutes : seconds** (range 0 to 86399 (0:00:00 to 23:59:59)).	U_32_R	151	RAM 8
MONTH	8-bit register. Holds the real-time clock **month** (range 1 to 12).	U_8	8243	RAM/NVRAM 8
SECONDS	8-bit register. Holds the real-time clock **seconds** count (range 0 to 59).	U_8	8238	RAM/NVRAM 8
YEAR	8-bit register. Holds the real-time clock **year** (range 0 to 99).	U_8	8244	RAM/NVRAM 8

2.16 Codes

Code Setup Registers – Registers 8193 to 8203
Registers 8193 to 8203 are 8-bit registers used to control the functionality of the controller. When reading or writing to these registers via the serial port, the data is treated in octal format so that it is identical to the value shown on the display of the controller when setting the codes up manually. The function selected in the 1st digit of each Code register is stored in bits 6 and 7. The function selected in the 2nd digit of each Code register is stored in bits 3, 4, and 5. The function selected in the 3rd digit of each Code register is stored in bits 0, 1, and 2.

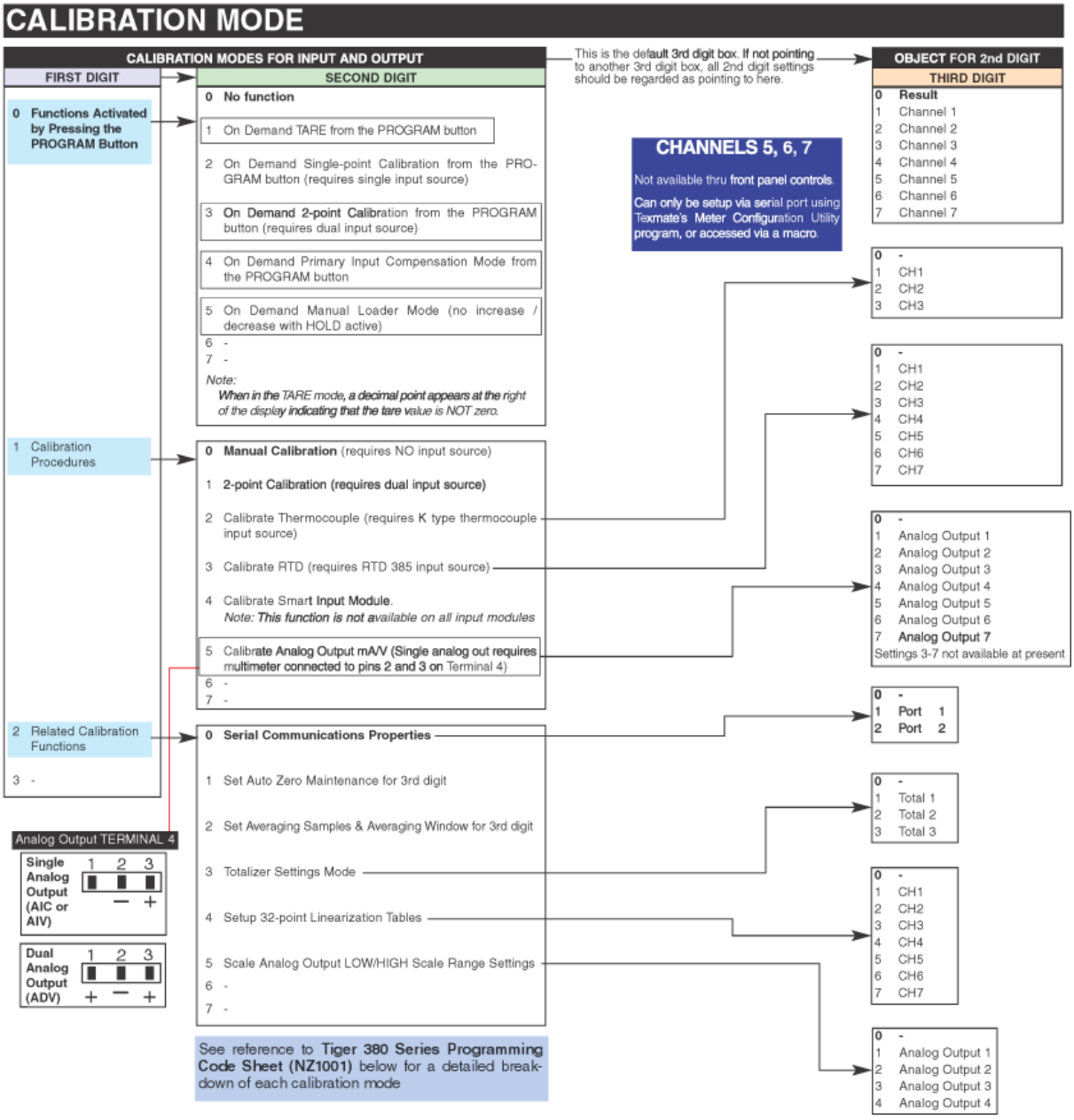
For example:
If the manual setup for Code 4 shows 241 on the display, then reading register 8197 in ASCII mode results in a value of 241. Converting this octal value to a binary equivalent of 10100001 or hexadecimal equivalent of 0A1.

	1st Digit	2nd Digit	3rd Digit
Octal 10	2	4	1
Binary	10	100	001

Name	Description	Symbol Type	Register Number	Memory Type
CAL 53	8-bit register. Holds the currently programmed calibration mode settings (Note, the meter display is in octal).	O_8 107	8193 53	RAM/EEPROM 8
CODE1 55	8-bit register. Holds the currently programmed settings for **Code 1** (Note, the display is in octal).	O_8 107	8194 55	RAM/EEPROM 8
CODE2 58	8-bit register. Holds the currently programmed settings for **Code 2** (Note, the display is in octal).	O_8 107	8195 58	RAM/EEPROM 8
CODE3 58	8-bit register. Holds the currently programmed settings for **Code 3** (Note, the display is in octal).	O_8 107	8196 58	RAM/EEPROM 8
CODE4 59	8-bit register. Holds the currently programmed settings for **Code 4** (Note, the display is in octal).	O_8 107	8197 59	RAM/EEPROM 8
CODE5 60	8-bit register. Holds the currently programmed settings for **Code 5** (Note, the display is in octal).	O_8 107	8198 60	RAM/EEPROM 8
CODE6 61	8-bit register. Holds the currently programmed settings for **Code 6** (Note, the display is in octal).	O_8 107	8199 61	RAM/EEPROM 8
CODE7 62	8-bit register. Holds the currently programmed settings for **Code 7** (Note, the display is in octal).	O_8 107	8200 62	RAM/EEPROM 8
CODE8 63	8-bit register. Holds the currently programmed settings for **Code 8** (Note, the display is in octal).	O_8 107	8201 63	RAM/EEPROM 8
CODE9 63	8-bit register. Holds the currently programmed settings for **Code 9** (Note, the display is in octal).	O_8 107	8202 63	RAM/EEPROM 8
CODE_BLANKING 54	16-bit register that controls which parameters are displayed when editing the code setups.	U_16	4434 54	EEPROM 8

2.16.1 Calibration

While programming through the front display, the calibration modes allow you to calibrate the selected channel for all input signals. They also allow you to set up on-demand functions, serial communications settings, auto zero maintenance settings, averaging samples and averaging window settings, set K factor and cutoff for totalizers, set up 32-point linearization tables, and calibrate and scale analog output signals.



2.16.2 Code Blanking

Register 4434 is a 16 bit register in EEPROM which controls the sequence of code setups that are displayed when the "Prog" and "Up" button are pressed. Each bit in the register controls a specific code display as shown below. If a specific bit is a "0" then the display of the associated function is disabled and that function will be skipped over. If a bit is a "1" the function will be displayed.

Bit 0 = not used
Bit 1 = Brightness
Bit 2 = "Lock" display
Bit 3 = Cal
Bit 4 = Code 1

Bit 5 = Code 2
Bit 6 = Code 3
Bit 7 = Code 4
Bit 8 = Code 5
Bit 9 = Code 6
Bit 10 = Code 7
Bit 11 = Code 8
Bit 12 = Code 9
Bit 13 = Code 10
Bit 14 = not used
Bit 15 = not used

When register 4434 is read it will be displayed as a 16 bit unsigned number. The default value will be 8191 (0x1FFF hex) which is all codes enabled. If, for example, you wished to display the following codes;

Brightness, Cal, Code 1, Code 4 & Code 7

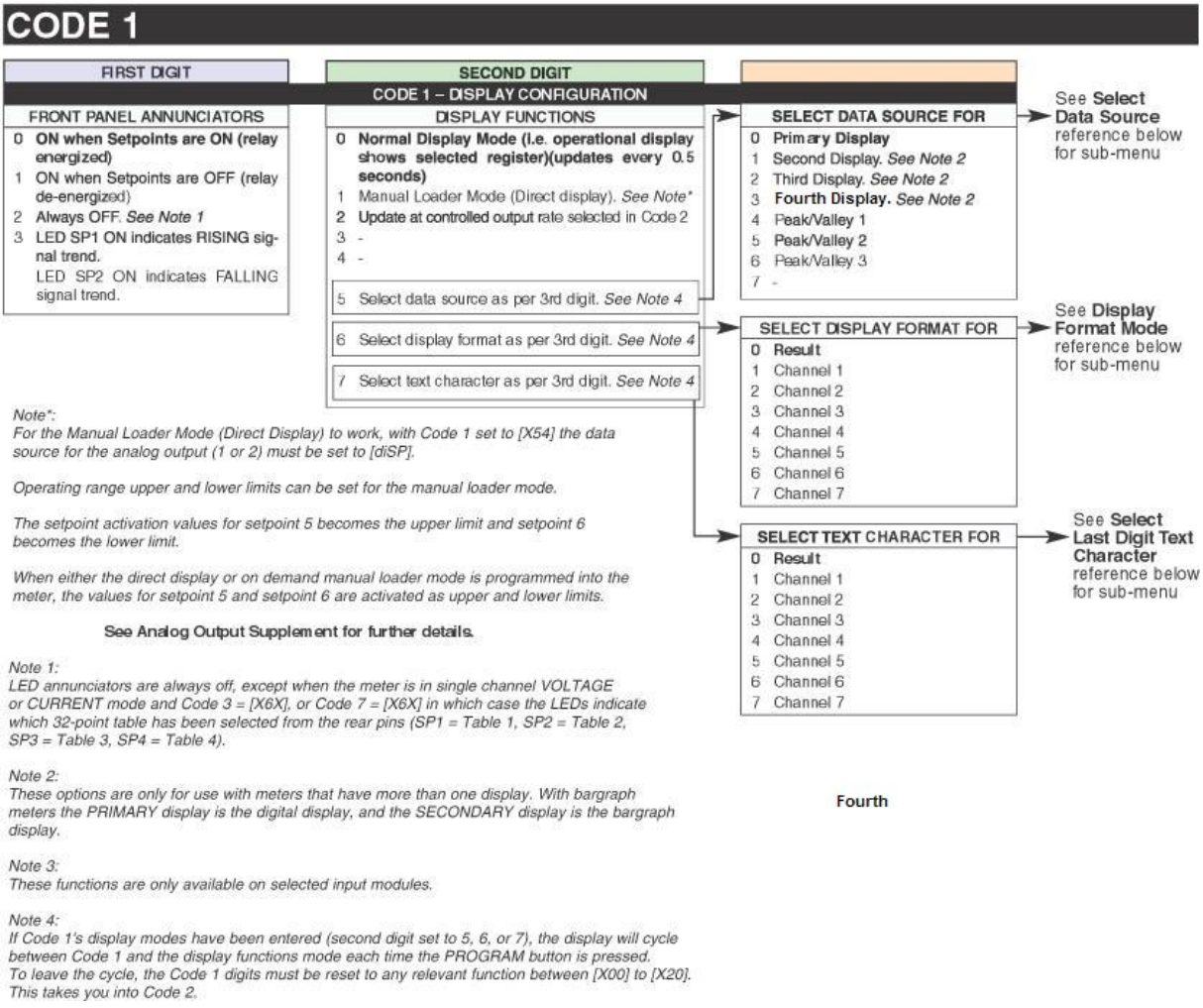
the resulting value for register 4434 would be;

0000 0100 1001 1010b (binary) = 0x049A (hex) = 1178 (decimal)

2.16.3 Code 1

While programming through the front display, the programming digits of Code 1 provide settings for the following display functions:

- Tendency indication thru SP1 & SP2 annunciators (shows rising / falling signal).
- Indication of setpoints / relays operation thru dedicated annunciators.
- Selecting the data source for the display, totalizers 1-6, analog output 1-7, peak / valley 1-3, CH1-CH7 and result.
- Selecting last digit rounding.
- Selecting display units.
- Selecting decimal point position.
- Display manual loader.
- Display with selected update rates (display shows selected register).



See also

Select Data Source 56

Display Format Mode 57

Last Digit Text Character 23

2.16.3.1 Select Data Source

Registers 1 to 16383 are available as the data source for the selected display (1 up to 3), peak / valley (1 to 3), or setpoint.

Following are the most commonly used named registers:

Register Name	Register Number	Register Name	Register Number
Display [DISP]	1	Peak [PEAK_3]	65
Result [RESULT]	7	Valley 3 [VALEY_3]	67
CH1 [CH_1]	9	Tare [TARE]	77
CH2 [CH_2]	11	Auxiliary 1 [AUX_1]	79
CH3 [CH_3]	13	Auxiliary 2 [AUX_2]	81
CH4 [CH_4]	15	Auxiliary 3 [AUX_3]	83
CH5 [CH_5]	17	Auxiliary 4 [AUX_4]	85
CH6 [CH_6]	19	Auxiliary 5 [AUX_5]	87
CH7 [CH_7]	21	Auxiliary 6 [AUX_6]	89
Total 1 [TOT_1]	37	Auxiliary 7 [AUX_7]	91
Total 2 [TOT_2]	39	Auxiliary 8 [AUX_8]	93
Total 3 [TOT_3]	41	Auxiliary 9 [AUX_9]	95
Total 4 [TOT_4]	43	Auxiliary 10 [AUX_10]	97
Total 5 [TOT_5]	45	Auxiliary 11 [AUX_11]	99
Total 6 [TOT_6]	47	Auxiliary 12 [AUX_12]	101
Peak 1 [PEAK_1]	57	Auxiliary 13 [AUX_13]	103
Valley 1 [VALEY_1]	59	Auxiliary 14 [AUX_14]	105
Peak 2 [PEAK_2]	61	Auxiliary 15 [AUX_15]	107
Valley 2 [VALEY_2]	63	Auxiliary 16 [AUX_16]	109

2.16.3.2 Display Format Mode

The three digits in the Display Format Mode allow you to:

- 1st digit Set last digit display rounding.
- 2nd digit Select the display units as octal format 107 or a range of clock 51 options.
- 3rd digit Set the position of the decimal point.

DISPLAY FORMAT MODE

FIRST DIGIT
LAST DIGIT ROUNDING

0 No rounding
1 Rounding by 2's
2 Rounding by 5's
3 Rounding by 10's

SECOND DIGIT
DISPLAY UNITS

0 Decimal
1 24-hour clock mode
Hours: Minutes:Seconds (6-digit version only)
2 12-hour clock mode (12:30 am is displayed as 12:30A. 12:30 pm is displayed as 12:30P)
3 24-hour clock mode
Days: Hours:Minutes (6-digit version only)

Note:
Selecting 1, 2, or 3 in the 2nd digit of this mode configures the display of the selected channel as a clock.

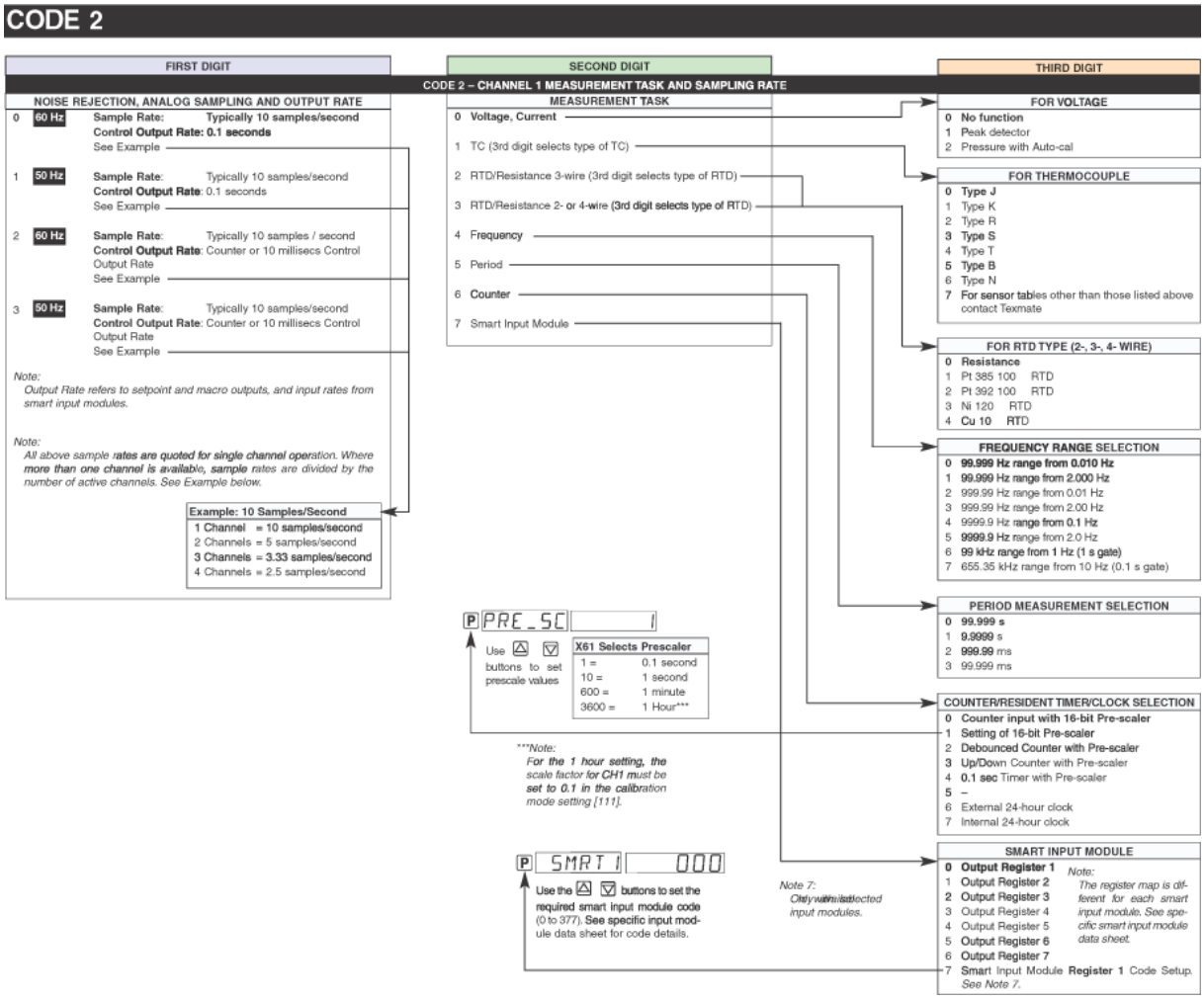
4 -
5 -
6 -
7 Octal

THIRD DIGIT
DECIMAL POINT PLACEMENT

0 No decimal point
1 XX.XX.XX (6 or 8-digit version only)
2 X.XXXXX (6 or 8-digit version only)
3 X.XXXX
4 X.XXX
5 X.XX
6 X.X
7 Decimal Point set from the rear (X.XXXXX to XXXXXX). *See Note 3. Also See Note 4*

2.16.4 Code 2

While programming through the front display, the programming digits of Code 2 provide the settings to select noise rejection, analog sampling rate, and output rate for all input channels. It also allows you to select the input signal type and range setting for CH1.



2.16.5 Code 3

While programming through the front display, the programming digits of Code 3 allow you to select a post processing function for CH1, as well as applying a selected linearization table.

CODE 3

FIRST DIGIT	SECOND DIGIT	THIRD DIGIT
CODE 3 – CHANNEL 1 POST PROCESSING FUNCTIONS		
CHANNEL 1 POST PROCESSING	32-POINT LINEARIZATION FOR CHANNEL 1	NOT USED
0 Direct Display of Input (no processing) 1 Square Root of Channel 1 2 Inverse of Channel 1 3 –	**0 No Linearization on CH1** 1 32-point Linearization on CH1 using Table 1 2 32-point Linearization on CH1 using Table 2. *See Note 5* 3 32-point Linearization on CH1 using Table 3. *See Note 5* 4 32-point Linearization on CH1 using Table 4. *See Note 5* 5 125-point Linearization on CH1 (Tables 1 to 4 cascaded). *See Note 5* 6 32-point Linearization on CH1 (Tables 1 to 4 selected from the rear pins of selected input modules). The selected table is not available if CH2, CH3, or CH4 is operating in the analog output mode. CH1 must be set to Voltage, Current in Code 2 [X0X]. *See Note 5* 7 -	NOT USED

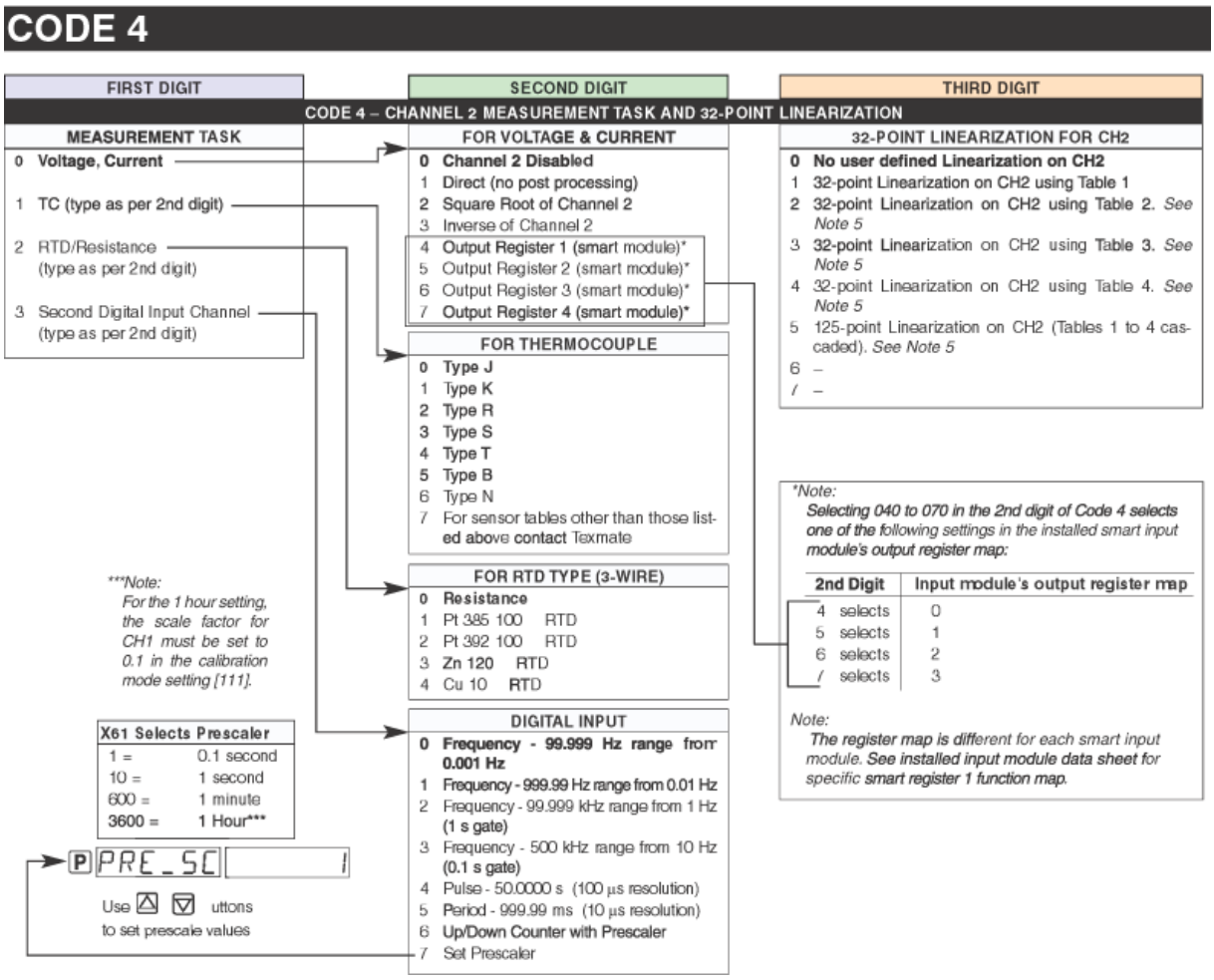
Note 5:
If only 4 kiobits of memory is installed, only Table 1 is available for:

- *CH1 in Code 3, 2nd digit.*
- *CH2 in Code 4, 3rd digit.*
- *CH3 in Code 5, 1st digit.*
- *CH4 in Code 6, 1st digit.*
- *RESULT in Code 7, 2nd digit.*

Note:
All linearization tables are set up in the Calibration Mode [24X].

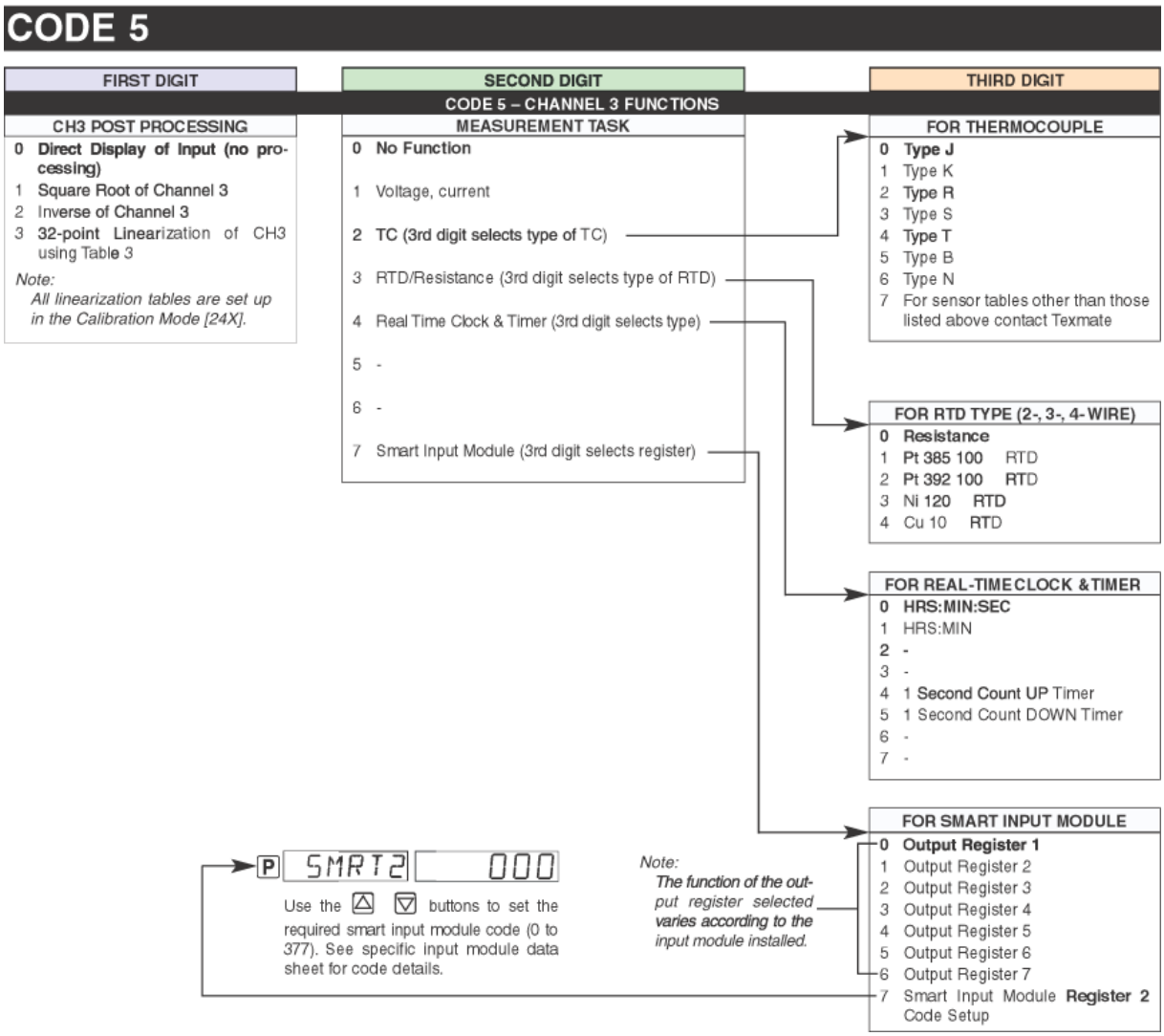
2.16.6 Code 4

While programming through the front display, the programming digits of Code 4 allow you to select the input signal type and range setting for CH2. It also allows you to apply a selected linearization table or set up a smart input module for CH2.



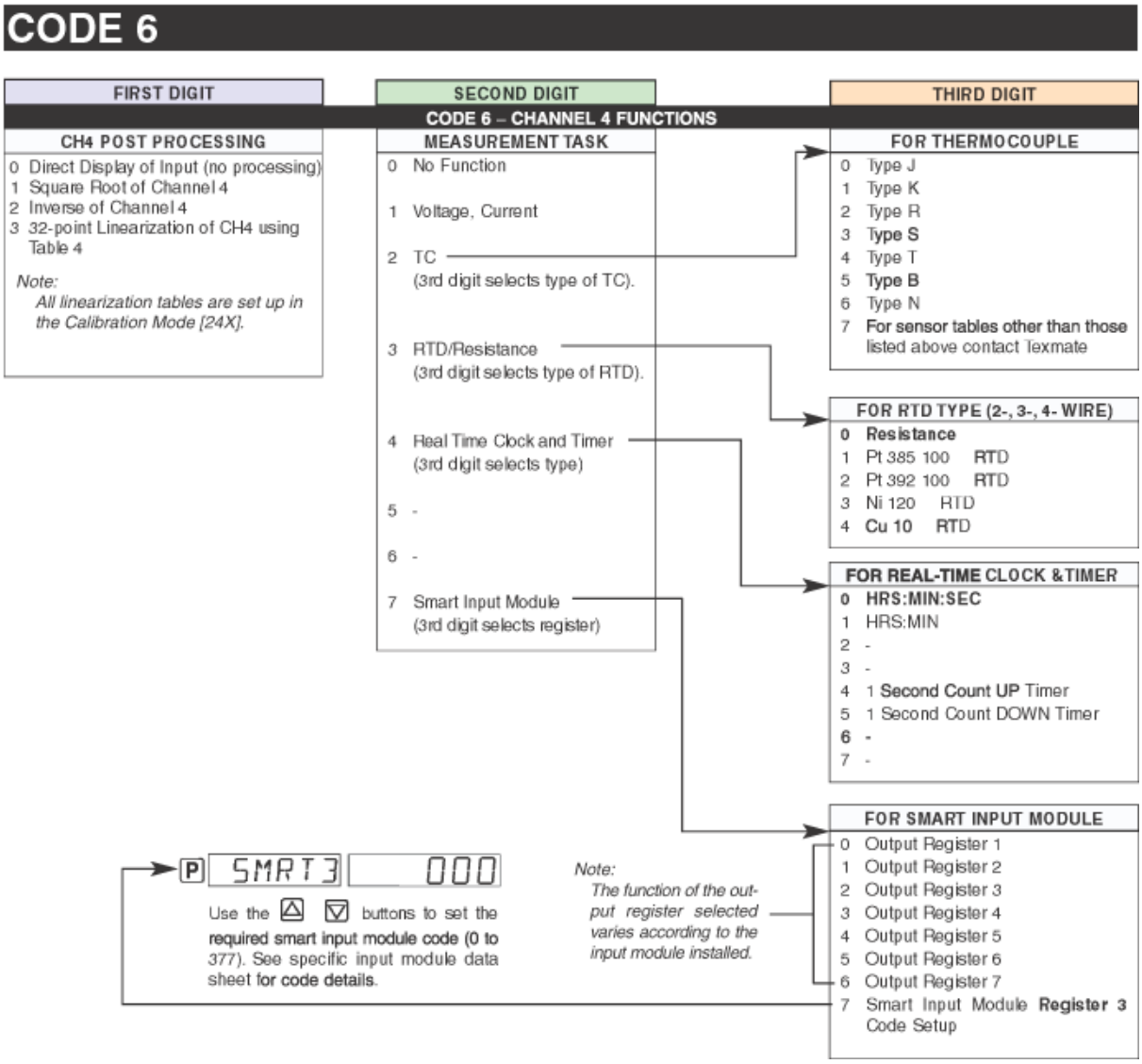
2.16.7 Code 5

While programming through the front display, the programming digits of Code 5 allow you to select a post processing function for CH3, as well as the input signal type and range setting.



2.16.8 Code 6

While programming through the front display, the programming digits of Code 6 allow you to select a post processing function for CH4, as well as the input signal type and range setting.



2.16.9 Code 7

While programming through the front display, the programming digits of Code 7 allow you to select a post processing function for the result. It also allows you to apply a selected linearization table and selected math's functions to the result.

CODE 7

CODE 7 – RESULT PROCESSING

FIRST DIGIT – RESULT PROCESSING

0 **Direct Display of Result as per processing performed in 2nd and 3rd digits**
1 Square Root of Result
2 Inverse of Result
3 -

SECOND DIGIT – 32-POINT LINEARIZATION FOR RESULT

0 **No Linearization on Result**
1 32-point Linearization on Result using Table 1
2 32-point Linearization on Result using Table 2. *See Note 5*
3 32-point Linearization on Result using Table 3. *See Note 5*
4 32-point Linearization on Result using Table 4. *See Note 5*
5 125-point Linearization on Result (Tables 1 to 4 cascaded). *See Note 5*
6 32-point Linearization on Result (Tables 1 to 4 selected from the rear of the meter).
The selected table is not available if CH2, CH3, or CH4 is operating in the analog mode. CH1 must be set to Voltage, Current in Code 2 [X0X].
See Note 5
7 –

THIRD DIGIT – MATHS FUNCTIONS FOR RESULT

0 **Result Register not Updated**
1 pH Meter (CH1 = Tbuff, CH2 = pH)
2 Result = CH1, Setpoint 2 = CH2
3 Result = CH1 + CH2
4 Result = CH1 - CH2
5 Result = CH1 x CH2/10 000
6 Result = (CH1 x 20 000)/CH2
7 Result = Parameter at result source register (#4450)
See Note 8

Note 8:
Register 4450 can only be setup via the serial port or a macro.

Note 5:
If only 4 kilobits of memory is installed, only Table 1 is available for:

- *CH1 in Code 3, 2nd digit.*
- *CH2 in Code 4, 3rd digit.*
- *CH3 in Code 5, 1st digit.*
- *CH4 in Code 6, 1st digit.*
- *RESULT in Code 7, 2nd digit.*

2.16.10 Code 8

While programming through the front display, the programming digits of Code 8 allow you to select data logging and print mode options.

CODE 8

CODE 8 – DATA LOGGING AND PRINT MODE OPTIONS

FIRST DIGIT – DATA LOG BUFFER TYPE

0 **No Data Logging**
1 Cyclic Buffer
2 Linear FIFO Buffer
3 Reset Buffer Number to 0

Note:
Setting Code 8 to [3XX] resets the data log buffer to 0. Once reset, Code 8 must be set back to the required data log buffer setting.

SECOND DIGIT – DATE & TIME STAMP OPTIONS

0 **Printer Format – No time stamp with print/log**
1 Printer Format – Time stamp format 1 [Mth-Day-Yr Hrs:Min:Sec] (with <CR><LF>)
2 Printer Format – Time stamp format 2 [Day-Mth-Yr Hrs:Min:Sec] (with <CR><LF>)
3 Printer Format – Time stamp format 3 [Hrs:Min:Sec] (with <CR><LF>)
4 Spreadsheet Format – No time stamp with print/log
5 Spreadsheet Format – Time stamp format 1 [Mth-Day-Yr Hrs:Min:Sec]
6 Spreadsheet Format – Time stamp format 2 [Day-Mth-Yr Hrs:Min:Sec]
7 Spreadsheet Format – Time stamp format 3 [Hrs:Min:Sec]

ALL ABOVE ARE REAL-TIME CLOCK OPTIONS

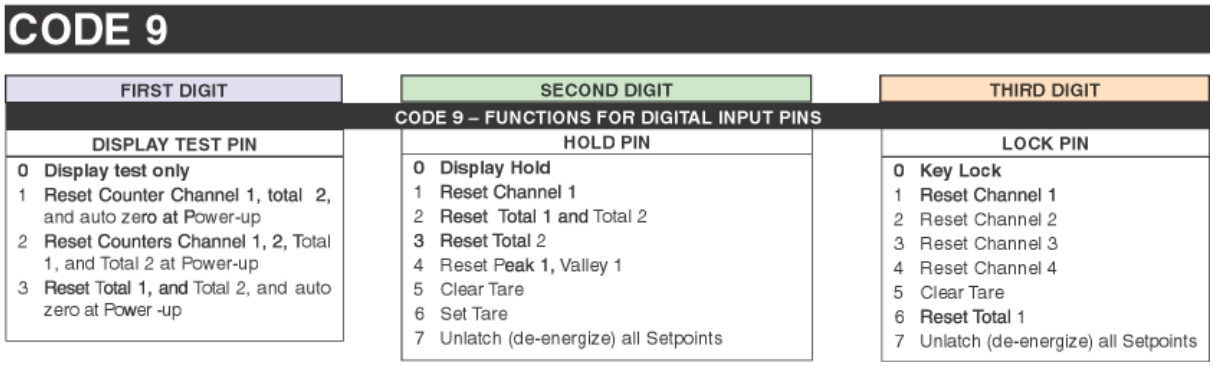
THIRD DIGIT – LOG OR PRINT TRIGGER

0 **No trigger**
1 Trigger on Demand from PROGRAM Button
2 Trigger on Demand from F1 Button
3 Trigger on Demand from F2 Button
4 Trigger on Demand from HOLD Pin
5 Trigger on Demand from LOCK Pin
6 -
7 -

Note:
Log and/or Print will only trigger if enabled.

2.16.11 Code 9

While programming through the front display, the programming digits of Code 9 allow you to set the functions for the DISPLAY TEST, HOLD, and LOCK pins located at the rear of the controller.



CODE 9

CODE 9 – FUNCTIONS FOR DIGITAL INPUT PINS

FIRST DIGIT	SECOND DIGIT	THIRD DIGIT
DISPLAY TEST PIN	HOLD PIN	LOCK PIN
0 Display test only	0 Display Hold	0 Key Lock
1 Reset Counter Channel 1, total 2, and auto zero at Power-up	1 Reset Channel 1	1 Reset Channel 1
2 Reset Counters Channel 1, 2, Total 1, and Total 2 at Power-up	2 Reset Total 1 and Total 2	2 Reset Channel 2
3 Reset Total 1, and Total 2, and auto zero at Power -up	3 Reset Total 2	3 Reset Channel 3
	4 Reset Peak 1, Valley 1	4 Reset Channel 4
	5 Clear Tare	5 Clear Tare
	6 Set Tare	6 Reset Total 1
	7 Unlatch (de-energize) all Setpoints	7 Unlatch (de-energize) all Setpoints

2.17 Data Logging

Most registers from register #1 to register #32765 can be logged. Registers are logged according to what type of register they , with floating point and text registers also able to be logged. See note on Sample Size 67 for more information on data and sample sizes). The ICC402 controllers have been expanded to log up to 32 different channels (prior to OS V4.04.01 only 16 different channels could be logged in one sample).

More than 24,500 samples (data records) can be stored (logged) in internal non-volatile memory for before and after analysis of any process condition. If the external SD card data logger module is used then the number of samples are related to the SD card size.

Note: the ICC402 controller must have either 3328 kilobits of on-board memory installed or an external SD data logging module connected for data logging to function.

Data logging can be triggered (activated) from a setpoint, a front panel button, an external switch, via the serial port or from a macro command. With a real-time clock installed, date and time stamps can be included.

See also
Data Logging Concepts 67

Name	Description	Symbol Type	Register Number	Memory Type
LOG_READ_COUNT	16-bit register. Sets the number of log samples to read using register 16555 (range 0 to 65535).	U_16	4462	RAM/EEPROM 8
LOG_WRITE_POINTER	32-bit register. Points to the most recent data log sample number written by the controller.	U_32	489	RAM/FLASH/SDcard 8
LOG_READ_POINTER	32-bit register. Pointer to the most recent data log sample number read by the controller.	U_32	491	RAM/FLASH/SDcard 8

LOG_REG1	16-bit register. Contains register number of 1st register logged in sample.	U_16	4417	RAM/EEPROM
LOG_REG2	16-bit register. Contains register number of 2nd register logged in sample.	U_16	4418	RAM/EEPROM
LOG_REG3	16-bit register. Contains register number of 3rd register logged in sample.	U_16	4419	RAM/EEPROM
LOG_REG4	16-bit register. Contains register number of 4th register logged in sample.	U_16	4420	RAM/EEPROM
LOG_REG5	16-bit register. Contains register number of 5th register logged in sample.	U_16	4421	RAM/EEPROM
LOG_REG6	16-bit register. Contains register number of 6th register logged in sample.	U_16	4422	RAM/EEPROM
LOG_REG7	16-bit register. Contains register number of 7th register logged in sample.	U_16	4423	RAM/EEPROM
LOG_REG8	16-bit register. Contains register number of 8th register logged in sample.	U_16	4424	RAM/EEPROM
LOG_REG9	16-bit register. Contains register number of 9th register logged in sample.	U_16	4425	RAM/EEPROM
LOG_REG10	16-bit register. Contains register number of 10th register logged in sample.	U_16	4426	RAM/EEPROM
LOG_REG11	16-bit register. Contains register number of 11th register logged in sample.	U_16	4427	RAM/EEPROM
LOG_REG12	16-bit register. Contains register number of 12th register logged in sample.	U_16	4428	RAM/EEPROM
LOG_REG13	16-bit register. Contains register number of 13th register logged in sample.	U_16	4429	RAM/EEPROM
LOG_REG14	16-bit register. Contains register number of 14th register logged in sample.	U_16	4430	RAM/EEPROM
LOG_REG15	16-bit register. Contains register number of 15th register logged in sample.	U_16	4431	RAM/EEPROM
LOG_REG16	16-bit register. Contains register number of 16th register logged in sample.	U_16	4432	RAM/EEPROM

LOG_REG17	16-bit register. Contains register number of 17th register logged in sample.	U_16	4491	RAM/EEPROM 8
LOG_REG18	16-bit register. Contains register number of 18th register logged in sample.	U_16	4492	RAM/EEPROM 8
LOG_REG19	16-bit register. Contains register number of 19th register logged in sample.	U_16	4493	RAM/EEPROM 8
LOG_REG20	16-bit register. Contains register number of 20th register logged in sample.	U_16	4494	RAM/EEPROM 8
LOG_REG21	16-bit register. Contains register number of 21th register logged in sample.	U_16	4495	RAM/EEPROM 8
LOG_REG22	16-bit register. Contains register number of 22th register logged in sample.	U_16	4496	RAM/EEPROM 8
LOG_REG23	16-bit register. Contains register number of 23th register logged in sample.	U_16	4497	RAM/EEPROM 8
LOG_REG24	16-bit register. Contains register number of 24th register logged in sample.	U_16	4498	RAM/EEPROM 8
LOG_REG25	16-bit register. Contains register number of 25th register logged in sample.	U_16	4499	RAM/EEPROM 8
LOG_REG26	16-bit register. Contains register number of 26th register logged in sample.	U_16	4500	RAM/EEPROM 8
LOG_REG27	16-bit register. Contains register number of 27th register logged in sample.	U_16	4501	RAM/EEPROM 8
LOG_REG28	16-bit register. Contains register number of 28th register logged in sample.	U_16	4502	RAM/EEPROM 8
LOG_REG29	16-bit register. Contains register number of 29th register logged in sample.	U_16	4503	RAM/EEPROM 8
LOG_REG30	16-bit register. Contains register number of 30th register logged in sample.	U_16	4504	RAM/EEPROM 8
LOG_REG31	16-bit register. Contains register number of 31st register logged in sample.	U_16	4505	RAM/EEPROM 8
LOG_REG32	16-bit register. Contains register number of 32nd register logged in sample.	U_16	4506	RAM/EEPROM 8

Name	Description	Symbol Type	Register Number	Memory Type
MAX_LOG_SAMPLES	This 32 bit unsigned read only register reports how many log samples are available for the current data logging configuration.	U_32_R	487	RAM 8

See also

Register 487 – Maximum Number Of Log Samples 72

Register 489 – Log Write Pointer 73

Register 491 – Log Read Pointer 73

Register 493 to 523 - Numeric Log Sample Values 73

Registers 4417 to 4432 – Log Register Source 74

Register 4462 – Number Of Log Sample Reads 74

Register 8443 to 8449 - Read Log Sample Data 74

Register 16553 – Read Single Log Data At Log Read Pointer 75

Register 16555 – Read Log Data At Log Read Pointer 77

Read Only Registers 69

2.17.1 Data Logging Concepts

The data logging function uses the concept of pointers to control where a sample is to be written to and from where one is to be read. These pointers are referred to as the log write pointer and the log read pointer.

Register 489 – Log Write Pointer
Register 489 is a 32-bit register that points to the most recent log sample written by the controller. It counts up from 0 each time a new sample is logged, with the maximum number of samples being limited by the size of non-volatile memory installed in the controller and also the number/size of registers to be logged. Before a new sample is written, the controller first checks to make sure that it is not overwriting a sample that has not been read. It does this by comparing the write pointer with the read pointer. If they are the same and the Linear 67 logging mode has been selected, data logging is halted until a read is actioned. If this occurs, new samples are lost. If the Cyclic 67 mode has been selected, the oldest sample will be overwritten with new data and the old sample will be lost. When the sample number reaches the maximum count it wraps around to 1.

Register 489 can be read from or written to. Make sure that any values written to this pointer are within the allowable range for the size of the installed memory.

Register 491 – Log Read Pointer
Register 491 is a 32-bit register that points to the most recent log sample read from the controller. It counts up from 0 each time log data is read from the controller, with the maximum number of samples being limited by the size of non-volatile memory installed in the controller and also the number/size of registers to be logged. When it reaches the maximum count it wraps around to 1. When it reaches the write pointer the log buffer is empty and no more data can be read out of the log.

Register 491 can be read from or written to. Make sure that any values written to this pointer are within the allowable range for the size of the installed memory.

Note: Although the log read and write pointers can be reset to zero, sample zero is never used to hold any real sample data. It is only used as "resting point" when the pointers are cleared. This is because the pointers are always pre-incremented before the sample is written. When pointers wrap around at the end of memory they wrap around to the value of 1.

Buffer Types
The controller has two types of buffer.

Cyclic Buffer. With the **cyclic** buffer selected in the Meter Configuration Utility program, the log write pointer (register 489) increments each time a sample is taken. When it exceeds the maximum sample number (determined by the amount of non-volatile memory installed and the number/size of registers to be logged) it wraps around to zero. If the write pointer equals the read pointer then oldest (unread) data will be overwritten with the new data and old data will lost. This means that when the cyclic buffer is full, the logged data is replaced on a first ON first OFF basis. This means that when the buffer is full, the first logged sample is discarded to make way for a new sample at the end of the logged data string. It then wraps around to sample number 1 again.

See description Register 491 - Log Read Pointer 67 for information about not overwriting old samples

that have not been read.

Linear Buffer. With the **linear** buffer selected in the Meter Configuration Utility program, the log write pointer increments each time a sample is taken until it reaches the read pointer. When it equals the read pointer the controller stops logging data and any new data is lost. If the sample number reaches the maximum sample number (determined by the amount of non-volatile memory installed and the number/size of registers to be logged) it will wrap around to zero. When the linear buffer is full it must either be read or reset to 0. See Reset Buffer 67.

Reset Buffer
With **reset buffer number to 0** set in the Meter Configuration Utility program, the log write and log read pointers are reset to zero when the PROGRAM button is pressed. The controller then reverts back to the same setting it had before the reset function was executed (either cyclic or linear). Note that when the reset function is executed, the contents of the buffer is not destroyed, only the pointers are changed.

Registers 4417 to 4432
Registers 4417 to 4432 can be read from and written to as normal registers. Registers 4417 to 4432 can only be configured in the Meter Configuration Utility program (via the serial port) or from the macro and are not accessible from the front panel buttons.

Registers 4417 to 4432 are used to specify which registers are to be logged. Register 4417 specifies the first register to be logged, 4418 the second, 4419 the third, and so on. Up to 16 registers in total can be logged in each sample (*see note below about additional 16 registers for later versions of OS). The controller will adjust the amount of memory required to store each sample based on the number of registers to be logged and the data size required for each register. As the sample size increases the maximum number samples available for a given amount of memory decreases.

***NOTE:**
On software versions V4.04.01 and higher, an additional 16 registers (4491 - 4506) are available to expand the data logging capability to log up to 32 registers per sample.

Writing a value of zero to one of these registers disables the register from taking any logs.

Sample Size
When logging with on-board memory the overall sample size can only be one of the following 4 possible options:
- 16 bytes
- 32 bytes
- 64 bytes
- 128 bytes

Each sample will always include:
- 1 byte required for trigger source.
- 3 bytes required for date stamp.
- 3 bytes required for time stamp.
- 1 byte required for checksum (last byte in sample).

The data in each sample can be made of a combination of any of the following data types:
- 8-bit registers use: 1 byte
- 16-bit registers use: 2 bytes
- 24-bit registers use: 3 bytes
- 32-bit registers use: 4 bytes
- Text registers: Number of bytes depends on length of text string.
- (Custom Text strings can be stored but only with special macro command "log_message" or via a

serial port write to register 16553. See Read Single Log Data at Log Read Pointer - Register 16553 75)

Non-volatile Memory Options
When the controller is fitted with the internal data logging memory option, then 3328 kilobits of non-volatile on-board memory (EEPROM) is installed. This allows up to 24,576 samples to be logged.

Maximum Number Of Samples
The maximum number of samples available depends on the size of the installed memory and the number/type of registers stored in each sample. Altering registers 4417 to 4432 may effect the sample size and the maximum number of samples. The maximum number of samples can be determined from the configuration Utility on the data logging page or by reading register 487.

Typical sample sizes are shown below.
- 24,576 samples logging 1-2 x 32bit registers + time stamp
- 12,288 samples logging 3-6 x 32bit registers + time stamp
- 6,144 samples logging 7-14 x 32bit registers + time stamp
- 3,072 samples logging 15-16 x 32bit registers + time stamp

NOTE: Altering registers 4417 to 4432 may effect the sample size and the maximum number of samples and may also render any previously stored log data as unreadable. All current log data should be read and saved before changes are made to these registers and they should be correctly configured before any new samples are taken.

Special Log Functions
Most data log samples are trigger by a setpoint or from an input pin but log samples can also be triggered from other sources such as special macro commands or from the serial port. There are 2 special macro commands which allow log samples to be triggered as shown below.

"force_log" command - This macro command trggiers a log sample to be taken in the standard format. The sample is taken the instant the command is executed.

"log_message" command - This macro command allows a text string to be logged by the data logger. It requires a following text string enclosed in quotation marks (") and allows the logging of custom messages from the macro. The maximum length of the text string is governed by the data logging settings for registers 4417 - 4432.

There are also 2 ways of triggering a data log from the serial port as shown below.

Write to register 8442 - a write to register 8442 via the serial port will trigger a log sample to be taken in the standard format. It allows log samples to be triggered from another serial device.

Write to register 16553 - a write to register 16553 via the serial port allows a text string to be logged in a similar manner to the "log_message" macro command. In this case the text string to be logged is included in the serial command in a similar manner to writing to other text registers.

2.17.2 Read Only Registers

These read only registers are provided to allow the user to selectively read a single parameter from a log sample, instead of reading all parameters in a sample. Only numeric parameters can be read (not text) and they will always be drawn from the sample which the log read pointer is currently pointing to. The user must ensure that the log read pointer is pointing to the correct sample of number before reading these registers. On previous firmware versions (earlier than 4.04.01), reading any of these registers does not alter the log read pointer position. On the ICC402 (versions 4.04.01 and later) an auto increment feature has been added.

Auto Increment Feature

The ICC402 incorporates an auto increment feature to simplify searches of the data log memory. If different log sample values are read from the same sample number the read pointer is unchanged (as with older versions). However, if the same log sample register is read consecutively (without accessing any other log sample register and without manually changing the log read pointer), then the log read pointer value is automatically incremented before each read.

Name	Description	Symbol Type	Register Number	Memory Type
LOG_SAMPLE_TRIGGER 74	Read only register. Trigger source of current log sample.	U_R	8443 74	EEPROM/SDcard 8
LOG_SAMPLE_DATE 74	Read only register. Returns 8-bit value for date of current log sample (range 1 to 31 days).	U_R	8444 74	EEPROM/SDcard 8
LOG_SAMPLE_MONTH 74	Read only register. Returns 8-bit value for months of current log sample (range 1 to 12 months).	U_R	8445 74	EEPROM/SDcard 8
LOG_SAMPLE_YEAR 74	Read only register. Returns 8-bit value for year of current log sample (range 00 to 99 years).	U_R	8446 74	EEPROM/SDcard 8
LOG_SAMPLE_HOUR 74	Read only register. Returns 8-bit value for hours of current log sample (range 0 to 23 hours).	U_R	8447 74	EEPROM/SDcard 8
LOG_SAMPLE_MINUTE 74	Read only register. Returns 8-bit value for minutes of current log sample (range 0 to 59 minutes).	U_R	8448 74	EEPROM/SDcard 8
LOG_SAMPLE_SECOND 74	Read only register. Returns 8-bit value for seconds of current log sample (range 0 to 59 seconds).	U_R	8449 74	EEPROM/SDcard 8
LOG_SAMPLE_HUNDREDTHS 74	Read only register. Returns 8-bit value for 1/100 of a second of current log sample (range 0 to 99 hundredths of a second).	U_R	8450 74	EEPROM/SDcard 8

LOG_SAMPLE_REG1 73	Read only register. Returns 1st data value for logged in current log sample (range depends on size of 1st register).	S_R	493 73	EEPROM/SDcard 8
LOG_SAMPLE_REG2 73	Read only register. Returns 2nd data value for logged in current log sample (range depends on size of 2nd register).	S_R	495 73	EEPROM/SDcard 8
LOG_SAMPLE_REG3 73	Read only register. Returns 3rd data value for logged in current log sample (range depends on size of 3rd register).	S_R	497 73	EEPROM/SDcard 8
LOG_SAMPLE_REG4 73	Read only register. Returns 4th data value for logged in current log sample (range depends on size of 4th register).	S_R	499 73	EEPROM/SDcard 8
LOG_SAMPLE_REG5 73	Read only register. Returns 5th data value for logged in current log sample (range depends on size of 5th register).	S_R	501 73	EEPROM/SDcard 8
LOG_SAMPLE_REG6 73	Read only register. Returns 6th data value for logged in current log sample (range depends on size of 6th register).	S_R	503 73	EEPROM/SDcard 8
LOG_SAMPLE_REG7 73	Read only register. Returns 7th data value for logged in current log sample (range depends on size of 7th register).	S_R	505 73	EEPROM/SDcard 8
LOG_SAMPLE_REG8 73	Read only register. Returns 8th data value for logged in current log sample (range depends on size of 8th register).	S_R	507 73	EEPROM/SDcard 8
LOG_SAMPLE_REG9 73	Read only register. Returns 9th data value for logged in current log sample (range depends on size of 9th register).	S_R	509 73	EEPROM/SDcard 8
LOG_SAMPLE_REG10 73	Read only register. Returns 10th data value for logged in current log sample (range depends on size of 10th register).	S_R	511 73	EEPROM/SDcard 8
LOG_SAMPLE_REG11 73	Read only register. Returns 11th data value for logged in current log sample (range depends on size of 11th register).	S_R	513 73	EEPROM/SDcard 8
LOG_SAMPLE_REG12 73	Read only register. Returns 12th data value for logged in current log sample (range depends on size of 12th register).	S_R	515 73	EEPROM/SDcard 8
LOG_SAMPLE_REG13 73	Read only register. Returns 13th data value for logged in current log sample (range depends on size of 13th register).	S_R	517 73	EEPROM/SDcard 8
LOG_SAMPLE_REG14 73	Read only register. Returns 14th data value for logged in current log sample (range depends on size of 14th register).	S_R	519 73	EEPROM/SDcard 8
LOG_SAMPLE_REG15 73	Read only register. Returns 15th data value for logged in current log sample (range depends on size of 15th register).	S_R	521 73	EEPROM/SDcard 8
LOG_SAMPLE_REG16 73	Read only register. Returns 16th data value for logged in current log sample (range depends on size of 16th register).	S_R	523 73	EEPROM/SDcard 8

LOG_SAMPLE_REG17 73	Read only register. Returns 17th data value for logged in current log sample (range depends on size of 17th register).	S_R	545 73	EEPROM/SDcard 8
LOG_SAMPLE_REG18 73	Read only register. Returns 18th data value for logged in current log sample (range depends on size of 18th register).	S_R	547 73	EEPROM/SDcard 8
LOG_SAMPLE_REG19 73	Read only register. Returns 19th data value for logged in current log sample (range depends on size of 19th register).	S_R	549 73	EEPROM/SDcard 8
LOG_SAMPLE_REG20 73	Read only register. Returns 20th data value for logged in current log sample (range depends on size of 20th register).	S_R	551 73	EEPROM/SDcard 8
LOG_SAMPLE_REG21 73	Read only register. Returns 21th data value for logged in current log sample (range depends on size of 21th register).	S_R	553 73	EEPROM/SDcard 8
LOG_SAMPLE_REG22 73	Read only register. Returns 22th data value for logged in current log sample (range depends on size of 22th register).	S_R	555 73	EEPROM/SDcard 8
LOG_SAMPLE_REG23 73	Read only register. Returns 23th data value for logged in current log sample (range depends on size of 23th register).	S_R	557 73	EEPROM/SDcard 8
LOG_SAMPLE_REG24 73	Read only register. Returns 24th data value for logged in current log sample (range depends on size of 24th register).	S_R	559 73	EEPROM/SDcard 8
LOG_SAMPLE_REG25 73	Read only register. Returns 25th data value for logged in current log sample (range depends on size of 25th register).	S_R	561 73	EEPROM/SDcard 8
LOG_SAMPLE_REG26 73	Read only register. Returns 26th data value for logged in current log sample (range depends on size of 26th register).	S_R	563 73	EEPROM/SDcard 8
LOG_SAMPLE_REG27 73	Read only register. Returns 27th data value for logged in current log sample (range depends on size of 27th register).	S_R	565 73	EEPROM/SDcard 8
LOG_SAMPLE_REG28 73	Read only register. Returns 28th data value for logged in current log sample (range depends on size of 28th register).	S_R	567 73	EEPROM/SDcard 8
LOG_SAMPLE_REG29 73	Read only register. Returns 29th data value for logged in current log sample (range depends on size of 29th register).	S_R	569 73	EEPROM/SDcard 8
LOG_SAMPLE_REG30 73	Read only register. Returns 30th data value for logged in current log sample (range depends on size of 30th register).	S_R	571 73	EEPROM/SDcard 8
LOG_SAMPLE_REG31 73	Read only register. Returns 31st data value for logged in current log sample (range depends on size of 31st register).	S_R	573 73	EEPROM/SDcard 8
LOG_SAMPLE_REG32 73	Read only register. Returns 32nd data value for logged in current log sample (range depends on size of 32nd register).	S_R	575 73	EEPROM/SDcard 8

2.17.3 Maximum Number Of Log Samples - Register 487

Register 487 is a 32-bit read only register that reports the maximum number of log samples available. The maximum number of log samples is defined by the amount of memory installed, how many registers are being logged, and the size of each register to be logged.

2.17.4 Log Write Pointer - Register 489

Register 489 is a 32-bit register that points to the most recent log sample written by the controller. It automatically increments by one count just before a new sample is loggged and counts up from 0 with the maximum number of samples being limited by the current memory size installed in the controller and the setting of registers 4417 to 4432. When it reaches the maximum count it wraps around to 1. When it catches up to the log read pointer (491) it either overwrites the old unread data or stops logging, depending on which type of buffer has been selected in the Meter Configuration Utility program (i.e. cyclic or linear).

Register 489 can be read or written to. You must make sure that any values written to this pointer are within the allowable range for the installed memory size.

Note: Although the log read and write pointers can be reset to zero, sample zero is never used to hold any real sample data. It is only used as "resting point" when the pointers are cleared. This is because the pointers are always pre-incremented before the sample is written. When pointers wrap around at the end of memory they wrap around to the value of 1.

2.17.5 Log Read Pointer - Register 491

Register 491 is a 32-bit register that points to the most recent log sample read from the controller. It counts up from 0 each time log data is read from the controller, with the maximum number of samples being limited by the current memory size installed in the controller and the setting of registers 4417 to 4432. When it reaches the maximum count it wraps around to 1. When it reaches the log write pointer, the data log buffer is empty and it stops.

Register 491 can be read or written to. You must make sure that any values written to this pointer are within the allowable range for the installed memory size.

Note: Although the log read and write pointers can be reset to zero, sample zero is never used to hold any real sample data. It is only used as "resting point" when the pointers are cleared. This is because the pointers are always pre-incremented before the sample is written. When pointers wrap around at the end of memory they wrap around to the value of 1.

2.17.6 Numeric Log Sample Values - Register 493 to 523

Registers 493 to 523 and 545 to 575 provide numeric values for the 1st through to the 32nd register logged in accordance with registers 4417 to 4432 64 and 4491 to 4506 64. The size and type varies depending on the size and data type of the registers addressed by registers 4417 to 4432 and 4491 to 4506. These registers give an unformatted numeric value that can be read in ASCII or any other serial mode.

The user should ensure that the log read pointer is set to the required sample number before accessing any of these registers. The ICC402 incorporates an Auto Increment Feature 69. (In firmware versions earlier than V4.04.01, a read of one of these registers did not modify the log read or the log write pointer).

If the sample does not contain the data that is requested, the controller responds by sending a null character in ASCII mode, or a data error in Modbus mode.

NOTE:
Registers 545 to 575 and registers 4491 to 4506 are not available on older code versions prior to V4.04.01.

2.17.7 Log Register Source - Registers 4417 to 4432

Registers 4417 to 4432 are used to specify which registers the data logger logs. Register 4417 specifies the first register to be logged, 4418 the second, 4419 the third and so on. Setting one of these registers to zero disables that register from logging any data. Changing registers 4417 to 4432 potentially changes the sample size and the maximum number of samples and may also render any previously stored log data as unreadable. All current log data should be read and saved before changes are made to these registers and they should be correctly configured before any new samples are taken.

Registers 4417 to 4432 can be read from and written to as normal registers. Most of the registers in the controller can be logged. This includes floating point and text registers.

NOTE: Text registers 16553, 16555 and 16543 should not be logged.

2.17.8 Number Of Log Sample Reads - Register 4462

Register 4462 defines the maximum number of log samples that will be output when register 16555 is read. This can be set to any number between 1 and 65535 with a default of 100 samples.

2.17.9 Read Log Sample Data - Register 8443 to 8450

Registers 8443 to 8450 are read only registers used to access data from a log sample. A read of one of these registers reads the appropriate data from the log sample which is addressed by the current value of the log read pointer (register 491). The user must setup the log read pointer to the required sample number before accessing registers 8443 to 8450. The ICC402 incorporates an Auto Increment Feature 69 which automatically increments the read pointer if consecutive reads are made to the same register. (In firmware versions earlier than V4.04.01, a read of one of these registers will not modify the log read or the log write pointer).

In each case the output is a numeric value only. Registers 8443 to 8450 can be read in ASCII or Modbus modes and can also be read from the macro.

NOTE: The information in registers 8443 to 8450 is logged in every sample, regardless of the settings in the Meter Configuration Utility program.

Register 8443 – Trigger type for sample
This register provides an 8-bit numeric value that defines the trigger point for the sample.

Numeric Value	Function
0	Triggered by reset
1	Triggered by setpoint 1
2	Triggered by setpoint 2
3	Triggered by setpoint 3
4	Triggered by setpoint 4
5	Triggered by setpoint 5
6	Triggered by setpoint 6
7	Triggered by setpoint 7
8	Triggered by setpoint 8
9	Triggered by setpoint 9
10	Triggered by setpoint 10
11	Triggered by setpoint 11
12	Triggered by setpoint 12
13 - 16	Reserved for future development
17	Triggered by Program button
18	Triggered by F1 button
19	Triggered by F2 button
20	Triggered by Hold pin
21	Triggered by Lock pin
22 - 23	Reserved for future development
24	Triggered by executing the macro instruction "force_log" or by a serial port write to register 8442.
25	Triggered by executing the macro instruction "log_message" or by a text string write to register 16553 via the serial port.

Register 8444 – Date of sample
This register provides an 8-bit numeric value for the date.

Register 8445 – Month of sample
This register provides an 8-bit numeric value for the month.

Register 8446 – Year of sample
This register provides an 8-bit numeric value for the last 2 digits of the year.

Register 8447 – Hour of sample
This register provides an 8-bit numeric value for the hours.

Register 8448 – Minute of sample
This register provides an 8-bit numeric value for the minutes.

Register 8449 – Second of sample
This register provides an 8-bit numeric value for the seconds.

Register 8450 – 1/100 Second of sample
This register provides an 8-bit numeric value for hundredths of a second.

2.17.10 Read Single Log Data at Log Read Pointer - Register 16553

Register 16553 is used to read the next sample of log data. It does this by comparing the log read pointer (register 491) with the log write pointer (register 489). If they are equal then there has been no new samples logged since the last read and the message **No New Log Data** is sent as a response. If they are not equal, the log read pointer (register 491) is incremented to point to the new sample and

the new log data is transmitted. Registers 489 and 491 can be used to control the data logger. To reset the data logger, both register 489 and 491 should be set to 0 (or any other value in the allowable range of memory).

Reading Register 16553 In ASCII Mode
Although register 16553 can be read in ASCII serial mode, it is not recommended if more than 10 channels of data are being logged (see note below on buffer overflow problems). Modbus mode is the recommended mode to read large amounts of data logging memory. To read log data in other serial modes, see **registers 493 to 523 and registers 8443 to 8450** 69.

Modbus Read Of 16553
Register 16553 can also be read in Modbus mode. Modbus mode should be used if logging more than 10 channels or if using an external data logging SD card with large numbers of samples. It provides a faster and more efficient method of downloading large amounts of data than via the ASCII mode and does not suffer from buffer overflow problems (see note below)

A read of register 16553 in Modbus mode produces a raw output format which needs to be decoded by the user. The data bytes within the Modbus packet is formatted as follows;

Fixed Format Section
Data bytes 1 to 7 of each sample have a fixed format as follows;

1st data byte = Log trigger (see Register 8443 – Trigger type for sample 74)
2nd data byte = Date (see Register 8444 – Date of sample 74)
3rd data byte = Month (see Register 8445 – Month of sample 74)
4th data byte = Year (see Register 8446 – Year of sample 74)
Data bytes 5-7 = Time stamp as 24 bit unsigned number in 1/100 second resolution.

Variable Format Section
Data bytes 8 and onwards do not have a fixed format. The format of these data bytes is determined by the data logging configuration (i.e. how many registers are being logged and which type of registers are being logged). The length of the data string is also effected by data logging configuration. To determine how many registers need to be read and how to decode them, the user should first read register 16551, the data logging format register.

Writing To Register 16553
A write to register 16553 via the serial port can be used to log a custom text string into data logging memory. The maximum length of the text string is governed by the values written to registers 4417 to 4432 as these effectively control the sample size. In ASCII mode the format would be;

SW16553,Hello World*

This allows text messages from another source to be time stamped and logged along with standard log samples. Data logging must be enabled in the Meter Configuration Utility program for this feature to function. (See Code 8 63)

A standard log sample can also be trigger from the serial port by writing any value (normally 0) to register #8442. This will cause the registers selected by 4417 - 4432 to be logged and time stamped in the standard format.

Data Log Format - Register 16551
Register 16551 can only be read in Modbus mode. A read of register 16551 will produce the following data.

Byte 1 = length (number of data bytes in packet)
Byte 2 = Sample size for data log
Byte 3,4 = Controller software version (e.g. 403)
Byte 5 = Not Used in ICC402 (always read as 0)
Byte 6 = Time format (current configuration of Code_8 63)

Byte 7,8 = Data log source register
Byte 9 = Register type
Byte 10 = Display format
Byte 11 = Alpha character
| repeated for each register that is logged

If the data logging configuration (see Data Logging 64 and Log Register Source - Registers 4417 to 4432) has been set up to log 4 registers per sample, then data bytes 7 - 11 will be repeated for each of the 4 logged registers.

The length byte defines how many bytes of data are contained in the data log format string.
The sample size byte gives the current data log sample size which is determined by the data logging configuration.
The time format byte is basically a copy of Code 8 and specifies how the time/date information is to be displayed.
The two data bytes which form the 16 bit data log source register define the register number of the logged data. The register type byte defines the size and format of the data bytes in the log sample. The different options available are;

0x00 = unsigned char
0x10 = signed char
0x20 = unsigned int
0x30 = signed int
0x40 = unsigned long
0x50 = signed long
0x60 = ASCII TEXT
0x70 = float
0x80 = unsigned 24 bit number(only 3 bytes long)
0x90 = signed 24 bit number(only 3 bytes long)

The display format and alpha character bytes define how the numeric value is to be displayed (i.e. were the decimal point should be and what the trailing text character (if any) should be). (see Display Format Mode for more info on how to interpret the display format byte)

NOTE:
When logging more than 10 registers per sample a buffer overflow can occur when downloading the log data in the printer format via ASCII mode, resulting in truncated output data. If this happens, try switching to spreadsheet format. If you are still experiencing problems then you should consider using Modbus mode as described above.

2.17.11 Read Log Data at Log Read Pointer - Register 16555

Register 16555 is a read only register similar to register 16553, except that it is used to read multiple log samples with a single command. A read of register 16555 outputs log data, starting at the sample pointed to by the read pointer, and continues to output log samples until the read pointer equals the write pointer, or the number of samples specified in register 4462 has been output. Each time it outputs a log sample the read pointer is automatically incremented. At the end of the sequence, the read pointer is equal to the write pointer. This command can also be used to read a selected block of

log samples by modifying the read pointer (register 491) and the write pointer (register 489) prior to reading register 16555.

NOTE: Register 16555 can only be read via the serial port in ASCII mode. To read log data in other serial modes or from the macro, see registers 493 to 523 and registers 8443 to 8450 69.

2.17.12 External Data Logger

If the ICC402 is being operated with the external data logging module (i.e. SD memory card with high precision clock) then the following additional registers are also available.

Register 8428 - Software Version Number
Register 8429 is an 8 bit unsigned read only register which reports the software version number of the external data logger module.

Register 8429 - Status
Register 8429 is an 8 bit unsigned read only register which reports the current status of the extended data logging/real time clock module. The bit functions for the status register are defined below;

Bit Number	Bit Function	
0	No RTC detected (0), RTC detected (1)	This bit defines whether the module has a high precision real time clock fitted or not.
1	RTC osc. fault	This bit is set when the real time clock oscillator has been stopped or corrupted in some way. This often indicates that the back up battery needs replacement.
2	Data range fault	This bit is set when one of the real time clock registers contains a value which is outside of the allowable range.
3	Read pointer error	This signifies that the read pointer in the extended data logging module has been corrupted. This does not mean that the data has been lost but you will need to reset the read pointer in order to recover the data.
4	Write pointer error	This signifies that the write pointer in the extended data logging module has been corrupted. This does not mean that the data has been lost but you will need to reset the write pointer and then read the data out before continuing to write any new data.
5	Incompatible card type	This indicates that the card currently inserted in the extended data logging module is incompatible. It may be that the card is too large or too small (128 MB or 256 MB is the preferred size) or it may be that the card is the wrong type. Also make sure that the contacts on the card are clean and free of dust and other residue.
6	Unformatted card	This indicates that the SD card has not been formatted correctly. Please ensure that the card contains a valid Master Boot Record (MBR) with its partition table and that the first partition is of type FAT16 or FAT32 (FAT16 preferred). Some cards are shipped without a partition table by default, instead starting the FAT root sector in the first sector of the card. A partition table can be created with fdisk (this will erase all data on the card) and then formatting the newly created partition. Also make sure that there is enough linear free space available on the card. Using a freshly formatted card is recommended, but if some files are already on the card we suggest to defrag the file system before inserting the card into the data logger module for the first time. The data logger will leave some free space at the end of the card, this is normal and intended. Please allow up to 60 seconds when inserting a card for the first time for the data logger to create the file used to store the samples.
7	Card ready	This indicates that the data logging card is ready for use.

Register 8432 - Data Log Memory Size
Register 8432 is an 8 bit unsigned read only register which indicates the size of the SD memory card which is currently inserted. This byte is structured in a similar manner to a floating point number with the most significant 4 bits containing the exponent and the least significant 4 bits containing the

mantissa. Here are some examples of how to decode this byte.

Bits 4 - 7 = EXPONENT, Bits 0 - 3 = MANTISSA
eg. 0x04 = 4 x 2^0 = 4 MB
0x08 = 8 x 2^0 = 8 MB
0x18 = 8 x 2^1 = 16 MB
0x28 = 8 x 2^2 = 32 MB
0x38 = 8 x 2^3 = 64 MB
0x48 = 8 x 2^4 = 128 MB
0x58 = 8 x 2^5 = 256 MB

2.18 Digital

Digital Output Registers

The following registers provide read/write access to the digital output pins on the 16/16 IO expansion modules. Access can be gained via 32 bit registers or via 16 bit registers for use with Modbus. These registers are designed to be accessed via the serial ports or a macro, and require one or more 16/16 IO expansion modules to be connected to the controller.

Name	Description	Symbol Type	Register Number	Memory Type
DIGITAL_OUT	32 bit register with bit flags for outputs DO_1 to DO_32 on multi I/O modules.	U_32	249	RAM
DIGITAL_OUT_LOW	16 bit register with bit flags for outputs DO_1 to DO_16 on multi I/O modules.	U_16	4509	RAM
DIGITAL_OUT_HIGH	16 bit register with bit flags for outputs DO_17 to DO_32 on multi I/O modules.	U_16	4510	RAM
DIGITAL_OUT2	32 bit register with bit flags for outputs DO_33 to DO_64 on multi I/O modules.	U_32	253	RAM
DIGITAL_OUT2_LOW	16 bit register with bit flags for outputs DO_33 to DO_48 on multi I/O modules.	U_16	4511	RAM
DIGITAL_OUT2_HIGH	16 bit register with bit flags for outputs DO_49 to DO_64 on multi I/O modules.	U_16	4512	RAM

Register 249 - Digital Out

Register 249 is a 32-bit register that contains bit flags to control up to 32 digital outputs. The bit functions for register 249 are as follows

Bit	Name	Description	Function
0	DO_1	Digital output 1. Bit flag that shows / controls the status of digital output 1 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DO_2	Digital output 2. Bit flag that shows / controls the status of digital output 2 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DO_3	Digital output 3. Bit flag that shows / controls the status of digital output 3 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DO_4	Digital output 4. Bit flag that shows / controls the status of digital output 4 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DO_5	Digital output 5. Bit flag that shows / controls the status of digital output 5 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DO_6	Digital output 6. Bit flag that shows / controls the status of digital output 6 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DO_7	Digital output 7. Bit flag that shows / controls the status of digital output 7 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_8	Digital output 8. Bit flag that shows / controls the status of digital output 8 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DO_9	Digital output 9. Bit flag that shows / controls the status of digital output 9 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DO_10	Digital output 10. Bit flag that shows / controls the status of digital output 10 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DO_11	Digital output 11. Bit flag that shows / controls the status of digital output 11 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DO_12	Digital output 12. Bit flag that shows / controls the status of digital output 12 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DO_13	Digital output 13. Bit flag that shows / controls the status of digital output 13 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DO_14	Digital output 14. Bit flag that shows / controls the status of digital output 14 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DO_15	Digital output 15. Bit flag that shows / controls the status of digital output 15 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DO_16	Digital output 16. Bit flag that shows / controls the status of digital output 16 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

16	DO_17	Digital output 17. Bit flag that shows / controls the status of digital output 17 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
17	DO_18	Digital output 18. Bit flag that shows / controls the status of digital output 18 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
18	DO_19	Digital output 19. Bit flag that shows / controls the status of digital output 19 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
19	DO_20	Digital output 20. Bit flag that shows / controls the status of digital output 20 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
20	DO_21	Digital output 21. Bit flag that shows / controls the status of digital output 21 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
21	DO_22	Digital output 22. Bit flag that shows / controls the status of digital output 22 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
22	DO_23	Digital output 23. Bit flag that shows / controls the status of digital output 23 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
23	DO_24	Digital output 24. Bit flag that shows / controls the status of digital output 24 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
24	DO_25	Digital output 25. Bit flag that shows / controls the status of digital output 25 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
25	DO_26	Digital output 26. Bit flag that shows / controls the status of digital output 26 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
26	DO_27	Digital output 27. Bit flag that shows / controls the status of digital output 27 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
27	DO_28	Digital output 28. Bit flag that shows / controls the status of digital output 28 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
28	DO_29	Digital output 29. Bit flag that shows / controls the status of digital output 29 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
29	DO_30	Digital output 30. Bit flag that shows / controls the status of digital output 30 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
30	DO_31	Digital output 31. Bit flag that shows / controls the status of digital output 31 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
31	DO_32	Digital output 32. Bit flag that shows / controls the status of digital output 32 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

Register 4509 - Digital Out Low

Register 4509 is a 16-bit register that contains bit flags to control digital outputs DO_1 to DO16. The bit functions for register 4509 are as follows.

Bit	Name	Description	Function
0	DO_1	Digital output 1. Bit flag that shows / controls the status of digital output 1 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DO_2	Digital output 2. Bit flag that shows / controls the status of digital output 2 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DO_3	Digital output 3. Bit flag that shows / controls the status of digital output 3 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DO_4	Digital output 4. Bit flag that shows / controls the status of digital output 4 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DO_5	Digital output 5. Bit flag that shows / controls the status of digital output 5 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DO_6	Digital output 6. Bit flag that shows / controls the status of digital output 6 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DO_7	Digital output 7. Bit flag that shows / controls the status of digital output 7 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_8	Digital output 8. Bit flag that shows / controls the status of digital output 8 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DO_9	Digital output 9. Bit flag that shows / controls the status of digital output 9 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DO_10	Digital output 10. Bit flag that shows / controls the status of digital output 10 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DO_11	Digital output 11. Bit flag that shows / controls the status of digital output 11 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DO_12	Digital output 12. Bit flag that shows / controls the status of digital output 12 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DO_13	Digital output 13. Bit flag that shows / controls the status of digital output 13 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DO_14	Digital output 14. Bit flag that shows / controls the status of digital output 14 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DO_15	Digital output 15. Bit flag that shows / controls the status of digital output 15 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DO_16	Digital output 16. Bit flag that shows / controls the status of digital output 16 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

Register 4510 - Digital Out High

Register 4510 is a 16-bit register that contains bit flags to control digital outputs DO_17 to DO_32. The bit functions for register 4510 are as follows.

Bit	Name	Description	Function
0	DO_17	Digital output 17. Bit flag that shows / controls the status of digital output 17 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DO_18	Digital output 18. Bit flag that shows / controls the status of digital output 18 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DO_19	Digital output 19. Bit flag that shows / controls the status of digital output 19 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DO_20	Digital output 20. Bit flag that shows / controls the status of digital output 20 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DO_21	Digital output 21. Bit flag that shows / controls the status of digital output 21 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DO_22	Digital output 22. Bit flag that shows / controls the status of digital output 22 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DO_23	Digital output 23. Bit flag that shows / controls the status of digital output 23 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_24	Digital output 24. Bit flag that shows / controls the status of digital output 24 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DO_25	Digital output 25. Bit flag that shows / controls the status of digital output 25 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DO_26	Digital output 26. Bit flag that shows / controls the status of digital output 26 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DO_27	Digital output 27. Bit flag that shows / controls the status of digital output 27 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DO_28	Digital output 28. Bit flag that shows / controls the status of digital output 28 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DO_29	Digital output 29. Bit flag that shows / controls the status of digital output 29 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DO_30	Digital output 30. Bit flag that shows / controls the status of digital output 30 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DO_31	Digital output 31. Bit flag that shows / controls the status of digital output 31 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DO_32	Digital output 32. Bit flag that shows / controls the status of digital output 32 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

Register 253 – Digital Output 2

Register 253 is a 32-bit register that contains bit flags to control up to 32 digital outputs. It is designed to be used with a macro and requires three 16/16 IO expansion modules to be connected to the controller. The bit functions for register 253 are as follows.

Bit	Name	Description	Function
0	DO_33	Digital output 33. Bit flag that shows / controls the status of digital output 33 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DO_34	Digital output 34. Bit flag that shows / controls the status of digital output 34 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DO_35	Digital output 35. Bit flag that shows / controls the status of digital output 35 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DO_36	Digital output 36. Bit flag that shows / controls the status of digital output 36 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DO_37	Digital output 37. Bit flag that shows / controls the status of digital output 37 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DO_38	Digital output 38. Bit flag that shows / controls the status of digital output 38 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DO_39	Digital output 39. Bit flag that shows / controls the status of digital output 39 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_40	Digital output 40. Bit flag that shows / controls the status of digital output 40 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DO_41	Digital output 41. Bit flag that shows / controls the status of digital output 41 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DO_42	Digital output 42. Bit flag that shows / controls the status of digital output 42 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DO_43	Digital output 43. Bit flag that shows / controls the status of digital output 43 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DO_44	Digital output 44. Bit flag that shows / controls the status of digital output 44 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DO_45	Digital output 45. Bit flag that shows / controls the status of digital output 45 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DO_46	Digital output 46. Bit flag that shows / controls the status of digital output 46 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DO_47	Digital output 47. Bit flag that shows / controls the status of digital output 47 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DO_48	Digital output 48. Bit flag that shows / controls the status of digital output 48 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

16	DO_49	Reserved for future development. Can be used as general purpose flag.
17	DO_50	Reserved for future development. Can be used as general purpose flag.
18	DO_51	Reserved for future development. Can be used as general purpose flag.
19	DO_52	Reserved for future development. Can be used as general purpose flag.
20	DO_53	Reserved for future development. Can be used as general purpose flag.
21	DO_54	Reserved for future development. Can be used as general purpose flag.
22	DO_55	Reserved for future development. Can be used as general purpose flag.
23	DO_56	Reserved for future development. Can be used as general purpose flag.
24	DO_57	Reserved for future development. Can be used as general purpose flag.
25	DO_58	Reserved for future development. Can be used as general purpose flag.
26	DO_59	Reserved for future development. Can be used as general purpose flag.
27	DO_60	Reserved for future development. Can be used as general purpose flag.
28	DO_61	Reserved for future development. Can be used as general purpose flag.
29	DO_62	Reserved for future development. Can be used as general purpose flag.
30	DO_63	Reserved for future development. Can be used as general purpose flag.
31	DO_64	Reserved for future development. Can be used as general purpose flag.

Register 4511 - Digital Output 2 Low

Register 4511 is a 16-bit register that contains bit flags to control digital outputs DO_33 to DO_48. The bit functions for register 4511 are as follows.

Bit	Name	Description	Function
0	DO_33	Digital output 33. Bit flag that shows / controls the status of digital output 33 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DO_34	Digital output 34. Bit flag that shows / controls the status of digital output 34 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DO_35	Digital output 35. Bit flag that shows / controls the status of digital output 35 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DO_36	Digital output 36. Bit flag that shows / controls the status of digital output 36 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DO_37	Digital output 37. Bit flag that shows / controls the status of digital output 37 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DO_38	Digital output 38. Bit flag that shows / controls the status of digital output 38 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DO_39	Digital output 39. Bit flag that shows / controls the status of digital output 39 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_40	Digital output 40. Bit flag that shows / controls the status of digital output 40 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DO_41	Digital output 41. Bit flag that shows / controls the status of digital output 41 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DO_42	Digital output 42. Bit flag that shows / controls the status of digital output 42 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DO_43	Digital output 43. Bit flag that shows / controls the status of digital output 43 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DO_44	Digital output 44. Bit flag that shows / controls the status of digital output 44 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DO_45	Digital output 45. Bit flag that shows / controls the status of digital output 45 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DO_46	Digital output 46. Bit flag that shows / controls the status of digital output 46 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DO_47	Digital output 47. Bit flag that shows / controls the status of digital output 47 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DO_48	Digital output 48. Bit flag that shows / controls the status of digital output 48 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

Register 4512 - Digital Output 2 High

Register 4512 is a 16-bit register that contains bit flags to control digital outputs DO_49 to DO_64. The bit functions for register 4512 are as follows.

(Note: Digital outputs DO_49 to DO_64 are not implemented in hardware as yet and are reserved for future development)

Bit	Name	Description	Function
0	DO_49	Reserved for future development. Can be used as general purpose flag.	
1	DO_50	Reserved for future development. Can be used as general purpose flag.	
2	DO_51	Reserved for future development. Can be used as general purpose flag.	
3	DO_52	Reserved for future development. Can be used as general purpose flag.	
4	DO_53	Reserved for future development. Can be used as general purpose flag.	
5	DO_54	Reserved for future development. Can be used as general purpose flag.	
6	DO_55	Reserved for future development. Can be used as general purpose flag.	
7	DO_56	Reserved for future development. Can be used as general purpose flag.	
8	DO_57	Reserved for future development. Can be used as general purpose flag.	
9	DO_58	Reserved for future development. Can be used as general purpose flag.	
10	DO_59	Reserved for future development. Can be used as general purpose flag.	
11	DO_60	Reserved for future development. Can be used as general purpose flag.	
12	DO_61	Reserved for future development. Can be used as general purpose flag.	
13	DO_62	Reserved for future development. Can be used as general purpose flag.	
14	DO_63	Reserved for future development. Can be used as general purpose flag.	
15	DO_64	Reserved for future development. Can be used as general purpose flag.	

See also

Digital Read Only 88

See Register 8430 I/O Module Type 97

2.18.1 Control Inputs

Register 4103 – Control Input Register

Register 4103 is a 16-bit read only register that contains bit flags to read the 10 digital control inputs on the 10/12 I/O module. It is designed to be used with a macro and requires one or more IO expansion modules to be connected to the controller.

Name	Description	Symbol Type	Register Number	Memory Type
CONTROL_INPUT	16 bit read only register with bit flags for 10/12 I/O modules	U_16	4103	RAM 8

The bit functions for register 4103 are as follows:

Bit	Name	Description	Function
0	CI_1	Control input 1. Read only bit flag that shows the status of control input 1 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	CI_2	Control input 2. Read only bit flag that shows the status of control input 2 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	CI_3	Control input 3. Read only bit flag that shows the status of control input 3 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	CI_4	Control input 4. Read only bit flag that shows the status of control input 4 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	CI_5	Control input 5. Read only bit flag that shows the status of control input 5 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	CI_6	Control input 6. Read only bit flag that shows the status of control input 6 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	CI_7	Control input 7. Read only bit flag that shows the status of control input 7 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	CI_8	Control input 8. Read only bit flag that shows the status of control input 8 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	CI_9	Control input 9. Read only bit flag that shows the status of control input 9 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	CI_10	Control input 10. Read only bit flag that shows the status of control input 10 on the 10/12-I/O module (1 = output active).	1 = Active State 0 = Inactive State

See also

Digital Input Register 88

Digital Output Register 79

See Register 8430 I/O Module Type 97

2.18.2 Digital Read Only

Digital Input Registers

The following registers provide read access to the digital input pins on the 16/16 IO expansion modules. Access can be gained via 32 bit registers or via 16 bit registers for use with Modbus. These registers are designed to be accessed via the serial ports or a macro, and require one or more 16/16 IO expansion modules to be connected to the controller.

Name	Description	Symbol Type	Register Number	Memory Type
DIGITAL_IN	32 bit read only register with bit flags for digital inputs DI_1 to DI_32 on multi I/O modules	U_32	251	RAM
DIGITAL_IN_LOW	16 bit read only register with bit flags for digital inputs DI_1 to DI_16 on multi I/O modules.	U_16	4515	RAM
DIGITAL_IN_HIGH	16 bit read only register with bit flags for digital inputs DI_17 to DI_32 on multi I/O modules.	U_16	4516	RAM
DIGITAL_IN2	32 bit read only register with bit flags for digital inputs DI_33 to DI_64 on multi I/O modules	U_32	255	RAM
DIGITAL_IN2_LOW	16 bit read only register with bit flags for digital inputs DI_33 to DI_48 on multi I/O modules.	U_16	4517	RAM
DIGITAL_IN2_HIGH	16 bit read only register with bit flags for digital inputs DI_49 to DI_64 on multi I/O modules.	U_16	4518	RAM

Register 251 - Digital Input

Register 251 is a 32-bit read only register that contains bit flags to read up to 32 digital inputs. It is designed to be used with a macro and requires one or more 16/16 IO expansion modules to be connected to the controller. The bit functions for register 251 are as follows:

Bit	Name	Description	Function
0	DI_1	Digital input 1. Read only bit flag that shows the status of digital input 1 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DI_2	Digital input 2. Read only bit flag that shows the status of digital input 2 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DI_3	Digital input 3. Read only bit flag that shows the status of digital input 3 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DI_4	Digital input 4. Read only bit flag that shows the status of digital input 4 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DI_5	Digital input 5. Read only bit flag that shows the status of digital input 5 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DI_6	Digital input 6. Read only bit flag that shows the status of digital input 6 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DI_7	Digital input 7. Read only bit flag that shows the status of digital input 7 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_8	Digital input 8. Read only bit flag that shows the status of digital input 8 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DI_9	Digital input 9. Read only bit flag that shows the status of digital input 9 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DI_10	Digital input 10. Read only bit flag that shows the status of digital input 10 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DI_11	Digital input 11. Read only bit flag that shows the status of digital input 11 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DI_12	Digital input 12. Read only bit flag that shows the status of digital input 22 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DI_13	Digital input 13. Read only bit flag that shows the status of digital input 13 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DI_14	Digital input 14. Read only bit flag that shows the status of digital input 14 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DI_15	Digital input 15. Read only bit flag that shows the status of digital input 15 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DI_16	Digital input 16. Read only bit flag that shows the status of digital input 16 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

16	DI_17	Digital input 17. Read only bit flag that shows the status of digital input 17 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
17	DI_18	Digital input 18. Read only bit flag that shows the status of digital input 18 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
18	DI_19	Digital input 19. Read only bit flag that shows the status of digital input 19 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
19	DI_20	Digital input 20. Read only bit flag that shows the status of digital input 20 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
20	DI_21	Digital input 21. Read only bit flag that shows the status of digital input 21 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
21	DI_22	Digital input 22. Read only bit flag that shows the status of digital input 22 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
22	DI_23	Digital input 23. Read only bit flag that shows the status of digital input 23 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
23	DI_24	Digital input 24. Read only bit flag that shows the status of digital input 24 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
24	DI_25	Digital input 25. Read only bit flag that shows the status of digital input 25 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
25	DI_26	Digital input 26. Read only bit flag that shows the status of digital input 26 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
26	DI_27	Digital input 27. Read only bit flag that shows the status of digital input 27 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
27	DI_28	Digital input 28. Read only bit flag that shows the status of digital input 28 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
28	DI_29	Digital input 29. Read only bit flag that shows the status of digital input 29 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
29	DI_30	Digital input 30. Read only bit flag that shows the status of digital input 30 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
30	DI_31	Digital input 31. Read only bit flag that shows the status of digital input 31 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
31	DI_32	Digital input 32. Read only bit flag that shows the status of digital input 32 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

Register 4515 - Digital Input Low

Register 4515 is a 16-bit read only register that contains bit flags to read digital inputs DI_1 to DI_16. It is designed to be used with a macro and requires one or more 16/16 IO expansion modules to be connected to the controller. The bit functions for register 4515 are as follows:

Bit	Name	Description	Function
0	DI_1	Digital input 1. Read only bit flag that shows the status of digital input 1 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DI_2	Digital input 2. Read only bit flag that shows the status of digital input 2 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DI_3	Digital input 3. Read only bit flag that shows the status of digital input 3 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DI_4	Digital input 4. Read only bit flag that shows the status of digital input 4 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DI_5	Digital input 5. Read only bit flag that shows the status of digital input 5 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DI_6	Digital input 6. Read only bit flag that shows the status of digital input 6 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DI_7	Digital input 7. Read only bit flag that shows the status of digital input 7 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_8	Digital input 8. Read only bit flag that shows the status of digital input 8 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DI_9	Digital input 9. Read only bit flag that shows the status of digital input 9 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DI_10	Digital input 10. Read only bit flag that shows the status of digital input 10 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DI_11	Digital input 11. Read only bit flag that shows the status of digital input 11 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DI_12	Digital input 12. Read only bit flag that shows the status of digital input 22 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DI_13	Digital input 13. Read only bit flag that shows the status of digital input 13 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DI_14	Digital input 14. Read only bit flag that shows the status of digital input 14 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DI_15	Digital input 15. Read only bit flag that shows the status of digital input 15 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DI_16	Digital input 16. Read only bit flag that shows the status of digital input 16 on the first 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

Register 4516 - Digital Input High

Register 4516 is a 16-bit read only register that contains bit flags to read digital inputs DI_17 to DI_32. It is designed to be used with a macro and requires one or more 16/16 IO expansion modules to be connected to the controller. The bit functions for register 4516 are as follows:

Bit	Name	Description	Function
0	DI_17	Digital input 17. Read only bit flag that shows the status of digital input 17 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DI_18	Digital input 18. Read only bit flag that shows the status of digital input 18 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DI_19	Digital input 19. Read only bit flag that shows the status of digital input 19 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DI_20	Digital input 20. Read only bit flag that shows the status of digital input 20 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DI_21	Digital input 21. Read only bit flag that shows the status of digital input 21 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DI_22	Digital input 22. Read only bit flag that shows the status of digital input 22 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DI_23	Digital input 23. Read only bit flag that shows the status of digital input 23 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DI_24	Digital input 24. Read only bit flag that shows the status of digital input 24 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DI_25	Digital input 25. Read only bit flag that shows the status of digital input 25 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DI_26	Digital input 26. Read only bit flag that shows the status of digital input 26 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DI_27	Digital input 27. Read only bit flag that shows the status of digital input 27 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DI_28	Digital input 28. Read only bit flag that shows the status of digital input 28 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DI_29	Digital input 29. Read only bit flag that shows the status of digital input 29 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DI_30	Digital input 30. Read only bit flag that shows the status of digital input 30 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DI_31	Digital input 31. Read only bit flag that shows the status of digital input 31 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DI_32	Digital input 32. Read only bit flag that shows the status of digital input 32 on the second 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

Register 255 – Digital Input Register #2

Register 255 is a 32-bit read only register that contains bit flags to read up to 32 digital inputs. It is designed to be used with a macro and requires three IO expansion modules to be connected to the controller. The bit functions for register 255 are as follows:

Bit	Name	Description	Function
0	DI_33	Digital input 33. Read only bit flag that shows the status of digital input 33 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DI_34	Digital input 34. Read only bit flag that shows the status of digital input 34 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DI_35	Digital input 35. Read only bit flag that shows the status of digital input 35 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DI_36	Digital input 36. Read only bit flag that shows the status of digital input 36 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DI_37	Digital input 37. Read only bit flag that shows the status of digital input 37 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DI_38	Digital input 38. Read only bit flag that shows the status of digital input 38 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DI_39	Digital input 39. Read only bit flag that shows the status of digital input 39 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_40	Digital input 40. Read only bit flag that shows the status of digital input 40 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DI_41	Digital input 41. Read only bit flag that shows the status of digital input 41 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DI_42	Digital input 42. Read only bit flag that shows the status of digital input 42 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DI_43	Digital input 43. Read only bit flag that shows the status of digital input 43 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DI_44	Digital input 44. Read only bit flag that shows the status of digital input 44 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DI_45	Digital input 45. Read only bit flag that shows the status of digital input 45 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DI_46	Digital input 46. Read only bit flag that shows the status of digital input 46 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DI_47	Digital input 47. Read only bit flag that shows the status of digital input 47 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DI_48	Digital input 48. Read only bit flag that shows the status of digital input 48 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

16	DI_49	Reserved for future development.
17	DI_50	Reserved for future development.
18	DI_51	Reserved for future development.
19	DI_52	Reserved for future development.
20	DI_53	Reserved for future development.
21	DI_54	Reserved for future development.
22	DI_55	Reserved for future development.
23	DI_56	Reserved for future development.
24	DI_57	Reserved for future development.
25	DI_58	Reserved for future development.
26	DI_59	Reserved for future development.
27	DI_60	Reserved for future development.
28	DI_61	Reserved for future development.
29	DI_62	Reserved for future development.
30	DI_63	Reserved for future development.
31	DI_64	Reserved for future development.

Register 4517 - Digital Input 2 Low

Register 4517 is a 16-bit read only register that contains bit flags to read digital inputs DI_33 to DI_48. It is designed to be used with a macro and requires one or more 16/16 IO expansion modules to be connected to the controller. The bit functions for register 4517 are as follows:

Bit	Name	Description	Function
0	DI_33	Digital input 33. Read only bit flag that shows the status of digital input 33 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
1	DI_34	Digital input 34. Read only bit flag that shows the status of digital input 34 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
2	DI_35	Digital input 35. Read only bit flag that shows the status of digital input 35 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
3	DI_36	Digital input 36. Read only bit flag that shows the status of digital input 36 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
4	DI_37	Digital input 37. Read only bit flag that shows the status of digital input 37 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
5	DI_38	Digital input 38. Read only bit flag that shows the status of digital input 38 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
6	DI_39	Digital input 39. Read only bit flag that shows the status of digital input 39 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
7	DO_40	Digital input 40. Read only bit flag that shows the status of digital input 40 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
8	DI_41	Digital input 41. Read only bit flag that shows the status of digital input 41 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
9	DI_42	Digital input 42. Read only bit flag that shows the status of digital input 42 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
10	DI_43	Digital input 43. Read only bit flag that shows the status of digital input 43 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
11	DI_44	Digital input 44. Read only bit flag that shows the status of digital input 44 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
12	DI_45	Digital input 45. Read only bit flag that shows the status of digital input 45 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
13	DI_46	Digital input 46. Read only bit flag that shows the status of digital input 46 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
14	DI_47	Digital input 47. Read only bit flag that shows the status of digital input 47 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State
15	DI_48	Digital input 48. Read only bit flag that shows the status of digital input 48 on the third 16/16 multi-I/O module (1 = output active).	1 = Active State 0 = Inactive State

Register 4518 - Digital Input 2 High

Register 4518 is a 16-bit read only register that contains bit flags to read digital inputs DI_49 to DI_64. It is designed to be used with a macro and requires one or more 16/16 IO expansion modules to be connected to the controller. The bit functions for register 4518 are as follows:

Note: These bit functions are not available in hardware at present and are reserved for future development.

Bit	Name	Description	Function
0	DI_49	Reserved for future development.	
1	DI_50	Reserved for future development.	
2	DI_51	Reserved for future development.	
3	DI_52	Reserved for future development.	
4	DI_53	Reserved for future development.	
5	DI_54	Reserved for future development.	
6	DI_55	Reserved for future development.	
7	DI_56	Reserved for future development.	
8	DI_57	Reserved for future development.	
9	DI_58	Reserved for future development.	
10	DI_59	Reserved for future development.	
11	DI_60	Reserved for future development.	
12	DI_61	Reserved for future development.	
13	DI_62	Reserved for future development.	
14	DI_63	Reserved for future development.	
15	DI_64	Reserved for future development.	

See also

Control Input Register 87

Digital Output Register 79

See Register 8430 I/O Module Type 97

2.18.3 I/O Module Type - Register 8430

Register 8430 is an 8-bit unsigned register in RAM that shows which I/O expansion modules are connected to the controller at switch on. If more than one expansion module is fitted then the number held in register 8430 will be the cumulative total of the options shown in the tables below.

Data Value	Function
0	No I/O expansion module fitted
1	16 digital outputs (DO_1 - DO_16)
2	16 digital inputs (DI_1 - DI_16)
4	16 digital outputs (DO_17 - DO_32)
8	16 digital inputs (DI_17 - DI_32)
16	12 setpoints, 10 control inputs (CI_1 to CI_10)
32	Not used
64	16 digital outputs (DO_33 - DO_48)
128	16 digital inputs (DI_33 - DI_48)

Typical I/O module type values for common configurations are shown below.

Data Value	Function
0	No I/O expansion modules fitted
16	10/12 module
19	10/12 module plus one 16/16 module
31	10/12 module plus two 16/16 modules
223	10/12 module plus three 16/16 modules

NOTE: Register 8430 is updated by the controller at power up, or if a different module type is detected. If the I/O expansion module is removed after power up, register 8430 retains its original value and keeps trying to locate the I/O module on the expansion bus. It will only change to a new value if another I/O module type is located on the bus. Although this is intended as a read only register, it can be written to. Writing a value of zero to this register disables the I/O expansion module. Writing any value other than zero forces the controller to search the bus for a connected I/O module.

2.19 Edit Mode

Edit mode registers are used in the Edit mode macro to control the editable range for a parameter and keep track of which parameter is currently being edited.

Name	Description	Symbol Type	Register Number	Memory Type
EDIT_DEF	32-bit register. Sets the default value when UP and DOWN buttons are pressed in edit mode.	S_32	237	RAM 8
EDIT_MAX	32-bit register. Sets the maximum allowable range in edit mode.	S_32	233	RAM 8
EDIT_MIN	32-bit register. Sets the minimum allowable range in edit mode.	S_32	235	RAM 8
EDIT_STATE 99	8-bit register. Defines which parameter type is currently being edited.	U_8	8223 99	RAM 8
EDIT_VALUE	32-bit register holds the currently displayed value when in any edit mode.	S_32	231	RAM 8
NON_VOLATILE_WRITE 98	When this flag is ON, the next register written will be saved in permanent memory (flag reset after each write).	B_5	8222 98	RAM 8

NOTE: Edit Mode registers are intended for use with the Edit Macro. Writing to these registers while the controller is in any of its setup modes is not recommended. This may allow parameters to be selected outside of their allowable ranges.

2.19.1 Non-volatile Write Flag - Register 8222

Bit 5 of register 8222 is only used by the macro to enable a write to non-volatile memory. Nearly all of the registers in the controller are situated in RAM (Random Access Memory) which looses its data when the controller is turned OFF. In some cases this is undesirable, so a second copy of the data is stored in non-volatile memory that retains its data even when the controller is turned OFF. However, a physical limitation of this sort of memory is that each memory location can only be written to 1,000,000 times in total.

The non-volatile write flag is provided so that those registers that have a copy in non-volatile memory can still be written to during each cycle by the macro without exceeding the maximum write limit. If bit 5 is set to a **1** then the next time the macro stores a value into a register it will write to RAM and update any associated non-volatile memory locations as well. If bit 5 is set to a **0** then the next time

the macro stores a value into a register, it will only write to RAM. After each store instruction, the macro engine automatically clears this flag to 0.

NOTE: The non-volatile write flag only functions for macro commands. Any write to a register via the serial port **always** updates the non-volatile memory.

2.19.2 Edit State - Register 8223

Register 8223 is an 8-bit register that gives the current operational state of the controller when it is in any edit mode.

The following table shows the different parameters that are being edited in each state.

Edit State	Operation
0	Not in edit mode
1	Brightness
2	Lock display (up)
3	Cal
4	Code 1
5	Code 2
6	Code 3
7	Code 4
8	Code 5
9	Code 6
10	Code 7
11	Code 8
12	Code 9
13	Code 10
14	Not used
15	Edit mode - Macro
16	Lock display (down)
17	Setpoint 1
18	Setpoint 2
19	Setpoint 3
20	Setpoint 4
21	Setpoint 5
22	Setpoint 6
23	Setpoint 1 control
24	Setpoint 2 control
25	Setpoint 3 control
26	Setpoint 4 control
27	Setpoint 5 control
28	Setpoint 6 control
29-31	Not used
32	Manual cal - offset
33	Manual cal - scale
34	Tbuff in PH mode or Cal TC, RTD
35	Auto cal - Zero
36	Auto cal - Span
37	Auto cal - 4 second delay and calculate scale and offset values
38	Baud rate
39	Parity
40	Transmit delay
41	Serial address
42	Analog output – cal low end
43	Analog output – cal high end
44	Analog output – zero
45	Analog output – full scale
46	K factor for totalizer
47	Cut off for totalizer
48	Prescaler for counter 1
49	Display format

Edit State	Operation
51	Increment/decrement, Manual loader mode from Prog button
52	Not used
53	Not used
54	Not used
55	Colour band 1
56	Colour band 2
57	Colour band 3
58	Colour band 4
59	Colour band 5
60	Colour band 6
61	Colour band 7
62	Prescaler for counter 2
63	Calibrate zero in Ph mode
64	Calibrate span in Ph mode
65	Display source.
66-68	Not used
69	Auto zero capture band
70	Auto zero motion band
71	Averaging samples
72	Averaging window
73	Smart input module Setup
74	Smart input module Setup 2
75	Smart input module Setup 3
76	Auto zero aperture band
77	Display input value for totalizer setup
78	Totalizer rate time selection
79	Totalizer roll over select
80	Select 7 or 8 data bits for serial port
81	Disable code blanking
82	Disable macro
83	Serial Mode
84	Not used
85	Not used
86	Not used
87	Not used
88	Not used
89	Not used
90	Not used
91	Not used
92	Not used
93	Not used
94	Not used
95	Not used
96	Select mode for 32-point linearization table
97	Select table for 32-point linearization table
98	Enter date for 32-point linearization table
99	Enter serial no. for 32-point linearization table
100-163	Edit 32-point input and output values.

2.20 Display

The majority of registers in the display category affect how a parameter is formatted and displayed. For example, where the decimal point (if any) is positioned, what number modulus is used (i.e. decimal, octal, time format, etc.), and whether the number requires a following text character to describe the units.

Also included in this category are registers that specify the data source for a particular display or other special registers.

Name	Description	Symbol Type	Register Number	Memory Type
BRIGHTNESS 105	8-bit register holds the display brightness/contrast setting (range 0-7).	U_8	8206	RAM/EEPROM 8
CURRENT_ALPHA_CHARACTER 106	8-bit register holds the ASCII value for the last digit text character (0= no character).	U_8	8392	RAM 8
CURRENT_DISPLAY_FORMAT 106	8-bit register holds the display format settings of the current display (displayed in octal 107 format).	U_8	8366	RAM
DATA_SOURCE_DISPLAY1 105	16-bit register holds the register number of the data source for the primary display.	U_8	4371	RAM/EEPROM
DATA_SOURCE_DISPLAY2 105	16-bit register holds the register number of the data source for the second display.	U_8	4372	RAM/EEPROM
DATA_SOURCE_DISPLAY3 105	16-bit register holds the register number of the data source for the third display.	U_8	4373	RAM/EEPROM
DATA_SOURCE_DISPLAY4 105	16-bit register holds the register number of the data source for the fourth display.	U_8	4377	RAM/EEPROM
DATA_SOURCE_PEAK_VALLEY1 105	16-bit register holds the register number of the data source for peak 1 & valley 1.	U_16	4374	RAM/EEPROM
DATA_SOURCE_PEAK_VALLEY2 105	16-bit register holds the register number of the data source for peak 2 & valley 2.	U_16	4375	RAM/EEPROM
DATA_SOURCE_PEAK_VALLEY3 105	16-bit register holds the register number of the data source for peak 3 & valley 3.	U_16	4376	RAM/EEPROM
DISPLAY	32-bit register for primary display data.	S_32	1	RAM 8
DISPLAY_FORMAT_AUX1 106	8-bit register controls the display format settings for auxiliary 1 (displayed in octal 107 format).	O_8	8382	RAM/EEPROM 8
DISPLAY_FORMAT_AUX2 106	8-bit register controls the display format settings for auxiliary 2 (displayed in octal 107 format).	O_8	8383	RAM/EEPROM
DISPLAY_FORMAT_AUX3 106	8-bit register controls the display format settings for auxiliary 3 (displayed in octal 107 format).	O_8	8384	RAM/EEPROM
DISPLAY_FORMAT_AUX4 106	8-bit register controls the display format settings for auxiliary 4 (displayed in octal 107 format).	O_8	8385	RAM/EEPROM
DISPLAY_FORMAT_AUX5 106	8-bit register controls the display format settings for auxiliary 5 (displayed in octal 107 format).	O_8	8386	RAM/EEPROM
DISPLAY_FORMAT_AUX6 106	8-bit register controls the display format settings for auxiliary 6 (displayed in octal 107 format).	O_8	8387	RAM/EEPROM
DISPLAY_FORMAT_AUX7 106	8-bit register controls the display format settings for auxiliary 7 (displayed in octal 107 format).	O_8	8388	RAM/EEPROM
DISPLAY_FORMAT_AUX8 106	8-bit register controls the display format settings for auxiliary 8 (displayed in octal 107 format).	O_8	8389	RAM/EEPROM
DISPLAY_FORMAT_AUX9 106	8-bit register controls the display format settings for auxiliary 9 (displayed in octal 107 format).	O_8	8390	RAM/EEPROM

DISPLAY_FORMAT_AUX10 106	8-bit register controls the display format settings for auxiliary 10 (displayed in octal 107 format).	O_8	8391	RAM/EEPROM 8
DISPLAY_FORMAT_AUX11 106	8-bit register controls the display format settings for auxiliary 11 (displayed in octal 107 format).	O_8	8392	RAM/EEPROM
DISPLAY_FORMAT_AUX12 106	8-bit register controls the display format settings for auxiliary 12 (displayed in octal 107 format).	O_8	8393	RAM/EEPROM
DISPLAY_FORMAT_AUX13 106	8-bit register controls the display format settings for auxiliary 13 (displayed in octal 107 format).	O_8	8394	RAM/EEPROM
DISPLAY_FORMAT_AUX14 106	8-bit register controls the display format settings for auxiliary 14 (displayed in octal 107 format).	O_8	8395	RAM/EEPROM
DISPLAY_FORMAT_AUX15 106	8-bit register controls the display format settings for auxiliary 15 (displayed in octal 107 format).	O_8	8396	RAM/EEPROM
DISPLAY_FORMAT_AUX16 106	8-bit register controls the display format settings for auxiliary 16 (displayed in octal 107 format).	O_8	8397	RAM/EEPROM
DISPLAY_FORMAT_CH1 106	8-bit register controls the display format settings for channel 1 (displayed in octal 107 format).	O_8	8368	RAM/EEPROM
DISPLAY_FORMAT_CH2 106	8-bit register controls the display format settings for channel 2 (displayed in octal 107 format).	O_8	8369	RAM/EEPROM 8
DISPLAY_FORMAT_CH3 106	8-bit register controls the display format settings for channel 3 (displayed in octal 107 format).	O_8	8370	RAM/EEPROM
DISPLAY_FORMAT_CH4 106	8-bit register controls the display format settings for channel 4 (displayed in octal 107 format).	O_8	8371	RAM/EEPROM
DISPLAY_FORMAT_CH5 106	8-bit register controls the display format settings for channel 5 (displayed in octal 107 format).	O_8	8372	RAM/EEPROM
DISPLAY_FORMAT_CH6 106	8-bit register controls the display format settings for channel 6 (displayed in octal 107 format).	O_8	8373	RAM/EEPROM
DISPLAY_FORMAT_CH7 106	8-bit register controls the display format settings for channel 7 (displayed in octal 107 format).	O_8	8374	RAM/EEPROM
DISPLAY_FORMAT_DEFAULT 106	8-bit register holds the default display format settings (displayed in octal 107 format).	O_8	8375	RAM/EEPROM
DISPLAY_FORMAT_RESULT 106	8-bit register controls the display format settings for the result (displayed in octal 107 format).	O_8	8367	RAM/EEPROM 8
DISPLAY_FORMAT_TOTAL1 106	8-bit register controls the display format settings for total 1 (displayed in octal 107 format).	O_8	8376	RAM/EEPROM
DISPLAY_FORMAT_TOTAL2 106	8-bit register controls the display format settings for total 2 (displayed in octal 107 format).	O_8	8377	RAM/EEPROM
DISPLAY_FORMAT_TOTAL3 106	8-bit register controls the display format settings for total 3 (displayed in octal 107 format).	O_8	8378	RAM/EEPROM
DISPLAY_FORMAT_TOTAL4 106	8-bit register controls the display format settings for total 4 (displayed in octal 107 format).	O_8	8379	RAM/EEPROM
DISPLAY_FORMAT_TOTAL5 106	8-bit register controls the display format settings for total 5 (displayed in octal 107 format).	O_8	8380	RAM/EEPROM 8
DISPLAY_FORMAT_TOTAL6 106	8-bit register controls the display format settings for total 6 (displayed in octal 107 format).	O_8	8381	RAM/EEPROM
PEAK1	32-bit register for peak 1 value.	S_32	57	RAM 8
PEAK2	32-bit register for peak 2 value.	S_32	61	RAM
PEAK3	32-bit register for peak 3 value.	S_32	65	RAM
TARE	32-bit register for tare value.	S_32	77	RAM

TEXT_CHARACTER_RESULT 108	8-bit register holds the ASCII value for the last digit text character for Result (0= no character).	U_8	8393	RAM/EEPROM
TEXT_CHARACTER_CH1 108	8-bit register holds the ASCII value for the last digit text character for channel 1 (0= no character).	U_8	8394	RAM/EEPROM
TEXT_CHARACTER_CH2 108	8-bit register holds the ASCII value for the last digit text character for channel 2 (0= no character).	U_8	8395	RAM/EEPROM
TEXT_CHARACTER_CH3 108	8-bit register holds the ASCII value for the last digit text character for channel 3 (0= no character).	U_8	8396	RAM/EEPROM
TEXT_CHARACTER_CH4 108	8-bit register holds the ASCII value for the last digit text character for channel 4 (0= no character).	U_8	8397	RAM/EEPROM 8
TEXT_CHARACTER_CH5 108	8-bit register holds the ASCII value for the last digit text character for channel 5 (0= no character).	U_8	8398	RAM/EEPROM
TEXT_CHARACTER_CH6 108	8-bit register holds the ASCII value for the last digit text character for channel 6 (0= no character).	U_8	8399	RAM/EEPROM
TEXT_CHARACTER_CH7 108	8-bit register holds the ASCII value for the last digit text character for channel 7 (0= no character).	U_8	8400	RAM/EEPROM
TEXT_CHARACTER_DEFAULT 108	8-bit register holds the ASCII value for the last digit text character for the default (0= no character).	U_8	8401	RAM/EEPROM
TEXT_CHARACTER_TOTAL1 108	8-bit register holds the ASCII value for the last digit text character for the total 1 (0= no character).	U_8	8402	RAM/EEPROM 8
TEXT_CHARACTER_TOTAL2 108	8-bit register holds the ASCII value for the last digit text character for the total 2 (0= no character).	U_8	8403	RAM/EEPROM
TEXT_CHARACTER_TOTAL3 108	8-bit register holds the ASCII value for the last digit text character for the total 3 (0= no character).	U_8	8404	RAM/EEPROM
TEXT_CHARACTER_TOTAL4 108	8-bit register holds the ASCII value for the last digit text character for the total 4 (0= no character).	U_8	8405	RAM/EEPROM
TEXT_CHARACTER_TOTAL5 108	8-bit register holds the ASCII value for the last digit text character for the total 5 (0= no character).	U_8	8406	RAM/EEPROM
TEXT_CHARACTER_TOTAL6 108	8-bit register holds the ASCII value for the last digit text character for the total 6 (0= no character).	U_8	8407	RAM/EEPROM
TEXT_CHARACTER_AUX1 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 1 (0= no character).	U_8	8408	RAM/EEPROM 8
TEXT_CHARACTER_AUX2 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 2 (0= no character).	U_8	8409	RAM/EEPROM
TEXT_CHARACTER_AUX3 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 3 (0= no character).	U_8	8410	RAM/EEPROM

TEXT_CHARACTER_AUX4 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 4 (0= no character).	U_8	8411	RAM/EEPROM
TEXT_CHARACTER_AUX5 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 5 (0= no character).	U_8	8412	RAM/EEPROM
TEXT_CHARACTER_AUX6 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 6 (0= no character).	U_8	8413	RAM/EEPROM
TEXT_CHARACTER_AUX7 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 7 (0= no character).	U_8	8414	RAM/EEPROM 8
TEXT_CHARACTER_AUX8 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 8 (0= no character).	U_8	8415	RAM/EEPROM
TEXT_CHARACTER_AUX9 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 9 (0= no character).	U_8	8416	RAM/EEPROM
TEXT_CHARACTER_AUX10 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 10 (0= no character).	U_8	8417	RAM/EEPROM
TEXT_CHARACTER_AUX11 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 11 (0= no character).	U_8	8418	RAM/EEPROM
TEXT_CHARACTER_AUX12 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 12 (0= no character).	U_8	8419	RAM/EEPROM
TEXT_CHARACTER_AUX13 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 13 (0= no character).	U_8	8420	RAM/EEPROM 8
TEXT_CHARACTER_AUX14 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 14 (0= no character).	U_8	8421	RAM/EEPROM
TEXT_CHARACTER_AUX15 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 15 (0= no character).	U_8	8422	RAM/EEPROM
TEXT_CHARACTER_AUX15 108	8-bit register holds the ASCII value for the last digit text character for auxiliary 15 (0= no character).	U_8	8423	RAM/EEPROM
VALLEY1	32-bit register for valley 1 value.	S_32	59	RAM 8
VALLEY2	32-bit register for valley 2 value.	S_32	63	RAM
VALLEY3	32-bit register for valley 3 value.	S_32	67	RAM

Registers 8393 to 8417

Registers 8393 to 8417 allow a text character to be displayed after the least significant digit of a parameter in order to describe the units for the parameter (e.g. C for degrees C or N for Newtons). When a text character is selected for a specific parameter, it is included when the parameter is displayed on the front panel of the controller or when the parameter is read via the serial port in ASCII mode. Any other registers that are referenced to the parameter (such as setpoints, peak & valley, etc.) also include the text parameter.

These registers can only accept printable ASCII characters from 0x20 to 0x7F hex. Control characters are not allowed and if used could produce unexpected results.

Writing a value of 0 to one of these text character registers disables any text characters from being with that parameter and the display shows the numeric value.

NOTE: Some controller models with dot matrix LCD displays use the extended ASCII set to allow special characters to be displayed as well. For these models the acceptable range is from 0x20 to 0xFF hex.

HINT: When setting these registers from the macro you can use ASC(" ") to load the appropriate ASCII character number.

For example, to select the text character "F" for channel 1 you would type:

&TEXT_CHARACTER_CH1=ASC("F")

See also
Text 108

Display Format 106

Octal Format 107

2.20.1 Display Data Source Selection - Register 4371 to 4373

Registers 4371 to 4373 and 4377 are 16-bit registers that specify the data source for various displays. The number they contain is the ASCII/Modbus register number for the required data source.

The ICC402 controller allows any register type to be displayed on the display. This includes floating point and text registers.

Register 4372, 4373 & 4377 are reserved for different models of controller with more than a single line display. The function of registers 4372, 4373 & 4377 with different controller types is shown as follows:

6 x 2 Line LED Display
4371 = Data source for the top display (also known as the primary display)
4372 = Data source for the 2nd line of display

16 x 2 Line LCD Display
4371 = Data source for the top display (primary display)
4372 = Data source for 2nd line of display
4373 = Data source for second variable on top display (primary display)
4377 = Data source for second variable on lower line of display

2.20.2 Peak/Valley Data Source Selection - Register 4374 to 4376

Registers 4374 to 4376 are 16-bit registers that specify the data source for the peak and valley displays. The number they contain is the ASCII/Modbus register number for the required data source.

NOTE: Only registers that hold integer values can be used as a data source for the display. Floating point and text registers can not be used.

2.20.3 Brightness/Contrast - Register 8206

Register 8206 is an 8-bit register used to control the brightness of an LED display or the contrast of some LCD display. A number between 0 and 7 can be written to register 8206 to control the brightness, with 0 being dull and 7 bright.

2.20.4 Display Options For Current Display - Register 8366

Register 8366 is an 8-bit register that shows the display options such as rounding, display mode, decimal point selection, that are currently active on the display. This can be used in the macro to modify the preset display format of a register just before it is scrolled across the display with the WRITE or APPEND command.

The function of each bit is exactly the same as that for registers 8367 to 8391 101.

2.20.5 Alphanumeric Character for Current Display - Register 8392

Register 8392 is an 8-bit register that shows the alphanumeric character that is currently active on the display. The value read from register 8392 is the ASCII code for the character. An ASCII null (0) indicates that no alphanumeric character is displayed.

Registers 8392 to 8417
These registers hold an ASCII character that determines which alphanumeric character is inserted in the Least Significant Digit (LSD) for each of the different data displays. To enable the display of an ASCII character in the LSD, write the appropriate ASCII code (See ASCII Characters for 14-Segment Display 23) to one of these registers. To return to a normal display (with no ASCII character), write an ASCII null (0) to the register.

2.20.6 Display Format

The display format registers control how a register is displayed on the screen. This includes features such as the rounding of the least significant digit, the display units (i.e. number modulus) used to display the parameter and the position of the decimal point.

Display format registers are represented in octal 107 format to allow 3 functions to be selected in one digit.

Display Digit	1st Digit	2nd Digit	3rd Digit
Function	Last Digit Rounding	Display Units	Decimal Point

1st Digit - Rounding
The first digit of a display format register (bits 6 & 7) controls the rounding applied to the last digit of the displayed value. The options available are:

0 = No rounding
1 = Rounding by 2's
2 = Rounding by 5's
3 = Rounding by 10's.

2nd Digit - Display Units
The 2nd digit of a display format register (bits 3, 4, 5) controls the display units or modulus applied to a parameter before it is displayed. The options available are:

0 = Decimal
1 = 24 hour clock mode (i.e. Hours:Minutes:Seconds)
2 = 12 hour clock mode (i.e. 12:30 am is displayed as 12:30A, 12:30 pm is displayed as 12:30P)
3 = 24 hour clock mode (i.e. Days:Hours:Minutes)
4 = Reserved for future development
5 = Reserved for future development
6 = Reserved for future development
7 = Octal.

3rd Digit - Decimal Point
The 3rd digit of a display format register (bits 0,1, 2) controls the position of the decimal point when parameter is displayed. The options available are shown below.

0 = No decimal point
1 = XX:XX:XX (6-digit displays only)
2 = X.XXXXX (6-digit displays only)
3 = X.XXXX
4 = X.XXX
5 = X.XX
6 = X.X
7 = Decimal point is set from the rear pins. (see the Program Code Sheet for more details)

NOTE: The features selected in a display format register are applied to a parameter prior to it being displayed or transmitted via the serial port. The actual contents of the parameter are not changed by the selection of the display format.

For example, if rounding is applied the displayed value includes rounding but internally the data in the parameter is without rounding. This means that any other functions that reference this parameter work on the original data value before rounding. This is important to consider when using setpoints, etc.

The same is true of the display units and the decimal point. The decimal point is really only a pseudo decimal that is superimposed on top of the original count value. It does not change a fixed point integer into a true floating point number. When writing parameters via the serial port, it is often useful to ignore the decimal point and just work in display counts.

See also
Text 108

2.20.7 Octal Format

When editing configuration parameters (i.e. Codes) in the controller from a display panel, many of the parameters are displayed in an octal format. This allows several different functions to be selected with 3 digits. When reading or writing to these configuration registers via the serial port or the macro, the data is treated in octal format so that it is identical to the value shown on the display when setting the codes up manually. The function selected in the 1st digit of each Code register is stored in bits 6 and 7. The function selected in the 2nd digit of each Code register is stored in bits 3, 4, and 5. The function selected in the 3rd digit of each Code register is stored in bits 0, 1, and 2.

For example:
If the manual setup for Code 4 shows 241 on the display, then reading register 8197 in ASCII mode results in a value of 241. Converting this octal value to a binary equivalent of 10100001 or hexadecimal equivalent of 0A1.

	1st Digit	2nd Digit	3rd Digit
Octal	2	4	1
Binary	10	100	001

See also
Codes 52

Display Format 106

Setpoint Control Registers 158

2.20.8 Text

Name	Description	Symbol Type	Register Number	Memory Type
AUX1_TEXT	Text display for Auxiliary 1.	L_8	16463	EEPROM 8
AUX10_TEXT	Text display for Auxiliary 10.	L_8	16465	EEPROM 8
AUX11_TEXT	Text display for Auxiliary 11.	L_8	16467	EEPROM 8
AUX12_TEXT	Text display for Auxiliary 12.	L_8	16469	EEPROM 8
AUX13_TEXT	Text display for Auxiliary 13.	L_8	16471	EEPROM 8
AUX14_TEXT	Text display for Auxiliary 14.	L_8	16473	EEPROM 8
AUX15_TEXT	Text display for Auxiliary 15.	L_8	16475	EEPROM 8
AUX16_TEXT	Text display for Auxiliary 16.	L_8	16477	EEPROM 8
AUX2_TEXT	Text display for Auxiliary 2.	L_8	16479	EEPROM 8
AUX3_TEXT	Text display for Auxiliary 3.	L_8	16481	EEPROM 8
AUX4_TEXT	Text display for Auxiliary 4.	L_8	16483	EEPROM 8
AUX5_TEXT	Text display for Auxiliary 5.	L_8	16485	EEPROM 8
AUX6_TEXT	Text display for Auxiliary 6.	L_8	16487	EEPROM 8
AUX7_TEXT	Text display for Auxiliary 7.	L_8	16489	EEPROM 8
AUX8_TEXT	Text display for Auxiliary 8.	L_8	16491	EEPROM 8
AUX9_TEXT	Text display for Auxiliary 9.	L_8	16493	EEPROM 8
CHANNEL1_TEXT	Text display for Channel 1.	L_8	16393	EEPROM 8
CHANNEL2_TEXT	Text display for Channel 2.	L_8	16395	EEPROM 8
CHANNEL3_TEXT	Text display for Channel 3.	L_8	16397	EEPROM 8
CHANNEL4_TEXT	Text display for Channel 4.	L_8	16399	EEPROM 8
CHANNEL5_TEXT	Text display for Channel 5.	L_8	16401	EEPROM 8
CHANNEL6_TEXT	Text display for Channel 6.	L_8	16403	EEPROM 8
CHANNEL7_TEXT	Text display for Channel 7.	L_8	16405	EEPROM 8
DISPLAY_STRING	Read Register 1, Write Display Text	L_8	16385	RAM 8
OVER_TEXT	Text display for over range.	L_8	16539	EEPROM 8
PEAK1_TEXT	Text display for Peak 1.	L_8	16441	EEPROM 8
PEAK2_TEXT	Text display for Peak 2.	L_8	16445	EEPROM 8
PEAK3_TEXT	Text display for Peak 3.	L_8	16449	EEPROM 8
Print_String	Print String.	L_62	16543	EEPROM 8
PRINT_STRING	Print String.	L_62	16543	EEPROM 8
RESULT_TEXT	Text display for Result.	L_8	16391	EEPROM 8
SETPOINT1_TEXT	Text display for Setpoint 1.	L_8	16495	EEPROM 8
SETPOINT2_TEXT	Text display for Setpoint 2.	L_8	16497	EEPROM 8

See also
Display Format 106

2.21 Linearization

Linearization registers contain the input and output data for the 32 input and 32 output points of each linearization table, as well as the table's date and serial number.

Name	Description	Symbol Type	Register Number	Memory Type
TABLE1_DATE	16-bit register. Date (Year/Week) when linearization table 1 was last modified (range 0000 - 9952).	U_16	4277	EEPROM 8
TABLE1_SERIAL_NO	16-bit register. Serial number of linearization table 1 (range 0-65535).	U_16	4285	EEPROM 8
TABLE2_DATE	16-bit register. Date (Year/Week) when linearization table 2 was last modified (range 0000 - 9952).	U_16	4278	EEPROM 8
TABLE2_SERIAL_NO	16-bit register. Serial number of linearization table 2 (range 0-65535).	U_16	4286	EEPROM 8
TABLE3_DATE	16-bit register. Date (Year/Week) when linearization table 3 was last modified (range 0000 - 9952).	U_16	4279	EEPROM 8
TABLE3_SERIAL_NO	16-bit register. Serial number of linearization table 3 (range 0-65535).	U_16	4287	EEPROM 8
TABLE4_DATE	16-bit register. Date (Year/Week) when linearization table 4 was last modified (range 0000 - 9952).	U_16	4280	EEPROM 8
TABLE4_SERIAL_NO	16-bit register. Serial number of linearization table 4 (range 0-65535).	U_16	4288	EEPROM 8

See Also
Linearization Table 1 109

Linearization Table 2 113

Linearization Table 3 116

Linearization Table 4 119

2.21.1 Linearization Table 1

Name	Description	Symbol Type	Register Number	Memory Type
TABLE1_INPUT1	24-bit register. Input point 1, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2049	RAM/EEPROM 8
TABLE1_INPUT10	24-bit register. Input point 10, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2067	RAM/EEPROM 8
TABLE1_INPUT11	24-bit register. Input point 11, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2069	RAM/EEPROM 8
TABLE1_INPUT12	24-bit register. Input point 12, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2071	RAM/EEPROM 8
TABLE1_INPUT13	24-bit register. Input point 13, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2073	RAM/EEPROM 8

TABLE1_INPUT14	24-bit register. Input point 14, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2075	RAM/EEPROM 8
TABLE1_INPUT15	24-bit register. Input point 15, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2077	RAM/EEPROM 8
TABLE1_INPUT16	24-bit register. Input point 16, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2079	RAM/EEPROM 8
TABLE1_INPUT17	24-bit register. Input point 17, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2081	RAM/EEPROM 8
TABLE1_INPUT18	24-bit register. Input point 18, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2083	RAM/EEPROM 8
TABLE1_INPUT19	24-bit register. Input point 19, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2085	RAM/EEPROM 8
TABLE1_INPUT2	24-bit register. Input point 2, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2051	RAM/EEPROM 8
TABLE1_INPUT20	24-bit register. Input point 20, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2087	RAM/EEPROM 8
TABLE1_INPUT21	24-bit register. Input point 21, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2089	RAM/EEPROM 8
TABLE1_INPUT22	24-bit register. Input point 22, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2091	RAM/EEPROM 8
TABLE1_INPUT23	24-bit register. Input point 23, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2093	RAM/EEPROM 8
TABLE1_INPUT24	24-bit register. Input point 24, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2095	RAM/EEPROM 8
TABLE1_INPUT25	24-bit register. Input point 25, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2097	RAM/EEPROM 8
TABLE1_INPUT26	24-bit register. Input point 26, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2099	RAM/EEPROM 8

TABLE1_INPUT27	24-bit register. Input point 27, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2101	RAM/EEPROM 8
TABLE1_INPUT28	24-bit register. Input point 28, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2103	RAM/EEPROM 8
TABLE1_INPUT29	24-bit register. Input point 29, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2105	RAM/EEPROM 8
TABLE1_INPUT3	24-bit register. Input point 3, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2053	RAM/EEPROM 8
TABLE1_INPUT30	24-bit register. Input point 30, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2107	RAM/EEPROM 8
TABLE1_INPUT31	24-bit register. Input point 31, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2109	RAM/EEPROM 8
TABLE1_INPUT32	24-bit register. Input point 32, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2111	RAM/EEPROM 8
TABLE1_INPUT4	24-bit register. Input point 4, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2055	RAM/EEPROM 8
TABLE1_INPUT5	24-bit register. Input point 5, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2057	RAM/EEPROM 8
TABLE1_INPUT6	24-bit register. Input point 6, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2059	RAM/EEPROM 8
TABLE1_INPUT7	24-bit register. Input point 7, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2061	RAM/EEPROM 8
TABLE1_INPUT8	24-bit register. Input point 8, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2063	RAM/EEPROM 8
TABLE1_INPUT9	24-bit register. Input point 9, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2065	RAM/EEPROM 8
TABLE1_OUTPUT1	24-bit register. Output point 1, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2113	RAM/EEPROM 8
TABLE1_OUTPUT10	24-bit register. Output point 10, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2131	RAM/EEPROM 8
TABLE1_OUTPUT11	24-bit register. Output point 11, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2133	RAM/EEPROM 8
TABLE1_OUTPUT12	24-bit register. Output point 12, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2135	RAM/EEPROM 8
TABLE1_OUTPUT13	24-bit register. Output point 13, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2137	RAM/EEPROM 8
TABLE1_OUTPUT14	24-bit register. Output point 14, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2139	RAM/EEPROM 8

TABLE1_OUTPUT15	24-bit register. Output point 15, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2141	RAM/EEPROM 8
TABLE1_OUTPUT16	24-bit register. Output point 16, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2143	RAM/EEPROM 8
TABLE1_OUTPUT17	24-bit register. Output point 17, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2145	RAM/EEPROM 8
TABLE1_OUTPUT18	24-bit register. Output point 18, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2147	RAM/EEPROM 8
TABLE1_OUTPUT19	24-bit register. Output point 19, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2149	RAM/EEPROM 8
TABLE1_OUTPUT2	24-bit register. Output point 2, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2115	RAM/EEPROM 8
TABLE1_OUTPUT20	24-bit register. Output point 20, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2151	RAM/EEPROM 8
TABLE1_OUTPUT21	24-bit register. Output point 21, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2153	RAM/EEPROM 8
TABLE1_OUTPUT22	24-bit register. Output point 22, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2155	RAM/EEPROM 8
TABLE1_OUTPUT23	24-bit register. Output point 23, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2157	RAM/EEPROM 8
TABLE1_OUTPUT24	24-bit register. Output point 24, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2159	RAM/EEPROM 8
TABLE1_OUTPUT25	24-bit register. Output point 25, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2161	RAM/EEPROM 8
TABLE1_OUTPUT26	24-bit register. Output point 26, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2163	RAM/EEPROM 8
TABLE1_OUTPUT27	24-bit register. Output point 27, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2165	RAM/EEPROM 8
TABLE1_OUTPUT28	24-bit register. Output point 28, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2167	RAM/EEPROM 8
TABLE1_OUTPUT29	24-bit register. Output point 29, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2169	RAM/EEPROM 8
TABLE1_OUTPUT3	24-bit register. Output point 3, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2117	RAM/EEPROM 8
TABLE1_OUTPUT30	24-bit register. Output point 30, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2171	RAM/EEPROM 8
TABLE1_OUTPUT31	24-bit register. Output point 31, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2173	RAM/EEPROM 8
TABLE1_OUTPUT32	24-bit register. Output point 32, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2175	RAM/EEPROM 8
TABLE1_OUTPUT4	24-bit register. Output point 4, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2119	RAM/EEPROM 8
TABLE1_OUTPUT5	24-bit register. Output point 5, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2121	RAM/EEPROM 8
TABLE1_OUTPUT6	24-bit register. Output point 6, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2123	RAM/EEPROM 8
TABLE1_OUTPUT7	24-bit register. Output point 7, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2125	RAM/EEPROM 8
TABLE1_OUTPUT8	24-bit register. Output point 8, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2127	RAM/EEPROM 8
TABLE1_OUTPUT9	24-bit register. Output point 9, 32-point linearization table 1 (range -8388607 - +8388607).	S_24	2129	RAM/EEPROM 8

2.21.2 Linearization Table 2

Name	Description	Symbol Type	Register Number	Memory Type
TABLE2_INPUT1	24-bit register. Input point 1, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2177	RAM/EEPROM 8
TABLE2_INPUT10	24-bit register. Input point 10, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2195	RAM/EEPROM 8
TABLE2_INPUT11	24-bit register. Input point 11, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2197	RAM/EEPROM 8
TABLE2_INPUT12	24-bit register. Input point 12, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2199	RAM/EEPROM 8
TABLE2_INPUT13	24-bit register. Input point 13, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2201	RAM/EEPROM 8
TABLE2_INPUT14	24-bit register. Input point 14, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2203	RAM/EEPROM 8
TABLE2_INPUT15	24-bit register. Input point 15, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2205	RAM/EEPROM 8
TABLE2_INPUT16	24-bit register. Input point 16, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2207	RAM/EEPROM 8
TABLE2_INPUT17	24-bit register. Input point 17, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2209	RAM/EEPROM 8
TABLE2_INPUT18	24-bit register. Input point 18, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2211	RAM/EEPROM 8
TABLE2_INPUT19	24-bit register. Input point 19, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2213	RAM/EEPROM 8
TABLE2_INPUT2	24-bit register. Input point 2, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2179	RAM/EEPROM 8
TABLE2_INPUT20	24-bit register. Input point 20, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2215	RAM/EEPROM 8
TABLE2_INPUT21	24-bit register. Input point 21, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2217	RAM/EEPROM 8
TABLE2_INPUT22	24-bit register. Input point 22, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2219	RAM/EEPROM 8
TABLE2_INPUT23	24-bit register. Input point 23, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2221	RAM/EEPROM 8
TABLE2_INPUT24	24-bit register. Input point 24, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2223	RAM/EEPROM 8
TABLE2_INPUT25	24-bit register. Input point 25, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2225	RAM/EEPROM 8
TABLE2_INPUT26	24-bit register. Input point 26, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2227	RAM/EEPROM 8

TABLE2_INPUT27	24-bit register. Input point 27, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2229	RAM/EEPROM 8
TABLE2_INPUT28	24-bit register. Input point 28, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2231	RAM/EEPROM 8
TABLE2_INPUT29	24-bit register. Input point 29, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2233	RAM/EEPROM 8
TABLE2_INPUT3	24-bit register. Input point 3, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2181	RAM/EEPROM 8
TABLE2_INPUT30	24-bit register. Input point 30, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2235	RAM/EEPROM 8
TABLE2_INPUT31	24-bit register. Input point 31, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2237	RAM/EEPROM 8
TABLE2_INPUT32	24-bit register. Input point 32, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2239	RAM/EEPROM 8
TABLE2_INPUT4	24-bit register. Input point 4, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2183	RAM/EEPROM 8
TABLE2_INPUT5	24-bit register. Input point 5, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2185	RAM/EEPROM 8
TABLE2_INPUT6	24-bit register. Input point 6, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2187	RAM/EEPROM 8
TABLE2_INPUT7	24-bit register. Input point 7, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2189	RAM/EEPROM 8
TABLE2_INPUT8	24-bit register. Input point 8, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2191	RAM/EEPROM 8
TABLE2_INPUT9	24-bit register. Input point 9, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2193	RAM/EEPROM 8
TABLE2_OUTPUT1	24-bit register. Output point 1, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2241	RAM/EEPROM 8
TABLE2_OUTPUT10	24-bit register. Output point 10, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2259	RAM/EEPROM 8
TABLE2_OUTPUT11	24-bit register. Output point 11, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2261	RAM/EEPROM 8
TABLE2_OUTPUT12	24-bit register. Output point 12, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2263	RAM/EEPROM 8
TABLE2_OUTPUT13	24-bit register. Output point 13, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2265	RAM/EEPROM 8
TABLE2_OUTPUT14	24-bit register. Output point 14, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2267	RAM/EEPROM 8

TABLE2_OUTPUT15	24-bit register. Output point 15, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2269	RAM/EEPROM 8
TABLE2_OUTPUT16	24-bit register. Output point 16, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2271	RAM/EEPROM 8
TABLE2_OUTPUT17	24-bit register. Output point 17, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2273	RAM/EEPROM 8
TABLE2_OUTPUT18	24-bit register. Output point 18, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2275	RAM/EEPROM 8
TABLE2_OUTPUT19	24-bit register. Output point 19, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2277	RAM/EEPROM 8
TABLE2_OUTPUT2	24-bit register. Output point 2, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2243	RAM/EEPROM 8
TABLE2_OUTPUT20	24-bit register. Output point 20, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2279	RAM/EEPROM 8
TABLE2_OUTPUT21	24-bit register. Output point 21, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2281	RAM/EEPROM 8
TABLE2_OUTPUT22	24-bit register. Output point 22, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2283	RAM/EEPROM 8
TABLE2_OUTPUT23	24-bit register. Output point 23, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2285	RAM/EEPROM 8
TABLE2_OUTPUT24	24-bit register. Output point 24, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2287	RAM/EEPROM 8
TABLE2_OUTPUT25	24-bit register. Output point 25, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2289	RAM/EEPROM 8
TABLE2_OUTPUT26	24-bit register. Output point 26, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2291	RAM/EEPROM 8
TABLE2_OUTPUT27	24-bit register. Output point 27, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2293	RAM/EEPROM 8
TABLE2_OUTPUT28	24-bit register. Output point 28, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2295	RAM/EEPROM 8
TABLE2_OUTPUT29	24-bit register. Output point 29, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2297	RAM/EEPROM 8
TABLE2_OUTPUT3	24-bit register. Output point 3, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2245	RAM/EEPROM 8
TABLE2_OUTPUT30	24-bit register. Output point 30, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2299	RAM/EEPROM 8
TABLE2_OUTPUT31	24-bit register. Output point 31, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2301	RAM/EEPROM 8
TABLE2_OUTPUT32	24-bit register. Output point 32, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2303	RAM/EEPROM 8
TABLE2_OUTPUT4	24-bit register. Output point 4, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2247	RAM/EEPROM 8
TABLE2_OUTPUT5	24-bit register. Output point 5, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2249	RAM/EEPROM 8
TABLE2_OUTPUT6	24-bit register. Output point 6, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2251	RAM/EEPROM 8
TABLE2_OUTPUT7	24-bit register. Output point 7, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2253	RAM/EEPROM 8
TABLE2_OUTPUT8	24-bit register. Output point 8, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2255	RAM/EEPROM 8
TABLE2_OUTPUT9	24-bit register. Output point 9, 32-point linearization table 2 (range -8388607 - +8388607).	S_24	2257	RAM/EEPROM 8

2.21.3 Linearization Table 3

Name	Description	Symbol Type	Register Number	Memory Type
TABLE3_INPUT1	24-bit register. Input point 1, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2305	RAM/EEPROM 8
TABLE3_INPUT10	24-bit register. Input point 10, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2323	RAM/EEPROM 8
TABLE3_INPUT11	24-bit register. Input point 11, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2325	RAM/EEPROM 8
TABLE3_INPUT12	24-bit register. Input point 12, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2327	RAM/EEPROM 8
TABLE3_INPUT13	24-bit register. Input point 13, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2329	RAM/EEPROM 8
TABLE3_INPUT14	24-bit register. Input point 14, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2331	RAM/EEPROM 8
TABLE3_INPUT15	24-bit register. Input point 15, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2333	RAM/EEPROM 8
TABLE3_INPUT16	24-bit register. Input point 16, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2335	RAM/EEPROM 8
TABLE3_INPUT17	24-bit register. Input point 17, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2337	RAM/EEPROM 8
TABLE3_INPUT18	24-bit register. Input point 18, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2339	RAM/EEPROM 8
TABLE3_INPUT19	24-bit register. Input point 19, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2341	RAM/EEPROM 8
TABLE3_INPUT2	24-bit register. Input point 2, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2307	RAM/EEPROM 8
TABLE3_INPUT20	24-bit register. Input point 20, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2343	RAM/EEPROM 8
TABLE3_INPUT21	24-bit register. Input point 21, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2345	RAM/EEPROM 8
TABLE3_INPUT22	24-bit register. Input point 22, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2347	RAM/EEPROM 8
TABLE3_INPUT23	24-bit register. Input point 23, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2349	RAM/EEPROM 8
TABLE3_INPUT24	24-bit register. Input point 24, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2351	RAM/EEPROM 8
TABLE3_INPUT25	24-bit register. Input point 25, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2353	RAM/EEPROM 8
TABLE3_INPUT26	24-bit register. Input point 26, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2355	RAM/EEPROM 8

TABLE3_INPUT27	24-bit register. Input point 27, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2357	RAM/EEPROM 8
TABLE3_INPUT28	24-bit register. Input point 28, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2359	RAM/EEPROM 8
TABLE3_INPUT29	24-bit register. Input point 29, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2361	RAM/EEPROM 8
TABLE3_INPUT3	24-bit register. Input point 3, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2309	RAM/EEPROM 8
TABLE3_INPUT30	24-bit register. Input point 30, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2363	RAM/EEPROM 8
TABLE3_INPUT31	24-bit register. Input point 31, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2365	RAM/EEPROM 8
TABLE3_INPUT32	24-bit register. Input point 32, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2367	RAM/EEPROM 8
TABLE3_INPUT4	24-bit register. Input point 4, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2311	RAM/EEPROM 8
TABLE3_INPUT5	24-bit register. Input point 5, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2313	RAM/EEPROM 8
TABLE3_INPUT6	24-bit register. Input point 6, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2315	RAM/EEPROM 8
TABLE3_INPUT7	24-bit register. Input point 7, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2317	RAM/EEPROM 8
TABLE3_INPUT8	24-bit register. Input point 8, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2319	RAM/EEPROM 8
TABLE3_INPUT9	24-bit register. Input point 9, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2321	RAM/EEPROM 8
TABLE3_OUTPUT1	24-bit register. Output point 1, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2369	RAM/EEPROM 8
TABLE3_OUTPUT10	24-bit register. Output point 10, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2387	RAM/EEPROM 8
TABLE3_OUTPUT11	24-bit register. Output point 11, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2389	RAM/EEPROM 8
TABLE3_OUTPUT12	24-bit register. Output point 12, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2391	RAM/EEPROM 8
TABLE3_OUTPUT13	24-bit register. Output point 13, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2393	RAM/EEPROM 8
TABLE3_OUTPUT14	24-bit register. Output point 14, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2395	RAM/EEPROM 8

TABLE3_OUTPUT15	24-bit register. Output point 15, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2397	RAM/EEPROM 8
TABLE3_OUTPUT16	24-bit register. Output point 16, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2399	RAM/EEPROM 8
TABLE3_OUTPUT17	24-bit register. Output point 17, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2401	RAM/EEPROM 8
TABLE3_OUTPUT18	24-bit register. Output point 18, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2403	RAM/EEPROM 8
TABLE3_OUTPUT19	24-bit register. Output point 19, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2405	RAM/EEPROM 8
TABLE3_OUTPUT2	24-bit register. Output point 2, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2371	RAM/EEPROM 8
TABLE3_OUTPUT20	24-bit register. Output point 20, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2407	RAM/EEPROM 8
TABLE3_OUTPUT21	24-bit register. Output point 21, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2409	RAM/EEPROM 8
TABLE3_OUTPUT22	24-bit register. Output point 22, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2411	RAM/EEPROM 8
TABLE3_OUTPUT23	24-bit register. Output point 23, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2413	RAM/EEPROM 8
TABLE3_OUTPUT24	24-bit register. Output point 24, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2415	RAM/EEPROM 8
TABLE3_OUTPUT25	24-bit register. Output point 25, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2417	RAM/EEPROM 8
TABLE3_OUTPUT26	24-bit register. Output point 26, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2419	RAM/EEPROM 8
TABLE3_OUTPUT27	24-bit register. Output point 27, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2421	RAM/EEPROM 8
TABLE3_OUTPUT28	24-bit register. Output point 28, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2423	RAM/EEPROM 8
TABLE3_OUTPUT29	24-bit register. Output point 29, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2425	RAM/EEPROM 8
TABLE3_OUTPUT3	24-bit register. Output point 3, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2373	RAM/EEPROM 8
TABLE3_OUTPUT30	24-bit register. Output point 30, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2427	RAM/EEPROM 8
TABLE3_OUTPUT31	24-bit register. Output point 31, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2429	RAM/EEPROM 8
TABLE3_OUTPUT32	24-bit register. Output point 32, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2431	RAM/EEPROM 8
TABLE3_OUTPUT4	24-bit register. Output point 4, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2375	RAM/EEPROM 8
TABLE3_OUTPUT5	24-bit register. Output point 5, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2377	RAM/EEPROM 8
TABLE3_OUTPUT6	24-bit register. Output point 6, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2379	RAM/EEPROM 8
TABLE3_OUTPUT7	24-bit register. Output point 7, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2381	RAM/EEPROM 8
TABLE3_OUTPUT8	24-bit register. Output point 8, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2383	RAM/EEPROM 8
TABLE3_OUTPUT9	24-bit register. Output point 9, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2385	RAM/EEPROM 8

2.21.4 Linearization Table 4

Name	Description	Symbol Type	Register Number	Memory Type
TABLE4_INPUT1	24-bit register. Input point 1, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2433	RAM/EEPROM 8
TABLE4_INPUT10	24-bit register. Input point 10, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2451	RAM/EEPROM 8
TABLE4_INPUT11	24-bit register. Input point 11, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2453	RAM/EEPROM 8
TABLE4_INPUT12	24-bit register. Input point 12, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2455	RAM/EEPROM 8
TABLE4_INPUT13	24-bit register. Input point 13, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2457	RAM/EEPROM 8
TABLE4_INPUT14	24-bit register. Input point 14, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2459	RAM/EEPROM 8
TABLE4_INPUT15	24-bit register. Input point 15, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2461	RAM/EEPROM 8
TABLE4_INPUT16	24-bit register. Input point 16, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2463	RAM/EEPROM 8
TABLE4_INPUT17	24-bit register. Input point 17, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2465	RAM/EEPROM 8
TABLE4_INPUT18	24-bit register. Input point 18, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2467	RAM/EEPROM 8
TABLE4_INPUT19	24-bit register. Input point 19, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2469	RAM/EEPROM 8
TABLE4_INPUT2	24-bit register. Input point 2, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2435	RAM/EEPROM 8
TABLE4_INPUT20	24-bit register. Input point 20, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2471	RAM/EEPROM 8
TABLE4_INPUT21	24-bit register. Input point 21, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2473	RAM/EEPROM 8
TABLE4_INPUT22	24-bit register. Input point 22, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2475	RAM/EEPROM 8
TABLE4_INPUT23	24-bit register. Input point 23, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2477	RAM/EEPROM 8
TABLE4_INPUT24	24-bit register. Input point 24, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2479	RAM/EEPROM 8
TABLE4_INPUT25	24-bit register. Input point 25, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2481	RAM/EEPROM 8
TABLE4_INPUT26	24-bit register. Input point 26, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2483	RAM/EEPROM 8

TABLE4_INPUT27	24-bit register. Input point 27, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2485	RAM/EEPROM 8
TABLE4_INPUT28	24-bit register. Input point 28, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2487	RAM/EEPROM 8
TABLE4_INPUT29	24-bit register. Input point 29, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2489	RAM/EEPROM 8
TABLE4_INPUT3	24-bit register. Input point 3, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2437	RAM/EEPROM 8
TABLE4_INPUT30	24-bit register. Input point 30, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2491	RAM/EEPROM 8
TABLE4_INPUT31	24-bit register. Input point 31, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2493	RAM/EEPROM 8
TABLE4_INPUT32	24-bit register. Input point 32, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2495	RAM/EEPROM 8
TABLE4_INPUT4	24-bit register. Input point 4, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2439	RAM/EEPROM 8
TABLE4_INPUT5	24-bit register. Input point 5, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2441	RAM/EEPROM 8
TABLE4_INPUT6	24-bit register. Input point 6, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2443	RAM/EEPROM 8
TABLE4_INPUT7	24-bit register. Input point 7, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2445	RAM/EEPROM 8
TABLE4_INPUT8	24-bit register. Input point 8, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2447	RAM/EEPROM 8
TABLE4_INPUT9	24-bit register. Input point 9, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2449	RAM/EEPROM 8
TABLE4_OUTPUT1	24-bit register. Output point 1, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2497	RAM/EEPROM 8
TABLE4_OUTPUT10	24-bit register. Output point 10, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2515	RAM/EEPROM 8
TABLE4_OUTPUT11	24-bit register. Output point 11, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2517	RAM/EEPROM 8
TABLE4_OUTPUT12	24-bit register. Output point 12, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2519	RAM/EEPROM 8
TABLE4_OUTPUT13	24-bit register. Output point 13, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2521	RAM/EEPROM 8
TABLE4_OUTPUT14	24-bit register. Output point 14, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2523	RAM/EEPROM 8

TABLE4_OUTPUT15	24-bit register. Output point 15, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2525	RAM/EEPROM 8
TABLE4_OUTPUT16	24-bit register. Output point 16, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2527	RAM/EEPROM 8
TABLE4_OUTPUT17	24-bit register. Output point 17, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2529	RAM/EEPROM 8
TABLE4_OUTPUT18	24-bit register. Output point 18, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2531	RAM/EEPROM 8
TABLE4_OUTPUT19	24-bit register. Output point 19, 32-point linearization table 3 (range -8388607 - +8388607).	S_24	2533	RAM/EEPROM 8
TABLE4_OUTPUT2	24-bit register. Output point 2, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2499	RAM/EEPROM 8
TABLE4_OUTPUT20	24-bit register. Output point 20, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2535	RAM/EEPROM 8
TABLE4_OUTPUT21	24-bit register. Output point 21, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2537	RAM/EEPROM 8
TABLE4_OUTPUT22	24-bit register. Output point 22, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2539	RAM/EEPROM 8
TABLE4_OUTPUT23	24-bit register. Output point 23, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2541	RAM/EEPROM 8
TABLE4_OUTPUT24	24-bit register. Output point 24, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2543	RAM/EEPROM 8
TABLE4_OUTPUT25	24-bit register. Output point 25, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2545	RAM/EEPROM 8
TABLE4_OUTPUT26	24-bit register. Output point 26, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2547	RAM/EEPROM 8
TABLE4_OUTPUT27	24-bit register. Output point 27, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2549	RAM/EEPROM 8
TABLE4_OUTPUT28	24-bit register. Output point 28, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2551	RAM/EEPROM 8
TABLE4_OUTPUT29	24-bit register. Output point 29, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2553	RAM/EEPROM 8
TABLE4_OUTPUT3	24-bit register. Output point 3, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2501	RAM/EEPROM 8
TABLE4_OUTPUT30	24-bit register. Output point 30, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2555	RAM/EEPROM 8
TABLE4_OUTPUT31	24-bit register. Output point 31, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2557	RAM/EEPROM 8
TABLE4_OUTPUT32	24-bit register. Output point 32, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2559	RAM/EEPROM 8
TABLE4_OUTPUT4	24-bit register. Output point 4, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2503	RAM/EEPROM 8
TABLE4_OUTPUT5	24-bit register. Output point 5, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2505	RAM/EEPROM 8
TABLE4_OUTPUT6	24-bit register. Output point 6, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2507	RAM/EEPROM 8
TABLE4_OUTPUT7	24-bit register. Output point 7, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2509	RAM/EEPROM 8
TABLE4_OUTPUT8	24-bit register. Output point 8, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2511	RAM/EEPROM 8
TABLE4_OUTPUT9	24-bit register. Output point 9, 32-point linearization table 4 (range -8388607 - +8388607).	S_24	2513	RAM/EEPROM 8

2.22 PID

PID registers contain the individual proportional, integral, and derivative data for each independent PID output.

See also

PID 1 123

PID 2 125

PID 3 127

PID 4 129

PID 5 131

PID 6 133

Register 1057 to 1079 - PID Control Registers 139

PID Manual Mode 134

NOTE: Setpoints 7 to 12 do not have any PID functions.

See Setpoints and Relays Supplement (NZ201) for a detailed description of the relationship between PID and the setpoints.

2.22.1 PID 1

Name	Description	Symbol Type	Register Number	Memory Type
PID1_IW_GAIN	16-bit register. PID integral windup gain for setpoint 1. (range 1 - 65535).	U_16	4245	RAM/EEPROM 8
PID1_CYCLE_TIME	16-bit register. PID cycle time for setpoint 1 (range 0-1000.0 seconds).	U_16	4261	RAM/EEPROM 8
PID1_DERIVATIVE_TIME	32-bit register sets the derivative time for PID1 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the derivative action.	U_32	725	RAM/EEPROM 8
PID1_ERRD_OLD 139	32-bit floating point register which contains the derivative error sum value for PID1.	FP_32	1057 139	RAM 8
PID1_INTEGRAL_TERM 139	32-bit floating point register that contains the integral term for PID1.	FP_32	1069 139	RAM 8
PID1_INTEGRAL_TIME	32-bit register sets the integral time for PID1 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the integral action.	U_32	701	RAM/EEPROM 8
PID1_OUTPUT	16-bit register holds the PID output value from -10000 to +10000 counts when SP1 is set to PID mode.	S_16	4562	RAM 8
PID1_KC	32-bit register sets the controller gain for PID1 loop from 0.01 to 999.99 (where 100 counts = gain of 1.00, so PID1_KC = Gain(Kc) x 100).	U_32	749	RAM/EEPROM 8
PID1_MAN_OUT 134	32-bit floating point register sets PID 1 output value when operating in manual output mode. (See PID Auto/Manual Mode 134)	FP_32	1207 134	RAM 8
PID1_ALPHA_SP 135	16-bit register which sets the time constant for filtering of setpoint 1. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4586	RAM/EEPROM 8
PID1_B_SPTW 136	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 1. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8573	RAM/EEPROM 8
PID1_C_SPTW 137	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 1. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8585	RAM/EEPROM 8
PID1_N_DER	8-bit register which sets the scale factor for the derivative of setpoint 1. Range is from 1-255 counts. Factory default value is 1.	U_8	8597	RAM/EEPROM 8
PID1_INC_DEC 134	16-bit signed register controls the PID increment/decrement rate when PID 1 is operating in manual mode. (range -32768 to +32767).	S_16	4526	RAM 8
PID1_SAT_HIGH 138	16-bit signed register that controls the high saturation output level for PID1_OUTPUT. (range -10,000 to +10,000. Note: must be > PID1_SAT_LOW).	S_16	4538	RAM/EEPROM 8
PID1_SAT_LOW 138	16-bit signed register that controls the low saturation output level for PID1_OUTPUT. (range -10,000 to +10,000. Note: must be < PID1_SAT_HIGH).	S_16	4550	RAM/EEPROM 8

See also

SP1 Setup 167

PID 122

2.22.2 PID 2

Name	Description	Symbol Type	Register Number	Memory Type
PID2_IW_GAIN	16-bit register. PID integral windup gain for setpoint 2. (range 1 - 65535).	U_16	4246	RAM/EEPROM 8
PID2_CYCLE_TIME	16-bit register. PID cycle time for setpoint 2 (range 0-1000.0 seconds).	U_16	4262	RAM/EEPROM 8
PID2_DERIVATIVE_TIME	32-bit register sets the derivative time for PID2 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the derivative action.	U_32	727	RAM/EEPROM 8
PID2_ERRD_OLD 139	32-bit floating point register which contains the derivative error sum value for PID2.	FP_32	1059 139	RAM 8
PID2_INTEGRAL_TERM 139	32-bit floating point register that contains the integral term for PID2.	FP_32	1071 139	RAM 8
PID2_INTEGRAL_TIME	32-bit register sets the integral time for PID2 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the integral action.	U_32	703	RAM/EEPROM 8
PID2_OUTPUT	16-bit register holds the PID output value from -10000 to +10000 counts when SP2 is set to PID mode.	S_16	4563	RAM 8
PID2_KC	32-bit register sets the controller gain for PID2 loop from 0.01 to 999.99 (where 100 counts = gain of 1.00, so PID2_KC = Gain(Kc) x 100).	U_32	751	RAM/EEPROM 8
PID2_MAN_OUT 134	32-bit floating point register sets PID 2 output value when operating in manual output mode. (See PID Auto/Manual Mode 134)	FP_32	1209 134	RAM 8
PID2_ALPHA_SP 135	16-bit register which sets the time constant for filtering of setpoint 2. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4587	RAM/EEPROM 8
PID2_B_SPTW 136	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 2. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8574	RAM/EEPROM 8
PID2_C_SPTW 137	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 2. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8586	RAM/EEPROM 8
PID2_N_DER	8-bit register which sets the scale factor for the derivative of setpoint 2. Range is from 1-255 counts. Factory default value is 1.	U_8	8598	RAM/EEPROM 8
PID2_INC_DEC 134	16-bit signed register controls the PID increment/decrement rate when PID 2 is operating in manual mode. (range -32768 to +32767).	S_16	4527	RAM 8
PID2_SAT_HIGH 138	16-bit signed register that controls the high saturation output level for PID2_OUTPUT. (range -10,000 to +10,000. Note: must be > PID2_SAT_LOW).	S_16	4539	RAM/EEPROM 8
PID2_SAT_LOW 138	16-bit signed register that controls the low saturation output level for PID2_OUTPUT. (range -10,000 to +10,000. Note: must be < PID2_SAT_HIGH).	S_16	4551	RAM/EEPROM 8

See also

SP2 Setup 169

PID 122

2.22.3 PID 3

Name	Description	Symbol Type	Register Number	Memory Type
PID3_IW_GAIN	16-bit register. PID integral windup gain for setpoint 3. (range 1 - 65535).	U_16	4247	RAM/EEPROM 8
PID3_CYCLE_TIME	16-bit register. PID cycle time for setpoint 3 (range 0-1000.0 seconds).	U_16	4263	RAM/EEPROM 8
PID3_DERIVATIVE_TIME	32-bit register sets the derivative time for PID3 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the derivative action.	U_32	729	RAM/EEPROM 8
PID3_ERRD_OLD 139	32-bit floating point register which contains the derivative error sum value for PID3.	FP_32	1061 139	RAM 8
PID3_INTEGRAL_TERM 139	32-bit floating point register that contains the integral term for PID3.	FP_32	1073 139	RAM 8
PID3_INTEGRAL_TIME	32-bit register sets the integral time for PID3 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the integral action.	U_32	705	RAM/EEPROM 8
PID3_OUTPUT	16-bit register holds the PID output value from -10000 to +10000 counts when SP3 is set to PID mode.	S_16	4564	RAM 8
PID3_KC	32-bit register sets the controller gain for PID3 loop from 0.01 to 999.99 (where 100 counts = gain of 1.00, so PID3_KC = Gain(Kc) x 100).	U_32	753	RAM/EEPROM 8
PID3_MAN_OUT 134	32-bit floating point register sets PID 3 output value when operating in manual output mode. (See PID Auto/Manual Mode 134)	FP_32	1211 134	RAM 8
PID3_ALPHA_SP 135	16-bit register which sets the time constant for filtering of setpoint 3. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4588	RAM/EEPROM 8
PID3_B_SPTW 136	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 3. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8575	RAM/EEPROM 8
PID3_C_SPTW 137	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 3. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8587	RAM/EEPROM 8
PID3_N_DER	8-bit register which sets the scale factor for the derivative of setpoint 3. Range is from 1-255 counts. Factory default value is 1.	U_8	8599	RAM/EEPROM 8
PID3_INC_DEC 134	16-bit signed register controls the PID increment/decrement rate when PID 3 is operating in manual mode. (range -32768 to +32767).	S_16	4528	RAM 8
3	16-bit signed register that controls the high saturation output level for PID3_OUTPUT. (range -10,000 to +10,000. Note: must be > PID3_SAT_LOW).	S_16	4540	RAM/EEPROM 8
PID3_SAT_LOW 138	16-bit signed register that controls the high saturation output level for PID3_OUTPUT. (range -10,000 to +10,000. Note: must be < PID3_SAT_HIGH).	S_16	4552	RAM/EEPROM 8

See also

SP3 Setup 171

PID 122

2.22.4 PID 4

Name	Description	Symbol Type	Register Number	Memory Type
PID4_IW_GAIN	16-bit register. PID integral windup gain for setpoint 4. (range 1 - 65535).	U_16	4248	RAM/EEPROM 8
PID4_CYCLE_TIME	16-bit register. PID cycle time for setpoint 4 (range 0-1000.0 seconds).	U_16	4264	RAM/EEPROM 8
PID4_DERIVATIVE_TIME	32-bit register sets the derivative time for PID4 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the derivative action.	U_32	731	RAM/EEPROM 8
PID4_ERRD_OLD 139	32-bit floating point register which contains the derivative error sum value for PID4.	FP_32	1063 139	RAM 8
PID4_INTEGRAL_TERM 139	32-bit floating point register that contains the integral term for PID4.	FP_32	1075 139	RAM 8
PID4_INTEGRAL_TIME	32-bit register sets the integral time for PID4 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the integral action.	U_32	707	RAM/EEPROM 8
PID4_OUTPUT	16-bit register holds the PID output value from -10000 to +10000 counts when SP4 is set to PID mode.	S_16	4565	RAM 8
PID4_KC	32-bit register sets the controller gain for PID4 loop from 0.01 to 999.99 (where 100 counts = gain of 1.00, so PID4_KC = Gain(Kc) x 100).	U_32	755	RAM/EEPROM 8
PID4_MAN_OUT 134	32-bit floating point register sets PID 4 output value when operating in manual output mode. (See PID Auto/Manual Mode 134)	FP_32	1213 134	RAM 8
PID4_ALPHA_SP 135	16-bit register which sets the time constant for filtering of setpoint 4. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4589	RAM/EEPROM 8
PID4_B_SPTW 136	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 4. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8576	RAM/EEPROM 8
PID4_C_SPTW 137	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 4. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8588	RAM/EEPROM 8
PID4_N_DER	8-bit register which sets the scale factor for the derivative of setpoint 4. Range is from 1-255 counts. Factory default value is 1.	U_8	8600	RAM/EEPROM 8
PID4_INC_DEC 134	16-bit signed register controls the PID increment/decrement rate when PID 4 is operating in manual mode. (range -32768 to +32767).	S_16	4529	RAM 8
PID4_SAT_HIGH 138	16-bit signed register that controls the high saturation output level for PID4_OUTPUT. (range -10,000 to +10,000. Note: must be > PID4_SAT_LOW).	S_16	4541	RAM/EEPROM 8
PID4_SAT_LOW 138	16-bit signed register that controls the high saturation output level for PID4_OUTPUT. (range -10,000 to +10,000. Note: must be < PID4_SAT_HIGH).	S_16	4553	RAM/EEPROM 8

See also

SP4 Setup 173

PID 122

2.22.5 PID 5

Name	Description	Symbol Type	Register Number	Memory Type
PID5_IW_GAIN	16-bit register. PID integral windup gain for setpoint 5. (range 1 - 65535).	U_16	4249	RAM/EEPROM 8
PID5_CYCLE_TIME	16-bit register. PID cycle time for setpoint 5 (range 0-1000.0 seconds).	U_16	4265	RAM/EEPROM 8
PID5_DERIVATIVE_TIME	32-bit register sets the derivative time for PID5 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the derivative action.	U_32	733	RAM/EEPROM 8
PID5_ERRD_OLD 139	32-bit floating point register which contains the derivative error sum value for PID5.	FP_32	1065 139	RAM 8
PID5_INTEGRAL_TERM 139	32-bit floating point register that contains the integral term for PID5	FP_32	1077 139	RAM 8
PID5_INTEGRAL_TIME	32-bit register sets the integral time for PID5 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the integral action.	U_32	709	RAM/EEPROM 8
PID5_OUTPUT	16-bit register holds the PID output value from -10000 to +10000 counts when SP5 is set to PID mode.	S_16	4566	RAM 8
PID5_KC	32-bit register sets the controller gain for PID5 loop from 0.01 to 999.99 (where 100 counts = gain of 1.00, so PID5_KC = Gain(Kc) x 100).	U_32	757	RAM/EEPROM 8
PID5_MAN_OUT 134	32-bit floating point register sets PID 5 output value when operating in manual output mode. (See PID Auto/Manual Mode 134)	FP_32	1215 134	RAM 8
PID5_ALPHA_SP 135	16-bit register which sets the time constant for filtering of setpoint 5. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4590	RAM/EEPROM 8
PID5_B_SPTW 136	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 5. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8577	RAM/EEPROM 8
PID5_C_SPTW 137	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 5. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8589	RAM/EEPROM 8
PID5_N_DER	8-bit register which sets the scale factor for the derivative of setpoint 5. Range is from 1-255 counts. Factory default value is 1.	U_8	8601	RAM/EEPROM 8
PID5_INC_DEC 134	16-bit signed register controls the PID increment/decrement rate when PID 5 is operating in manual mode. (range -32768 to +32767).	S_16	4530	RAM 8
PID5_SAT_HIGH 138	16-bit signed register that controls the high saturation output level for PID5_OUTPUT. (range -10,000 to +10,000. Note: must be > PID5_SAT_LOW).	S_16	4542	RAM/EEPROM 8
PID5_SAT_LOW 138	16-bit signed register that controls the high saturation output level for PID5_OUTPUT. (range -10,000 to +10,000. Note: must be < PID5_SAT_HIGH).	S_16	4554	RAM/EEPROM 8

See also

SP5 Setup 175

PID 122

2.22.6 PID 6

Name	Description	Symbol Type	Register Number	Memory Type
PID6_IW_GAIN	16-bit register. PID integral windup gain for setpoint 6. (range 1 - 65535).	U_16	4250	RAM/EEPROM 8
PID6_CYCLE_TIME	16-bit register. PID cycle time for setpoint 6 (range 0-1000.0 seconds).	U_16	4266	RAM/EEPROM 8
PID6_DERIVATIVE_TIME	32-bit register sets the derivative time for PID6 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the derivative action.	U_32	735	RAM/EEPROM 8
PID6_ERRD_OLD 139	32-bit floating point register which contains the derivative error sum value for PID6.	FP_32	1067 139	RAM 8
PID6_INTEGRAL_TERM 139	32-bit floating point register that contains the integral term for PID5	FP_32	1079 139	RAM 8
PID6_INTEGRAL_TIME	32-bit register sets the integral time for PID6 from 0.01 to 999.99 seconds in 0.01 second resolution (100 counts = 1 second). Setting time to 0 turns off the integral action.	U_32	711	RAM/EEPROM 8
PID6_OUTPUT	16-bit register holds the PID output value from -10000 to +10000 counts when SP6 is set to PID mode.	S_16	4567	RAM 8
PID6_KC	32-bit register sets the controller gain for PID6 loop from 0.01 to 999.99 (where 100 counts = gain of 1.00, so PID6_KC = Gain(Kc) x 100).	U_32	759	RAM/EEPROM 8
PID6_MAN_OUT 134	32-bit floating point register sets PID 6 output value when operating in manual output mode. (See PID Auto/Manual Mode 134)	FP_32	1217 134	RAM 8
PID6_ALPHA_SP 135	16-bit register which sets the time constant for filtering of setpoint 6. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4591	RAM/EEPROM 8
PID6_B_SPTW 136	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 6. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8578	RAM/EEPROM 8
PID6_C_SPTW 137	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 6. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8590	RAM/EEPROM 8
PID6_N_DER	8-bit register which sets the scale factor for the derivative of setpoint 6. Range is from 1-255 counts. Factory default value is 1.	U_8	8602	RAM/EEPROM 8
PID6_INC_DEC 134	16-bit signed register controls the PID increment/decrement rate when PID 6 is operating in manual mode. (range -32768 to +32767).	S_16	4531	RAM 8
PID6_SAT_HIGH 138	16-bit signed register that controls the high saturation output level for PID6_OUTPUT. (range -10,000 to +10,000. Note: must be > PID6_SAT_LOW).	S_16	4543	RAM/EEPROM 8
PID6_SAT_LOW 138	16-bit signed register that controls the high saturation output level for PID6_OUTPUT. (range -10,000 to +10,000. Note: must be < PID6_SAT_HIGH).	S_16	4555	RAM/EEPROM 8

See also
SP6 Setup 177

PID 122

2.22.7 PID Auto/Manual/Off Mode

Each setpoint that is configured to operate in PID mode will start up in PID auto mode at power on. Each PID loop can be switched to operate in manual mode or PID off mode by setting the appropriate |SPx_REMOTE bit and |SPx bits in the ALARM_STATUS register.

The following table shows the different options available.

|SPx_REMOTE bit	|SPx bit	PID function
Off (0)	Don't care	Auto Mode - normal mode of operation where PID output is under control of the PID algorithm. (Default mode at power on)
On(1)	On(1)	Manual Output Mode - manual mode of operation where PID output is controlled by manually writing to PIDx_MAN_OUT register.
On(1)	Off (0)	Off Mode - PID algorithm and output are turned off, PID registers (PIDx_ERRD_OLD, PIDx_INTEGRAL_TERM, PIDx_OUTPUT) are reset to zero.

(for more information on setting to remote bits see Alarm Status Write 211).

PIDx_MAN_OUT Registers
When running in manual output mode, each PID loop has an associated PIDx_MAN_OUT register which sets the current PID output value. Each PID loop has bumpless transfer from between auto and manual mode.

INC_DEC Registers
When running in manual mode, each PID loop has an associated INC_DEC register which controls the rate at which the PIDx_MAN_OUT (PID output) register is changed by. The following table shows the various manual mode registers available.

Name	Description	Symbol Type	Register Number	Memory Type
PID1_MAN_OUT	32-bit floating point register sets PID 1 output value when operating in manual output mode.	FP_32	1207	RAM 8
PID2_MAN_OUT	32-bit floating point register sets PID 2 output value when operating in manual output mode.	FP_32	1209	RAM 8
PID3_MAN_OUT	32-bit floating point register sets PID 3 output value when operating in manual output mode.	FP_32	1211	RAM 8
PID4_MAN_OUT	32-bit floating point register sets PID 4 output value when operating in manual output mode.	FP_32	1213	RAM 8
PID5_MAN_OUT	32-bit floating point register sets PID 5 output value when operating in manual output mode.	FP_32	1215	RAM 8
PID6_MAN_OUT	32-bit floating point register sets PID 6 output value when operating in manual output mode.	FP_32	1217	RAM 8
PID1_INC_DEC	16-bit signed register controls the PID increment/decrement rate when PID 1 is operating in manual mode. (range -32768 to +32767).	S_16	4526	RAM 8
PID2_INC_DEC	16-bit signed register controls the PID increment/decrement rate when PID 1 is operating in manual mode. (range -32768 to +32767).	S_16	4527	RAM 8
PID3_INC_DEC	16-bit signed register controls the PID increment/decrement rate when PID 1 is operating in manual mode. (range -32768 to +32767).	S_16	4528	RAM 8
PID4_INC_DEC	16-bit signed register controls the PID increment/decrement rate when PID 1 is operating in manual mode. (range -32768 to +32767).	S_16	4529	RAM 8
PID5_INC_DEC	16-bit signed register controls the PID increment/decrement rate when PID 1 is operating in manual mode. (range -32768 to +32767).	S_16	4530	RAM 8
PID6_INC_DEC	16-bit signed register controls the PID increment/decrement rate when PID 1 is operating in manual mode. (range -32768 to +32767).	S_16	4531	RAM 8

2.22.8 Setpoint Filter

When configured to run in PID mode, each setpoint (or PID loop) includes setpoint filtering to allow setpoint changes to be made on the fly without causing the PID loop to become unstable. Each PID loop has an individual setpoint filter constant called PIDx_ALPHA_SP which sets the time constant of the setpoint filter. Although these time constants are not accessible directly via the menu system on the front panel of the remote display, they can be changed from the macro or via the serial port by writing to specific registers.

Each register shown in the table below is an 16 bit unsigned register which holds a value from 0 to 65535. The actual filter constant is calculated by dividing the register value by 65535 to obtain a fraction.

For example, if a filter constant register holds a value of 39977 then the actual filter constant is;

Filter constant = register value/65535
=> = 39977/65535

=> = 0.61

The larger the value of PIDx_ALPHA_SP, the longer the setpoint filter time constant. With PIDx_ALPHA_SP=39977 (0.61), the filter time constant is a couple of seconds.

The table below shows the various registers used for this purpose.

Name	Description	Symbol Type	Register Number	Memory Type
PID1_ALPHA_SP	16-bit register which sets the time constant for filtering of setpoint 1. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant = value/65535). Factory default value is 0.61 (39977 counts).	U_16	4586	RAM/EEPROM
PID2_ALPHA_SP	16-bit register which sets the time constant for filtering of setpoint 2. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4587	RAM/EEPROM
PID3_ALPHA_SP	16-bit register which sets the time constant for filtering of setpoint 3. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4588	RAM/EEPROM
PID4_ALPHA_SP	16-bit register which sets the time constant for filtering of setpoint 4. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4589	RAM/EEPROM
PID5_ALPHA_SP	16-bit register which sets the time constant for filtering of setpoint 5. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4590	RAM/EEPROM
PID6_ALPHA_SP	16-bit register which sets the time constant for filtering of setpoint 6. Range is from 0-65535 counts where a count of 65535 gives a time constant of 1.0 (i.e. time constant =value/65535). Factory default value is 0.61 (39977 counts).	U_16	4591	RAM/EEPROM

2.22.9 Setpoint Weighting Factor - P

When configured to run in PID mode, each setpoint (or PID loop) includes a setpoint weighting factor which is used in the calculation of the proportional error as per the following formula.

Proportional_Error =Setpoint_Weighting_Factor x Filtered_Setpoint_Value - Process_Variable

Each PID loop has an individual setpoint weighting factor register called PIDx_B_SPTW. Although these constants are not accessible directly via the menu system on the front panel of the remote display, they can be changed from the macro or via the serial port by writing to specific registers. The table below shows the various registers used for this purpose.

Each register shown below is an 8 bit unsigned register which holds a value from 0 to 255. The actual filter constant is calculated by dividing the register value by 255 to obtain a fraction.

For example, if a weighting factor register holds a value of 255 then the actual weighting factor is;

Weighting Factor = register value/255

=> = 255/255
=> = 1.0

Name	Description	Symbol Type	Register Number	Memory Type
PID1_B_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 1. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8573	RAM/EEPROM
PID2_B_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 2. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8574	RAM/EEPROM
PID3_B_SPTWP	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 3. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8575	RAM/EEPROM
PID4_B_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 4. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8576	RAM/EEPROM
PID5_B_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 5. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8577	RAM/EEPROM
PID6_B_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the proportional error in setpoint 6. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 1.0 (255 counts).	U_8	8578	RAM/EEPROM

2.22.10 Setpoint Weighting Factor - D

When configured to run in PID mode, each setpoint (or PID loop) includes a setpoint weighting factor which is used in the calculation of the derivative error as per the following formula.

Derivative_Error =Setpoint_Weighting_Factor x Filtered_Setpoint_Value - Process_Variable

Each PID loop has an individual setpoint weighting factor register called PIDx_C_SPTW. Although these constants are not accessible directly via the menu system on the front panel of the remote display, they can be changed from the macro or via the serial port by writing to specific registers. The table below shows the various registers used for this purpose.

Each register shown below is an 8 bit unsigned register which holds a value from 0 to 255. The actual filter constant is calculated by dividing the register value by 255 to obtain a fraction.

For example, if a weighting factor register holds a value of 255 then the actual weighting factor is;

Weighting Factor = register value/255
=> = 255/255
=> = 1.0

Name	Description	Symbol Type	Register Number	Memory Type
PID1_C_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 1. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8585	RAM/EEPROM 8
PID2_C_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 2. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8586	RAM/EEPROM 8
PID3_C_SPTWP	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 3. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8587	RAM/EEPROM 8
PID4_C_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 4. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8588	RAM/EEPROM 8
PID5_C_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 5. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8589	RAM/EEPROM 8
PID6_C_SPTW	8-bit register which sets the setpoint weighting factor used in the calculation of the derivative error in setpoint 6. Range is from 0-255 counts where a count of 255 gives a factor of 1.0 (i.e. factor=value/255). Factory default value is 0.	U_8	8590	RAM/EEPROM 8

2.22.11 PID Output Saturation

When configured to run in PID mode, the output value of each PID loop is stored in a register called PIDx_OUTPUT. Maximum allowable range for this output value is from -10,000 counts (0%) to +10,000 counts (100%). However, in some control applications it may be desirable to limit this range. For this reason each PID loop has 2 saturation registers called PIDx_SAT_LOW and PIDx_SAT_HIGH. These registers are 16 bit signed registers with an allowable range from -10,000 counts to +10,000 counts. The PID output value will be limited to the range of PIDx_SAT_LOW to PIDx_SAT_HIGH. The factory default values for these registers are -9,950 and +9,950.

NOTE: The user must ensure that PIDx_SAT_LOW is always less than PIDx_SAT_HIGH. Failure to do this will result in abnormal PID output values.

Name	Description	Symbol Type	Register Number	Memory Type
PID1_SAT_HIGH	16-bit signed register that controls the high saturation output level for PID1_OUTPUT. (range -10,000 to +10,000. Note: must be > PID1_SAT_LOW).	S_16	4538	RAM/EEPROM
PID2_SAT_HIGH	16-bit signed register that controls the high saturation output level for PID2_OUTPUT. (range -10,000 to +10,000. Note: must be > PID2_SAT_LOW).	S_16	4539	RAM/EEPROM
PID3_SAT_HIGH	16-bit signed register that controls the high saturation output level for PID3_OUTPUT. (range -10,000 to +10,000. Note: must be > PID3_SAT_LOW).	S_16	4540	RAM/EEPROM
PID4_SAT_HIGH	16-bit signed register that controls the high saturation output level for PID4_OUTPUT. (range -10,000 to +10,000. Note: must be > PID4_SAT_LOW).	S_16	4541	RAM/EEPROM
PID5_SAT_HIGH	16-bit signed register that controls the high saturation output level for PID5_OUTPUT. (range -10,000 to +10,000. Note: must be > PID5_SAT_LOW).	S_16	4542	RAM/EEPROM
PID6_SAT_HIGH	16-bit signed register that controls the high saturation output level for PID6_OUTPUT. (range -10,000 to +10,000. Note: must be > PID6_SAT_LOW).	S_16	4543	RAM/EEPROM
PID1_SAT_LOW	16-bit signed register that controls the low saturation output level for PID1_OUTPUT. (range -10,000 to +10,000. Note: must be < PID1_SAT_HIGH).	S_16	4550	RAM/EEPROM
PID2_SAT_LOW	16-bit signed register that controls the low saturation output level for PID2_OUTPUT. (range -10,000 to +10,000. Note: must be < PID2_SAT_HIGH).	S_16	4551	RAM/EEPROM
PID3_SAT_LOW	16-bit signed register that controls the low saturation output level for PID3_OUTPUT. (range -10,000 to +10,000. Note: must be < PID3_SAT_HIGH).	S_16	4552	RAM/EEPROM
PID4_SAT_LOW	16-bit signed register that controls the low saturation output level for PID4_OUTPUT. (range -10,000 to +10,000. Note: must be < PID4_SAT_HIGH).	S_16	4553	RAM/EEPROM
PID5_SAT_LOW	16-bit signed register that controls the low saturation output level for PID5_OUTPUT. (range -10,000 to +10,000. Note: must be < PID5_SAT_HIGH).	S_16	4554	RAM/EEPROM
PID6_SAT_LOW	16-bit signed register that controls the low saturation output level for PID6_OUTPUT. (range -10,000 to +10,000. Note: must be < PID6_SAT_HIGH).	S_16	4555	RAM/EEPROM

2.22.12 PID Control Registers - Register 1057 to 1079

Registers 1057 to 1079 are 32-bit floating point registers used by the PID control algorithms. Registers 1057 to 1067 show the Error Sum term for PID outputs 1 to 6. Registers 1069 to 1079 show the Integral term for PID outputs 1 to 6.

2.23 Digital Pins

Register 4108 Digital Input/Output Register
Register 4108 is a 16-bit register that provides read and write access to the digital pins on the top of the controller and on selected input modules. These pins are multi purpose pins and are used in conjunction with other features and functions of the controller. Some pins are present on all models while others are only available on specific input modules.

The HOLD, LOCK and CAPTURE pins are present on all controllers but share functions with Code 8, Code 9, and SPC_1 to SPC_6 (setpoints 1 to 6). You should check the ICC402 programming sheet before writing to these inputs.

The remaining pins are normally used by the controller to interface with the input module but can be used as extra digital I/O with special input modules. If these are used you must ensure that the controller configuration is set to a mode that will not interfere with the pins being used.

Name	Description	Symbol Type	Register Number	Memory Type
DIGITAL_IO	16 bit register contains flags for the digital inputs on the top of the controller.	U_16	4108	RAM 8

The function of each bit is shown as follows:

Bit	Name	Description	Function
0	HOLD_PIN	Flag shows the current status of the hold pin on the top of the controller.	ON = hold pin connected to common
1	LOCK_PIN	Flag shows the current status of the lock pin on the top of the controller.	ON = lock pin connected to common
2	CAPTURE_PIN	Flag shows the current status of the capture pin on the top of the controller.	ON = capture pin connected to common
3	D1	Read only flag shows the current status of the D1 pin on selected input modules.	ON = D1 pin connected to common
4	D2	Read only flag shows the current status of the D2 pin on selected input modules.	ON = D2 pin connected to common
5	D3	Read only flag shows the current status of the D3 on selected input modules.	ON = D3 pin connected to common
6	COUNT2_INPUT	Read only flag shows the current status of the CH2 count input pin on selected input modules.	ON = COUNT2_INPUT pin connected to common
7	COUNT2_DIRECTION	Read only flag shows the current status of the CH2 count direction pin on selected input modules.	ON = COUNT2_DIRECTION pin connected to common
8	MUX0	Read only flag shows the current status of the MUX0 pin on selected input modules.	ON = MUX0 pin connected to common
9	MUX1	Read only flag shows the current status of the MUX1 pin on selected input modules.	ON = MUX1 pin connected to common
10 to 16	Not Used	-	-

NOTE: Because of the port structure of the microprocessor used in the controller, a **0** written to a

digital pin configures it as an input with a weak pull-up. A **1** written to a pin configures the pin as an output only, which is driven low. Care must be taken when writing to register 4108 (either as a 16-bit word or as bit writes via the macro) to ensure that a **0** is written to each bit that is required to be an input. This can be a trap if 4108 is read, modified, and written out again. If an input pin is active when it is read (i.e. a **1**) and is output as a **1**, the pin is no longer an input pin but now becomes an output only pin. The solution to this problem is to always mask out those pins which are inputs before writing a value out to 4108.

When register 4108 is read the status of all pins is sampled directly from the input pin. If register 4108 is written to, the values specified are also written directly to the corresponding pins.

Also note that the state of each bit in register 4108 is not the same as the actual voltage on the connector pin of the controller. This is because the inputs are high in their inactive state and are pulled low when active.

2.24 Result

Result registers contain the result information of cross channel math functions, including the setup information for auto zero maintenance.

Name	Description	Symbol Type	Register Number	Memory Type
RESULT	Non volatile 32-bit register for result data.	S_32	7	RAM/FLASH 8
RESULT_PROCESSED	32-bit register holds the processed data for the result channel.	S_32	257	RAM/FLASH 8
RESULT_RAW	32-bit register holds the raw result data, prior to scaling and post processing.	S_32	281	RAM 8
RESULT_SCALED	32-bit register holds the scaled result data, before any post processing.	S_32	305	RAM 8

The above registers are normally updated by the operating system of the controller after a new input sample is processed. If disabled, it is also possible to modify the contents of the register by writing to it from the setpoint reset logic, the Macro, or via the serial port. A write to these registers in any other operational mode may result in the newly written value being over written by the operating system in the controller.

A write to the RESULT_PROCESSED, RESULT_RAW, or RESULT_SCALED registers modifies only that register. A write to the RESULT register automatically updates RESULT_PROCESSED, RESULT_RAW, and RESULT_SCALED. The controller takes into account scale and offset and calculates the RESULT_RAW value.

NOTE: 32-point linearization is not supported by the above feature.

A read of RESULT register is identical to a read of RESULT_PROCESSED register.

See also
Result Setup Registers 141

2.24.1 Result Setup Registers

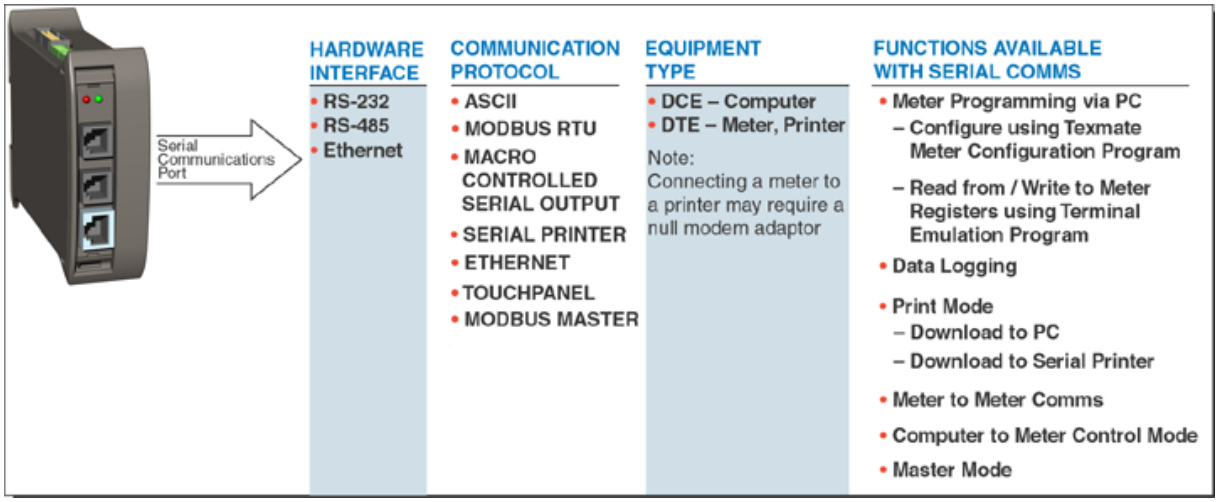
Name	Description	Symbol Type	Register Number	Memory Type
AUTO_ZERO_RESULT	16-bit register contains the auto zero correction offset for the result.	S_16	4133	RAM/FLASH 8

2.25 Serial Port

Serial port registers contain the serial communications settings required for ports 1 to 3, as well as Modbus master information.

The following functions are available via the serial ports:

- controller programming via PC. Configure using Texmate NZ Meter Configuration Utility program, or Read from / Write to controller registers using terminal emulation program.
- Data logging.
- Print mode. Download to PC, download to serial printer.
  NOTE: Connecting a controller to a serial printer may require a null modem adjustor.
- controller to controller communications.
- Computer to controller control mode.
- Master mode.



Serial Port Modes - Registers 8215 - 8217
Registers 8215 to 8217 are 8 bit registers which control the functionality of serial ports 1 to 3 respectively. The following table shows the register value for currently available serial port protocols.

Value	Mode
0	ASCII 141
1	Modbus RTU slave
2	Macro
3	Printer
4	Ethernet
5	Devicenet
6	LCD touch panel
7	Modbus RTU Master 147

Note: Some of the above protocols require special hardware support and are only available for some serial ports. Please check with modes for Port 1 146, Port 2 152, and Port 3 154.

Register 8219 – Serial Port In Use
Register 8219 is an 8-bit register that reports which serial port is currently in use. Because the ICC402 Series controllers have multiple serial ports it may be necessary for an external device to know which one it is currently using. For example, a read of 8219 via serial port 1 results in a number

1 being returned.

Name	Description	Symbol Type	Register Number
SERIAL_PORT_NO	8-bit read only register shows which serial port is being accessed.	U_8_R	8219

See also
ASCII Mode Format 14

ASCII Text Registers 19

Modbus Master 147

Port 1 146

Port 2 152

Port 3 154

2.25.1 Serial Port Settings - Registers 8207 - 8209

Registers 8207 to 8209 are 8-bit registers used to store the serial port settings for serial ports 1 to 3. Bits 0 to 2 are used to hold the baud rate information. Bit 3 is used to select between 7 or 8 data bits. Bits 4 and 5 are used to select the parity type. Bits 6 and 7 allow different transmit delay times to be selected. The various options available are shown as follows:

Bits 7, 6 **Transmit delay**
00 = 2 milliseconds
01 = 20 milliseconds
10 = 50 milliseconds
11 = 100 milliseconds

Bits 5, 4 **Parity**
00 = no parity
01 = odd parity
10 = even parity

Bit 3 **Data bits**
0 = 8 data bits
1 = 7 data bits

Bits 2, 1, 0 **Baud rate**
000 = 1200 baud
001 = 2400 baud
010 = 4800 baud
011 = 9600 baud
100 = 19200 baud
101 = 38400 baud
110 = 57600 baud
111 = 115200 baud

NOTE: The baud rates shown above are not available on all ports. Please check bauds rates for Port 1 146, Port 2 152 and Port 3 154

NOTE: If these registers are modified via the serial port, the controller response (and any subsequent

communications) is issued at the new modified baud rate/parity settings and may result in a communications error at the master device.

2.25.2 Serial Address - Registers 8211 - 8213

These are 8-bit registers that set the serial address for serial ports 1 - 3 respectively. The controller address can be set from 1 to 255. The controller address should not be set to 0 as this address is reserved (all controllers respond to a request at address zero).

2.25.3 Start / End of String Characters - Registers 8226 to 8228

The protocol for master mode is not set in software, it is user defined in the macro. Registers 8226 to 8229 in conjunction with registers 8230 to 8233 are used for this purpose. The following table shows which register pairs are used together.

Register Pair	Serial Port	Memory Type
8226/8230	Port 1	RAM/EEPROM 8
8227/8231	Port 2	RAM/EEPROM 8
8228/8232	Port 3	RAM/EEPROM 8
8229/8233	Reserved for future development of Port 4	RAM/EEPROM 8

Note: The following explanation uses the registers for port 1 but the same logic applies to all ports.

Register 8226 is an 8-bit register which is used in conjunction with register 8230. Register 8226 is called STRING_LENGTH1 and register 8230 is called STRING_CHARACTER1 144. The function of the serial port in master mode will vary depending on the values contained in these two registers. There are 3 possible different modes of operation as described below.

1) String Length Zero
If register 8230 is zero, (i.e. string length = zero), then register 8226 functions as an end of string character (EOS) for master mode serial com's, allowing the user to define their own terminating character for the received string. In this mode the controller will keep receiving data until it finds the EOS character defined in register 8226. At this point it will stop receiving any further data and set the RECEIVE_READY1 flag to indicate that a new string is ready for interrogation. Note: the received string cannot be longer than 255 bytes. If the EOS character is not received within 255 bytes, the receive buffer is flushed and reception starts all over again, with all previously received data being lost.

2) String Length > Zero, String Character > Zero
If register 8230 contains some value other than zero and the string character is set to some value other than zero, then register (8226) functions as a Start Of String (SOS) character. In this mode the serial port searches incoming data for the SOS character (defined in register 8226) and when it receives this character it loads this character into its buffer and then continues to input data until it has received the number of characters defined by register 8230. At this point it will stop receiving any further data and set the RECEIVE_READY1 flag to indicate that a new string is ready for interrogation.

3) String Length > Zero, String Character Zero
If register 8230 contains some value other than zero, and the string character is set to zero (ASCII null), then in this mode the serial port will start receiving data until it has received the number of bytes specified by the string length (8230). At this point it will stop receiving any further data and set the RECEIVE_READY1 flag to indicate that a new string is ready for interrogation.

Note: Mode 3 above was only introduced in ICC402 firmware version 4.04.12. In older versions of firmware a string character of zero functions in mode 2.

2.25.4 String Length - Registers 8230 to 8232

Registers 8230 to 8233 are 8-bit registers that set the string length for the receive respective serial port in master mode. If these registers are greater than zero, the respective serial port searches for the start character defined in the appropriate start of string register (see 8226 to 8229 144), and then it inputs the specified number of bytes.

2.25.5 Serial Receive Count- Registers 8454 - 8456

Registers 8454 to 8456 are 8-bit registers that show the received message length for serial ports 1 - 3 respectively (i.e. how many bytes have been received by the serial port in a message). Although these registers can be read in all serial modes, their main purpose is for use in master mode under macro control of the serial ports.

2.25.6 Serial Transmit Count - Registers 8465 - 8467

Registers 8465 to 8467 are 8-bit unsigned registers that relate to the number of bytes to be transmitted by serial ports 1 - 3 respectively. They are only intended for use with the serial port set in Macro master mode.

Writing To The Transmit Count Register
Normally the 'PRINT' command is used in a macro to send an ASCII string out via one of the serial ports. However sometimes it may be necessary to send non ASCII strings or complicated strings which need extra processing for checksums etc. In this case the macro would write directly to the serial buffer and load the outgoing string byte by byte. When this process has been completed, the string is transmitted by writing the length of the string to the appropriate TRANSMIT_COUNT register.

NOTE: The serial port must be operating in Macro Master mode for correct operation.

Reading The Transmit Count Register
Reading the transmit count register while a string is being transmitted will show the progress of the transmit process by pointing to the next byte to be transmitted.

2.25.7 Port 1

Name	Description	Symbol Type	Register Number	Memory Type
BAUDRATE1 143	8-bit register sets the serial port 1 baud rate (0 = 1200, 1 = 2400, 2 = 4800, 3 = 9600, 4 = 19.2 k, 5 = 38.4 k, 6 = 57.6 k, 7 = 115.2 k).	U_8	8207 143	RAM/EEPROM 8
SERIAL_BUFFER1 147	Start of serial transmit/receive buffer for port 1 (255 bytes long).	U_8	12289 147	RAM 8
RECEIVE_COUNT1 145	8-bit register which shows how many characters have been received by the serial port 1.	U_8	8454 145	RAM 8
TRANSMIT_COUNT1 145	8-bit register which sets how many characters are to transmitted by the serial port 1.	U_8	8465 145	RAM 8
RECEIVE_FLAGS1	8-bit register. Serial receive flags. Used in master mode.	U_8	8234	RAM 8
RECEIVE_READY1	This flag shows that a new message string has been received on port 1 in master mode.	B_0	8234	RAM 8
RECEIVE_RESULT1	32-bit register holds the 1st numeric value received in a string via serial port 1.	S_32	349	RAM 8
SERIAL_ADDRESS1 144	8-bit register holds the serial address of the controller.	U_8	8211 144	RAM/EEPROM 8
SERIAL_MODE1	8-bit register sets the serial mode for port 1.	U_8	8215	RAM/EEPROM 8
SERIAL_POINTER1	8-bit pointer used for string compare commands with serial port 1.	U_8	8458	RAM 8
STRING_CHARACTER1 144	8-bit register. Sets ASCII character for the start or end of serial receive string in master mode for port 1.	U_8	8226 144	RAM/EEPROM 8
STRING_LENGTH1	8-bit register. Sets string length of serial receive string in master mode for port 1.	U_8	8230	RAM/EEPROM 8
RECEIVE_IDLE_TIME1	8-bit register. Shows seconds of inactivity on serial port 1. (max. count = 255 seconds	U_8	8505	RAM 8

Baud Rates - Port 1

The baud rates for port 1 are controlled by bits 0 to 2 of register 8207 and the available options are shown below.

Bits 2, 1, 0	Baud rate
	000 = 1200 baud
	001 = 2400 baud
	010 = 4800 baud
	011 = 9600 baud
	100 = 19200 baud
	101 = 38400 baud
	110 = 57600 baud
	111 = 115200 baud.

Serial Port Modes - Registers 8215

Registers 8215 is an 8 bit register which controls the functionality of serial port 1. The following table shows the register value for currently available serial port protocols.

Value	Mode
0	ASCII 14

1	Modbus RTU slave
2	Macro
3	Printer
4	Ethernet
5	Not available
6	LCD touch panel
7	Modbus RTU Master 147

2.25.7.1 Serial Buffer Port 1 - Registers 12289 to 12544

Registers 12289 to 12544 are all 8-bit unsigned registers that are used as a buffer for serial port 1 received and transmitted data. They are used in all serial port modes, but their intended use is in master mode under macro control. By accessing these registers individually, a message string can be built up or interrogated, byte by byte.

NOTE: Although registers 12289 to 12544 can be written to, it is not recommended unless you have a thorough knowledge of how the serial port operates. Writing the wrong value to these registers could cause the serial port to lock up.

See Also
Serial Transmit Count - Registers 8465 - 8467 145

Serial Receive Count- Registers 8454 - 8456 145

2.25.8 ModBus Master

The Modbus master macro is a special macro area which can be used to configure the ICC402 controller as a Modbus master which is capable of reading and writing to other Modbus slave devices. Several special Modbus macro commands are included which can only be used in the MODBUS_MASTER_MACRO. In addition, all other macro commands can be used as well. (Note: Although the MODBUS_MASTER_MACRO is primarily designed for the purpose of providing a Modbus master function, it can be used for other purposes and does not strictly require the inclusion of any Modbus macro commands).

When implementing a Modbus master function the desired serial port(s) must also be set to Modbus master mode. See serial port modes 142.

Modbus Master Registers
The Modbus Master mode can only be used under the control of the Modbus Master macro. The registers shown below are only intended for this use.

Name	Description	Symbol Type	Register Number
CRC_ERROR	Read only flag - response from slave received with CRC checksum error.	B_5	8464
DATA_ERROR	Read only flag - Modbus attempted to read/write incorrect data type to slave.	B_6	8464
MESSAGE_COMPLETE	Read only flag - previous message transaction is completed correctly.	B_7	8464
MESSAGE_TIMEOUT	Read only flag - no response received from slave.	B_4	8464
MODBUS_MASTER_FLAGS	8-bit read only register which contains status flags for the Modbus master macro.	U_8_R	8464
POLL_TIME	8-bit register which sets the polling time for the Modbus master macro (1 count = 0.01 seconds).	U_8	8462
RESPONSE_TIME	8-bit register which sets the message response timeout for the Modbus master macro (1 count = 0.1 seconds).	U_8	8463

Simple Modbus Master Example

The following example shows a typical Modbus master implementation in the MODBUS_MASTER_MACRO. This example shows the simplest form of the command which allows one register to be read or written in each Modbus command. (See the section below on ICC402 enhancements)

```
Modbus_Master_Macro:
  &POLL_TIME =10                              //100mS (1 COUNT = 10mS)
  &RESPONSE_TIME=5                            //0.5S (1 count = 0.1S)
  MODBUS_READ 1 (1,40004,&CH3,MB_LONG)          //
  GOSUB CHECK_MESSAGE
  MODBUS_WRITE 1 (1,&CH4,40005,MB_LONG)
  GOSUB CHECK_MESSAGE
  MODBUS_READ 1 (1,30005,&CH2,MB_LONG)
  GOSUB CHECK_MESSAGE
RETURN

CHECK_MESSAGE:
  IF (&MODBUS_MASTER_FLAGS AND 0x7F) != 0 THEN
    WRITE "   ___ERROR - "+&MODBUS_MASTER_FLAGS+"      "
  ENDIF
RETURN
```

As shown in the above example, the MODBUS_READ and MODBUS_WRITE commands have a similar format. The number following the command (i.e. outside the brackets) specifies the serial port number to be used for the Modbus master mode. Then inside the brackets the format is as follows;

([slave device address],[Modbus source register],[Modbus destination register],[register type])

[slave device address] = controller address of the Modbus slave can be a number from 0 to 247.

[Modbus source register] = in a read command this can be a number from 30001 to 49999. For a write command you can specify the register name and the compiler while calculate the register number.

[Modbus destination register] = in a write command this can be a number from 30001 to 49999. For a read command you can specify the register name and the compiler while calculate the register number.

[register type] = This specifies the size and type of the register being accessed with the following options being available.

MB_BIT	Not supported at present
MB_BYTE	8 bit register
MB_SHORT	16 bit register
MB_24	24 bit register
MB_24_SWAPPED	24 bit register with MSW and LSW swapped
MB_LONG	32 bit register
MB_LONG_SWAPPED	32 bit register with MSW and LSW swapped
MB_FLOAT	32 bit single precision floating point register
MB_FLOAT_SWAPPED	32 bit single precision floating point register with MSW and LSW swapped
MB_STRING	Text string register (only supported with enhanced command format - see note below)

NOTE:
If the register types for the source and destination registers in the Modbus command do not match, the ICC402 will attempt to correct this if possible. In the case of a float/fixed point mismatch, the ICC402 will attempt to type cast the value into the different format, but this will not always be possible in the case of large floating point numbers so the user should be careful if using this feature to ensure that range problems do not occur.

Currently Supported Functions
At present input registers (30000 range) and holding registers (40000 range) are supported in the Modbus master mode.

Modbus Master Flags
Register 8464 (MODBUS_MASTER_FLAGS) can be used in Modbus Master mode to determine if a transmission error occurred in the previous communication. The following errors are possible.

Bits 0 - 3 (Standard Modbus Exception Error Codes)
1 = Illegal function call (function call not supported by slave)
2 = Illegal data address (the data address specified in the command is not available in the slave)
3 = Illegal data value (a data value specified in the command is not in the acceptable range)
4 = Slave device failure
5 = Acknowledge
6 = Slave device busy
7 = Negative acknowledge
8 = Memory parity errors

Bit 4 = Message timeout
Bit 5 = CRC receive error
Bit 6 = data type error
Bit 7 = Reception complete and ready for new command

Poll Time
Register 8462 is an 8 bit register that defines the rate at which the Modbus master macro is executed. Each count of the POLL_TIME register represents a time interval of 10mS so a value of 100 would result in the Modbus master macro being executed once a second. If a value of 0 is written to POLL_TIME the Modbus master macro will execute as fast as the operating system will allow.

Register 8462 defaults to a value of 10 (i.e. 0.1S) each time the controller is powered up and any

writes to this register are stored in volatile memory which is lost at power down. For this reason the register &POLL_TIME should be written in either the RESET_MACRO or the MODBUS_MASTER_MACRO.

Response Time
Register 8463 is an 8 bit register that defines the maximum time the Modbus master will wait for a slave to respond. Each count of the RESPONSE_TIME register represents a time interval of 100mS so a value of 10 would result in the Modbus master waiting for up to a second for a slave response. If the slave device fails to respond within the set time, bit 4 of the MODBUS_MASTER_FLAGS register is set and the Modbus master macro continues execution at the next line of macro code.

Register 8463 defaults to a value of 10 (i.e. 1S) each time the controller is powered up and any writes to this register are stored in volatile memory which is lost at power down. For this reason the register &RESPONSE_TIME should be written in either the RESET_MACRO or the MODBUS_MASTER_MACRO.

ICC402 Modbus Master Enhancements
The Modbus master example shown above is the simplest from of Modbus master command. The ICC402 has been enhanced to allow block reads and writes of registers and also to allow variable expressions in the Modbus master command. This allows the implemenetation of more efficient and more compact Modbus master macros, saving processing time and macro memory space. The enhanced form of the Modbus master commands are shown below.

MODBUS_READ [serial port number] ([slave address],[remote source register],[local destination register],[register type],[number of registers to be read])
MODBUS_WRITE [serial port number] ([slave address],[local source register],[remote destination register],[register type],[number of registers to be written])

In the enhanced command;
[slave device address] = controller address of the Modbus slave. Can either be a number from 0 to 247 or can also be a variable register which holds a number from 1 to 247.

[remote source register] = this can be a constant (i.e. number from 1 to 19999) or it can be a variable register which holds a number from 1 to 19999 or it can be an expression.

[remote destination register] = this can be a constant (i.e. number from 1 to 19999) or it can be a variable register which holds a number from 1 to 19999 or it can be an expression.

Note: With the enhanced command format you no longer need to specificy the address in the 30000 or 40000 format for remote registers. You should now just specify the register number only without the 30000 or 40000 offset. The [register type] feild now specifies whether the register is an input or holding register.

[local source register] = this must be specified as a register, with or without array index.

[local destination register] = this must be specified as a register, with or without array index.

[register type] = this is similar to the simple example above but it must also have either +MB_HOLD or +MB_INPUT added to the end to specify whether the register is a holding register or an input register. So a typical example would be MB_LONG+MB_HOLD. The enhanced command also includes the new register type of MB_STRING which allows the reading and writing to text string registers via the Modbus master mode. See note below on Text Strings.

[number of registers to be read] = This field is mandatory for the enhanced Modbus master command and must be a constant greater than zero. The following range is allowed for different commands with

different register types. (See note below regarding use of this field for with the register type MB_STRING)

Read Commands
1 to 250 for MB_TEXT
1 to 125 for MB_BYTE , MB_SHORT
1 to 62 for MB_24, MB_24_SWAPPED, MB_LONG ,MB_LONG_SWAPPED, MB_FLOAT, MB_FLOAT_SWAPPED

Write Commands
1 to 246 for MB_TEXT
1 to 123 for MB_BYTE , MB_SHORT
1 to 61 for MB_24, MB_24_SWAPPED, MB_LONG ,MB_LONG_SWAPPED, MB_FLOAT, MB_FLOAT_SWAPPED

Block Reads/Writes In Modbus Master Comands
The first Modbus master example shown above only allows 1 register to be read or written to in each Modbus master command line. The ICC402 controller now allows blocks of registers to be read or written to using a single Modbus read or write command. The following example shows the new form.

```
MODBUS_READ 2 (1, 111, &AUX1, MB_LONG+MB_HOLD, 6)
MODBUS_WRITE 1 (1, &SETPOINT1, 111, MB_LONG+MB_HOLD, 6)
```

Using Variables In Modbus Master Commands
With the simplest form of the Modbus master command the parameters are specified directly in the command (i.e. with fixed values). The ICC402 controller has some new enhancements which allow variables and expressions to be used in the Modbus master command. This allows for next loops to be used with Modbus master commands and greatly reduces the amount of macro code needed. The following example shows how variables and expressions can be used with the new Modbus master command.

```
#src = addr(&SETPOINT1)
for #address = 1 to 6
  MODBUS_READ 2 (#address, #src, &AUX1[#address-1], MB_LONG+MB_HOLD, 1)
  #src = #src + 2
next #address
```

The example above could be used to read data from 6 other Texmate controllers with addresses 1 - 6. It would read the value of setpoint1 in controller 1 and store it in &AUX1. Then it would read the value of setpoint2 in controller 2 and store it in &AUX2 and so on.

Text Strings In Modbus Master Mode
The simple form of the Modbus master command did not allow text strings to be accessed via the Modbus master command but the enhanced command does. To read or write to a text string register the register type should be specified as MB_STRING. In this mode the field [number of registers to be read] is interpreted as [number of characters to be read] and any where from 1 to 250 characters can be read. Only one text string register can be read with each command - reading of successive text registers is supported.

Note: If [number of characters to be read] is set to a value which is greater than the actual size of the text string in the remote register, the excess characters are padded out with ASCII nulls (0x00). Although this condition is acceptable it is not very efficient as extends the length of the string by adding unneccessary bytes. For this reason the [number of characters to be read] should match the size of the remote text register.

2.25.9 Port 2

Name	Description	Symbol Type	Register Number	Memory Type
BAUDRATE2 143	8-bit register sets the serial port 2 baud rate (0-2 not available, 3 = 9600, 4 = 19.2 k, 5 = 38.4 k, 6 = 57.6 k, 7 = 115.2 k).	U_8	8208 143	RAM/EEPROM 8
SERIAL_BUFFER2 153	Start of serial transmit/receive buffer for port 2 (255 bytes long).	U_8	12545 153	RAM 8
RECEIVE_COUNT2 145	8-bit register which shows how many characters have been received by the serial port 2.	U_8	8455 145	RAM 8
TRANSMIT_COUNT2 145	8-bit register which sets how many characters are to transmitted by the serial port 2.	U_8	8466 145	RAM 8
RECEIVE_FLAGS2	8-bit register. Serial receive flags. Used in master mode.	U_8	8235	RAM 8
RECEIVE_READY2	This flag shows that a new message string has been received on port 2 in master mode.	B_0	8235	RAM 8
RECEIVE_RESULT2	32-bit register holds the 1st numeric value received in a string via serial port 2.	S_32	351	RAM 8
SERIAL_ADDRESS2 144	8-bit register holds the serial address of the controller.	U_8	8212 144	RAM/EEPROM 8
SERIAL_MODE2	8-bit register sets the serial mode for port 2.	U_8	8216	RAM/EEPROM 8
SERIAL_POINTER2	8-bit pointer used for string compare commands with serial port 2.	U_8	8459	RAM 8
STRING_CHARACTER2 144	8-bit register. Sets ASCII character for the start or end of serial receive string in master mode for port 2.	U_8	8227 144	RAM/EEPROM 8
STRING_LENGTH2	8-bit register. Sets string length of serial receive string in master mode for port 2.	U_8	8231	RAM/EEPROM 8
RECEIVE_IDLE_TIME2	8-bit register. Shows seconds of inactivity on serial port 2. (max. count = 255 seconds	U_8	8506	RAM 8

Baud Rates - Port 2

The baud rates for port 2 are controlled by bits 0 to 2 of register 8208 and the available options are shown below.

Bits 2, 1, 0	Baud rate
	000 = Not available (4800 baud)
	001 = Not available (4800 baud)
	010 = 4800 baud
	011 = 9600 baud
	100 = 19200 baud
	101 = 38400 baud
	110 = 57600 baud
	111 = 115200 baud.

Serial Port Modes - Registers 8216

Registers 8216 is an 8 bit register which controls the functionality of serial port 2. The following table shows the register value for currently available serial port protocols.

Value	Mode
0	ASCII 14
1	Modbus RTU slave

2	Macro
3	Printer
4	Not available
5	Not available
6	LCD touch panel
7	Modbus RTU Master 147

2.25.9.1 Serial Buffer Port 2 - Registers 12545 to 12800

Registers 12545 to 12800 are all 8-bit unsigned registers that are used as a buffer for serial port 2 received and transmitted data. They are used in all serial port modes, but their intended use is in master mode under macro control. By accessing these registers individually, a message string can be built up or interrogated, byte by byte.

NOTE: Although registers 12545 to 12800 can be written to, it is not recommended unless you have a thorough knowledge of how the serial port operates. Writing the wrong value to these registers could cause the serial port to lock up.

See Also

Serial Transmit Count - Registers 8465 - 8467 145

Serial Receive Count- Registers 8454 - 8456 145

2.25.10 Port 3

Name	Description	Symbol Type	Register Number	Memory Type
BAUDRATE3 143	8-bit register sets the serial port 3 baud rate (0 = 1200, 1 = 2400, 2 = 4800, 3 = 9600, 4 = 19.2 k, 5 - 7 not available).	U_8	8209 143	RAM/EEPROM 8
SERIAL_BUFFER3 153	Start of serial transmit/receive buffer for port 3 (255 bytes long).	U_8	12801 153	RAM 8
RECEIVE_COUNT3 145	8-bit register which shows how many characters have been received by the serial port 3.	U_8	8456 145	RAM 8
TRANSMIT_COUNT3 145	8-bit register which sets how many characters are to transmitted by the serial port 3.	U_8	8467 145	RAM 8
RECEIVE_FLAGS3	8-bit register. Serial receive flags. Used in master mode.	U_8	8236	RAM 8
RECEIVE_READY3	This flag shows that a new message string has been received on port 3 in master mode.	B_0	8236	RAM 8
RECEIVE_RESULT3	32-bit register holds the 1st numeric value received in a string via serial port 3.	S_32	353	RAM 8
SERIAL_ADDRESS3 144	8-bit register holds the serial address of the controller.	U_8	8213 144	RAM/EEPROM 8
SERIAL_MODE3	8-bit register sets the serial mode for port 3.	U_8	8217	RAM/EEPROM 8
SERIAL_POINTER3	8-bit pointer used for string compare commands with serial port 3.	U_8	8460	RAM 8
STRING_CHARACTER3 144	8-bit register. Sets ASCII character for the start or end of serial receive string in master mode for port 3.	U_8	8228 144	RAM/EEPROM 8
STRING_LENGTH3	8-bit register. Sets string length of serial receive string in master mode for port 3.	U_8	8232	RAM/EEPROM 8
RECEIVE_IDLE_TIME3	8-bit register. Shows seconds of inactivity on serial port 3. (max. count = 255 seconds	U_8	8507	RAM 8

Baud Rates - Port 3

The baud rates for port 3 are controlled by bits 0 to 2 of register 8209 and the available options are shown below.

Bits 2, 1, 0	Baud rate
	000 = 1200 baud
	001 = 2400 baud
	010 = 4800 baud
	011 = 9600 baud
	100 = 19200 baud
	101 = Not available (19200 baud)
	110 = Not available (19200 baud)
	111 = Not available (19200 baud).

Serial Port Modes - Registers 8217

Registers 8217 is an 8 bit register which controls the functionality of serial port 3. The following table shows the register value for currently available serial port protocols.

Value	Mode
0	ASCII 14

1	Modbus RTU slave
2	Macro
3	Printer
4	Not available
5	Not available
6	LCD touch panel (19200 baud only)
7	Modbus RTU Master 147

2.25.10.1 Serial Buffer Port 3 - Registers 12801 - 13056

Registers 12801 to 13056 are all 8-bit unsigned registers that are used as a buffer for serial port 3 received and transmitted data. They are used in all serial port modes, but their intended use is in master mode under macro control. By accessing these registers individually, a message string can be built up or interrogated, byte by byte.

NOTE: Although registers 12801 to 13056 can be written to, it is not recommended unless you have a thorough knowledge of how the serial port operates. Writing the wrong value to these registers could cause the serial port to lock up.

See Also
Serial Transmit Count - Registers 8465 - 8467 145

Serial Receive Count- Registers 8454 - 8456 145

2.26 Setpoint

Setpoint registers contain all individual setpoint activation, control, and setup information for the 12 setpoints that are available for macro and front panel programming.

See
Setpoint 1 166
Setpoint 2 168
Setpoint 3 170
Setpoint 4 172
Setpoint 5 174
Setpoint 6 176
Setpoint 7 178
Setpoint 8 180
Setpoint 9 182
Setpoint 10 184
Setpoint 11 186
Setpoint 12 188
PID 122

See also
Setpoint Control Registers 156
Setpoint Latch Mask - Register 4100 157
Relay De-energize Mask - Register 4101 160
Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160
Reset Destination - Register 4229 to 4234
Setpoint Data Source Selection - Register 4401 to 4406 161

Setpoint Tracking - Register 8261 to 8272 161
Delay Type - Register 8277 to 8288 162
Setpoint Trigger Functions - Register 8293 to 8304 163
Setpoint Status Flags - Register 4097 164
Setpoint Trigger Flags - Register 4098 165
Setpoint Blanking - Register 4435 165

See Setpoints & Relays Supplement (NZ201) for a detailed description of the functionality of setpoints and relays.

2.26.1 Setpoint Control Registers

Registers 8245 to 8256
These are 8-bit registers used to control setpoint functionality. When reading or writing to these registers from the macro or via the serial port, the data is treated in octal format 107 so that it is identical to the value shown on the display of the controller when setting these codes up manually. This allows 3 function groups to be controlled with one 3-digit number. The functional groups for the setpoint control registers are:

Display Digit	**1st Digit**	**2nd Digit**	**3rd Digit**
Function	Relay Energize Function	SP Activation Source	SP Functions

See graphic of setpoint 3-digit settings.

1st Digit - Relay Energize Function
The 1st digit of a setpoint control register (bits 6 & 7) controls when the relays energize in response to the input condition. The options available are:

0 = Energizes at or above setpoint value
1 = Energizes below setpoint value
2 = Energizes at or above setpoint value with falling input signal initial startup inhibit
3 = Energizes below setpoint value with rising input signal initial startup inhibit.

See detailed description of 1st digit Relay Energize Functions 158 options.

2nd Digit - SP Activation Source
The 2nd digit of a setpoint control register (bits 3, 4, 5) selects the data source for the setpoint control logic. The options available are:

0 = Activate setpoint from selected source register
1 = Select source register for setpoint
2 = Activate setpoint from digital input source - Capture Pin
3 = Activate setpoint from digital input source - D1 Pin
4 = Activate setpoint from digital input source - D2 Pin
5 = Activate setpoint from digital input source - D3 Pin
6 = Activate setpoint from digital input source - Hold Pin
7 = Activate setpoint from digital input source - Lock Pin.

See list of most commonly used named registers for the Setpoint Activation Source.

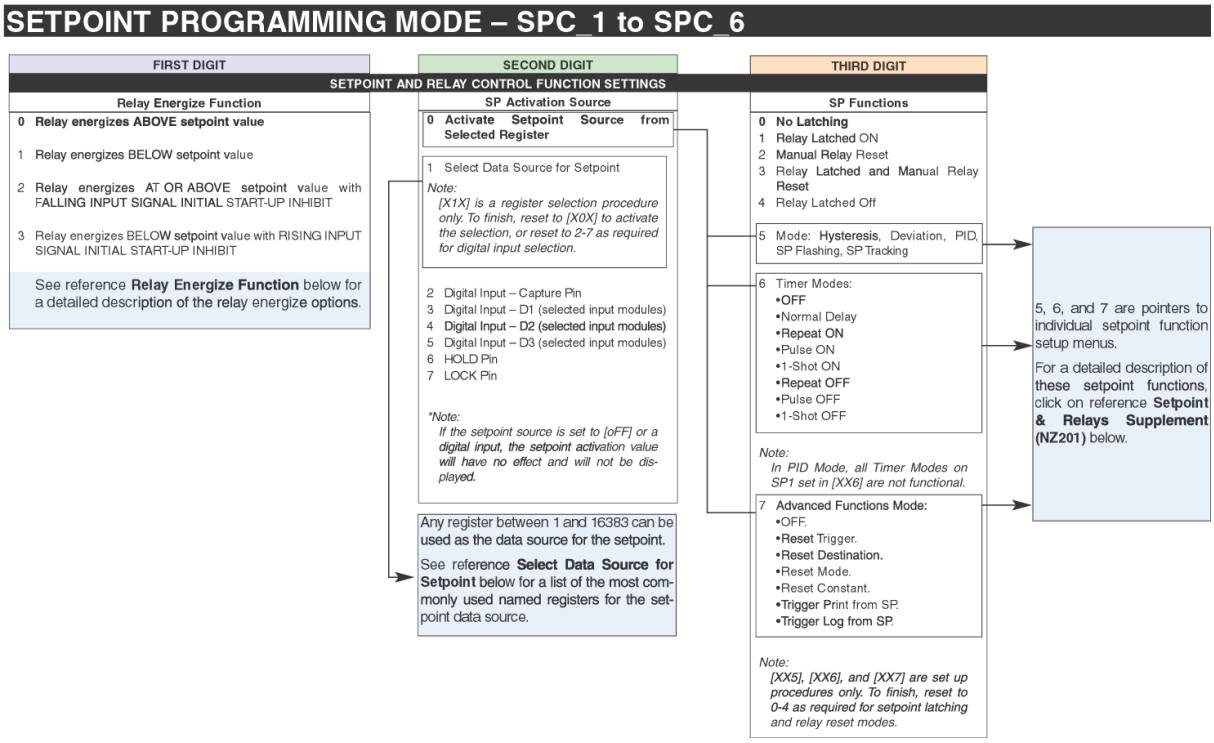
3rd Digit - SP Functions
The 3rd digit of a setpoint control register (bits 0,1,2) selects special setpoint functions and gives access to higher level setpoint functions from the display panel of the controller. The options available are:

0 = No latching
1 = Relay latched ON
2 = Manual relay reset
3 = Relay latched ON and manual relay reset
4 = Relay latched OFF
5 = Entry into Hysteresis, deviation and PID menus
6 = Entry into Timer menu
7 = Entry into trigger menu

See Setpoints & Relays Supplement (NZ201) for a detailed description of the functionality of setpoints and relays, including timer modes; reset and trigger modes; hysteresis, deviation, and PID modes; setpoint tracking and more.

See also
Octal Format 107

2.26.1.1 Setpoint 3-digit Graphic



2.26.1.1.1 Setpoint Latch Mask - Register 4100

This is a 16-bit register in RAM that controls the latching feature for the setpoints. If latching is selected for a setpoint and the appropriate bit of register 4100 is set, then the setpoint is latched (either above or below the setpoint value as selected). The setpoint can be unlatched by clearing the appropriate bit to zero. This holds true regardless of whether the latching is in the ON state or the OFF state. Unlatching the controller from the front panel buttons or from the LOCK or HOLD pin does exactly the same thing.

The function of each bit is as follows:

Bit	Name	Description	Function
0	SP1_LATCH	Flag shows/controls the latch status of setpoint 1.	0 = Setpoint unlatched 1 = Setpoint Latched
1	SP2_LATCH	Flag shows/controls the latch status of setpoint 2.	0 = Setpoint unlatched 1 = Setpoint Latched
2	SP3_LATCH	Flag shows/controls the latch status of setpoint 3.	0 = Setpoint unlatched 1 = Setpoint Latched
3	SP4_LATCH	Flag shows/controls the latch status of setpoint 4.	0 = Setpoint unlatched 1 = Setpoint Latched
4	SP5_LATCH	Flag shows/controls the latch status of setpoint 5.	0 = Setpoint unlatched 1 = Setpoint Latched
5	SP6_LATCH	Flag shows/controls the latch status of setpoint 6.	0 = Setpoint unlatched 1 = Setpoint Latched
6	SP7_LATCH	Flag shows/controls the latch status of setpoint 7.	0 = Setpoint unlatched 1 = Setpoint Latched
7	SP8_LATCH	Flag shows/controls the latch status of setpoint 8.	0 = Setpoint unlatched 1 = Setpoint Latched
8	SP9_LATCH	Flag shows/controls the latch status of setpoint 9.	0 = Setpoint unlatched 1 = Setpoint Latched
9	SP10_LATCH	Flag shows/controls the latch status of setpoint 10.	0 = Setpoint unlatched 1 = Setpoint Latched
10	SP11_LATCH	Flag shows/controls the latch status of setpoint 11.	0 = Setpoint unlatched 1 = Setpoint Latched
11	SP12_LATCH	Flag shows/controls the latch status of setpoint 12.	0 = Setpoint unlatched 1 = Setpoint Latched
12 to 15	Don't care	-	-

2.26.1.2 Relay Energize Functions

Following is a detailed description of the options available on the 1st digit of the setpoint programming mode's setpoint control settings. Each description shows how the relay energize function operates when the setpoint has been set up for either hysteresis, deviation, or PID modes.

1st Digit Of Setpoint Control	Setpoint Mode	Hysteresis Type	Description
0	Normal	n/a	Relay energises at or above the setpoint value.
	Hysteresis	Temperature	Cooling mode - Relay de-energises at or below the setpoint value and energises above (setpoint value + hysteresis value).
		Alarm	Relay energises at or above the setpoint value and de-energises below (setpoint value - hysteresis value).
	Deviation	n/a	Relay energises inside the deviation band (setpoint value +/- deviation counts) and de-energises outside the deviation band.
	PID	n/a	Controls above the setpoint value.
1	Normal	n/a	Relay energises below the setpoint value.
	Hysteresis	Temperature	Heating mode - Relay de-energises at or above the setpoint value and energises below (setpoint value - hysteresis value).
		Alarm	Relay energises at or below the setpoint value and de-energises above (setpoint value + hysteresis value).
	Deviation	n/a	Relay energises outside the deviation band (setpoint value +/- deviation counts) and de-energises inside the deviation band.
	PID	n/a	Controls below the setpoint value.
2	Normal	n/a	Relay energises at or above the setpoint value with falling input startup inhibit. (see note below on Falling Input Startup Inhibit)
	Hysteresis	Temperature	Cooling mode - Relay de-energises at or below the setpoint value and energises above (setpoint value + hysteresis value) with falling input startup inhibit. (see note below on Falling Input Startup Inhibit)
		Alarm	Relay energises at or above the setpoint value and de-energises below (setpoint value - hysteresis value) with falling input startup inhibit.
	Deviation	n/a	Relay energises inside the deviation band (setpoint value +/- deviation counts) and de-energises outside the deviation band with falling input startup inhibit. (see note below on Falling Input Startup Inhibit)
	PID	n/a	Controls above the setpoint value.
3	Normal	n/a	Relay energises below the setpoint value with rising input startup inhibit.
	Hysteresis	Temperature	Heating mode - Relay de-energises at or above the setpoint value and energises below (setpoint value - hysteresis value) with rising input startup inhibit. (see note below on Rising Input Startup Inhibit)
		Alarm	Relay energises at or below the setpoint value and de-energises above (setpoint value + hysteresis value) with rising input startup inhibit. (see note below on Rising Input Startup Inhibit)
	Deviation	n/a	Relay energises outside the deviation band (setpoint value +/- deviation counts) and de-energises inside the deviation band with rising input startup inhibit. (see note below on Rising Input Startup Inhibit)
	PID	n/a	Controls below the setpoint value.

Falling Input Startup Inhibit

Falling input startup inhibit means that if the input signal is above the setpoint value at power up the relay will not be energised. The input signal must first fall below the setpoint value and rise again before the relay will be energised.

Rising Input Startup Inhibit

Rising input startup inhibit means that if the input signal is below the setpoint value at power up the relay will not be energised. The input signal must first rise above the setpoint value and fall again

before the relay will be energised.

2.26.2 Relay De-energize Mask - Register 4101

This is a 16-bit register in RAM that controls the de-energizing feature for the relays. If the de-energize feature is selected for a setpoint, then when that setpoint is in its inactive state, the appropriate bit of register 4101 is set by the software. When the setpoint becomes active, register 4101 is used as a mask and the appropriate bit is ANDed with the relay output state. If the result is a **1**, then the relay is energized. If the appropriate bit of register 4101 is cleared to a **0** (while the setpoint is active), the relay is de-energized. As soon as the setpoint returns to its inactive state the appropriate bit of register 4101 is set to a **1** again. If a relay is de-energized by the front panel buttons, register 4101 is modified in the same way.

The function of each bit is shown as follows:

Bit	Name	Description	Function
0	RLY1_DE_ENERGISE	Flag shows/controls the de-energized status of relay 1.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
1	RLY2_DE_ENERGISE	Flag shows/controls the de-energized status of relay 2.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
2	RLY3_DE_ENERGISE	Flag shows/controls the de-energized status of relay 3.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
3	RLY4_DE_ENERGISE	Flag shows/controls the de-energized status of relay 4.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
4	RLY5_DE_ENERGISE	Flag shows/controls the de-energized status of relay 5.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
5	RLY6_DE_ENERGISE	Flag shows/controls the de-energized status of relay 6.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
6	RLY7_DE_ENERGISE	Flag shows/controls the de-energized status of relay 7.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
7	RLY8_DE_ENERGISE	Flag shows/controls the de-energized status of relay 8.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
8	RLY9_DE_ENERGISE	Flag shows/controls the de-energized status of relay 9.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
9	RLY10_DE_ENERGISE	Flag shows/controls the de-energized status of relay 10.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
10	RLY11_DE_ENERGISE	Flag shows/controls the de-energized status of relay 11.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
11	RLY12_DE_ENERGISE	Flag shows/controls the de-energized status of relay 12.	0 = Relay De-energize (Activate) 1 = Relay De-energize (Inactive)
12 to 15	Don't care	-	-

2.26.3 Setpoint Reset Delay (Power-On Inhibit) - Register 4102

Register 4102 is a 16-bit register in RAM that contains flags for the reset delay function of the setpoints.

The function of each bit is as follows:

Bit	Name	Description	Function
0	POWERON_INHIBIT_SP1	Bit flag shows that setpoint 1 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
1	POWERON_INHIBIT_SP2	Bit flag shows that setpoint 2 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
2	POWERON_INHIBIT_SP3	Bit flag shows that setpoint 3 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
3	POWERON_INHIBIT_SP4	Bit flag shows that setpoint 4 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
4	POWERON_INHIBIT_SP5	Bit flag shows that setpoint 5 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
5	POWERON_INHIBIT_SP6	Bit flag shows that setpoint 6 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
6	POWERON_INHIBIT_SP7	Bit flag shows that setpoint 7 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
7	POWERON_INHIBIT_SP8	Bit flag shows that setpoint 8 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
8	POWERON_INHIBIT_SP9	Bit flag shows that setpoint 9 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
9	POWERON_INHIBIT_SP10	Bit flag shows that setpoint 10 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
10	POWERON_INHIBIT_SP11	Bit flag shows that setpoint 11 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
11	POWERON_INHIBIT_SP12	Bit flag shows that setpoint 12 has been in-active since power-on.	0 = power-on inhibit ACTIVE 1 = power-on inhibit INACTIVE
12 to 15	-	Don't care	-

After power on, register 4102 is initially cleared to zero. As each setpoint is examined, the appropriate bit of register 4102 is set only if the setpoint is inactive. A setpoint that has the reset delay feature selected can only activate the relay if the appropriate power-on inhibit bit is set. This means that after reset, the setpoint must first enter the in-active state before it can be activated.

This register can be read or written to.

2.26.4 Reset Destination - Register 4229 to 4240

Registers 4229 to 4240 are 16-bit registers that specify the destination register that will be modified by each setpoint reset function.

2.26.5 Setpoint Data Source Selection - Register 4401 to 4406

Registers 4401 to 4412 are 16-bit registers that specify the data source for the setpoints. The number they contain is the ASCII/Modbus register number for the required data source.

NOTE: Only registers that hold integer values can be used as a data source for the display. Floating point and text registers can not be used.

2.26.6 Setpoint Tracking - Register 8261 to 8272

Registers 8261 to 8272 are 8-bit registers used for selecting setpoint tracking. The function of each bit is as follows:

Bit Position								Description
7	6	5	4	3	2	1	0	
				0	0	0	0	Tracking disabled
				0	0	0	1	Setpoint tracks SP1
				0	0	1	0	Setpoint tracks SP2
				0	0	1	1	Setpoint tracks SP3
				0	1	0	0	Setpoint tracks SP4
				0	1	0	1	Setpoint tracks SP5
				0	1	1	0	Setpoint tracks SP6
				0	1	1	1	Setpoint tracks SP7
				1	0	0	0	Setpoint tracks SP8
				1	0	0	1	Setpoint tracks SP9
				1	0	1	0	Setpoint tracks SP10
				1	0	1	1	Setpoint tracks SP11
				1	1	0	0	Setpoint tracks SP12

2.26.7 Delay Type - Register 8277 to 8288

Registers 8277 to 8288 are 8-bit registers used to control the delay type, display flashing, and mode of each setpoint SP1 to SP12.

The function of each bit is as follows:

Bits 0 to 2: Delay type

Bit Position								Description
7	6	5	4	3	2	1	0	
					0	0	0	OFF
					0	0	1	Normal
					0	1	0	1-Shot
					0	1	1	Pulse
					1	0	0	Repeat
					1	0	1	Negative 1-Shot
					1	1	0	Negative Pulse
					1	1	1	Negative Repeat

Bit 3: Display flash on setpoint
0 = no flash
1 = flash on setpoint active

Bit 4, 5: Hysteresis / Deviation / PID mode

Bit Position								Description
7	6	5	4	3	2	1	0	
		0	0					OFF
		0	1					Hysteresis
		1	0					Deviation
		1	1					PID (SP1 to SP6 only)

Bit 6 Delay resolution (SP1 to SP6 only)
0 = 0.1 second resolution
1 = 1 millisecond resolution

Bit 7 **Hysteresis Type**
0 = Temperature control
1 = Alarm
(Note: see Hysteresis Type 163 for a full explanation of options)

2.26.8 Hysteresis Type

When a setpoint is operated in Hysteresis mode, two types of hysteresis action can be selected depending on the application. The hysteresis can be set to operate in a manner suitable for temperature control applications or it can be set to operate for use with alarms. A more detailed explanation of each mode is given below.

1st Digit Of Setpoint Control	Hysteresis Type	Description
0, 2	Temperature	Cooling mode - Relay de-energises at or below the setpoint value and energises above (setpoint value + hysteresis value).
	Alarm	Relay energises at or above the setpoint value and de-energises below (setpoint value - hysteresis value).
1, 3	Temperature	Heating mode - Relay de-energises at or above the setpoint value and energises below (setpoint value - hysteresis value).
	Alarm	Relay energises at or below the setpoint value and de-energises above (setpoint value + hysteresis value).

See Also
Delay Type - Register 8277 to 8288 162

2.26.9 Setpoint Trigger Functions - Register 8293 to 8304

Registers 8293 to 8304 are 8-bit registers used for selecting the setpoint trigger functions on the following setpoints.

The function of each bit is as follows:

Bits 0 to 2: Trigger functions

Bit Position								Description
7	6	5	4	3	2	1	0	
					0	0	0	All trigger functions disabled
					0	0	1	Trigger on make edge
					0	1	0	Trigger on break edge
					0	1	1	Trigger on both make & break edge
					1	0	0	Trigger when energized

Bit 3 & 4: Reset mode

Bit Position								Description
7	6	5	4	3	2	1	0	
			0	0				Destination register = user defined constant
			0	1				Destination register = Input_data - Setpoint_value+Constant
			1	0				Destination register = Destination + Constant
			1	1				Destination register = Source register

Bit 5: **Reserved for future development**

Bit 6: **Log on selected edge**
0 = no log
1 = log

Bit 7: **Print on selected edge**
0 = no print
1 = print

2.26.10 Setpoint Status Flags - Register 4097

Register 4097 is a single 16-bit read only register that contains 12 flags showing the status of setpoints 1 to 12 in normal and remote mode. It differs from the ALARM_STATUS 207 register (239) which allows the setpoints to be remotely controlled as well. If a setpoint is forced into remote mode by a write to register 239, the setpoint will no longer respond to input changes based on the setpoint logic but will now only change state when 239 is written. However, the setpoint logic (i.e. comparison between the setpoint activation value and the data input value) is still operational in the background even though it is not used. Register 4097 displays the status of the comparison at this point before the final control is diverted to remote mode. It includes features such hysteresis, deviation, and setpoint tracking.

This can be useful in a macro that needs to control the relay in a special way and still use one of the above standard features.

The function of each bit is as follows:

Bit	Name	Description	Function
0	SP1_STATUS	Read only flag shows the status of setpoint 1 in normal & remote mode.	(1 = setpoint activated)
1	SP2_STATUS	Read only flag shows the status of setpoint 2 in normal & remote mode.	(1 = setpoint activated)
2	SP3_STATUS	Read only flag shows the status of setpoint 3 in normal & remote mode.	(1 = setpoint activated)
3	SP4_STATUS	Read only flag shows the status of setpoint 4 in normal & remote mode.	(1 = setpoint activated)
4	SP5_STATUS	Read only flag shows the status of setpoint 5 in normal & remote mode.	(1 = setpoint activated)
5	SP6_STATUS	Read only flag shows the status of setpoint 6 in normal & remote mode..	(1 = setpoint activated)
6	SP7_STATUS	Read only flag shows the status of setpoint 7 in normal & remote mode.	(1 = setpoint activated)
7	SP8_STATUS	Read only flag shows the status of setpoint 8 in normal & remote mode.	(1 = setpoint activated)
8	SP9_STATUS	Read only flag shows the status of setpoint 9 in normal & remote mode.	(1 = setpoint activated)
9	SP10_STATUS	Read only flag shows the status of setpoint 10 in normal & remote mode.	(1 = setpoint activated)
10	SP11_STATUS	Read only flag shows the status of setpoint 11 in normal & remote mode.	(1 = setpoint activated)
11	SP12_STATUS	Read only flag shows the status of setpoint 12 in normal & remote mode.	(1 = setpoint activated)
12 to 15	-	Reserved for future development.	-

See also

ALARM_STATUS 207

2.26.11 Setpoint Trigger Flags - Register 4098

Register 4098 is a single 16-bit read only register that contains 12 flags showing the trigger status for each of the 12 setpoints. Each flag is set if the trigger condition selected for that setpoint (i.e. make, break, both, level) is satisfied, and cleared if the trigger condition is false.

The function of each bit is as follows:

Bit	Name	Description	Function
0	TRIGGER1	Read only flag shows the trigger status for setpoint 1.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
1	TRIGGER2	Read only flag shows the trigger status for setpoint 2.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
2	TRIGGER3	Read only flag shows the trigger status for setpoint 3.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
3	TRIGGER4	Read only flag shows the trigger status for setpoint 4.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
4	TRIGGER5	Read only flag shows the trigger status for setpoint 5.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
5	TRIGGER6	Read only flag shows the trigger status for setpoint 6.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
6	TRIGGER7	Read only flag shows the trigger status for setpoint 7.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
7	TRIGGER8	Read only flag shows the trigger status for setpoint 8.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
8	TRIGGER9	Read only flag shows the trigger status for setpoint 9.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
9	TRIGGER10	Read only flag shows the trigger status for setpoint 10.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
10	TRIGGER11	Read only flag shows the trigger status for setpoint 11.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
11	TRIGGER12	Read only flag shows the trigger status for setpoint 12.	0 = Trigger INACTIVE 1 = Trigger ACTIVATED
12 to 15	-	Reserved for future development.	-

NOTE: These flags only remain **alive** for one output cycle following the active edge which caused the trigger. They are intended to be used by the macro.

2.26.12 Setpoint Blanking

Register 4435 is a 16 bit register in EEPROM which controls the sequence of setpoint setup parameters that are displayed when the "Prog" and "Down" button are pressed. Each bit in the register controls a specific parameter display as shown below. If a specific bit is a "0" then the display of the associated parameter is disabled and that parameter will be skipped over. If a bit is a "1" the parameter will be displayed.

Bit 0 = "Lock" display
Bit 1 = Setpoint 1
Bit 2 = Setpoint 2
Bit 3 = Setpoint 3
Bit 4 = Setpoint 4
Bit 5 = Setpoint 5
Bit 6 = Setpoint 6

Bit 7 = Setpoint control 1
Bit 8 = Setpoint control 2
Bit 9 = Setpoint control 3
Bit 10 = Setpoint control 4
Bit 11 = Setpoint control 5
Bit 12 = Setpoint control 6
Bit 13 = not used
Bit 14 = not used
Bit 15 = not used

When register 4435 is read it will be displayed as a 16 bit unsigned number. The default value will be 8191 (0x1FFF hex) which is all codes enabled.

2.26.13 Setpoint 1

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT1	32-bit register for setpoint 1 value.	S_32	111	RAM/EEPROM 8
SP1	This flag shows/controls the status of setpoint 1 (ON = setpoint activated).	B_0	239	RAM 8
SP1_REMOTE	Setting this bit to ON places setpoint 1 in remote mode.	B_8	239	RAM 8
SP1_STATUS 197	Read only flag shows the status of setpoint 1 in normal & remote mode.	B_0_R	4097 197	RAM 8
TRIGGER1 165	8-bit read only register which contains status flags for the Modbus master macro.	B_0_R	4098 165	RAM 8
RELAY1 197	Flag which shows/controls the instantaneous state of relay 1 (ON=energized).	B_0	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP1 Setup 167

2.26.13.1 SP1 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP1 160	Bit flag shows that setpoint 1 has been in-active since power-on.	B_0	4102 160	RAM 8
RLY1_DE_ENERGISE 160	Bit flag shows that setpoint 1 has been in-active since power-on.	B_0	4101 160	RAM 8
SETPOINT1_TEXT	Text display for setpoint 1.	L_8_T	16495	EEPROM 8
SETPOINT1_BREAK_DELAY	16-bit register holds the break delay time for setpoint 1 (0.1s or 0.001s resolution).	U_16	4213	RAM/EEPROM 8
SP1_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 1(note: controller display is in octal).	O_8	8245 156	RAM/EEPROM 8
SP1_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 1.	U_16	4401 161	RAM/EEPROM 8
SP1_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 1.	U_8	8277 162	RAM/EEPROM 8
SP1_HYST	16-bit register holds the hysteresis/passband value for setpoint 1.	U_16	4181	RAM/EEPROM 8
SP1_LATCH 157	Flag shows/controls the latch status of setpoint 1(ON = setpoint latched).	B_0	4100 157	RAM 8
SP1_MAKE_DELAY	16-bit register holds the make delay time for setpoint 1 (0.1s or 0.001s resolution).	U_16	4197	RAM/EEPROM 8
SP1_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 1 trigger functions.	U_8	4229 161	RAM/EEPROM 8
SP1_RESET_VALUE	32-bit register holds the reset value used with setpoint 1 trigger functions.	S_32	429	RAM/EEPROM 8
SP1_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 1.	U_8	8261 161	RAM/EEPROM 8
SP1_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 1.	U_8	8293 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

PID 122

PID 1 123

Octal Format 107

2.26.14 Setpoint 2

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT2	32-bit register for setpoint 2 value.	S_32	113	RAM/EEPROM 8
SP2	This flag shows/controls the status of setpoint 2 (ON = setpoint activated).	B_1	239	RAM 8
SP2_REMOTE	Setting this bit to ON places setpoint 2 in remote mode.	B_9	239	RAM 8
SP2_STATUS 197	Read only flag shows the status of setpoint 2 in normal & remote mode.	B_1_R	4097 197	RAM 8
TRIGGER2 165	8-bit read only register which contains status flags for the Modbus master macro.	B	4098 165	RAM 8
RELAY2 197	Flag which shows/controls the instantaneous state of relay 2 (ON=energized).	B_1	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP2 Setup 169

2.26.14.1 SP2 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP2 160	Bit flag shows that setpoint 2 has been in-active since power-on.	B_1	4102 160	RAM 8
RLY2_DE_ENERGISE 160	Bit flag shows that setpoint 2 has been in-active since power-on.	B_1	4101 160	RAM 8
SETPOINT2_TEXT	Text display for setpoint 2.	L_8_T	16497	EEPROM 8
SETPOINT2_BREAK_DELAY	16-bit register holds the break delay time for setpoint 2 (0.1s or 0.001s resolution).	U_16	4214	RAM/EEPROM 8
SP2_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 2 (note: controller display is in octal).	O_8	8246 156	RAM/EEPROM 8
SP2_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 2.	U_16	4402 161	RAM/EEPROM 8
SP2_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 2.	U_8	8278 162	RAM/EEPROM 8
SP2_HYST	16-bit register holds the hysteresis/passband value for setpoint 2.	U_16	4182	RAM/EEPROM 8
SP2_LATCH 157	Flag shows/controls the latch status of setpoint 2 (ON = setpoint latched).	B_1	4100 157	RAM 8
SP2_MAKE_DELAY	16-bit register holds the make delay time for setpoint 2 (0.1s or 0.001s resolution).	U_16	4198	RAM/EEPROM 8
SP2_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 2 trigger functions.	U_8	4230 161	RAM/EEPROM 8
SP2_RESET_VALUE	32-bit register holds the reset value used with setpoint 2 trigger functions.	S_32	431	RAM/EEPROM 8
SP2_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 2.	U_8	8262 161	RAM/EEPROM 8
SP2_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 2.	U_8	8294 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

PID 122

PID2 125

Octal Format 107

2.26.15 Setpoint 3

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT3	32-bit register for setpoint 3 value.	S_32	115	RAM/EEPROM 8
SP3	This flag shows/controls the status of setpoint 3 (ON = setpoint activated).	B_2	239	RAM 8
SP3_REMOTE	Setting this bit to ON places setpoint 3 in remote mode.	B_10	239	RAM 8
SP3_STATUS 197	Read only flag shows the status of setpoint 3 in normal & remote mode.	B_2_R	4097 197	RAM 8
TRIGGER3 165	8-bit read only register which contains status flags for the Modbus master macro.	B_2_R	4098 165	RAM 8
RELAY3 197	Flag which shows/controls the instantaneous state of relay 3 (ON=energized).	B_2	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP3 Setup 171

2.26.15.1 SP3 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP3 160	Bit flag shows that setpoint 3 has been in-active since power-on.	B_2	4102 160	RAM 8
RLY3_DE_ENERGISE 160	Bit flag shows that setpoint 3 has been in-active since power-on.	B_2	4101 160	RAM 8
SETPOINT3_TEXT	Text display for setpoint 3.	L_8_T	16499	EEPROM 8
SETPOINT3_BREAK_DELAY	16-bit register holds the break delay time for setpoint 3 (0.1s or 0.001s resolution).	U_16	4215	RAM/EEPROM 8
SP3_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 3 (note: controller display is in octal).	O_8	8247 156	RAM/EEPROM 8
SP3_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 3.	U_16	4403 161	RAM/EEPROM 8
SP3_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 3.	U_8	8279 162	RAM/EEPROM 8
SP3_HYST	16-bit register holds the hysteresis/passband value for setpoint 3.	U_16	4183	RAM/EEPROM 8
SP3_LATCH 157	Flag shows/controls the latch status of setpoint 3 (ON = setpoint latched).	B_2	4100 157	RAM 8
SP3_MAKE_DELAY	16-bit register holds the make delay time for setpoint 3 (0.1s or 0.001s resolution).	U_16	4199	RAM/EEPROM 8
SP3_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 3 trigger functions.	U_8	4231 161	RAM/EEPROM 8
SP3_RESET_VALUE	32-bit register holds the reset value used with setpoint 3 trigger functions.	S_32	433	RAM/EEPROM 8
SP3_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 3.	U_8	8263 161	RAM/EEPROM 8
SP3_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 3.	U_8	8295 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

PID 122

PID 3 127

Octal Format 107

2.26.16 Setpoint 4

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT4	32-bit register for setpoint 4 value.	S_32	117	RAM/EEPROM 8
SP4	This flag shows/controls the status of setpoint 4 (ON = setpoint activated).	B_3	239	RAM 8
SP4_REMOTE	Setting this bit to ON places setpoint 4 in remote mode.	B_11	239	RAM 8
SP4_STATUS 197	Read only flag shows the status of setpoint 4 in normal & remote mode.	B_3_R	4097 197	RAM 8
TRIGGER4 165	8-bit read only register which contains status flags for the Modbus master macro.	B_3_R	4098 165	RAM 8
RELAY4 197	Flag which shows/controls the instantaneous state of relay 4 (ON=energized).	B_3	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP4 Setup 173

2.26.16.1 SP4 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP4 160	Bit flag shows that setpoint 4 has been in-active since power-on.	B_3	4102 160	RAM 8
RLY4_DE_ENERGISE 160	Bit flag shows that setpoint 4 has been in-active since power-on.	B_3	4101 160	RAM 8
SETPOINT4_TEXT	Text display for setpoint 4.	L_8_T	16501	EEPROM 8
SETPOINT4_BREAK_DELAY	16-bit register holds the break delay time for setpoint 4 (0.1s or 0.001s resolution).	U_16	4216	RAM/EEPROM 8
SP4_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 4 (note: controller display is in octal).	O_8	8248 156	RAM/EEPROM 8
SP4_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 4.	U_16	4404 161	RAM/EEPROM 8
SP4_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 4.	U_8	8280 162	RAM/EEPROM 8
SP4_HYST	16-bit register holds the hysteresis/passband value for setpoint 4.	U_16	4184	RAM/EEPROM 8
SP4_LATCH 157	Flag shows/controls the latch status of setpoint 4 (ON = setpoint latched).	B_3	4100 157	RAM 8
SP4_MAKE_DELAY	16-bit register holds the make delay time for setpoint 4 (0.1s or 0.001s resolution).	U_16	4200	RAM/EEPROM 8
SP4_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 4 trigger functions.	U_8	4232 161	RAM/EEPROM 8
SP4_RESET_VALUE	32-bit register holds the reset value used with setpoint 4 trigger functions.	S_32	435	RAM/EEPROM 8
SP4_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 4.	U_8	8264 161	RAM/EEPROM 8
SP4_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 4.	U_8	8296 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

PID 122

PID 4 129

Octal Format 107

2.26.17 Setpoint 5

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT5	32-bit register for setpoint 5 value.	S_32	119	RAM/EEPROM 8
SP5	This flag shows/controls the status of setpoint 5 (ON = setpoint activated).	B_4	239	RAM 8
SP5_REMOTE	Setting this bit to ON places setpoint 5 in remote mode.	B_12	239	RAM 8
SP5_STATUS 197	Read only flag shows the status of setpoint 5 in normal & remote mode.	B_4_R	4097 197	RAM 8
TRIGGER5 165	8-bit read only register which contains status flags for the Modbus master macro.	B_4_R	4098 165	RAM 8
RELAY5 197	Flag which shows/controls the instantaneous state of relay 5 (ON=energized).	B_4	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP5 Setup 175

2.26.17.1 SP5 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP5 161	Bit flag shows that setpoint 5 has been in-active since power-on.	B_4	4102 160	RAM 8
RLY5_DE_ENERGISE 160	Bit flag shows that setpoint 5 has been in-active since power-on.	B_4	4101 160	RAM 8
SETPOINT5_TEXT	Text display for setpoint 5.	L_8_T	16503	EEPROM 8
SETPOINT5_BREAK_DELAY	16-bit register holds the break delay time for setpoint 5 (0.1s or 0.001s resolution).	U_16	4217	RAM/EEPROM 8
SP5_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 5 (note: controller display is in octal).	O_8	8249 156	RAM/EEPROM 8
SP5_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 5.	U_16	4405 161	RAM/EEPROM 8
SP5_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 5.	U_8	8281 162	RAM/EEPROM 8
SP5_HYST	16-bit register holds the hysteresis/passband value for setpoint 5.	U_16	4185	RAM/EEPROM 8
SP5_LATCH 157	Flag shows/controls the latch status of setpoint 5 (ON = setpoint latched).	B_4	4100 157	RAM 8
SP5_MAKE_DELAY	16-bit register holds the make delay time for setpoint 5 (0.1s or 0.001s resolution).	U_16	4201	RAM/EEPROM 8
SP5_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 5 trigger functions.	U_8	4233 161	RAM/EEPROM 8
SP5_RESET_VALUE	32-bit register holds the reset value used with setpoint 5 trigger functions.	S_32	437	RAM/EEPROM 8
SP5_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 5.	U_8	8265 161	RAM/EEPROM 8
SP5_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 5.	U_8	8297 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

PID 122

PID 5 131

Octal Format 107

2.26.18 Setpoint 6

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT6	32-bit register for setpoint 6 value.	S_32	121	RAM/EEPROM 8
SP6	This flag shows/controls the status of setpoint 6 (ON = setpoint activated).	B_5	239	RAM 8
SP6_REMOTE	Setting this bit to ON places setpoint 6 in remote mode.	B_13	239	RAM 8
SP6_STATUS 197	Read only flag shows the status of setpoint 6 in normal & remote mode.	B_5_R	4097 197	RAM 8
TRIGGER6 165	8-bit read only register which contains status flags for the Modbus master macro.	B_5_R	4098 165	RAM 8
RELAY6 197	Flag which shows/controls the instantaneous state of relay 6 (ON=energized).	B_5	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP6 Setup 177

Register 4099 - Relay output 197

2.26.18.1 SP6 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP6 160	Bit flag shows that setpoint 6 has been in-active since power-on.	B_5	4102 160	RAM 8
RLY6_DE_ENERGISE 160	Bit flag shows that setpoint 6 has been in-active since power-on.	B_5	4101 160	RAM 8
SETPOINT6_TEXT	Text display for setpoint 6.	L_8_T	16505	EEPROM 8
SETPOINT6_BREAK_DELAY	16-bit register holds the break delay time for setpoint 6 (0.1s or 0.001s resolution).	U_16	4218	RAM/EEPROM 8
SP6_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 6 (note: controller display is in octal).	O_8	8250 156	RAM/EEPROM 8
SP6_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 6.	U_16	4406 161	RAM/EEPROM 8
SP6_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 6.	U_8	8282 162	RAM/EEPROM 8
SP6_HYST	16-bit register holds the hysteresis/passband value for setpoint 6.	U_16	4186	RAM/EEPROM 8
SP6_LATCH 157	Flag shows/controls the latch status of setpoint 6 (ON = setpoint latched).	B_5	4100 157	RAM 8
SP6_MAKE_DELAY	16-bit register holds the make delay time for setpoint 6 (0.1s or 0.001s resolution).	U_16	4202	RAM/EEPROM 8
SP6_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 6 trigger functions.	U_8	4234 161	RAM/EEPROM 8
SP6_RESET_VALUE	32-bit register holds the reset value used with setpoint 6 trigger functions.	S_32	439	RAM/EEPROM 8
SP6_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 6.	U_8	8266 161	RAM/EEPROM 8
SP6_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 6.	U_8	8298 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

PID 122

PID 6 133

Octal Format 107

2.26.19 Setpoint 7

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT7	32-bit register for setpoint 7 value.	S_32	123	RAM/EEPROM 8
SP7	This flag shows/controls the status of setpoint 7 (ON = setpoint activated).	B_16	239	RAM 8
SP7_REMOTE	Setting this bit to ON places setpoint 7 in remote mode.	B_24	239	RAM 8
SP7_STATUS 197	Read only flag shows the status of setpoint 7 in normal & remote mode.	B_6_R	4097 197	RAM 8
TRIGGER7 165	8-bit read only register which contains status flags for the Modbus master macro.	B_6_R	4098 165	RAM 8
RELAY7 197	Flag which shows/controls the instantaneous state of relay 7 (ON=energized).	B_6	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP7 Setup 179

Register 4099 - Relay output 197

2.26.19.1 SP7 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP7 160	Bit flag shows that setpoint 7 has been in-active since power-on.	B_6	4102 160	RAM 8
RLY7_DE_ENERGISE 160	Bit flag shows that setpoint 7 has been in-active since power-on.	B_6	4101 160	RAM 8
SETPOINT7_TEXT	Text display for setpoint 7.	L_8_T	16507	EEPROM 8
SETPOINT7_BREAK_DELAY	16-bit register holds the break delay time for setpoint 7 (0.1s resolution).	U_16	4219	RAM/EEPROM 8
SP7_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 7 (note: controller display is in octal).	O_8	8251 156	RAM/EEPROM 8
SP7_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 7.	U_16	4407 161	RAM/EEPROM 8
SP7_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 7.	U_8	8283 162	RAM/EEPROM 8
SP7_HYST	16-bit register holds the hysteresis/passband value for setpoint 7.	U_16	4187	RAM/EEPROM 8
SP7_LATCH 157	Flag shows/controls the latch status of setpoint 7 (ON = setpoint latched).	B_6	4100 157	RAM 8
SP7_MAKE_DELAY	16-bit register holds the make delay time for setpoint 7 (0.1s resolution).	U_16	4203	RAM/EEPROM 8
SP7_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 7 trigger functions.	U_8	4235 161	RAM/EEPROM 8
SP7_RESET_VALUE	32-bit register holds the reset value used with setpoint 7 trigger functions.	S_32	441	RAM/EEPROM 8
SP7_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 7.	U_8	8267 161	RAM/EEPROM 8
SP7_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 7.	U_8	8299 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

Octal Format 107

2.26.20 Setpoint 8

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT8	32-bit register for setpoint 8 value.	S_32	125	RAM/EEPROM 8
SP8	This flag shows/controls the status of setpoint 8 (ON = setpoint activated).	B_17	239	RAM 8
SP8_REMOTE	Setting this bit to ON places setpoint 8 in remote mode.	B_25	239	RAM 8
SP8_STATUS 197	Read only flag shows the status of setpoint 8 in normal & remote mode.	B_7_R	4097 197	RAM 8
TRIGGER8 165	8-bit read only register which contains status flags for the Modbus master macro.	B_7_R	4098 165	RAM 8
RELAY8 197	Flag which shows/controls the instantaneous state of relay 8 (ON=energized).	B_7	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP8 Setup 181

Register 4099 - Relay output 197

2.26.20.1 SP8 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP8 160	Bit flag shows that setpoint 8 has been in-active since power-on.	B_7	4102 160	RAM 8
RLY8_DE_ENERGISE 160	Bit flag shows that setpoint 8 has been in-active since power-on.	B_7	4101 160	RAM 8
SETPOINT8_TEXT	Text display for setpoint 8.	L_8_T	16509	EEPROM 8
SETPOINT8_BREAK_DELAY	16-bit register holds the break delay time for setpoint 8 (0.1s resolution).	U_16	4220	RAM/EEPROM 8
SP8_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 8 (note: controller display is in octal).	O_8	8252 156	RAM/EEPROM 8
SP8_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 8.	U_16	4408 161	RAM/EEPROM 8
SP8_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 8.	U_8	8284 162	RAM/EEPROM 8
SP8_HYST	16-bit register holds the hysteresis/passband value for setpoint 8.	U_16	4188	RAM/EEPROM 8
SP8_LATCH 157	Flag shows/controls the latch status of setpoint 8 (ON = setpoint latched).	B_7	4100 157	RAM 8
SP8_MAKE_DELAY	16-bit register holds the make delay time for setpoint 8 (0.1s resolution).	U_16	4204	RAM/EEPROM 8
SP8_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 8 trigger functions.	U_8	4236 161	RAM/EEPROM 8
SP8_RESET_VALUE	32-bit register holds the reset value used with setpoint 8 trigger functions.	S_32	443	RAM/EEPROM 8
SP8_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 8.	U_8	8268 161	RAM/EEPROM 8
SP8_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 8.	U_8	8300 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

Octal Format 107

2.26.21 Setpoint 9

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT9	32-bit register for setpoint 9 value.	S_32	127	RAM/EEPROM 8
SP9	This flag shows/controls the status of setpoint 9 (ON = setpoint activated).	B_18	239	RAM 8
SP9_REMOTE	Setting this bit to ON places setpoint 9 in remote mode.	B_26	239	RAM 8
SP9_STATUS 197	Read only flag shows the status of setpoint 9 in normal & remote mode.	B_8_R	4097 197	RAM 8
TRIGGER9 165	8-bit read only register which contains status flags for the Modbus master macro.	B_8_R	4098 165	RAM 8
RELAY9 197	Flag which shows/controls the instantaneous state of relay 9 (ON=energized).	B_8	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP9 Setup 183

Register 4099 - Relay output 197

2.26.21.1 SP9 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP9 160	Bit flag shows that setpoint 9 has been in-active since power-on.	B_8	4103 160	RAM 8
RLY9_DE_ENERGISE 160	Bit flag shows that setpoint 9 has been in-active since power-on.	B_8	4102 160	RAM 8
SETPOINT9_TEXT	Text display for setpoint 9.	L_8_T	16511	EEPROM 8
SETPOINT9_BREAK_DELAY	16-bit register holds the break delay time for setpoint 9 (0.1s resolution).	U_16	4221	RAM/EEPROM 8
SP9_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 9 (note: controller display is in octal).	O_8	8253 156	RAM/EEPROM 8
SP9_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 9.	U_16	4409 161	RAM/EEPROM 8
SP9_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 9.	U_8	8285 162	RAM/EEPROM 8
SP9_HYST	16-bit register holds the hysteresis/passband value for setpoint 9.	U_16	4189	RAM/EEPROM 8
SP9_LATCH 157	Flag shows/controls the latch status of setpoint 9 (ON = setpoint latched).	B_8	4101 157	RAM 8
SP9_MAKE_DELAY	16-bit register holds the make delay time for setpoint 9 (0.1s resolution).	U_16	4205	RAM/EEPROM 8
SP9_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 9 trigger functions.	U_8	4237 161	RAM/EEPROM 8
SP9_RESET_VALUE	32-bit register holds the reset value used with setpoint 9 trigger functions.	S_32	445	RAM/EEPROM 8
SP9_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 9.	U_8	8269 161	RAM/EEPROM 8
SP9_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 9.	U_8	8301 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

Octal Format 107

2.26.22 Setpoint 10

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT10	32-bit register for setpoint 10 value.	S_32	129	RAM/EEPROM 8
SP10	This flag shows/controls the status of setpoint 10 (ON = setpoint activated).	B_19	239	RAM 8
SP10_REMOTE	Setting this bit to ON places setpoint 10 in remote mode.	B_27	239	RAM 8
SP10_STATUS 197	Read only flag shows the status of setpoint 10 in normal & remote mode.	B_9_R	4097 197	RAM 8
TRIGGER10 165	8-bit read only register which contains status flags for the Modbus master macro.	B_9_R	4098 165	RAM 8
RELAY10 197	Flag which shows/controls the instantaneous state of relay 10 (ON=energized).	B_9	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP10 Setup 185

Register 4099 - Relay output 197

2.26.22.1 SP10 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP10 160	Bit flag shows that setpoint 10 has been in-active since power-on.	B_9	4103 160	RAM 8
RLY10_DE_ENERGISE 160	Bit flag shows that setpoint 10 has been in-active since power-on.	B_9	4102 160	RAM 8
SETPOINT10_TEXT	Text display for setpoint 10.	L_8_T	16513	EEPROM 8
SETPOINT10_BREAK_DELAY	16-bit register holds the break delay time for setpoint 10 (0.1s resolution).	U_16	4222	RAM/EEPROM 8
SP10_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 10 (note: controller display is in octal).	O_8	8254 156	RAM/EEPROM 8
SP10_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 10.	U_16	4410 161	RAM/EEPROM 8
SP10_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 10.	U_8	8286 162	RAM/EEPROM 8
SP10_HYST	16-bit register holds the hysteresis/passband value for setpoint 10.	U_16	4190	RAM/EEPROM 8
SP10_LATCH 157	Flag shows/controls the latch status of setpoint 10 (ON = setpoint latched).	B_9	4102 157	RAM 8
SP10_MAKE_DELAY	16-bit register holds the make delay time for setpoint 10 (0.1s resolution).	U_16	4206	RAM/EEPROM 8
SP10_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 10 trigger functions.	U_8	4238 161	RAM/EEPROM 8
SP10_RESET_VALUE	32-bit register holds the reset value used with setpoint 10 trigger functions.	S_32	447	RAM/EEPROM 8
SP10_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 10.	U_8	8270 161	RAM/EEPROM 8
SP10_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 10.	U_8	8302 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

Octal Format 107

2.26.23 Setpoint 11

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT11	32-bit register for setpoint 11 value.	S_32	131	RAM/EEPROM 8
SP11	This flag shows/controls the status of setpoint 11 (ON = setpoint activated).	B_20	239	RAM 8
SP11_REMOTE	Setting this bit to ON places setpoint 11 in remote mode.	B_28	239	RAM 8
SP11_STATUS 197	Read only flag shows the status of setpoint 11 in normal & remote mode.	B_10_R	4097 197	RAM 8
TRIGGER11 165	8-bit read only register which contains status flags for the Modbus master macro.	B_10_R	4098 165	RAM 8
RELAY11 197	Flag which shows/controls the instantaneous state of relay 11 (ON=energized).	B_10	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP11 Setup 187

Register 4099 - Relay output 197

2.26.23.1 SP11 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP11 160	Bit flag shows that setpoint 11 has been in-active since power-on.	B_10	4103 160	RAM 8
RLY11_DE_ENERGISE 160	Bit flag shows that setpoint 11 has been in-active since power-on.	B_10	4102 160	RAM 8
SETPOINT11_TEXT	Text display for setpoint 11.	L_8_T	16515	EEPROM 8
SETPOINT11_BREAK_DELAY	16-bit register holds the break delay time for setpoint 11 (0.1s resolution).	U_16	4223	RAM/EEPROM 8
SP11_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 11 (note: controller display is in octal).	O_8	8255 156	RAM/EEPROM 8
SP11_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 11.	U_16	4411 161	RAM/EEPROM 8
SP11_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 11.	U_8	8287 162	RAM/EEPROM 8
SP11_HYST	16-bit register holds the hysteresis/passband value for setpoint 11.	U_16	4191	RAM/EEPROM 8
SP11_LATCH 157	Flag shows/controls the latch status of setpoint 11 (ON = setpoint latched).	B_10	4102 157	RAM 8
SP11_MAKE_DELAY	16-bit register holds the make delay time for setpoint 11 (0.1s resolution).	U_16	4207	RAM/EEPROM 8
SP11_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 11 trigger functions.	U_8	4239 161	RAM/EEPROM 8
SP11_RESET_VALUE	32-bit register holds the reset value used with setpoint 11 trigger functions.	S_32	449	RAM/EEPROM 8
SP11_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 11.	U_8	8271 161	RAM/EEPROM 8
SP11_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 11.	U_8	8303 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

Octal Format 107

2.26.24 Setpoint 12

Name	Description	Symbol Type	Register Number	Memory Type
SETPOINT12	32-bit register for setpoint 12 value.	S_32	133	RAM/EEPROM 8
SP12	This flag shows/controls the status of setpoint 12 (ON = setpoint activated).	B_21	239	RAM 8
SP12_REMOTE	Setting this bit to ON places setpoint 11 in remote mode.	B_29	239	RAM 8
SP12_STATUS 197	Read only flag shows the status of setpoint 12 in normal & remote mode.	B_11_R	4097 197	RAM 8
TRIGGER12 165	8-bit read only register which contains status flags for the Modbus master macro.	B_11_R	4098 165	RAM 8
RELAY12 197	Flag which shows/controls the instantaneous state of relay 12 (ON=energized).	B_11	4099 197	RAM 8

See also

Setpoint Status Flags - Register 4097 164

Setpoint Trigger Flags - Register 4098 165

Register 239 - Alarm Status 207

SP12 Setup 189

Register 4099 - Relay output 197

2.26.24.1 SP12 Setup

Name	Description	Symbol Type	Register Number	Memory Type
POWERON_INHIBIT_SP12 160	Bit flag shows that setpoint 12 has been in-active since power-on.	B_11	4103 160	RAM 8
RLY12_DE_ENERGISE 160	Bit flag shows that setpoint 12 has been in-active since power-on.	B_11	4102 160	RAM 8
SETPOINT12_TEXT	Text display for setpoint 12.	L_8_T	16517	EEPROM 8
SETPOINT12_BREAK_DELAY	16-bit register holds the break delay time for setpoint 12 (0.1s resolution).	U_16	4224	RAM/EEPROM 8
SP12_CONTROL 156	8-bit register holds the setpoint & relay control setting for setpoint 12 (note: controller display is in octal).	O_8	8256 156	RAM/EEPROM 8
SP12_DATA_SOURCE 161	16-bit register holds the register number of the data source for setpoint 12.	U_16	4412 161	RAM/EEPROM 8
SP12_DELAY_TYPE 162	8-bit register controls the delay type settings for setpoint 12.	U_8	8288 162	RAM/EEPROM 8
SP12_HYST	16-bit register holds the hysteresis/passband value for setpoint 12.	U_16	4192	RAM/EEPROM 8
SP12_LATCH 157	Flag shows/controls the latch status of setpoint 12 (ON = setpoint latched).	B_11	4102 157	RAM 8
SP12_MAKE_DELAY	16-bit register holds the make delay time for setpoint 12 (0.1s resolution).	U_16	4208	RAM/EEPROM 8
SP12_RESET_DESTINATION 161	8-bit register holds the destination register number for setpoint 12 trigger functions.	U_8	4240 161	RAM/EEPROM 8
SP12_RESET_VALUE	32-bit register holds the reset value used with setpoint 12 trigger functions.	S_32	451	RAM/EEPROM 8
SP12_TRACKING 161	8-bit register controls the setpoint tracking for setpoint 12.	U_8	8272 161	RAM/EEPROM 8
SP12_TRIGGER 163	8-bit register. Controls trigger functions of setpoint 12.	U_8	8304 163	RAM/EEPROM 8

See also

Setpoint Latch Mask - Register 4100 157

Relay De-energize Mask - Register 4101 160

Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160

Octal Format 107

2.27 Smart Module

Smart modules are intelligent input modules that include their own microprocessor to perform faster and higher resolution measurements. When a smart input module is connected to a controller the following result and setup registers become available and can be read and written to via the macro or the serial port.

Smart Result Registers

When accessing smart result registers the controller interrogates the smart input module for the data. Any attempt to access these registers without a smart input module being present causes the

controller to respond with an error so these registers should only be used with a smart input module connected.

Name	Description	Symbol Type	Register Number	Memory Type
SMART_RESULT1	32-bit register holds result 1 when a smart input module is used.	S_32	329	RAMinputModule 8
SMART_RESULT2	32-bit register holds result 2 when a smart input module is used.	S_32	331	RAMinputModule 8
SMART_RESULT3	32-bit register holds result 3 when a smart input module is used.	S_32	333	RAMinputModule 8
SMART_RESULT4	32-bit register holds result 4 when a smart input module is used.	S_32	335	RAMinputModule 8
SMART_RESULT5	32-bit register holds result 5 when a smart input module is used.	S_32	337	RAMinputModule 8
SMART_RESULT6	32-bit register holds result 6 when a smart input module is used.	S_32	339	RAMinputModule 8
SMART_RESULT7	32-bit register holds result 7 when a smart input module is used.	S_32	341	RAMinputModule 8
SMART_RESULT8	32-bit register holds result 8 when a smart input module is used.	S_32	343	RAMinputModule 8

NOTE: Because some smart input registers reside in the smart input module, extra time is required to access the information. In most cases this is irrelevant, but it should be taken into account when accessing these registers repeatedly from a macro that is running at the fast update speed of 100 Hz. If the macro loop time becomes too large the effective update rate is reduced and in some cases the controller becomes slow to respond.

See also

Smart Module Setup 191

Smart Input Expander 192

2.27.1 Smart Module Setup

Name	Description	Symbol Type	Register Number	Memory Type
SMART_CAL1	32-bit floating point register. Calibration constant 1 in smart input module.	F_32	1123	EEPROMinputModule/RAMinputModule 8
SMART_CAL2	32-bit floating point register. Calibration constant 2 in smart input module.	F_32	1125	EEPROMinputModule/RAMinputModule 8
SMART_CAL3	32-bit floating point register. Calibration constant 3 in smart input module.	F_32	1127	EEPROMinputModule/RAMinputModule 8
SMART_CAL4	32-bit floating point register. Calibration constant 4 in smart input module.	F_32	1129	EEPROMinputModule/RAMinputModule 8
SMART_RESET_OFFSET1	32-bit register sets the reset offset 1 for smart input modules.	S_32	465	EEPROM/RAMinputModule 8
SMART_RESET_OFFSET2	32-bit register sets the reset offset 2 for smart input modules.	S_32	467	EEPROM/RAMinputModule 8
SMART_SETUP1	8-bit register. Setup 1 data for smart input module. (Note, the display is in octal)	O_8 107	8423	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8
SMART_SETUP2	8-bit register. Setup 2 data for smart input module. (Note, the display is in octal)	O_8 107	8424	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8
SMART_SETUP3	8-bit register. Setup 3 data for smart input module. (Note, the display is in octal)	O_8 107	8425	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8
Read Only				
SMART_ID	8 bit unsigned read only register. Defines the type of smart input module currently in use.	U_8_R	8422	RAM/RAMinputModule 8
SMART_VERSION 235	8-bit unsigned read only register shows version number of smart input module.	U_8_R	8433 235	RAM/RAMinputModule 8
SMART_ERROR	Read only bit flag that is set for a read/write error to the smart input module.	B_0_R	8514	RAM 8

Registers 465 & 467 – Reset Offset Registers
Registers 465 & 467 are reset offset registers. They are used with certain types of smart input modules. A write to register 465 or 467 causes the smart input module to reset result register 1 or result register 2 respectively. If the data value written to these registers is zero, then the smart input module writes a zero into the appropriate result register. If the data value written to these registers is other than zero, then the smart input module treats this as an offset and subtracts the data value from the appropriate result register.

A read of these registers reads the last offset value that was written to the input module.

NOTE: These registers are only for use with selected types of smart input module installed in the controller.

Register 1123 to 1129 – Smart Input Module Calibration Registers
Registers 1123 to 1129 are 32-bit floating point registers used to store calibration values for smart input modules. This calibration data is stored in an EEPROM on the smart input module. This data is read by the controller at power on and transferred to the smart input module. Reading these registers directly accesses the data in the EEPROM on the input module. Because these registers hold

calibration data, they are protected from inadvertent writes by a write protection scheme which requires a key lock value (170) to be written to the lock register (8439) first before any writes to the EEPROM are enabled. Any calibration changes written to these registers are only transferred to the smart input module at power on. After recalibration, please re-power the controller to load new calibration data.

The actual function of each register varies depending on which smart input module is being used. Some smart input modules do not use these registers and any attempt to access them produces an error (null) response. If there are no smart input modules operating, these registers cannot be read.

Some older smart input modules are not fitted with an EEPROM. If the controller cannot find an EEPROM on the smart input module it use the default value of 1.0 for the each of the calibration registers.

2.27.2 Smart Input Expander

The ICC402 series controllers also allow another two smart inputs modules to be connected to the system y connecting an input expander module to the expansion bus. This increases the maximum number of analog inputs channels to 12 and gives added flexibility to input configurations.

Each additional smart input module in the input expander module has it's own result and setup registers in a similar format to the original slot on the main ICC402 module. The table below shows the additional result registers for smart input modules in the input expander module.

Name	Description	Symbol Type	Register Number	Memory Type
SMART2_RESULT1	32-bit register holds result 1 when a smart input module 2 is used.	S_32	577	RAMinputModule 8
SMART2_RESULT2	32-bit register holds result 2 when a smart input module 2 is used.	S_32	579	RAMinputModule 8
SMART2_RESULT3	32-bit register holds result 3 when a smart input module 2 is used.	S_32	581	RAMinputModule 8
SMART2_RESULT4	32-bit register holds result 4 when a smart input module 2 is used.	S_32	583	RAMinputModule 8
SMART2_RESULT5	32-bit register holds result 5 when a smart input module 2 is used.	S_32	585	RAMinputModule 8
SMART2_RESULT6	32-bit register holds result 6 when a smart input module 2 is used.	S_32	587	RAMinputModule 8
SMART2_RESULT7	32-bit register holds result 7 when a smart input module 2 is used.	S_32	589	RAMinputModule 8
SMART2_RESULT8	32-bit register holds result 8 when a smart input module 2 is used.	S_32	591	RAMinputModule 8
SMART3_RESULT1	32-bit register holds result 1 when a smart input module 3 is used.	S_32	597	RAMinputModule 8
SMART3_RESULT2	32-bit register holds result 2 when a smart input module 3 is used.	S_32	599	RAMinputModule 8
SMART3_RESULT3	32-bit register holds result 3 when a smart input module 3 is used.	S_32	601	RAMinputModule 8
SMART3_RESULT4	32-bit register holds result 4 when a smart input module 3 is used.	S_32	603	RAMinputModule 8
SMART3_RESULT5	32-bit register holds result 5 when a smart input module 3 is used.	S_32	605	RAMinputModule 8
SMART3_RESULT6	32-bit register holds result 6 when a smart input module 3 is used.	S_32	607	RAMinputModule 8
SMART3_RESULT7	32-bit register holds result 7 when a smart input module 3 is used.	S_32	609	RAMinputModule 8
SMART3_RESULT8	32-bit register holds result 8 when a smart input module 3 is used.	S_32	611	RAMinputModule 8

See also

Smart Module 2 Setup 193

Smart Module 3 Setup 195

2.27.3 Smart Module 2 Setup

The following table shows the setup registers available for the smart input module 2 when used with a smart input expander module.

NOTE: The following registers are only available when using an input expander module with a smart input module fitted to slot 2. A read of some of these registers will produce a read error if the smart input module is not connected.

Name	Description	Symbol Type	Register Number	Memory Type
SMART2_CAL1	32-bit floating point register. Calibration constant 1 in smart input module 2.	F_32	1163	EEPROMinputModule/RAMinputModule 8
SMART2_CAL2	32-bit floating point register. Calibration constant 2 in smart input module 2.	F_32	1165	EEPROMinputModule/RAMinputModule 8
SMART2_CAL3	32-bit floating point register. Calibration constant 3 in smart input module 2.	F_32	1167	EEPROMinputModule/RAMinputModule 8
SMART2_CAL4	32-bit floating point register. Calibration constant 4 in smart input module 2.	F_32	1169	EEPROMinputModule/RAMinputModule 8
SMART2_RESET_OFFSET1	32-bit register sets the reset offset 1 for smart input module 2.	S_32	593	EEPROM/RAMinputModule 8
SMART2_RESET_OFFSET2	32-bit register sets the reset offset 2 for smart input module 2.	S_32	595	EEPROM/RAMinputModule 8
SMART2_SETUP1	8-bit register. Setup 1 data for smart input module 2. (Note, the display is in octal)	O_8 107	8528	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8
SMART2_SETUP2	8-bit register. Setup 2 data for smart input module 2. (Note, the display is in octal)	O_8 107	8529	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8
SMART2_SETUP3	8-bit register. Setup 3 data for smart input module 2. (Note, the display is in octal)	O_8 107	8530	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8

Read Only

SMART2_ID	8 bit unsigned read only register. Defines the type of smart input module currently in use in slot 2 of the input expander module.	U_8_R	8527	RAM/RAMinputModule 8
SMART2_VERSION 235	8-bit unsigned read only register shows version number of smart input module 2.	U_8_R	8531 235	RAM/RAMinputModule 8
INPUT_MODULE2_CAL_DATE	32-bit unsigned read only register shows the calibration date for smart input module 2. (Date format is DDMMYY)	U_32_R	693	EEPROMinputModule 8
INPUT_MODULE2_CONFIGURATION	8-bit unsigned read only register shows the configuration type option for smart input module 2.	U_8_R	8557	EEPROMinputModule 8
INPUT_MODULE2_NO	16-bit unsigned read only register shows the PCB board number for smart input module 2.	U_16_R	4522	EEPROMinputModule 8
INPUT_MODULE2_REVISION	8-bit unsigned read only register shows the PCB board number revision code for smart input module 2. (ASCII format 'A'=65 to 'Z'=90)	U_8_R	8552	EEPROMinputModule 8
INPUT_MODULE2_SERIAL_NO	32-bit unsigned read only register shows the serial number for smart input module 2.	U_32_R	683	EEPROMinputModule 8

Registers 593 & 595 – Reset Offset Registers

Registers 593 & 595 are reset offset registers for smart input module 2. They are used with certain

types of smart input modules. A write to register 593 or 595 causes the smart input module 2 to reset result register 1 or result register 2 respectively. If the data value written to these registers is zero, then the smart input module writes a zero into the appropriate result register. If the data value written to these registers is other than zero, then the smart input module treats this as an offset and subtracts the data value from the appropriate result register.

A read of these registers reads the last offset value that was written to the input module.

NOTE: These registers are only for use with selected types of smart input module installed in the controller.

Register 1163 to 1169 – Smart Input Module Calibration Registers
Registers 1163 to 1169 are 32-bit floating point registers used to store calibration values for smart input module 2. This calibration data is stored in an EEPROM on the smart input module. This data is read by the controller at power on and transferred to the smart input module. Reading these registers directly accesses the data in the EEPROM on the input module. Because these registers hold calibration data, they are protected from inadvertent writes by a write protection scheme which requires a key lock value (170) to be written to the lock register (8439) first before any writes to the EEPROM are enabled. Any calibration changes written to these registers are only transferred to the smart input module at power on. After recalibration, please re-power the controller to load new calibration data.

The actual function of each register varies depending on which smart input module is being used. Some smart input modules do not use these registers and any attempt to access them produces an error (null) response. If there are no smart input modules operating, these registers cannot be read.

Some older smart input modules are not fitted with an EEPROM. If the controller cannot find an EEPROM on the smart input module it use the default value of 1.0 for the each of the calibration registers.

2.27.4 Smart Module 3 Setup

The following table shows the setup registers available for the smart input module 3 when used with a smart input expander module.

NOTE: The following registers are only available when using an input expander module with a smart input module fitted to slot 3. A read of some of these registers will produce a read error if the smart input module is not connected.

Name	Description	Symbol Type	Register Number	Memory Type
SMART3_CAL1	32-bit floating point register. Calibration constant 1 in smart input module 3.	F_32	1171	EEPROMinputModule/RAMinputModule 8
SMART3_CAL2	32-bit floating point register. Calibration constant 2 in smart input module 3.	F_32	1173	EEPROMinputModule/RAMinputModule 8
SMART3_CAL3	32-bit floating point register. Calibration constant 3 in smart input module 3.	F_32	1175	EEPROMinputModule/RAMinputModule 8
SMART3_CAL4	32-bit floating point register. Calibration constant 4 in smart input module 3.	F_32	1177	EEPROMinputModule/RAMinputModule 8
SMART3_RESET_OFFSET1	32-bit register sets the reset offset 1 for smart input module 3.	S_32	613	EEPROM/RAMinputModule 8
SMART3_RESET_OFFSET2	32-bit register sets the reset offset 2 for smart input module 3.	S_32	615	EEPROM/RAMinputModule 8
SMART3_SETUP1	8-bit register. Setup 1 data for smart input module 3. (Note, the display is in octal)	O_8 107	8534	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8
SMART3_SETUP2	8-bit register. Setup 2 data for smart input module 3. (Note, the display is in octal)	O_8 107	8535	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8
SMART3_SETUP3	8-bit register. Setup 3 data for smart input module 3. (Note, the display is in octal)	O_8 107	8536	RAM/EEPROM/RAMinputModule/EEPROMinputModule 8

Read Only

SMART3_ID	8 bit unsigned read only register. Defines the type of smart input module currently in use in slot 3 of the input expander module.	U_8_R	8533	RAM/RAMinputModule 8
SMART3_VERSION 235	8-bit unsigned read only register shows version number of smart input module 3.	U_8_R	8537 235	RAM/RAMinputModule 8
INPUT_MODULE3_CAL_DATE	32-bit unsigned read only register shows the calibration date for smart input module 3. (Date format is DDMMYY)	U_32_R	695	EEPROMinputModule 8
INPUT_MODULE3_CONFIGURATION	8-bit unsigned read only register shows the configuration type option for smart input module 3.	U_8_R	8558	EEPROMinputModule 8
INPUT_MODULE3_NO	16-bit unsigned read only register shows the PCB board number for smart input module 3.	U_16_R	4523	EEPROMinputModule 8
INPUT_MODULE3_REVISION	8-bit unsigned read only register shows the PCB board number revision code for smart input module 3. (ASCII format 'A'=65 to 'Z'=90)	U_8_R	8553	EEPROMinputModule 8
INPUT_MODULE3_SERIAL_NO	32-bit unsigned read only register shows the serial number for smart input module 3.	U_32_R	685	EEPROMinputModule 8

Registers 613 & 615 – Reset Offset Registers

Registers 613 & 615 are reset offset registers for smart input module 3. They are used with certain

types of smart input modules. A write to register 613 or 615 causes the smart input module 3 to reset result register 1 or result register 2 respectively. If the data value written to these registers is zero, then the smart input module writes a zero into the appropriate result register. If the data value written to these registers is other than zero, then the smart input module treats this as an offset and subtracts the data value from the appropriate result register.

A read of these registers reads the last offset value that was written to the input module.

NOTE: These registers are only for use with selected types of smart input module installed in the controller.

Register 1171 to 1177 – Smart Input Module Calibration Registers
Registers 1171 to 1177 are 32-bit floating point registers used to store calibration values for smart input module 3. This calibration data is stored in an EEPROM on the smart input module. This data is read by the controller at power on and transferred to the smart input module. Reading these registers directly accesses the data in the EEPROM on the input module. Because these registers hold calibration data, they are protected from inadvertent writes by a write protection scheme which requires a key lock value (170) to be written to the lock register (8439) first before any writes to the EEPROM are enabled. Any calibration changes written to these registers are only transferred to the smart input module at power on. After recalibration, please re-power the controller to load new calibration data.

The actual function of each register varies depending on which smart input module is being used. Some smart input modules do not use these registers and any attempt to access them produces an error (null) response. If there are no smart input modules operating, these registers cannot be read.

Some older smart input modules are not fitted with an EEPROM. If the controller cannot find an EEPROM on the smart input module it use the default value of 1.0 for the each of the calibration registers.

2.28 Status Registers

Apart from configuration and working registers the controller also contains various status registers which contain flags relating to key functions in the controllers operation. Some flags are read only while others also allow the user to take control of functions remotely or via the macro.

Register 4099 Relay Output Image
This 16-bit register shows the current status of the relays after setpoint processing has been done. The difference between this register and register 239 (alarm status) is that the alarm status register shows the current status of the setpoints as opposed to the relays. For example a setpoint may have been activated but the relay may not yet be turned on because of a 10 second delay on make. In this case, reading alarm status register 239 shows the setpoint as active, but reading 4099 would show that the relay had not yet turned ON.

Writing To Register 4099
Register 4099 is normally used to read the current status of a relay as it is controlled by the setpoint logic. However it is possible to write directly to these flags as well, but the user should first understand the operation of the setpoint logic thoroughly before doing so. Under normal situations the setpoint logic calculates the new status of the relays at the control output rate (10mS or 100mS) and writes a new value to register 4099. This means that writes to this register from any other sources (macro or serial ports) will be overwritten by the setpoint logic. If you wish to write directly to a relay you should ensure that the setpoint logic for that relay is disabled by setting the setpoint source to "OFF". This means that all of the setpoint functions will be disabled and the relay will be totally under your control.

Speed Of Write To 4099

Under normal conditions (excluding PID and high resolution modes) the relay states are only updated at the control output rate. However, for setpoints 1 - 6, a write from the macro or serial port to register 4099 will cause the new value to appear directly on the relay output pins. (This only applies to code versions 4.01e and later).

Name	Description	Symbol Type	Register Number	Memory Type
RELAY_STATUS	16-bit register contains flags showing the instantaneous status of each relay (Note, this may be different to the setpoint status).	U_16	4099	RAM 8

The function of each bit of register 4099 is shown as follows.

Bit	Name	Description	Function
0	RELAY1	Flag shows the instantaneous state of relay 1 (ON = energized).	0 = OFF 1 = ON
1	RELAY2	Flag shows the instantaneous state of relay 2 (ON = energized).	0 = OFF 1 = ON
2	RELAY3	Flag shows the instantaneous state of relay 3 (ON = energized).	0 = OFF 1 = ON
3	RELAY4	Flag shows the instantaneous state of relay 4 (ON = energized).	0 = OFF 1 = ON
4	RELAY5	Flag shows the instantaneous state of relay 5 (ON = energized).	0 = OFF 1 = ON
5	RELAY6	Flag shows the instantaneous state of relay 6 (ON = energized).	0 = OFF 1 = ON
6	RELAY7	Flag shows the instantaneous state of relay 7 (ON = energized).	0 = OFF 1 = ON
7	RELAY8	Flag shows the instantaneous state of relay 8 (ON = energized).	0 = OFF 1 = ON
8	RELAY9	Flag shows the instantaneous state of relay 9 (ON = energized).	0 = OFF 1 = ON
9	RELAY10	Flag shows the instantaneous state of relay 10 (ON = energized).	0 = OFF 1 = ON
10	RELAY11	Flag shows the instantaneous state of relay 11 (ON = energized).	0 = OFF 1 = ON
11	RELAY12	Flag shows the instantaneous state of relay 12 (ON = energized).	0 = OFF 1 = ON
12 to 15	-	Reserved for future development.	Don't Care.

Register 4108

Name	Description	Symbol Type	Register Number	Memory Type
DIGITAL_IO 146	16-bit register contains flags for the digital inputs on the top of the controller.	U_16	4108 146	RAM 8

Register 4110 Remote LED Annunciators

Register 4110 is a 16-bit register that allows the user to take remote control of the LED annunciators on the front panel via the macro or the serial port. A read of this register will always show the current LED status of all LEDs whether in normal or remote mode.

Note: Some display options do not support all 6 LED annunciators. (See 1602 LCD Display 197)

Name	Description	Symbol Type	Register Number	Memory Type
ANNUNCIATORS	16-bit register contains status & control flags for the annunciator LEDs.	U_16	4110	RAM 8

The function of each bit of register 4110 is shown as follows:

Bit	Name	Description	Function
0	LED6	Bit shows/controls the status of the annunciator LED 6. (See 1602 LCD Display 197)	0 = OFF 1 = ON
1	LED5	Bit shows/controls the status of the annunciator LED 5. (See 1602 LCD Display 197)	0 = OFF 1 = ON
2	LED4	Bit shows/controls the status of the annunciator LED 4.	0 = OFF 1 = ON
3	LED3	Bit shows/controls the status of the annunciator LED 3.	0 = OFF 1 = ON
4	LED2	Bit shows/controls the status of the annunciator LED 2.	0 = OFF 1 = ON
5	LED1	Bit shows/controls the status of the annunciator LED 1.	0 = OFF 1 = ON
6 & 7	-	-	Don't care
8	REMOTE_LED6	Bit sets the control mode for LED 6. (See 1602 LCD Display 197)	ON = Remote Control OFF = controller Control
9	REMOTE_LED5	Bit sets the control mode for LED 5. (See 1602 LCD Display 197)	ON = Remote Control OFF = controller Control
10	REMOTE_LED4	Bit sets the control mode for LED 4.	ON = Remote Control OFF = controller Control
11	REMOTE_LED3	Bit sets the control mode for LED 3.	ON = Remote Control OFF = controller Control
12	REMOTE_LED2	Bit sets the control mode for LED 2.	ON = Remote Control OFF = controller Control
13	REMOTE_LED1	Bit sets the control mode for LED 1.	ON = Remote Control OFF = controller Control
14 & 15	-	Reserved for future use	These bits should be written as 0

1602 LCD Display

The 16x2 LCD display option only supports 4 annunciator LEDs (LED's 1 - 4). The controls for LEDs 5&6 are used to control the buzzer and the backlight respectively. The table below shows the function of register 4110 for the 1602 display option.

Bit	Name	Description	Function
0	BACKLIGHT	Bit shows/controls the status of the backlight for the 1602 LCD display.	0 = OFF 1 = ON Default = ON
1	BUZZER	Bit shows/controls the status of the buzzer for the 1602 LCD display	0 = OFF 1 = ON
2	LED4	Bit shows/controls the status of the annunciator LED 4.	0 = OFF 1 = ON
3	LED3	Bit shows/controls the status of the annunciator LED 3.	0 = OFF 1 = ON
4	LED2	Bit shows/controls the status of the annunciator LED 2.	0 = OFF 1 = ON
5	LED1	Bit shows/controls the status of the annunciator LED 1.	0 = OFF 1 = ON
6 & 7	-	-	Don't care
8	REMOTE_BACKLIGHT	Bit sets the control mode for the backlight on the 1602 LCD display.	ON = Remote Control OFF = controller Control Default = ON (Remote Control)
9	REMOTE_BUZZER	Bit sets the control mode for the buzzer on the 1602 LCD display.	ON = Remote Control OFF = controller Control Default = ON (Remote Control)
10	REMOTE_LED4	Bit sets the control mode for LED 4.	ON = Remote Control OFF = controller Control
11	REMOTE_LED3	Bit sets the control mode for LED 3.	ON = Remote Control OFF = controller Control
12	REMOTE_LED2	Bit sets the control mode for LED 2.	ON = Remote Control OFF = controller Control
13	REMOTE_LED1	Bit sets the control mode for LED 1.	ON = Remote Control OFF = controller Control
14 & 15	-	Reserved for future use	These bits should be written as 0

NOTE: If the remote bits for the backlight and the buzzer are set to OFF, then the buzzer and backlight will be controlled by setpoints 5&6 respectively.

Other Status Registers

The table below shows other status registers available in the ICC402.

Name	Description	Symbol Type	Register Number	Memory Type
ALARM_STATUS 207	32 bit value showing status and allowing control of setpoints.	U_32	239 207	RAM 8
EEPROM_MEMORY_SIZE	16-bit register shows how much EEPROM memory is fitted in the controller (value in Kbytes).	U_16_R	4437	RAM 8
RELAY_DE_ENERGIZE_FLAGS 160	16-bit register holds the de-energized status for relays.	U_16	4101 160	RAM 8
SETPOINT_STATUS_FLAGS 164	16 bit read only register shows the status of the setpoints	U_16_R	4097 164	RAM 8
DIGITAL_IO 140	16-bit register contains flags for the digital inputs on the top of the controller.	U_16	4108 140	RAM 8
STATE	16-bit register. Used to control the state number of the macro (cleared to 0 when returning to operational display).	U_16	4109	RAM 8
VIEW_POINTER	8-bit register indicates which value is currently being displayed in view mode (i.e. pressing UP or DOWN button).	U_8	8220	RAM 8
TEMPERATURE	8-bit signed register indicates the internal temperature of the module in degrees C.	S_8	8221	RAM 8
VIEW_MODE_BLANKING 201	16-bit register. Used to control which parameters are displayed when in view mode	U_16	4436 201	EEPROM 8
LAST_ERROR 202	8 bit register shows the last error encountered by the controller during power up.	U_8	8431 202	EEPROM 8
ERROR_STATUS 202	8 bit register shows the current error status of the controller during power up (0=no errors).	U_8	8435 202	RAM 8
SOFTWARE_VERSION_NO 235	16-bit register which displays the software version number currently operating the controller.	U_16_R	4106 235	RAM 8
DEVICE_TYPE 235	Text register which displays the device type	L_8_T	16565 235	RAM 8
SERIAL_NO 235	32 bit read only register. Contains product serial number (V4.02a onwards)	U_32_R	541 235	EEPROM 8

See also
Status Switches 206

2.28.1 View Mode Blanking

Register 4436 is a 16 bit register in EEPROM which controls the sequence of parameters which can be viewed when the "Up" or "Down" buttons are pressed. Each bit in the register controls a specific parameter display as shown below. If a specific bit is a "0" then the display of the associated parameter is disabled and that parameter will be skipped over. If a bit is a "1" the parameter will be displayed.

Down Button:

Bit 0 = Setpoint 2
Bit 1 = Setpoint 4
Bit 2 = Setpoint 6
Bit 3 = Valley
Bit 4 = Total 2
Bit 5 = Channel 2
Bit 6 = Channel 4
Bit 7 = Enable view mode macro on Down button

Up Button:

Bit 8 = Setpoint 1
Bit 9 = Setpoint 3
Bit 10 = Setpoint 5
Bit 11 = Peak
Bit 12 = Total 1
Bit 13 = Channel 1
Bit 14 = Channel 3
Bit 15 = Enable view mode macro on Up button

When register 4436 is read it will be displayed as a 16 bit signed number. The default value will be 32639 (0x7F7F hex) which is all codes enabled except the view macro.

2.28.2 Error Status

Last Error - Register 8431
Register 8431 is an 8 bit unsigned register which records the first error encountered after power up. Register 8431 is also stored in non volatile memory so that it can be viewed even after a power down. Register 8431 can be read or written to for clearing an error number if required, but unlike register 8435 below, clearing register 8431 has no other function. It will not remove or fix any error condition or enable the controller to continue operating. It's purpose is purely for a diagnostics tool. See the table below for an explanation of the error codes.

Current Error Status at Power up - Register 8435
Register 8435 is an 8 bit unsigned register which reports the current status of any errors at power up. It is useful for controller models which do not have a display or are run with a different display option. After power up with no errors this register will display a value of zero.

Writing a value of 0 to this register will clear the error condition and cause the controller to continue it's power up procedure, but it will not necessarily fix the cause of the error. Errors of 100 or greater will we cleared and the controller will restore th factory default values to the effected registers. This will mean that any previously stored data in these registers will be lost and they may need to be setup again. The remedy for each error condition will vary widely. If you encounter one of these errors you should contact your supplier for advice, quoting the error code.

The table below shows the possible error conditions and suggested causes.

Error #	Error Description
0	Successful power up with no errors.
1	The controller has attempted to read the EEPROM but the SCL line has been held low. This could be a device busy but is more likely to be a hardware fault. (check for shorts and check pull up resistor on SCL)
2	The controller has attempted to read the EEPROM but didn't receive any acknowledgement that it was there. Either there is no EEPRPOM installed in the required position or the wrong type of EEPROM has been installed (must be 24xC128 or greater)
3	During a read attempt of EEPROM the controller received a negative acknowledge error.
4	The controller managed to read data out of the EEPROM but the checksum for the data was incorrect.
5	Memory size error – checksum is ok but the memory size is either 0 or some other value which is not a valid size for this controller.
6	Checksum error when reading setups Cal – Code10.
7	Checksum error when reading brightness.
8	Checksum error when reading Model selection byte.
9	Checksum error when reading baud rate & serial address.
10	Checksum error when reading scaling data.
11	Checksum error when reading offset data.
12	Checksum error when reading hidden scaling.
13	Checksum error when reading set-point control.
14	Checksum error when reading set-point tracking/reset.
15	Checksum error when reading K factor data.
16	Checksum error when reading D/A calibration low data.
17	Checksum error when reading D/A calibration high data.
18	Checksum error when reading set-point data.
19	Checksum error when reading hysteresis data.

20 Checksum error when reading delay_on_make data.

21 Checksum error when reading delay_on_beak data.

22 Checksum error when reading data for delay type.

23 Checksum error when reading display format data.

24 Checksum error when reading display text character data.

25 Checksum error when reading set-point reset values.

26 Checksum error when reading bar graph data.

27 Checksum error when reading pre-scaler values for Ch1 & CH2 counter.

28 Checksum error when reading display source values.

29 Checksum error when reading analog output source values.

30 Checksum error when reading totalisator source values.

31 Checksum error when reading setpoint source values.

32 Checksum error when reading channel source values.

33 Checksum error when reading peak/valley source values.

34 Checksum error when reading reset destination values.

35 Checksum error when reading RTD gain values.
Note: These registers are initialized in the factory during testing. They are not changed when the controller is re-initialized. To clear this error you need to write any value to one of the RTD gain registers via the serial port. Then you will need to recalibrate the RTD input channels.

36 Checksum error when reading thermocouple calibration data.
Note: These registers are initialized in the factory during testing. They are not changed when the controller is re-initialized. To clear this error you need to write any value to one of the thermocouple gain registers via the serial port. Then you will need to recalibrate the thermocouple input channels.

37 Checksum error when reading PID cycle time data.

38 Checksum error when reading PID anti reset windup data.

39 Checksum error when reading auto zero window values

40 Checksum error when reading auto zero sample values

41 Checksum error when reading averaging window values

42 Checksum error when reading averaging sample values

43 Checksum error when reading smart setup values

44 Checksum error when reading auto zero motion band values

45 Checksum error when reading setpoint trigger values

46 Checksum error when reading End Of String character

47 Checksum error when reading data log register pointers

48 Checksum error when reading data log read size

49 EEPROM memory size has changed from the size originally installed in the controller. It now has no internal data logging memory installed. To accept this change, press the Prog button and the new memory size will be saved.

52 Checksum error when reading user memory band parameters

60 Checksum error when reading linearization table data

100 RAM error

101 Checksum error when reading Auxiliary and Floating point variables from on board FLASH

102 Checksum error when reading user memory from on board FLASH

103 Checksum error when reading auto zero tare values from on board FLASH

104 Checksum error when reading data log write pointer from on board FLASH

105 Checksum error when reading data log read pointer from on board FLASH

200 No real time clock device detected but internal data memory has been installed. This combination is not a standard option and suggests that the RTC device may be installed but not operating correctly.

201 Oscillator error on internal real time clock

202 Oscillator error on external real time clock

203 Data range error on external real time clock

204 A checksum error has been found when trying to read the data logging read/write pointers from the SD card. This does not necessarily mean that all the logged data on the SD card has been lost but you will have to reset either the read or write pointer (or both) in order to access any data which is still held on the card. You should do this before logging any new data as it may overwrite previously stored samples. For more information on which pointers are corrupted see RTC status 78.

205 Incompatible card type. It may be that the card is too large or too small (128 MB or 256 MB is the preferred size) or it may be that the card is the wrong type. Also make sure that the contacts on the card are clean and free of dust and other residue.

206 Unformatted card. The SD card has not been formatted correctly. Please ensure that the card contains a valid Master Boot Record (MBR) with its partition table and that the first partition is of type FAT16 or FAT32 (FAT16 preferred). Some cards are shipped without a partition table by default, instead starting the FAT root sector in the first sector of the card. A partition table can be created with fdisk (this will erase all data on the card) and then formatting the newly created partition. Also make sure that there is enough linear free space available on the card. Using a freshly formatted card is recommended, but if some files are already on the card we suggest to defrag the file system before inserting the card into the data logger module for the first time. The data logger will leave some free space at the end of the card, this is normal and intended. Please allow up to 60 seconds when inserting a card for the first time for the data logger to create the file used to store the samples.

207 Card not ready. The data logger cannot log data because either there is no card inserted or because a new card has been inserted and the data logger is busy creating the required file structure.

2.28.3 Status Switches

The following registers allow access to the status of the front panel switches and the rear pins which can be used with external switches.

Name	Description	Symbol Type	Register Number	Memory Type
SWITCHES 215	32-bit register contains flags for front panel switches.	U_32	247 215	RAM 8
DIGITAL_IO 140	16 bit register contains flags for the digital inputs on the top of the controller.	U_16	4108 140	RAM 8

See also

Switches 215

Status Registers 197

2.28.4 Register 239 - Alarm Status

Register 239 is a 32-bit register that contains flags to indicate the status of the 12 setpoints and rising or falling tendency. In the normal mode of operation, the status of each setpoint is controlled by the controller, based on a comparison between the input value and the setpoint value. Each setpoint can be individually placed into remote mode by setting the appropriate mode control bit (bits 8 to 13 and 24 to 29).

In remote mode, the input value and setpoint values have no effect on the setpoint status. Instead, the status of the setpoint is controlled directly by setting or clearing the appropriate status bit. This can be done from the serial port or from a macro.

Note: See Alarm Status 16 bit 207 for information about accessing these status bits via two 16 bit registers instead of one 32 bit register.

See also

Alarm Status Read 209

Alarm Status Write 211

Setpoint Status Flags 164

2.28.4.1 Alarm Status 16 bit

When using Modbus communications it is sometimes difficult to access 32 bit registers so the Alarm Status register can also be accessed by two 16 bit registers. Registers 4507 and 4508 duplicate the functions of the Alarm Status register 239 but allow it to be accessed in two 16 bit words instead of one 32 bit register.

Register 4507 - Alarm Status Low

Register 4507 is a 16 bit register which allows access to status/control flags for setpoints 1 - 6. The bit functions for register 4507 are shown below.

Bit	Name	Description	Function
0	SP1	This flag shows/controls the status of setpoint 1	0 = OFF 1 = ON
1	SP2	This flag shows/controls the status of setpoint 2	0 = OFF 1 = ON
2	SP3	This flag shows/controls the status of setpoint 3	0 = OFF 1 = ON
3	SP4	This flag shows/controls the status of setpoint 4	0 = OFF 1 = ON
4	SP5	This flag shows/controls the status of setpoint 5	0 = OFF 1 = ON
5	SP6	This flag shows/controls the status of setpoint 6	0 = OFF 1 = ON
6	TREND_UP	When On, this flag indicates an **upward** trend in the display data value.	1 = Rising
7	TREND_DOWN	When On, this flag indicates a **downward** trend in the display data value.	1 = Falling
8	SP1_REMOTE	Setting this bit to ON places setpoint 1 in remote mode. (If SP1 is operating in PID mode then setting this bit causes the PID1 to operate in manual mode and clearing this bit causes PID1 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual mode for PID operation)
9	SP2_REMOTE.	Setting this bit to ON places setpoint 1 in remote mode. (If SP1 is operating in PID mode then setting this bit causes the PID1 to operate in manual mode and clearing this bit causes PID1 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual mode for PID operation)
10	SP3_REMOTE.	Setting this bit to ON places setpoint 1 in remote mode. (If SP1 is operating in PID mode then setting this bit causes the PID1 to operate in manual mode and clearing this bit causes PID1 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual mode for PID operation)
11	SP4_REMOTE	Setting this bit to ON places setpoint 1 in remote mode. (If SP1 is operating in PID mode then setting this bit causes the PID1 to operate in manual mode and clearing this bit causes PID1 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual mode for PID operation)
12	SP5_REMOTE	Setting this bit to ON places setpoint 1 in remote mode. (If SP1 is operating in PID mode then setting this bit causes the PID1 to operate in manual mode and clearing this bit causes PID1 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual mode for PID operation)
13	SP6_REMOTE	Setting this bit to ON places setpoint 1 in remote mode. (If SP1 is operating in PID mode then setting this bit causes the PID1 to operate in manual mode and clearing this bit causes PID1 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual mode for PID operation)
14 & 15	-	Reserved for future use. These bits are always read as 0.	-

Register 4508 - Alarm Status High

Register 4508 is a 16 bit register which allows access to status/control flags for setpoints 7-12. The bit functions for register 4508 are shown below.

Bit	Name	Description	Function
0	SP7	This flag shows/controls the status of setpoint 7	0 = OFF 1 = ON
1	SP8	This flag shows/controls the status of setpoint 8	0 = OFF 1 = ON
2	SP9	This flag shows/controls the status of setpoint 9	0 = OFF 1 = ON
3	SP10	This flag shows/controls the status of setpoint 10	0 = OFF 1 = ON
4	SP11	This flag shows/controls the status of setpoint 11	0 = OFF 1 = ON
5	SP12	This flag shows/controls the status of setpoint 12	0 = OFF 1 = ON
6 & 7	-	Reserved for future use. These bits are always read as 0.	-
8	SP7_REMOTE	Setting this bit to ON places setpoint 7 in remote mode.	0 = Normal Mode 1 = Remote Mode
9	SP8_REMOTE	Setting this bit to ON places setpoint 8 in remote mode.	0 = Normal Mode 1 = Remote Mode
10	SP9_REMOTE	Setting this bit to ON places setpoint 9 in remote mode.	0 = Normal Mode 1 = Remote Mode
11	SP10_REMOTE	Setting this bit to ON places setpoint 10 in remote mode.	0 = Normal Mode 1 = Remote Mode
12	SP11_REMOTE	Setting this bit to ON places setpoint 11 in remote mode.	0 = Normal Mode 1 = Remote Mode
13	SP12_REMOTE	Setting this bit to ON places setpoint 12 in remote mode.	0 = Normal Mode 1 = Remote Mode
14 & 15	-	Reserved for future use. These bits are always read as 0.	-

2.28.4.2 Alarm Status Read

The following table shows the function of each bit when **reading** the alarm status.

Bit	Name	Description	Function
0	SP1	This flag shows/controls the status of setpoint 1	0 = OFF 1 = ON
1	SP2	This flag shows/controls the status of setpoint 2	0 = OFF 1 = ON
2	SP3	This flag shows/controls the status of setpoint 3	0 = OFF 1 = ON
3	SP4	This flag shows/controls the status of setpoint 4	0 = OFF 1 = ON
4	SP5	This flag shows/controls the status of setpoint 5	0 = OFF 1 = ON
5	SP6	This flag shows/controls the status of setpoint 6	0 = OFF 1 = ON
6	TREND_UP	When On, this flag indicates an **upward** trend in the display data value.	1 = Rising
7	TREND_DOWN	When On, this flag indicates a **downward** trend in the display data value.	1 = Falling
8	SP1_REMOTE	When this bit is ON setpoint 1 is in remote mode. (If setpoint 1 is operating in PID mode then PID 1 is in manual mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
9	SP2_REMOTE.	When this bit is ON setpoint 2 is in remote mode. (If setpoint 1 is operating in PID mode then PID 2 is in manual mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
10	SP3_REMOTE.	When this bit is ON setpoint 3 is in remote mode. (If setpoint 1 is operating in PID mode then PID 3 is in manual mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
11	SP4_REMOTE	When this bit is ON setpoint 4 is in remote mode. (If setpoint 1 is operating in PID mode then PID 4 is in manual mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
12	SP5_REMOTE	When this bit is ON setpoint 5 is in remote mode. (If setpoint 1 is operating in PID mode then PID 5 is in manual mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
13	SP6_REMOTE	When this bit is ON setpoint 6 is in remote mode. (If setpoint 1 is operating in PID mode then PID 6 is in manual mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
14 & 15	-	Reserved for future use. These bits are always read as 0.	-

16	SP7	This flag shows/controls the status of setpoint 7	0 = OFF 1 = ON
17	SP8	This flag shows/controls the status of setpoint 8	0 = OFF 1 = ON
18	SP9	This flag shows/controls the status of setpoint 9	0 = OFF 1 = ON
19	SP10	This flag shows/controls the status of setpoint 10	0 = OFF 1 = ON
20	SP11	This flag shows/controls the status of setpoint 11	0 = OFF 1 = ON
21	SP12	This flag shows/controls the status of setpoint 12	0 = OFF 1 = ON
22 & 23	-	Reserved for future use. These bits are always read as 0.	-
24	SP7_REMOTE	When this bit is ON setpoint 7 is in remote mode.	0 = Normal Mode 1 = Remote Mode
25	SP8_REMOTE	When this bit is ON setpoint 8 is in remote mode.	0 = Normal Mode 1 = Remote Mode
26	SP9_REMOTE	When this bit is ON setpoint 9 is in remote mode.	0 = Normal Mode 1 = Remote Mode
27	SP10_REMOTE	When this bit is ON setpoint 10 is in remote mode.	0 = Normal Mode 1 = Remote Mode
28	SP11_REMOTE	When this bit is ON setpoint 11 is in remote mode.	0 = Normal Mode 1 = Remote Mode
29	SP12_REMOTE	When this bit is ON setpoint 12 is in remote mode.	0 = Normal Mode 1 = Remote Mode
30 & 31	-	Reserved for future use. These bits are always read as 0.	-

NOTE: Bits 0 to 5 and 16 to 21 indicate the setpoint status only, not the relay status. Setpoint timer and manual reset settings could cause the relay status to be different from the setpoint status. For relay status, see register number 4099 197.

See also
Alarm Status Write 211

2.28.4.3 Alarm Status Write

The following table shows the function of each bit when **writing** to the alarm status.

Bit	Name	Description	Function
0	SP1	This flag shows/controls the status of setpoint 1	Setpoint Mode (with SP1_REMOTE on) 0 = OFF 1 = ON PID Mode (with SP1_REMOTE on) 0 = Output off 1 = Manual output
1	SP2	This flag shows/controls the status of setpoint 2	Setpoint Mode (with SP2_REMOTE on) 0 = OFF 1 = ON PID Mode (with SP2_REMOTE on) 0 = Output off 1 = Manual output
2	SP3	This flag shows/controls the status of setpoint 3	Setpoint Mode (with SP3_REMOTE on) 0 = OFF 1 = ON PID Mode (with SP3_REMOTE on) 0 = Output off 1 = Manual output
3	SP4	This flag shows/controls the status of setpoint 4	Setpoint Mode (with SP4_REMOTE on) 0 = OFF 1 = ON PID Mode (with SP4_REMOTE on) 0 = Output off 1 = Manual output
4	SP5	This flag shows/controls the status of setpoint 5	Setpoint Mode (with SP5_REMOTE on) 0 = OFF 1 = ON PID Mode (with SP5_REMOTE on) 0 = Output off 1 = Manual output
5	SP6	This flag shows/controls the status of setpoint 6	Setpoint Mode (with SP6_REMOTE on) 0 = OFF 1 = ON PID Mode (with SP6_REMOTE on) 0 = Output off 1 = Manual output
6	Don't Care	Read only bits. See Alarm Status Read 209	-
7	Don't Care	Read only bits. See Alarm Status Read 209	-

8	SP1_REMOTE	Setting this bit to ON places setpoint 1 in remote mode. (If SP1 is operating in PID mode then setting this bit causes the PID1 to operate in manual or off mode and clearing this bit causes PID1 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
9	SP2_REMOTE.	Setting this bit to ON places setpoint 2 in remote mode. (If SP2 is operating in PID mode then setting this bit causes the PID2 to operate in manual or off mode and clearing this bit causes PID2 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
10	SP3_REMOTE.	Setting this bit to ON places setpoint 3 in remote mode. (If SP3 is operating in PID mode then setting this bit causes the PID3 to operate in manual or off mode and clearing this bit causes PID3 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
11	SP4_REMOTE	Setting this bit to ON places setpoint 4 in remote mode. (If SP4 is operating in PID mode then setting this bit causes the PID4 to operate in manual or off mode and clearing this bit causes PID4 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
12	SP5_REMOTE	Setting this bit to ON places setpoint 5 in remote mode. (If SP5 is operating in PID mode then setting this bit causes the PID5 to operate in manual or off mode and clearing this bit causes PID5 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
13	SP6_REMOTE	Setting this bit to ON places setpoint 6 in remote mode. (If SP6 is operating in PID mode then setting this bit causes the PID6 to operate in manual or off mode and clearing this bit causes PID6 to revert back to auto mode)	0 = Normal Mode (auto mode for PID operation) 1 = Remote Mode (manual or zero output mode for PID operation)
14 &15	-	Reserved for future use. These bits should be written as 0.	-

16	SP7	This flag shows/controls the status of setpoint 7	0 = OFF 1 = ON
17	SP8	This flag shows/controls the status of setpoint 8	0 = OFF 1 = ON
18	SP9	This flag shows/controls the status of setpoint 9	0 = OFF 1 = ON
19	SP10	This flag shows/controls the status of setpoint 10	0 = OFF 1 = ON
20	SP11	This flag shows/controls the status of setpoint 11	0 = OFF 1 = ON
21	SP12	This flag shows/controls the status of setpoint 12	0 = OFF 1 = ON
22 & 23	-	Reserved for future use. These bits should be written as 0.	-
24	SP7_REMOTE	Setting this bit to ON places setpoint 7 in remote mode.	0 = Normal Mode 1 = Remote Mode
25	SP8_REMOTE	Setting this bit to ON places setpoint 8 in remote mode.	0 = Normal Mode 1 = Remote Mode
26	SP9_REMOTE	Setting this bit to ON places setpoint 9 in remote mode.	0 = Normal Mode 1 = Remote Mode
27	SP10_REMOTE	Setting this bit to ON places setpoint 10 in remote mode.	0 = Normal Mode 1 = Remote Mode
28	SP11_REMOTE	Setting this bit to ON places setpoint 11 in remote mode.	0 = Normal Mode 1 = Remote Mode
29	SP12_REMOTE	Setting this bit to ON places setpoint 12 in remote mode.	0 = Normal Mode 1 = Remote Mode
30 & 31	-	Reserved for future use. These bits should be written as 0.	-

NOTE: Bits 0 to 5 and 16 to 21 indicate the setpoint status only, not the relay status. Setpoint timer and manual reset settings could cause the relay status to be different from the setpoint status. For relay status, see register number 4099 197.

If setpoints are configured to operate in PID mode (setpoints 1 - 6 only) they will start up in PID auto mode at power on. Each PID loop can then be switched to operate in manual mode or PID off mode by setting the appropriate |SPx_REMOTE bit and |SPx bits in the ALARM_STATUS register.

The following table shows the different options available.

|SPx_REMOTE bit	|SPx bit	PID function
Off (0)	Don't care	Auto Mode - normal mode of operation where PID output is under control of the PID algorithm. (Default mode at power on)
On(1)	On(1)	Manual Output Mode - manual mode of operation where PID output is controlled by manually writing to PIDx_MAN_OUT register.
On(1)	Off (0)	Off Mode - PID algorithm and output are turned off, PID registers (PIDx_ERRD_OLD, PIDx_INTEGRAL_TERM, PIDx_OUTPUT) are reset to zero.

See also
Alarm Status Read 209

2.29 Switches

Register 247 Serial Switch Control

Register 247 is a 32-bit register used to read the status of the switches or to disable the front panel switches and manually control switch depressions from the serial port.

Name	Description	Symbol Type	Register Number	Memory Type
SWITCHES	32-bit register contains flags for front panel switches.	U_32	247	RAM 8

The function of each bit is shown in the table below. Please note that bits 0 - 6 & bit 15 are standard on all display models, but the functionality of bits 7 - 14 and 16 - 31 will only be available on specific models of display.

Bit	Name	Description	Function
0	PROGRAM_BUTTON	Flag shows the current status of the program button (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
1	DOWN_BUTTON	Flag shows the current status of the DOWN button (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
2	UP_BUTTON	Flag shows the current status of the UP button (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
3	LOCK_UP_SWITCH	Flag shows the current status of lock switch 2 (ON = main program locked).	0 = OFF (open), 1 = ON (closed)
4	LOCK_DOWN_SWITCH	Flag shows the current status of lock switch 1 (ON = setpoint access locked).	0 = OFF (open), 1 = ON (closed)
5	F1_BUTTON	Flag shows the current status of the function 1 button (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
6	F2_BUTTON	Flag shows the current status of the function 2 button (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
7	F3_BUTTON	Flag shows the current status of the function 3 button (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
8	F4_BUTTON	Flag shows the current status of the function 4 button (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
9	F5_BUTTON	Flag shows the current status of the function 5 button (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
10 to 14		Reserved for future use. These bits are always read as 0.	
15	REMOTE_SWITCH	When set, this bit disables operation of front panel switches bits0 - 7.	0 = Normal Mode, 1 = Remote Switch Mode

16	NUMBER1_BUTTON	Flag shows the current status of the numeric button 1 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
17	NUMBER2_BUTTON	Flag shows the current status of the numeric button 2 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
18	NUMBER3_BUTTON	Flag shows the current status of the numeric button 3 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
19	NUMBER4_BUTTON	Flag shows the current status of the numeric button 4 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
20	NUMBER5_BUTTON	Flag shows the current status of the numeric button 5 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
21	NUMBER6_BUTTON	Flag shows the current status of the numeric button 6 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
22	NUMBER7_BUTTON	Flag shows the current status of the numeric button 7 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
23	NUMBER8_BUTTON	Flag shows the current status of the numeric button 8 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
24	NUMBER9_BUTTON	Flag shows the current status of the numeric button 9 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
25	NUMBER0_BUTTON	Flag shows the current status of the numeric button 0 (ON = button pressed).	0 = OFF (open), 1 = ON (closed)
26 to 31		Reserved for future use. These bits are always read as 0.	

A read of register 247 will always show the debounced status of the front panel buttons and lock switches regardless of whether bit 15 is set to remote or normal mode.

Normal Switch Mode

In normal mode all buttons presses are transferred through to the operating system of the controller for processing. A write to bits 0 to 14 or bits 16 to 31 is **not recommended** while in the normal mode and can produce an unreliable results as these bits are also modified by the OS.

Remote Switch Mode

In remote switch mode the state of bits 0 to 7 (the Prog, Up, Down, F1, F2, F3 buttons and lock switches) are not transferred through to the operating system. They can still be obtained by reading register 247 but the controller will not respond to any changes in the normal way. Instead the pressing of buttons is simulated by setting and clearing the appropriate bits with a write to register 247.

NOTE: When operating in remote mode must be manually turned off as well. Leaving a button set to on permanently will cause a scrolling of parameters every 2 seconds.

Function Button Macros

Function buttons F1, F2 and F3 all have dedicated macro subroutines which are called once only each time the button is pressed. These macros are only called when the controller is in it's normal operating mode (i.e. not in editing modes) If any of the editing modes are operational when one of these buttons are pressed, the macro is not called, however the appropriate bit in register 247 is still set enabling the main macro to take an alternative action if required.

Function buttons F4 and F5 and number buttons 0-9 do not have any special macro subroutines. To action these button changes the main macro needs to poll register 247 for changes.

2.30 Timers

The timer registers shown below are software timers that are managed by the operating system of the controller and run continuously in the background with no user intervention required. Apart from being automatically incremented, they are not used by the operating system or any other standard functions

in the controller, so you have complete freedom to use them as required.

The timer count is incremented by the operating system at set intervals (usually 0.1 seconds). Timers can be read or written to from the macro or via the serial port. Timer values will not change within a macro loop (with the exception of the Modbus master macro).

Name	Description	Symbol Type	Register Number	Memory Type
SHORT_TIMER1	16-bit timer counts up every 0.1 seconds (109 minutes). It can be set/reset by macro.	U_16	4111	RAM
SHORT_TIMER2	16-bit timer counts up every 0.1 seconds (109 minutes). It can be set/reset by macro.	U_16	4112	RAM
SHORT_TIMER3	16-bit timer counts up every 0.1 seconds (109 minutes). It can be set/reset by macro.	U_16	4113	RAM
SHORT_TIMER4	16-bit timer counts up every 0.1 seconds (109 minutes). It can be set/reset by macro.	U_16	4114	RAM
FAST_TIMER1	16-bit timer counts up every 0.01 seconds (10.9 minutes). It can be set/reset by macro.	U_16	4471	RAM
FAST_TIMER2	16-bit timer counts up every 0.01 seconds (10.9 minutes). It can be set/reset by macro.	U_16	4472	RAM
FAST_TIMER3	16-bit timer counts up every 0.01 seconds (10.9 minutes). It can be set/reset by macro.	U_16	4473	RAM
FAST_TIMER4	16-bit timer counts up every 0.01 seconds (10.9 minutes). It can be set/reset by macro.	U_16	4474	RAM
TIMER1	32-bit timer counts up every 0.1 seconds (13.6 years) can be set/reset by macro.	U_32	143	RAM
TIMER2	32-bit timer counts up every 0.1 seconds (13.6 years) can be set/reset by macro.	U_32	145	RAM
TIMER3	32-bit timer counts up every 0.1 seconds (13.6 years) can be set/reset by macro.	U_32	147	RAM
TIMER4	32-bit timer counts up every 0.1 seconds (13.6 years) can be set/reset by macro.	U_32	149	RAM

NOTE: If a timer reaches its maximum count it overflows to 0 without any warning so you must ensure that you use the correct timer (long or short) for the task.

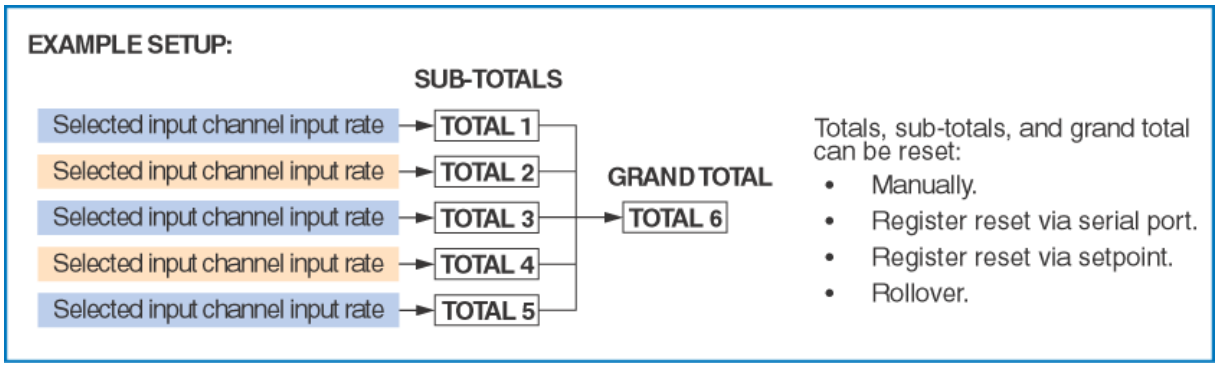
2.31 Total

A totalizer is a user selectable software function of the controller that converts an input rate to an input total over time.

For example:

You have a settling tank being filled with water. The flow rate is metered and input to a ICC402 Series controller. The flow rate indicates the speed at which the volume of water travels past a set point, but not the total volume accumulated in the tank. The controller's totalizer performs this function and provides you with the total amount of water currently in the tank. This then allows you to make control decisions, such as when to turn the tap off before the tank overflows!

Each ICC402 Series controller has six independent totalizers suitable for a wide variety of totaling and batching applications. Each totalizer can operate independently or combine to generate a sub-total and grand total. Totals can be reset using one of a number of methods. Setpoints can be used to reset a sub-total and increment a grand total.



Totalizer Diagram

See also

Total 1 219

Total 2 220

Total 3 221

Total 4 222

Total 5 223

Total 6 224

See Totalizing Supplement (NZ208) for details of totalizer functionality.

NOTE, the Totalizer Supplement describes the totalizer function of Tiger 320 Series controllers. The totalizer function of ICC402 Series controllers is similar, but has six totalizers available.

2.31.1 Total 1

Name	Description	Symbol Type	Register Number	Memory Type
CUTOFF1	16-bit register. Cutoff value for totalizer 1 (range -32768 to 32767).	S_16	4325	RAM/EEPROM 8
DATA_SOURCE_TOTAL1 224	16-bit register holds the register number of the data source for totalizer 1.	U_16	4391 224	RAM/EEPROM 8
DISPLAY_FORMAT_TOTAL1 106	8-bit register controls the display format settings for totalizer 1 (display is in octal 107 format).	O_8 107	8376 106	RAM/EEPROM 8
INPUT_RATE1 224	16-bit unsigned register. Input rate for K factor calculation totalizer 1.	U_16	4305 224	RAM/EEPROM 8
K_FACTOR1 224	32-bit register. K factor for totalizer 1.	S_16	4315 224	RAM/EEPROM 8
RATE_TOTALIZER1 225	8-bit register for totalizer 1 rate calculation & rollover.	U_8	8345 225	RAM/EEPROM 8
TEXT_CHARACTER_TOTAL1 108	8-bit register holds the ASCII value for the last digit text character for totalizer 1 (0 = no character).	U_8	8402 108	RAM/EEPROM 8
TOTAL1	Non-volatile 32-bit register for Totalizer 1 value.	S_32	37	RAM/FLASH 8
TOTAL1_TEXT	Text display for total 1.	L_8	16421	EEPROM 8

See also

K Factor 224

Input Rate Value - Registers 4305 to 4310 224

Totalizer Data Source Selection - Register 4391 to 4396 224

Totalizer Time Period and Rollover - Register 8345 to 8350 225

2.31.2 Total 2

Name	Description	Symbol Type	Register Number	Memory Type
CUTOFF2	16-bit register. Cutoff value for totalizer 2 (range -32768 to 32767).	S_16	4326	RAM/EEPROM 8
DATA_SOURCE_TOTAL2 224	16-bit register holds the register number of the data source for totalizer 2.	U_16	4392 224	RAM/EEPROM 8
DISPLAY_FORMAT_TOTAL2 106	8-bit register controls the display format settings for totalizer 2 (display is in octal 107 format).	O_8 107	8377 106	RAM/EEPROM 8
INPUT_RATE2 224	16-bit unsigned register. Input rate for K factor calculation totalizer 2.	U_16	4306 224	RAM/EEPROM 8
K_FACTOR2 224	32-bit register. K factor for totalizer 2.	S_16	4316 224	RAM/EEPROM 8
RATE_TOTALIZER2 225	8-bit register for totalizer 2 rate calculation & rollover.	U_8	8346 225	RAM/EEPROM 8
TEXT_CHARACTER_TOTAL2 108	8-bit register holds the ASCII value for the last digit text character for totalizer 2 (0 = no character).	U_8	8403 108	RAM/EEPROM 8
TOTAL2	Non-volatile 32-bit register for totalizer 2 value.	S_32	39	RAM/FLASH 8
TOTAL2_TEXT	Text display for total 2.	L_8	16423	EEPROM 8

See also

K Factor 224

Input Rate Value - Registers 4305 to 4310 224

Totalizer Data Source Selection - Register 4391 to 4396 224

Totalizer Time Period and Rollover - Register 8345 to 8350 225

2.31.3 Total 3

Name	Description	Symbol Type	Register Number	Memory Type
CUTOFF3	16-bit register. Cutoff value for totalizer 3 (range -32768 to 32767).	S_16	4327	RAM/EEPROM 8
DATA_SOURCE_TOTAL3 224	16-bit register holds the register number of the data source for totalizer 3.	U_16	4393 224	RAM/EEPROM 8
DISPLAY_FORMAT_TOTAL3 106	8-bit register controls the display format settings for totalizer 3 (display is in octal 107 format).	O_8 107	8378 106	RAM/EEPROM 8
INPUT_RATE3 224	16-bit unsigned register. Input rate for K factor calculation totalizer 3.	U_16	4307 224	RAM/EEPROM 8
K_FACTOR3 224	32-bit register. K factor for totalizer 3.	S_16	4317 224	RAM/EEPROM 8
RATE_TOTALIZER3 225	8-bit register for totalizer 3 rate calculation & rollover.	U_8	8347 225	RAM/EEPROM 8
TEXT_CHARACTER_TOTAL3 108	8-bit register holds the ASCII value for the last digit text character for totalizer 3 (0 = no character).	U_8	8404 108	RAM/EEPROM 8
TOTAL3	Non-volatile 32-bit register for totalizer 3 value.	S_32	41	RAM/FLASH 8
TOTAL3_TEXT	Text display for total 3.	L_8	16425	EEPROM 8

See also

K Factor 224

Input Rate Value - Registers 4305 to 4310 224

Totalizer Data Source Selection - Register 4391 to 4396 224

Totalizer Time Period and Rollover - Register 8345 to 8350 225

2.31.4 Total 4

Name	Description	Symbol Type	Register Number	Memory Type
CUTOFF4	16-bit register. Cutoff value for totalizer 4 (range -32768 to 32767).	S_16	4328	RAM/EEPROM 8
DATA_SOURCE_TOTAL4 224	16-bit register holds the register number of the data source for totalizer 4.	U_16	4394 224	RAM/EEPROM 8
DISPLAY_FORMAT_TOTAL4 106	8-bit register controls the display format settings for totalizer 4 (display is in octal 107 format).	O_8 107	8379 106	RAM/EEPROM 8
INPUT_RATE4 224	16-bit unsigned register. Input rate for K factor calculation totalizer 4.	U_16	4308 224	RAM/EEPROM 8
K_FACTOR4 224	32-bit register. K factor for totalizer 4.	S_16	4318 224	RAM/EEPROM 8
RATE_TOTALIZER4 225	8-bit register for totalizer 4 rate calculation & rollover.	U_8	8348 225	RAM/EEPROM 8
TEXT_CHARACTER_TOTAL4 108	8-bit register holds the ASCII value for the last digit text character for totalizer 4 (0 = no character).	U_8	8405 108	RAM/EEPROM 8
TOTAL4	Non-volatile 32-bit register for totalizer 4 value.	S_32	43	RAM/FLASH 8
TOTAL4_TEXT	Text display for total 4.	L_8	16427	EEPROM 8

See also

K Factor 224

Input Rate Value - Registers 4305 to 4310 224

Totalizer Data Source Selection - Register 4391 to 4396 224

Totalizer Time Period and Rollover - Register 8345 to 8350 225

2.31.5 Total 5

Name	Description	Symbol Type	Register Number	Memory Type
CUTOFF5	16-bit register. Cutoff value for totalizer 5 (range -32768 to 32767).	S_16	4329	RAM/EEPROM 8
DATA_SOURCE_TOTAL5 224	16-bit register holds the register number of the data source for totalizer 5.	U_16	4395 224	RAM/EEPROM 8
DISPLAY_FORMAT_TOTAL5 106	8-bit register controls the display format settings for totalizer 5 (display is in octal 107 format).	O_8 107	8380 106	RAM/EEPROM 8
INPUT_RATE5 224	16-bit unsigned register. Input rate for K factor calculation totalizer 5.	U_16	4309 224	RAM/EEPROM 8
K_FACTOR5 224	32-bit register. K factor for totalizer 5.	S_16	4319 224	RAM/EEPROM 8
RATE_TOTALIZER5 225	8-bit register for totalizer 5 rate calculation & rollover.	U_8	8349 225	RAM/EEPROM 8
TEXT_CHARACTER_TOTAL5 108	8-bit register holds the ASCII value for the last digit text character for totalizer 5 (0 = no character).	U_8	8406 108	RAM/EEPROM 8
TOTAL5	Non-volatile 32-bit register for totalizer 5 value.	S_32	45	RAM/FLASH 8
TOTAL5_TEXT	Text display for total 5.	L_8	16429	EEPROM 8

See also

K Factor 224

Input Rate Value - Registers 4305 to 4310 224

Totalizer Data Source Selection - Register 4391 to 4396 224

Totalizer Time Period and Rollover - Register 8345 to 8350 225

2.31.6 Total 6

Name	Description	Symbol Type	Register Number	Memory Type
CUTOFF6	16-bit register. Cutoff value for totalizer 6 (range -32768 to 32767).	S_16	4330	RAM/EEPROM 8
DATA_SOURCE_TOTAL6 224	16-bit register holds the register number of the data source for totalizer 6.	U_16	4396 224	RAM/EEPROM 8
DISPLAY_FORMAT_TOTAL6 106	8-bit register controls the display format settings for totalizer 6 (display is in octal 107 format).	O_8 107	8381 106	RAM/EEPROM 8
INPUT_RATE6 224	16-bit unsigned register. Input rate for K factor calculation totalizer 6.	U_16	4310 224	RAM/EEPROM 8
K_FACTOR6 224	32-bit register. K factor for totalizer 6.	S_16	4320 224	RAM/EEPROM 8
RATE_TOTALIZER6 225	8-bit register for totalizer 6 rate calculation & rollover.	U_8	8350 225	RAM/EEPROM 8
TEXT_CHARACTER_TOTAL6 108	8-bit register holds the ASCII value for the last digit text character for totalizer 6 (0 = no character).	U_8	8407 108	RAM/EEPROM 8
TOTAL6	Non-volatile 32-bit register for totalizer 6 value.	S_32	47	RAM/FLASH 8
TOTAL6_TEXT	Text display for total 6.	L_8	16431	EEPROM 8

See also
K Factor 224

Input Rate Value - Registers 4305 to 4310 224

Totalizer Data Source Selection - Register 4391 to 4396 224

Totalizer Time Period and Rollover - Register 8345 to 8350 225

2.31.7 K Factor

The registers 4315 to 4320 are used to calculate the totalizer K factor. They are used by the controller, in conjunction with Input rate 224 and Rate Totalizer 225 to calculate the true K factor. These registers are loaded with the desired value of the totalizer after the selected rate time has elapsed with the specified input rate value.

NOTE: Registers 4315 to 4320 are not the true K factor value but are only used in the calculation of the K factor. The actual value of the K factor is calculated by the controller.

2.31.8 Input Rate Value - Registers 4305 to 4310

Registers 4305 to 4310 are 16-bit unsigned registers that hold the numeric value for the input rate used during the totalizer calibration procedure.

2.31.9 Totalizer Data Source Selection - Register 4391 to 4396

Registers 4391 to 4396 are 16-bit registers that specify the data source for the totalizer channels. The number they contain is the ASCII/Modbus register number for the required data source.

NOTE: Only registers that hold integer values can be used as a data source for the display. Floating

point and text registers can not be used.

2.31.10 Totalizer Time Period and Rollover - Register 8345 to 8350

Registers 8345 to 8350 are 8-bit registers that control the time period for K factor calculation and rollover features for totalizers 1 to 6 respectively.

The function of each bit is shown as follows:

Bits 0 to 3

Bit Position								Description
7	6	5	4	3	2	1	0	
				0	0	0	0	1 second period for K factor calculations
				0	0	0	1	10 seconds period for K factor calculations
				0	0	1	0	1 minute period for K factor calculations
				0	0	1	1	10 minute period for K factor calculations
				0	1	0	0	1 hour period for K factor calculations
				0	1	0	1	10 hours period for K factor calculations
				0	1	1	0	1 day period for K factor calculations
				0	1	1	1	1 week period for K factor calculations

Bit 4 **Totalizer rollover**
0 = inactive
1 = rollover active

Bit 5 to 7 **Unused at present**

2.32 User

The controller includes a section of memory reserved for the storage of user data. This memory is not used by the operating system or any of the other standard functions in the controller, so you are free to allocate this as required. It is normally used in conjunction with a macro to store data tables or other setup parameters for the macro, but it could equally be used via the serial port.

NOTE: Some of this memory is non volatile EEPROM and write restrictions apply. See user memory 227 for more details.

See also
Auxiliary 225

Memory 227

TextMemory 231

Variables 232

2.32.1 Auxiliary

Registers 79 to 109 are signed 32-bit auxiliary registers that are intended for use with the macro. They can be used to hold calculated result values that can then be displayed or saved in the data logger. Each auxiliary register also has user definable 8 character text that can be used as a description of the function. Also the Display Format 106 and Text Character 108 for each auxiliary register can be independently setup.

Auxiliary registers are stored in RAM and also stored in non volatile FLASH memory at power down.

Auxiliary registers are not used by the operating system of the controller or by any other function so they can be used freely in the macro for any purpose.

Name	Description	Symbol Type	Register Number	Memory Type
AUX1	Non-volatile 32-bit auxiliary register.	S_32	79	RAM/FLASH
AUX2	Non-volatile 32-bit auxiliary register.	S_32	81	RAM/FLASH
AUX3	Non-volatile 32-bit auxiliary register.	S_32	83	RAM/FLASH
AUX4	Non-volatile 32-bit auxiliary register.	S_32	85	RAM/FLASH
AUX5	Non-volatile 32-bit auxiliary register.	S_32	87	RAM/FLASH
AUX6	Non-volatile 32-bit auxiliary register.	S_32	89	RAM/FLASH
AUX7	Non-volatile 32-bit auxiliary register.	S_32	91	RAM/FLASH
AUX8	Non-volatile 32-bit auxiliary register.	S_32	93	RAM/FLASH
AUX9	Non-volatile 32-bit auxiliary register.	S_32	95	RAM/FLASH
AUX10	Non-volatile 32-bit auxiliary register.	S_32	97	RAM/FLASH
AUX11	Non-volatile 32-bit auxiliary register.	S_32	99	RAM/FLASH
AUX12	Non-volatile 32-bit auxiliary register.	S_32	101	RAM/FLASH
AUX13	Non-volatile 32-bit auxiliary register.	S_32	103	RAM/FLASH
AUX14	Non-volatile 32-bit auxiliary register.	S_32	105	RAM/FLASH
AUX15	Non-volatile 32-bit auxiliary register.	S_32	107	RAM/FLASH
AUX16	Non-volatile 32-bit auxiliary register.	S_32	109	RAM/FLASH

AUX1_TEXT	Text display for Auxiliary 1	L_8	16463	EEPROM 8
AUX2_TEXT	Text display for Auxiliary 2	L_8	16465	EEPROM 8
AUX3_TEXT	Text display for Auxiliary 3	L_8	16467	EEPROM 8
AUX4_TEXT	Text display for Auxiliary 4	L_8	16469	EEPROM 8
AUX5_TEXT	Text display for Auxiliary 5	L_8	16471	EEPROM 8
AUX6_TEXT	Text display for Auxiliary 6	L_8	16473	EEPROM 8
AUX7_TEXT	Text display for Auxiliary 7	L_8	16475	EEPROM 8
AUX8_TEXT	Text display for Auxiliary 8	L_8	16477	EEPROM 8
AUX9_TEXT	Text display for Auxiliary 9	L_8	16479	EEPROM 8
AUX10_TEXT	Text display for Auxiliary 10	L_8	16481	EEPROM 8
AUX11_TEXT	Text display for Auxiliary 11	L_8	16483	EEPROM 8
AUX12_TEXT	Text display for Auxiliary 12	L_8	16485	EEPROM 8
AUX13_TEXT	Text display for Auxiliary 13	L_8	16487	EEPROM 8
AUX14_TEXT	Text display for Auxiliary 14	L_8	16489	EEPROM 8
AUX15_TEXT	Text display for Auxiliary 15	L_8	16491	EEPROM 8
AUX16_TEXT	Text display for Auxiliary 16	L_8	16493	EEPROM 8

See also
Display 101

2.32.1.1 Setup (Auxiliary)

See Display 101 for Auxiliary setup registers.

2.32.2 Memory

User memories are provided for non volatile storage of user data or look up tables, etc. These registers can be accessed either by the macro or via the serial port. They are not used by the operating system or any other functions of the controller so they can be used freely for any purpose required.

User memories are stored in RAM to give fast access and are backed up to non volatile FLASH memory at power down so that data is retained even after the power to the controller has been disconnected. There are no restrictions on the number of writes to user memory with the ICC402.

User memories can be addressed as either 8 bit unsigned registers or as 16 bit signed registers, however they share the same physical memory area and overlap each other. When using both types in the same application care should be taken avoid using the memory area for different variables.

User Memory Display Format/Text Character
By default, all user memory is displayed without any decimal point or additional text character. However, on software versions 4.02.07 onwards, the user memory area can be divided up into 3 different bands, each with different decimal point/text character configurations.

This is achieved by programming the user band registers USER_MEMORY16_BANDx (or USER_MEMORY8_BANDx for 8 bit user memories) with a register number which effectively defines

the end of a group of registers which share the same display format/text settings. The display format settings for each band (or group) are specified in DISPLAY_FORMAT_USER16_BANDx (or DISPLAY_FORMAT_USER8_BANDx) while the text character selection for each band are is specified in TEXT_CHARACTER_USER16_BANDx (or TEXT_CHARACTER_USER8_BANDx).

Band 1 starts from the beginning of user memory (register 5121 for 16 bit user memories and 10241 for 8 bit user memories) and finishes after the register specified for band 1.

Band 2 starts with the next register after that specified for band 1 and finishes after the register specified for band 2.

Band 3 starts with the next register after that specified for band 2 and finishes after the register specified for band 3.

Any user memory after band 3 will be displayed without any decimal point or text character.

When using these features you should always begin by using band 1 first. The factory default setting for these bands is zero which effectively means that the whole of user memory is displayed without any decimal point or text character.

16-bit User Memory: Registers 5121 to 6144 230

Name	Description	Symbol Type	Register Number	Memory Type
USER_MEMORY1	signed 16-bit non-volatile memory for user defined data/tables (range -32768 to 32767).	S_16	5121	RAM/FLASH 8
User memory 2 through to 1023	User memories 2 to 1023 are signed 16-bit non-volatile memory for user defined data/tables (range -32768 to 32767). The register numbers begin at 5122 for user memory 2 and end at 6143 for user memory 1023, increasing by 1 register number each time.	S_16	5122 to 6143	RAM/FLASH 8
USER_MEMORY1024	signed 16-bit non-volatile memory for user defined data/tables (range -32768 to 32767).	S_16	6144	RAM/FLASH 8
DISPLAY_FORMAT_USER16_BAND 1	8-bit register. Controls the display format settings for 16 bit user memory band 1 (displayed in octal 107 format).	O_8 107	8483 106	RAM/EEPROM 8
DISPLAY_FORMAT_USER16_BAND 2	8-bit register. Controls the display format settings for 16 bit user memory band 1 (displayed in octal 107 format).	O_8 107	8484 106	RAM/EEPROM 8
DISPLAY_FORMAT_USER16_BAND 3	8-bit register. Controls the display format settings for 16 bit user memory band 1 (displayed in octal 107 format).	O_8 107	8485 106	RAM/EEPROM 8
TEXT_CHARACTER_USER16_BAND1	8-bit register. Holds the ASCII value for the last digit text character for 16 bit user memory band 1 (0 = no character).	U_8	8491 108	RAM/EEPROM 8
TEXT_CHARACTER_USER16_BAND2	8-bit register. Holds the ASCII value for the last digit text character for 16 bit user memory band 2 (0 = no character).	U_8	8492 108	RAM/EEPROM 8
TEXT_CHARACTER_USER16_BAND3	8-bit register. Holds the ASCII value for the last digit text character for 16 bit user memory band 3 (0 = no character).	U_8	8493 108	RAM/EEPROM 8
USER_MEMORY16_BAND1	Unsigned 16-bit register that defines the last (highest) register number for band 1 of 16bit user memory.	U_16	4485	RAM/EEPROM 8
USER_MEMORY16_BAND2	Unsigned 16-bit register that defines the last (highest) register number for band 2 of 16bit user memory.	U_16	4486	RAM/EEPROM 8
USER_MEMORY16_BAND3	Unsigned 16-bit register that defines the last (highest) register number for band 3 of 16bit user memory.	U_16	4487	RAM/EEPROM 8

CAUTION **16-bit user memories overlap 8-bit user memories.**

8-bit User Memory Bytes: Registers 10241 to 12288 231

Name	Description	Symbol Type	Register Number	Memory Type
USER_MEMORY_BYTE_1	unsigned 8-bit non-volatile memory for macro use (range 0 to 255).	U_8	10241	RAM/FLASH 8
User memory bytes 2 through to 2047	User memory bytes 2 to 2047 are unsigned 8-bit non-volatile memory for macro use. The register numbers begin at 10241 for user memory byte 1 and end at 12288 for user memory byte 2048, increasing by 1 register number for each byte.	U_8	10242 to 12287	RAM/FLASH 8
USER_MEMORY_BYTE_2048	unsigned 8-bit non-volatile memory for macro use (range 0 to 255).	U_8	12288	RAM/FLASH 8
DISPLAY_FORMAT_USER8_BAND1	8-bit register. Controls the display format settings for 8 bit user memory band 1 (displayed in octal 107 format).	O_8 107	8486 108	RAM/EEPROM 8
DISPLAY_FORMAT_USER8_BAND2	8-bit register. Controls the display format settings for 8 bit user memory band 2 (displayed in octal 107 format).	O_8 107	8487 108	RAM/EEPROM 8
DISPLAY_FORMAT_USER8_BAND3	8-bit register. Controls the display format settings for 8 bit user memory band 3 (displayed in octal 107 format).	O_8 107	8488 108	RAM/EEPROM 8
TEXT_CHARACTER_USER8_BAND 1	8-bit register. Holds the ASCII value for the last digit text character for 8 bit user memory band 1 (0 = no character).	U_8	8494 108	RAM/EEPROM 8
TEXT_CHARACTER_USER8_BAND 2	8-bit register. Holds the ASCII value for the last digit text character for 8 bit user memory band 2 (0 = no character).	U_8	8495 108	RAM/EEPROM 8
TEXT_CHARACTER_USER8_BAND 3	8-bit register. Holds the ASCII value for the last digit text character for 8 bit user memory band 3 (0 = no character).	U_8	8496 108	RAM/EEPROM 8
USER_MEMORY8_BAND1	Unsigned 16-bit register that defines the last (highest) register number for band 1 of 8 bit user memory.	U_16	4488	RAM/EEPROM 8
USER_MEMORY8_BAND2	Unsigned 16-bit register that defines the last (highest) register number for band 2 of 8 bit user memory.	U_16	4489	RAM/EEPROM 8
USER_MEMORY8_BAND3	Unsigned 16-bit register that defines the last (highest) register number for band 3 of 8 bit user memory.	U_16	4490	RAM/EEPROM 8

CAUTION **8-bit user memory bytes overlap 16-bit user memories.**

2.32.2.1 16-bit User Memory - Registers 5121 to 6144

Registers 5121 to 6144 are 16-bit signed registers that can be used for non volatile storage of user data or look up tables, etc. These registers can be accessed either by the macro or via the serial port. They are not used by the operating system or any other functions of the controller so they can be used freely for any purpose required.

User memories are stored in RAM to give fast access and are backed up to non volatile FLASH memory at power down so that data is retained even after the power to the controller has been disconnected. There are no restrictions on the number of writes to user memory with the ICC402.

CAUTION: These registers overlap the 8-bit User Memories and share the same physical memory space. The only difference is that registers 5121 to 6144 address 1024 user memories, each 16 bits wide. For example;

Register 10241 occupies the same memory space as the most significant byte of register 5121
Register 10242 occupies the same memory space as the least significant byte of register 5121

Care should be taken when using both types of user memories to ensure that different memory areas are used.

For information on how to set the display format and text character of user memories see;
User Memory Display Format/Text Characters 227

2.32.2.2 8-bit User Memories - Registers 10241 to 12288

Registers 10241 to 12288 are 8-bit unsigned registers that can be used for non volatile storage of user data or look up tables, etc. These registers can be accessed either by the macro or via the serial port. They are not used by the operating system or any other functions of the controller so they can be used freely for any purpose required.

User memories are stored in RAM to give fast access and are backed up to non volatile FLASH memory at power down so that data is retained even after the power to the controller has been disconnected. There are no restrictions on the number of writes to user memory with the ICC402.

CAUTION: These registers overlap the 16-bit User Memory and share the same physical memory space. The only difference is that registers 10241 to 12288 address 2048 user memories, each 8 bits wide. For example;

Register 10241 occupies the same memory space as the most significant byte of register 5121
Register 10242 occupies the same memory space as the least significant byte of register 5121

Care should be taken when using both types of user memories to ensure that different memory areas are used.

For information on how to set the display format and text character of user memories see;
User Memory Display Format/Text Characters 227

2.32.3 Text Memory

Registers 16567 to 16693 are used to store user defined text strings of up to 30 characters long. Text strings are stored in EEPROM non volatile memory and are retained at power down. They can be used in a macro to store text which may need to changed by the end user. A good example of this would be to store a company name or phone number or maybe an email number.

Name	Description	Symbol Type	Register Number	Memory Type
USER_TEXT1	Non-volatile 30 character text string for user defined text storage.	L_30	16567	EEPROM 8
User text 1 through to 64	User text strings 1 to 64 are non-volatile 30 character text strings for user defined text storage. The register numbers begin at 16567 for text string 1 and end at 16693 for text string 64, increasing by 2 register numbers for each text string.	L_30	16567 to 16693	EEPROM 8
USER_TEXT64	Non volatile 30 character text string for user defined text storage.	L_30	16693	EEPROM 8

NOTE: Because user text memories are stored in EEPROM there is a limitation of 1x10^6 writes allowed to these registers (see Memory Types 8 for more information on maximum write limitation). If you need to write continuously to a text register then you should use Text Variables 235 instead)

See also
ASCII Text Registers 19

Text Variables 235

Startup Text 232

2.32.3.1 Startup Text

The ICC402 series controllers allow the user to change the text that is displayed on the top and bottom line of the 1602 LCD display at startup. This enables the user to display their own title or greeting message. Registers 16545 and 16547 are used to store user defined startup text strings of up to 16 characters long. Startup strings are stored in EEPROM non volatile memory and are retained at power down.

Name	Description	Symbol Type	Register Number	Memory Type
STARTUP_TEXT_LINE1	Non-volatile 16 character text string for user defined startup text on line 1 of 1602 LCD display.	L_16	16545	EEPROM 8
STARTUP_TEXT_LINE2	Non-volatile 16 character text string for user defined startup text on line 2 of 1602 LCD display.	L_16	16547	EEPROM 8

NOTE: Because startup text memories are stored in EEPROM there is a limitation of 1x10^6 writes allowed to these registers (see Memory Types 8 for more information on maximum write limitation).

See also
ASCII Text Registers 19

Text Variables 235

2.32.4 Variables

Variable registers are provided as a means of storing temporary data in a macro application. Variable registers are not used by the operating system or any other standard controller functions. However

some variable registers are assigned by the compiler. The different types of variable registers are shown below.

See Bit Flags 233

Floating Point 233

Integers 234

Text 235

See also
ASCII Text Registers 19

2.32.4.1 Bit Flags

Register 241 is a special purpose register which contains 32 bit flags which can be set or cleared in the macro. When using a remote LCD touch display panel these flags can also be set or cleared using the RPC command in the html code of the panel.

These flags are stored in RAM and are lost at power down.

Name	Description	Symbol Type	Register Number	Memory Type
GPF1	General purpose bit flag for macro use. Also settable via an LCD panel using the RPC (101) command.	B_0	241	RAM 8
General purpose bit flags 1 through to 32	All general purpose bit flags from 1 to 32 are for macro use and are also settable via an LCD panel using the RPC (101) command. The general purpose bit flags fall under register number 241 and are identified by their bit number: GPF1 = B_0 through to GPF32 = B_31.	B_0 to B_31	241	RAM 8
GPF32	General purpose bit flag for macro use. Also settable via an LCD panel using the RPC (101) command.	B_31	241	RAM 8

2.32.4.2 Floating Point

Registers 1025 to 1055 are non volatile 32-bit floating point variables that can be used in the macro to save floating point parameters. They hold a single precision floating point number that is formatted according to the IEEE-745 standard. They are stored in RAM for fast access and then saved to non volatile FLASH memory at power down.

Registers 1025 to 1055 are not used by the operating system or any other standard functions in the controller. However, they are assigned as floating point variables by the compiler when a new variable is declared using the % symbol. For example, if the following macro code is compiled:

```
RESET_MACRO:
%TEMP1 = 1.5
%TEMP2 = 1.234e-2
%TEMP3 = 1.0
END
```

The variable TEMP1 would be assigned to register FLOAT_VARIABLE1 (1025), TEMP2 would be assigned to register FLOAT_VARIABLE2 (1027), and TEMP3 would be assigned to register FLOAT_VARIABLE3 (1029). In this case the same data can be accessed by either register name. You need to ensure that the same memory area is not used for different variable functions.

Register 1121 is a 32-bit floating point register that can also be accessed as a 32-bit fixed point number via register 479. This can be a useful feature in some macros that decode an incoming serial string by reading data in a fixed point number but then need to treat it as a floating point number. See also Miscellaneous Registers 235

Name	Description	Symbol Type	Register Number	Memory Type
FLOAT_VARIABLE1	32-bit floating point register used by the macro for variable space.	F_32	1025	RAM/FLASH 8
Floating point variables 1 through to 16	Floating point variables 1 to 16 are 32-bit for macro use. The register numbers begin at 1025 for floating point register 1 and end at 1055 for floating point register 16, increasing by 2 register numbers each time.	F_32	1025 to 1055	RAM/FLASH 8
FLOAT_VARIABLE16	32-bit floating point register used by the macro for variable space.	F_32	1055	RAM/FLASH 8
VARIABLE_A_FP	32-bit register for variable A, accessed in floating point format.	F_32	1121	RAM 8

2.32.4.3 Integers

Registers 155 to 217 are 32-bit signed variables which can be used by in the macro to temporarily store parameters. They are stored in RAM for fast access and their contents is lost at power down. They default to 0 when the controller is turned ON.

Name	Description	Symbol Type	Register Number	Memory Type
INTEGER_VARIABLE1	32-bit integer used for macro variable space.	S_32	155	RAM 8
Integer variables 1 through to 32	Integer variables 1 to 32 are 32-bit for macro variable space. The register numbers begin at 155 for integer variable 1 and end at 217 for integer variable 32, increasing by 2 register numbers each time.	S_32	155 to 217	RAM 8
INTEGER_VARIABLE32	32-bit integer used for macro variable space.	S_32	217	RAM 8
VARIABLE_A_INT	32-bit register for variable A, accessed in fixed point format.	S_32	479	RAM 8

Registers 155 to 217 are not used by the operating system or any other standard functions in the controller. However, they are assigned as variables by the TDS when a new variable is declared using the # symbol. For example, if the following macro code is compiled:

```
RESET_MACRO:
#TEMP1 = 15
#TEMP2 = 12345
#TEMP3 = 1
END
```

The variable TEMP1 would be assigned to register INTEGER_VARIABLE1 (155), TEMP2 would be assigned to register INTEGER_VARIABLE2 (157), and TEMP3 would be assigned to register INTEGER_VARIABLE3 (159). In this case the same data can be accessed by either register name.

Bit Variables
When using the TDS, bit variables can also be declared in a macro in a similar method as above. The

following macro code shows an example of how to declare bit variables in your macro source code.

```
RESET_MACRO:
 |MY_FLAG1 = FALSE
 |MY_FLAG2 = FALSE
 |MY_FLAG3 = FALSE
END
```

When bit variables are declared as shown above, the TDS will take the highest unused integer variable (usually INTEGER_VARIABLE32) and allocate this as a temporary bit flag register. It will then assign |MY_FLAG1 to be bit 0 of that register, MY_FLAG2 to be bit 1, MY_FLAG3 to be bit 2 and so on. If more than 32 bit variables are declared the TDS will start using the next integer variable down and assign bit flags there.

Note:
You need to ensure that the same memory area is not used for different variable functions. If you are allowing the compiler to allocate variables by declaring them with the '#' symbol or bit flags with the '|' symbol and also referencing the same variable by it's predefined variable name (INTEGER_VARIABLEx) then the compiler will issue a warning to make you aware that you could be overwriting the same variable space.

Register 479 is a 32-bit fixed point register that can also be accessed as a 32-bit floating point number via register 1121. This can be a useful feature in some macros that decode an incoming serial string by reading data in a fixed point number but then need to treat it as a floating point number. See also Miscellaneous Registers 235

2.32.4.4 Text Variables

Registers 16897 to 16911 are used to store user defined text strings of up to 30 characters long. Text strings are stored in RAM for fast access and are lost at power down. They can be used in a macro to store temporary text which is received via the serial port.

Name	Description	Symbol Type	Register Number	Memory Type
TEXT_VARIABLE1	30 character text string variable in RAM.	L_30	16897	RAM 8
Text variables 1 through to 8	Text variables 1 to 8 are 30 character text string variables in RAM. The register numbers begin at 16897 for text variable 1 and end at 16911 for text variable 8, increasing by 2 register numbers each time.	L_30	16897 to 16911	RAM 8
TEXT_VARIABLE8	30 character text string variable in RAM.	L_30	16911	RAM 8

See also
Text Memory 231

2.33 Miscellaneous Registers

Variable A Register – 479 and 1121
Registers 479 and 1121 both address the same physical 32-bit register in memory (Variable A), but differ in the way the register is interpreted. Accessing Variable A through register 1121 assumes the contents have been stored in a 32-bit single precision floating point format. Accessing Variable A through register 479 assumes the contents have been stored in a 32-bit fixed point long format.

This pair of registers is only intended for use with the macro.

Power-up Reset Counter – Register 4348
Register 4348 is a 16-bit read/write register that is incremented each time the controller is powered up or reset from the rear test pin. It is used for diagnostic purposes only.

Memory Size for External Data Logger – Register 8432
Register 8432 is an 8-bit unsigned register that shows if an external data logger module is connected to the controller and how much memory is installed. (See External Data Logger 78 for more information on how to interpret this register).

Smart Module Software Version Number – Register 8433
Register 8433 is an 8-bit read only unsigned register that shows the software version number of the smart input module installed in the controller. The controller must have at least one input channel set to smart input mode before this register is updated.

CPU Loading – Register 8434
This register is an 8-bit read only unsigned register that shows the current processing load on the CPU in the controller from 1 to 100%. A value over 90% indicates that the controller is running out of processing time to complete all of the required functions within the selected update time. As a result input samples or output functions may be skipped and software timers or totalizers may become inaccurate. The solution is to set to OFF all unused functions and setpoints, reduce the size of any macros that are currently running, and select an update rate of 0.1 seconds.

OEM Control Register - Register 8513
On the ICC402 series of controllers the macro space can be split to allow an open area of macro space for the end user application and a closed (or locked) area of macro space for an OEM macro. Register 8513 is an 8 bit unsigned register which controls the size and lock status of the OEM macro area. Bits 0 to 4 select the size of the OEM macro in 1k byte blocks so that a size of 0 to 31k bytes can be selected. Bits 0 - 4 can only be written when their original value is 0. If any value other than 0 has been written to these bytes they cannot be overwritten until the OEM macro is erased. The OEM macro can be erased from the macro development system.

Name	Description	Symbol Type	Register Number	Memory Type
OEM_CONTROL 235	8 bit register that shows the size and lock status of the OEM macro area.	U_8	8513 235	FLASH 8

Bit 7 locks all read/write access to the OEM macro area and also locks write access to the OEM control byte. Once bit 7 is set the OEM control byte cannot be modified until the OEM macro is erased.

NOTE:
1) The actual size of an OEM macro is limited by the amount of macro code space available in the controller. So the maximum OEM macro size of 31k in the OEM control byte is a theoretical maximum only and allows for future development. (See note below on Macro Size)

2) On firmware releases 4.03.12 and earlier only bits 0-3 and bit 7 of the OEM control byte were active, limiting the maximum OEM macro size to 15k.

Macro Size – Register 4433
This register is a 16-bit read only register that defines the amount of macro code space available in the controller. Reading this register produces a number from 1 to 65535 that relates to the number of ASCII/Modbus registers allocated for macro code storage. This may change with model or version number.

Software Version Number - Register 4106

This is a 16 bit read only register that defines the currently installed software version number.

Device Type - Register 16565
This is a read only text register that defines the model or controller type.

Display Type - Register 16557
This is a read only text register that defines the type of display to be used with this controller. The ICC402 controller can be supplied with several different display options. Before connecting a display to the controller you should ensure you are using the correct type.

Product Serial Number - Register 541
This is a 32 bit read only register that holds the product serial number. This is only available on versions 4.02a+.

Memory Reset - Register 16559
This is write only text register which can be used to reset the controller or parts of the controllers memory. All text written to this register must be in upper case. The following functions can be accessed via this register;

Factory defaults - The controller configuration can be returned to factory defaults by writing "INIT" to register 16559.

Erase Macro - All programmed macros can be erased by writing "ERASE MACRO" to register 16559.

Controller Reset - The controller can be totally reset by writing "RESET" to register 16559.

WARNING: Using this register could erase any currently installed macro or cause the loss of custom configuration data by returning the meter to factory defaults. We recommend care when using this register!

Index

- 1 -

16bit 230
16-bit Fixed point (4097 to 8192) 8

- 2 -

24-bit Fixed Point (2049 to 4095) 8

- 3 -

32-bit Fixed Point (1 to 1023) 7
32-bit Floating Point (1025 to 2047) 8

- 4 -

400 Series 7

- 8 -

8bit 231
8-bit Fixed point (8193 to 16384) 8

- A -

Alarm Status Read 209
Alarm Status Write 211
Alphanumeric Character for Current Display - Register 8392 106
Analog Output 1 28
Analog Output 2 28
Analog Output 3 28
Analog Output 4 29
Analog Output 5 29
Analog Output 6 29
Analog Output 7 30
Analog Output 8 30
Analog Output Data Source Selection - Register 4379 to 4386 30
AnalogOutput 26
ASCII Characters for 14-Segment Display 23
ASCII Mode 9
ASCII Mode Format 14
ASCII Read/Write Examples 16
ASCII Text Registers 19
Auxiliary 225
AuxiliaryAnalogInputs 30

- B -

BitFlags 233
Brightness/Contrast - Register 8206 105

- C -

Calibration 53
CH1 Setup Registers 33
CH10 Setup Registers 49
CH11 Setup Registers 51
CH2 Setup Registers 35
CH3 Setup Registers 36
CH4 Setup Registers 37
CH5 Setup Registers 39
CH6 Setup Registers 41
CH7 Setup Registers 43
CH8 Setup Registers 45
CH9 Setup Registers 47
Channel1 32
Channel10 47
Channel11 49
Channel2 33
Channel3 35
Channel4 37
Channel5 38
Channel6 39
Channel7 41
Channel8 43
Channel9 45
Character Frame Formats 12
Clock 51
Code 1 55
Code 2 58
Code 3 58
Code 4 59
Code 5 60
Code 6 61
Code 7 62
Code 8 63
Code 9 63

Codes 52
Command Response Time 13
Communication Formats 9
ControlInputs 87

- D -

Data Logging Concepts 67
DataLogging 64
Delay Type - Register 8277 to 8288 162
Digital 79
Digital Read Only 88
DIGITAL_IO 140
DigitalPins 140
Display 101
Display Data Source Selection - Register 4371 to 4373 105
Display Options For Current Display - Register 8366 106

- E -

Edit State - Register 8223 99
EditMode 98
ExternalDataLogger 78

- F -

FloatingPoint 233

- H -

Hysteresis Type 163

- I -

I/O Module Type - Register 8430 97
Input Rate Value - Registers 4305 to 4310 224
InputModule 140
Integers 234

- L -

Linearization 109
Linearization Table 1 109
Linearization Table 2 113
Linearization Table 3 116
Linearization Table 4 119
Log Read Pointer - Register 491 73
Log Register Source - Registers 4417 to 4432 74
Log Write Pointer - Register 489 73

- M -

Macro Code Registers (32769 to 65536) 8
Macro Compiling Uploading 17
Manual Mode 134
Maximum Number Of Log Samples - Register 487 72
Memory 227
Memory types 8
Miscellaneous Registers 235
Modbus Mode 11
ModBusMaster 147
Multiple Write 16

- N -

Non-volatile Write Flag - Register 8222 98
Number Of Log Sample Reads - Register 4462 74
Numeric Log Sample Values - Register 493 to 523 73

- O -

Octal 107
Output Saturation 138

- P -

Peak/Valley Data Source Selection - Register 4374 to 4376 105
PID 122
PID 1 123
PID 2 125
PID 3 127
PID 4 129
PID 5 131
PID 6 133
PID Control Registers - Register 1057 to 1079 139
Port1 146
Port2 152
Port3 154

Print String - Register 16543 24

- R -

Read Log Data at Log Read Pointer - Register 16555 77
Read Log Sample Data - Register 8443 to 8449 74
Read Only - Analog Output 1 28
Read Only - Analog Output 2 28
Read Only - Analog Output 3 28
Read Only - Analog Output 4 29
Read Only - Analog Output 5 29
Read Only - Analog Output 6 29
Read Only - Analog Output 7 30
Read Only - Analog Output 8 30
Read Only - Digital 88
Read Only - Smart Module 191
Read Only Registers 69
Read Only Registers - Data Logging 69
Read Single Log Data at Log Read Pointer - Register 16553 75
Register 16385 23
Register 239 - Alarm Status 207
Register List 19
Register Types 7
Relay De-energize Mask - Register 4101 160
Relay Energize Functions 158
Reset Destination - Register 4229 to 4240 161
Result 141
Result Setup Registers 141

- S -

Select Data Source 56
Serial 142
Serial Address - Register 8211 / 8212 144
Serial Buffer Port 1 - Registers 12289 to 12544 147
Serial Buffer Port 2 - Registers 12545 to 12800 153
Serial Count Receive - Registers 8454 & 8455 145
Serial Port Settings - Register 8207 / 8208 143
Setpoint 155
Setpoint 1 166
Setpoint 2 168
Setpoint 3 170
Setpoint 4 172
Setpoint 5 174
Setpoint 6 176
Setpoint Control Registers 156
Setpoint Data Source Selection - Register 4401 to 4406 161
Setpoint Digit Settings 157
Setpoint Filter 135
Setpoint Latch Mask - Register 4100 157
Setpoint Reset Delay (Power-On Inhibit) - Register 4102 160
Setpoint Tracking - Register 8261 to 8272 161
Setpoint Trigger Flags - Register 4098 165
Setpoint Trigger Functions - Register 8293 to 8304 163
Setpoint7 178
Setpoint8 180
Setup - Auxiliary 227
Setup - CH1 33
Setup - CH10 49
Setup - CH11 51
Setup - CH2 35
Setup - CH3 36
Setup - CH4 37
Setup - CH5 39
Setup - CH6 41
Setup - CH7 43
Setup - CH8 45
Setup - CH9 47
Setup - Result Registers 141
Setup - Smart Module 191
Setup - SP1 167
Setup - SP2 169
Setup - SP3 171
Setup - SP4 173
Setup - SP5 175
Setup - SP6 177
Smart Module Setup 191
Smart2 192
Smart2Setup 193
Smart3 192
Smart3Setup 195
SmartModule 189
SP1 Setup 167
SP2 Setup 169
SP3 Setup 171
SP4 Setup 173
SP5 Setup 175
SP6 Setup 177

SP7Setup 179
Start / End of String Characters - Register 8226 to 8229 144
StartupText 232
StartupTextLine1 232
StartupTextLine2 232
Status 197
Status Registers 197
Status Switches 206
String Length - Register 8230 to 8233 145
Switches 215

- T -

Text 108
Text - Display 108
Text Registers (16385 to 20479) 8
Text Variables 235
TextMemory 231
Timers 216
Total 217
Total 1 219
Total 2 220
Total 3 221
Total 4 222
Total 5 223
Total 6 224
Totalizer 217
Totalizer Data Source Selection - Register 4391 to 4396 224
Totalizer Time Period and Rollover - Register 8345 to 8350 225

- U -

User 225

- V -

VALLEY1 101
Variables 232

- W -

Weighting Factor 136