I-87026PW
Command Sets

ICP DAS CO., LTD.
Revision: 1.0
2012/03/09

Table of Contents

1. Introduction

1.1 Default Settings

- Module address: 01
- Analog input type: Type 08, -10 V to 10 V
- Analog output type: Type 33, -10 V to 10 V
- Baud Rate: 115200 bps
- Checksum disabled
- Engineering units format
- Filter set at 60 Hz rejection

1.2 Calibration

Warning: *It is not recommended that calibration be performed until the process is fully understood.*

1.2.1 Analog Input

The calibration procedure is as follows:

1. Warm up the module for 30 minutes.
2. Set the type code to the type you want to calibrate. Refer to Section 2.19 for details.
3. Enable calibration. Refer to Section 2.48 for details.
4. Apply the zero calibration voltage/current.
5. Send the analog input "zero calibration" command. Refer to Section 2.7 for details.
6. Apply the span calibration voltage/current.
7. Send the analog input "span calibration" command. Refer to Section 2.6 for details.
8. Repeat steps 3 to 7 three times.

Notes:

1. Connect the calibration voltage/current to channel 0.
2. When calibrating type 0D, the jumper for channel 0 should be set to the "current input" position.
3. Calibration voltages and currents are shown below.

Calibration voltages/currents:

Type Code	08	09	0A	0B	0C	0D
Zero Input	0 V	0 V	0 V	0 mV	0 mV	0 mA
Span Input	+10 V	+5 V	+1 V	+500 mV	+150 mV	+20 mA

1.2.2 Analog Output

The calibration procedure is as follows:

1. Warm up the module for 30 minutes.
2. Set the type code to the type you want to calibrate. Refer to Section 2.23 for details.
3. Set the zero analog output. Refer to Section 2.5 for details.
4. Check the meter and trim the output until the zero output is achieved. Refer to Section 2.11 for details.
5. Send the analog output "zero calibration" command. Refer to Section 2.8 for details.
6. Set the span analog output. Refer to Section 2.5 for details.
7. Check the meter and trim the output until the span output is achieved. Refer to Section 2.11 for details.
8. Send the analog output "span calibration" command. Refer to Section 2.9 for details.

Notes:

1. Connect the calibration voltage/current to the specific channel that is to be calibrated.
2. When calibrating types 30 and 31, the jumper should be set to the "current input" position.
3. Calibration voltages and currents are shown below.

Calibration voltages/current:

Type Code	30	31	32	33	34	35
Zero Input	0 mA	4 mA	0 V	0 V	0 V	0 V
Span Input	+20 mA	+20 mA	+10 V	+10 V	+5 V	+5 V

1.3 Configuration Tables

Baud Rate Settings (CC)

Bits 5:0

Code	03	04	05	06	07	08	09	0A
Baud Rate	1200	2400	4800	9600	19200	38400	57600	115200

Bits 7:6

00: no parity, 1 stop bit
01: no parity, 2 stop bits
10: even parity, 1 stop bit
11: odd parity, 1 stop bit

Analog Input Type Settings (TT)

Type Code	Analog Input Type	Range
07	+4 to +20 mA	4mA ~ 20mA
08	+/-10 V	-10 V ~ 10 V
09	+/-5 V	-5 V ~ 5 V
0A	+/-1 V	-1 V ~ 1 V
0B	+/-500 mV	-500 mV ~ 500 mV
0C	+/-150 mV	-150 mV ~ 150 mV
0D	+/-20 mA	-20 mA ~ 20 mA
1A	0 to +20 mA	0 ~ 20 mA

Note:
When types 07, 0D or 1A are selected, the jumper for the corresponding channel should be set to the “current input” position.

Data Format Setting (FF)

7	6	5	4	3	2	1	0
FS	CS	MS	Reserved			DF	

Key	Description
DF	Data format 00: Engineering units 01: % of FSR (full scale range) 10: 2's complement hexadecimal
MS	Mode settings 0: Normal mode (16 bits) 1: Fast mode (12 bits)
CS	Checksum settings 0: Disabled 1: Enabled
FS	Filter settings 0: 60Hz rejection 1: 50Hz rejection

Note: The reserved bits should be zero.

Analog Input Type and Data Format Table

Type Code	Input Type	Data Format	+F.S	-F.S.
07	+4 to +20 mA	Engineering units	+20.000	+04.000
		% of FSR	+100.00	+000.00
		2's comp HEX	FFFF	0000
08	-10 to +10 V	Engineering units	+10.000	-10.000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
09	-5 to +5 V	Engineering units	+5.0000	-5.0000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
0A	-1 to +1 V	Engineering units	+1.0000	-1.0000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
0B	-500 to +500 mV	Engineering units	+500.00	-500.00
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
0C	-150 to +150 mV	Engineering units	+150.00	-150.00
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
0D	-20 to +20 mA	Engineering units	+20.000	-20.000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
		2's comp HEX	7FFF	A99A
1A	0 to +20 mA	Engineering units	+20.000	+00.000
		% of FSR	+100.00	+000.00
		2's comp HEX	FFFF	0000

Analog Output Type and Data Format Table

Type Code	Input Type	Data Format	Max.	Min.
30	0 to 20 mA	Engineering units	+20.000	+00.000
31	4 to 20 mA	Engineering units	+20.000	+04.000
32	0 to 10 V	Engineering units	+10.000	+00.000
33	-10 to 10 V	Engineering units	+10.000	-10.000
34	0 to 5 V	Engineering units	+05.000	+00.000
35	-5 to 5 V	Engineering units	+05.000	-05.000

Analog Output Slew Rate Control

0 Immediate change
1 0.0625V/Second or 0.125mA/Second
2 0.125V/Second or 0.25mA/Second
3 0.25V/Second or 0.5mA/Second
4 0.5V/Second or 1.0mA/Second
5 1.0V/Second or 2.0mA/Second
6 2.0V/Second or 4.0mA/Second
7 4.0V/Second or 8.0mA/Second
8 8.0V/Second or 16mA/Second
9 16V/Second or 32mA/Second
A 32V/Second or 64mA/Second
B 64V/Second or 128mA/Second
C 128V/Second or 256mA/Second
D 256V/Second or 512mA/Second
E 512V/Second or 1024mA/Second

2. DCON Protocol

All communication with I-87K I/O modules consists of commands generated by the host and responses transmitted by the modules. Each module has a unique ID number that is used for addressing purposes and is stored in non-volatile memory. The ID is 01 by default and can be changed using a user command. All commands to the module contain the ID address, meaning that only the addressed module will respond. The only exception to this is command ~** (Section 2.35), which is sent to all modules, but in this case, the modules do not reply to the command.

Command Format:

Leading Character	Module Address	Command	[CHKSUM]	CR

Response Format:

Leading Character	Module Address	Data	[CHKSUM]	CR

CHKSUM A 2-character checksum that is present when the checksum setting is enabled. See Sections 1.3 and 2.1 for details.

CR End of command character, carriage return (0x0D)

Checksum Calculation:

1. Calculate the ASCII code sum of all the characters in the command/response string, except for the carriage return character (CR).
2. The checksum is equal to the sum masked by 0ffh.

Example:

Command string: $012(CR)

1. Sum of the string = “$”+”0”+”1”+”2” = 24h+30h+31h+32h = B7h
2. Therefore the checksum is B7h, and so CHKSUM = “B7”
3. The command string with the checksum = $012B7(CR)

Response string: !01200600(CR)

1. Sum of the string = “!”+”0”+”1”+”2”+”0”+”0”+”6”+”0”+”0” = 21h+30h+31h+32h+30h+30h+36h+30h+30h = 1AAh
2. Therefore the checksum is AAh, and so CHKSUM = “AA”
3. The response string with the checksum = !01200600AA(CR)

Note:

All characters should be in upper case.

2.1 %AANNTTCCFF

Description:

This command is used to set the configuration of a specific module.

Syntax:

%AANNTTCCFF[CHKSUM](CR)

%	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
NN	The new address of the module in hexadecimal format (00 to FF)
TT	Not used by the I-87026PW and should be set to 00.
CC	The new Baud Rate code, see Section 1.3 for details. To change the Baud Rate, the module should first be switched to INIT* mode.
FF	The command used to set the data format, checksum, and filter settings (see Section 1.3 for details). To change the checksum settings, the module should first be switched to INIT* mode.

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter for a valid response
?	Delimiter for an invalid response (If the Baud Rate or checksum settings are changed without first switching to INIT* mode, the module will return an invalid response.)
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Note:

Changes to the address, Type Code, Data Format and Filter settings take effect immediately after a valid command is received. Changes to the Baud Rate and checksum settings take effect at the next power-on reset.

Examples:

Command: %0102000600 Response: !02

Changes the address of module 01 to 02 and returns a valid response.

Command: %0202000602 Response: !02

Sets the data format of module 02 to type 2 (2's complement hexadecimal). The module returns a valid response.

Command: %0101000A00 Response: ?01

Attempts to change the Baud Rate of module 01 to 115200bps and returns an invalid response, because it was not switched to INIT* mode before setting the command.

Command: %0101000A00 Response: !01

Changes the Baud Rate of module 01 to 115200bps and the module is in INIT* mode. The module returns a valid response.

Related Command:

Section 2.10 $AA2

Related Topics:

Section 1.3 Configuration Tables

2.2 #**

Description:

When this command is received, it allows every analog input module to read data from every input channel and the data will be stored in the buffer for later retrieval.

Syntax:

#[CHKSUM](CR)**

#	Delimiter character
**	The synchronized sampling command

Response:

There is no response to this command. To access the data, another command, $AA4, must be sent, see Section 2.12 for details.

Examples:

Command: #** No response

Sends the synchronized sampling command.

Command: $014 Response:
>011+025.12+020.45+012.78+018.97+003.24+015.35

Sends the command to read the synchronized data. The status byte of the response is 1, which means that it is the first time the synchronized data has been read since the previous #** command was sent.

Command: $014 Response:
>010+025.12+020.45+012.78+018.97+003.24+015.35

Sends the command to read the synchronized data. The status byte of the response is 0, which means that it is not the first time the synchronized data has been read since the previous #** command was sent.

Related Command:

Section 2.12 $AA4

2.3 #AA

Description:

This command is used to read the data from all analog input channels of a specified module.

Syntax:

#AA[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)

Response:

Valid Response: **>(Data)[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

>	Delimiter character for a valid response
?	Delimiter character for an invalid response
(Data)	The data from all analog input channels, see Section 1.3 for details of the data format.
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #01 Response:
>+025.12+020.45+012.78+018.97+003.24+015.35

Reads module 01 and receives the data in engineering units format.

Command: #02 Response:
>4C532628E2D683A20F2ADBA1

Reads module 02 and receives the data in hexadecimal format.

Command: #03 Response:
>-9999.9-9999.9-9999.9-9999.9-9999.9-9999.9

Attempts to read module 03 and returns a invalid response indicating that the data is out of range.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.10 $AA2, Section 2.4 #AAN

Related Topics:

Section 1.3 Configuration Tables

2.4 #AAN

Description:

This command is used to read the analog input data from channel N of a specified module.

Syntax:

#AAN[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
N	The channel to be read, zero based.

Response:

Valid Response: **>(Data)[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

>	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The analog input data from the specified channel, see Section 1.3 for details of the data format.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #032 Response: >+025.13

Reads data from channel 2 of module 03 and returns a valid response indicating a value of +025.13.

Command: #02F Response: ?02

Attempts to read data from channel 15 of module 02 and returns an invalid response because channel 15 does not exist.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.3 #AA, Section 2.10 $AA2

Related Topics:

Section 1.3 Configuration Tables

2.5 #AAN(Data)

Description:

This command is used to set the analog output value for channel N of a specified module.

Syntax:

#AAN(Data)[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
N	The channel to be set, zero based.
(Data)	The analog output value, see Section 1.3 for details of the data format.

Response:

Valid Response: **>[CHKSUM](CR)**
Out of Range: ?**[CHKSUM](CR)**
Invalid Response: !**[CHKSUM](CR)**

>	Delimiter character for a valid response
?	Delimiter character to indicate that the (Data) value is out of range. The output will revert to the closest value in the range set for the module.
!	Delimiter character to indicate that the Host Watchdog flag for the module has been set. The output command will be ignored and the output will be set to define the safe value.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $0190300 Response: !01

Sets the configuration for channel 0 of module 01 to an output range of 0 to 20mA, sets a slew rate of immediate change, and returns a valid response.

Command: #010+05.000 Response: >

Sets the analog output for channel 0 of module 01 to 5mA and returns a valid response.

Command: #010+25.000 Response: ?

Sets the analog output for channel 0 of module 01 to 25.0mA and returns an invalid response, meaning that the output value is out of range. Channel 0 will revert to the closest valid output value of +20.000mA.

Command: #018+05.000 No Response

Sets the analog output for channel 8 of module 01 to 5.0mA and returns no response, which indicates that channel 8 does not exist.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.10 $AA2, Section 2.22 $AA9N, Section 2.23 $AA9NTTS

Related Topics:

Section 1.3 Configuration Tables

2.6 $AA0

Description:

This command is used to perform an analog input span calibration on a specified module.

Syntax:

$AA0[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be calibrated in hexadecimal format (00 to FF)
0	The command to perform the analog input span calibration

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Note:

The “enable calibration” command, ~AAEV Section 2.48, must be sent before this command is used. See Section 1.2.1 for details.

Examples:

Command: $010 Response: !01

Performs an analog input span calibration on module 01 and returns a valid response.

Command: $020 Response: ?02

Attempts to perform an analog input span calibration on module 02 and returns an invalid response because the “enable calibration” command, “~AAEV” was not sent in advance.

Related Commands:

Section 2.7 $AA1, Section 2.48 ~AAEV

Related Topics:

Section 1.2.1 Calibration

2.7 $AA1

Description:

This command is used to perform an analog input zero calibration on a specified module.

Syntax:

$AA1[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be calibrated in hexadecimal format (00 to FF)
1	The command to perform the analog input zero calibration

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Notes:

The “enable calibration” command, ~AAEV Section 2.48, must be sent before this command is used, see Section 1.2.1 for details.

Examples:

Command: $011 Response: !01

Performs an analog input zero calibration on module 01 and returns a valid response.

Command: $021 Response: ?02

Attempts to perform an analog input zero calibration on module 02 and returns an invalid response because the “enable calibration” command, “~AAEV” was not sent in advance.

Related Commands:

Section 2.6 $AA0, Section 2.48 ~AAEV

Related Topics:

Section 1.2.1 Calibration

2.8 $AA0N

Description:

The command is used to perform an offset calibration on analog output channel N of a specified module.

Syntax:

$AA1[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be calibrated in hexadecimal format (00 to FF)
0	The command to perform the analog output offset calibration
N	The channel to be calibrated, zero based

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to 1F)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: $0101 Response: !01

Performs an offset calibration on analog output channel 1 of module 01 and returns a valid response.

Command: $010F Response: ?01

Attempts to perform an offset calibration on analog output channel 15 of module 01 and returns a invalid response because channel 5 does not exist.

Related Commands:

Section 2.9 $AA1N, Section 2.11 $AA3NVV

Related Topics:

Section 1.2.2 Calibration

2.9 $AA1N

Description:

This command is used to perform a span calibration on analog output channel N of a specified module.

Syntax:

$AA1N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be calibrated in hexadecimal format (00 to FF)
1	The command to perform the analog output span calibration
N	The channel to be calibrated, zero based

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command:$0112 Response: !01

Performs a span calibration on analog output channel 2 of module 01 and returns a valid response.

Command: $011F Response: ?01

Attempts to perform a span calibration on analog output channel 15 of module 01 and returns a invalid response because channel 15 does not exist.

Related Commands:

Section 2.8 $AA0N, Section 2.11 $AA3NVV

Related Topics:

Section 1.2.2 Calibration

2.10 $AA2

Description:

This command is used to read the module configuration of a specified module.

Syntax:

$AA2[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
2	The command to read the configuration of the module

Response:

Valid Response: **!AATTCCFF[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
TT	Not used by the I-87026PW and should be 00.
CC	The Baud Rate code for the module, see Section 1.3 for details.
FF	The data format, checksum and filter settings for the module, see Section 1.3 for details.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $012 Response: !01000A00

Reads the configuration of module 01 and returns a valid response indicating that the Baud Rate is 115200, the data format is engineering format and checksum is disabled.

Command: $022 Response: !02000A02

Reads the configuration of module 02 and returns a valid response indicating that the Baud Rate is 115200, data format is hex format and checksum is disabled.

Related Command:

Section 2.1 %AANNTTCCFF

Related Topics:

Section 1.3 Configuration Tables

2.11 $AA3NVV

Description:

This command is used to trim the calibration for analog output channel N of a specified module.

Syntax:

$AA3NVV[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
3	The command to read the configuration of the module
N	The channel to be calibrated, zero based
VV	A two hexadecimal digits to represent the trim calibration value. A value of 01 to 5F will increase the calibration value by 1 to 95, and values FF to A1 will decrease the calibration value by 1 to 95.

Response:

Valid Response: **!AA [CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01301F Response: !01

Trims the output for channel 0 of module 01 and increases the calibration value by 31 and returns a valid response.

Command: $013060 Response: ?01

Attempts to trim the output for channel 0 of module 01 and increase the calibration value by 96, but returns an invalid response indicating that trim the value is out of range.

Related Commands:

Section 2.8 $AA0N, Section 2.9 $AA1N

Related Topics:

Section 1.2.2 Calibration

2.12 $AA4

Description:

This command is used to read the synchronization data from a specified module that was stored when the last #** Section 2.2 command was sent.

Syntax:

$AA4[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
4	The command to read the synchronization data

Response:

Valid Response: **!AAS(Data)[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
S	The status of the synchronization data 1: The data is being read for the first time 0: The data has been read before
(Data)	The synchronization data. See Section 1.3 for details of the data format.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #** No response

Sends the synchronized sampling command. There will be no response to this command.

Command: $014 Response:
>011+00.000+00.100+01.000+10.000-01.000+05.000

Reads the synchronization data from module 01 and returns a valid response containing the data, and sets the status byte to 1 to indicate that the synchronized data is being read for the first time.

Command: $014 Response:
>010+00.000+00.100+01.000+10.000-01.000+05.000

Reads the synchronized data from module 01 and returns a valid response containing the data, and sets the status byte to 0 to indicate that the synchronized data has already been read before.

Related Command:

Section 2.2 #**

Related Topics:

Section 1.3 Configuration Tables

2.13 $AA4N

Description:

This command is used to set the analog output power-on value for channel N of a specified module.

Syntax:

$AA4N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
4	The command to set the power-on value, and store the current output value as the power-on value
N	The channel to be set, zero based

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #012+00.000 Response: >

Sets the analog output for channel 2 of module 01 to 0.0V and returns a valid response.

Command: $0142 Response: !01

Sets the power-on value for channel 2 of module 01to 0.0V and returns a valid response. The power-on value for channel 2 is set to 0.0V immediately.

Command: $014F Response: ?01

Attempts to set the power-on value for channel 15 of module 01 and returns an invalid response because channel 15 does not exist.

Related Commands:

Section 2.5 #AAN(Data), Section 2.18 $AA7N

2.14 $AA5

Description:

This command is used to read the reset status of a specified module.

Syntax:

$AA5[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
5	The command to read the reset status

Response:

Valid Response: **!AAS[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
S	The reset status of the module: 0: This is not the first time the command has been sent since the module was powered on, which denotes that there has been no module reset since the last $AA5 command was sent. 1: This is the first time the command has been sent since the module was powered on.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $015 Response: !011

Reads the reset status of module 01. The response indicates that it is the first time the $AA5 command has been sent since the module was powered on.

Command: $015 Response: !010

Reads the reset status of module 01. The response indicates that there has been no module reset since the last $AA5 command was sent.

2.15 $AA5VV

Description:

This command is used to specify which channel(s) of a specified module are to be enabled.

Syntax:

$AA5VVVV[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
5	The command to set the channel(s) to enabled
VV	A two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it means that the channel is disabled and 1 means that the channel is enabled.

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response. An invalid response is returned if an attempt is made to enable a channel that is not present.
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $0153A Response: !01

Enables channels 1, 3, 4, and 5 of module 01 and disables all other channels. The module returns a valid response.

Command: $016 Response: !013A

Reads the channel status of module 01 and returns a valid response of “3A”, meaning that channels 1, 3, 4, and 5 are enabled and all other channels are disabled.

Related Command:

Section 2.16 $AA6

2.16 $AA6

Description:

This command is used to read whether each channel of a specified module is enabled or disabled.

Syntax:

$AA6[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
6	The command to read the channel status

Response:

Valid Response: **!AAVV[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
VV	A two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0 it means that the channel is disabled, and 1 means that the channel is enabled.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $0153A Response: !01

Enables channels 1, 3, 4, and 5 of module 01 and disables all other channels. The module returns a valid response.

Command: $016 Response: !013A

Reads the channel status of module 01 and returns a valid response of “3A”, meaning that channels 1, 3, 4, and 5 are enabled and all other channels are disabled.

Related Command:

Section 2.15 $AA5VV

2.17 $AA6N

Description:

This command is used to read the value of the last analog output command from channel N of a specified module.

Syntax:

$AA6N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
6	The command to read the value of the last analog output command
N	The channel to be read, zero based

Response:

Valid Response: **!AA(Data)[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response or an invalid type code
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The value of the last analog output command

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #011+10.000 Response: !01

Sets the analog output value for channel 1 of module 01 to +10.000 and returns a valid response.

Command: $0161 Response: !01+10.000

Reads the last analog output value for channel 1 of module 01 and returns a valid response with a value of +10.000.

Command: $016F Response: ?01

Attempts to read the last analog output value for channel 15 of module 01 and returns an invalid response because channel 15 does not exist.

Related Commands:

Section 2.5 #AAN(Data), Section 2.20 $AA8N, Section 2.23 $AA9NTTS

2.18 $AA7N

Description:

This command is used to read the analog output power-on value for channel N of a specified module.

Syntax:

$AA7N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
7	The command to read the analog output power-on value
N	The channel to be read, zero based

Response:

Valid Response: **!AA(Data)[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for a invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The analog output power-on value

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $0170 Response: !01+10.000

Reads the analog output power-on value for channel 0 of module 01, and returns a valid response with a value of +10.000

Command: $017F Response: ?01

Attempts to read the analog output power-on value for channel 15 of module 01, and returns an invalid response indicating analog output channel 15 does not exist.

Related Commands:

Section 2.5 #AAN(Data), Section 2.13 $AA4N

2.19 $AA7CiRrr

Description:

This command is used to set the type code for a specific channel of a specified module.

Syntax:

$AA7CiRrr[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
7	The command to set the channel range code
Ci	The channel to be set, zero based
Rrr	rr represents the type code of the channel to be set. Refer to the Analog Input Type Settings table in Section 1.3 for details.

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response or an invalid type code
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $017C0R08 Response: !01

Sets the type code for channel 0 of module 01 to 08 (-10 ~ +10V) and the module returns a valid response.

Command: $037C1RFF Response: ?03

Attempts to set the type code for channel 1 of module 03 to FF, but returns an invalid response because the type code is invalid.

Related Command:

Section 2.21 $AA8Ci

Related Topics:

Section 1.3 Configuration Tables

2.20 $AA8N

Description:

This command is used to read the current analog output value for channel N of a specified module.

Syntax:

$AA8N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
8	The command to read the current analog output value
N	The channel to be read, zero based

Response:

Valid Response: **!AA(Data)[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for a invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The current analog output value

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $0180 Response: !01+01.000

Reads the current analog output value for channel 0 of module 01, and returns a valid response with a value of +01.000V.

Command: $018F Response: ?01

Attempts to read the current analog output value for channel 15 of module 01, and returns an invalid response indicating that analog output channel 15 does not exist.

Related Commands:

Section 2.5 #AAN(Data), Section 2.17 $AA6N, Section 2.23 $AA9NTTS

2.21 $AA8Ci

Description:

This command is used to read the analog input type code information for a specific channel of a specified module.

Syntax:

$AA8Ci[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
8	The command to read the type code of the channel
Ci	Specifies which analog input channel to access for the type code information

Response:

Valid Response: **!AACiRrr[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response or an invalid channel
AA	The address of the responding module in hexadecimal format (00 to FF)
Ci	Specifies the analog input channel that was accessed to retrieve the type code information.
Rrr	Represents the type code of the specified analog input channel. Refer to the Analog Input Type Settings table in Section 1.3 for details.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $018C0 Response: !01C0R09

Reads the analog input type code information for channel 0 of module 01 and returns a valid response of 09 which means that the input type is -5~+5V.

Command: $018CF Response: ?01

Attempts to read the analog input type code information for channel 15 of module 01 and returns an invalid response indicating that analog input channel 15 does not exist.

Related Command:

Section 2.19 $AA7CiRrr

Related Topics:

Section 1.3 Configuration Tables

2.22 $AA9N

Description:

This command is used to read the analog output configuration for channel N of a specified module.

Syntax:

$AA9N[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
9	The command to read the analog output configuration
N	The channel to be read, zero based

Response:

Valid Response: **!AATS[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
T	The analog output type. Refer to the Analog Output Type Settings table in Section 1.3 for details.
S	The analog output slew rate. Refer to the Analog Output Type Settings table in Section 1.3 for details.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $0190330 Response: !01

Sets the analog output configuration for channel 0 of module 01 to an output range of -10 to +10V and any changes are immediate. The module returns a valid response.

Command: $0190 Response: !01330

Reads the analog output configuration for channel 0 of module 01 and returns a valid response indicating that the output is -10V to +10V and any changes are immediate

Command: $019F Response: ?01

Attempts to read the analog output configuration for channel 15 of module 01 and returns an invalid response indicating that analog output channel 15 does not exist.

Related Command:

Section 2.23 $AA9NTTS

Related Topics:

Section 1.3 Configuration Tables

2.23 $AA9NTTS

Description:

This command is used to set the analog output configuration for channel N of a specified module.

Syntax:

$AA9NTTS[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
9	The command to set the analog output configuration
N	The channel to be set, zero based
T	The analog output type. Refer to the Analog Output Type Settings table in Section 1.3 for details.
S	The analog output slew rate. Refer to the Analog Output Type Settings table in Section 1.3 for details.

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $0191331 Response: !01

Sets the analog output configuration for channel 1 of module 01 to an output range of -10 to 10V and a slew rate of 0.625V/Second, and returns a valid response.

Command: $0191 Response: !01331

Reads the analog output configuration for channel 1 of module 01 and returns a valid response indicating that the output range is -10V to +10V and the slew rate is 0.625V/Second

Command: $019F331 Response: !01

Attempts to set the analog output configuration for channel 15 of module 01 to an output range of -10 to 10V and a slew rate of 0.625V/Second, and returns an valid response indicating that analog output channel 15 does not exist.

Related Command:

Section 2.22 $AA9N

Related Topics:

Section 1.3 Configuration Tables

2.24 $AAB

Description:

This command is used to detect the wiring status for the analog outputs of a specified module.

Syntax:

$AAB[CHKSUM](CR)

$	Delimiter character
AA	The address of the module for which the wiring status is to be detected in hexadecimal format (00 to FF)
B	The command to detect the wiring status of the analog outputs

Response:

Valid Response: **!AANN[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
NN	Represents the detection results for all the analog output channels (00 to FF), where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. If the bit is 0, it denotes that the channel is disabled or normal. If the bit is 1, it denotes that the channel is enabled and there is no wire connected.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: $01B Response: !0101

Detects the wiring status for the analog outputs of module 01 and returns a valid response denoting that no wire is connected to channel 1.

Note:

This command is only applicable to type 30 (0 to 20mA) and type 31 (4 to 20mA).

2.25 $AAC

Description:

This command is used to clear the digital input/output latch data of a specified module.

Syntax:

$AAC[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be cleared in hexadecimal format (00 to FF)
C	The command to clear the digital input/output latch data

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: $01C Response: !01

Clears the latched data for module 01 and returns a valid response.

Related Command:

Section 2.33 $AALS

2.26 $AAD

Description:

This command is used to read whether the counter for each channel of a specified module is enabled or disabled.

Syntax:

$AAD[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
D	The command to read the status of the counter for each channel

Response:

Valid Response: **!AAnn[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
nn	A two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that the channel is disabled, and 1 denotes that the channel is enabled.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01D3A Response: !01

Enables the counters for channels 1, 3, 4, and 5 and disables the counters for all other channels on module 01. The module returns a valid response.

Command: $01D Response: !013A

Reads the status of the counters for module 01 and returns a valid response of “3A”, which denotes that the counters for channels 1, 3, 4, and 5 are enabled and the counters for all other channels are disabled.

Related Command:

Section 2.27 $AADnn

2.27 $AADnn

Description:

This command is used to specify whether the counter for a particular channel of a specified module is to be enabled.

Syntax:

$AADnn[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
D	The command to set the counter for the channel to enabled
nn	A two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that the channel is disabled, and 1 denotes that the channel is enabled.

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response. An invalid response is returned if an attempt is made to enable a channel that is not present.
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01D3A Response: !01

Enables the counters for channels 1, 3, 4, and 5 and disables the counters for all other channels on module 01. The module returns a valid response.

Command: $01D Response: !013A

Reads the status of the counters for module 01 and returns a valid response of “3A”, which denotes that the counters for channels 1, 3, 4, and 5 are enabled and the counters for all other channels are disabled.

Related Command:

Section 2.26 $AAD

2.28 $AAE

Description:

This command is used to read whether the edge of the counter for each channel of a specified module is rising or falling.

Syntax:

$AAE[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
E	The command to read the status of the edge of the counter for each channel

Response:

Valid Response: **!AAnn[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
nn	A two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that the edge of the counter for the channel is rising, and 1 denotes that the edge is falling.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01E3A Response: !01

Sets the edges of the counters for channels 1, 3, 4, and 5 on module 01 to falling and the sets edges of the counters for all other channels to rising. The module returns a valid response.

Command: $01E Response: !013A

Reads the status of the edges of the counters for module 01 and returns a valid response of “3A”, which denotes that the edges of the counters for channel 1, 3, 4, and 5 are falling and the edges of the counters for all other channels are rising.

Related Command:

Section 2.29 $AAEnn

2.29 $AAEnn

Description:
This command is used to set whether the edge of the counter for each channel of a specified module should be rising or falling.

Syntax:
$AAEnn[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
E	The command to set the counter for each channel to enabled
nn	A two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that the edge of the counter for the channel is rising, and 1 denotes that the edge is falling.

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01E3A Response: !01

Sets the edges of the counters for channels 1, 3, 4, and 5 on module 01 are falling and sets the edges of the counters for all other channels to rising. The module returns a valid response.

Command: $01E Response: !013A

Reads the status of the edges of the counters for module 01 and returns a valid response of “3A”, which denotes that the edges of the counters for channel 1, 3, 4, and 5 are falling and the edges of the counters for all other channels are rising.

Related Command:

Section 2.28 $AAE

2.30 $AAF

Description:

This command is used to read the firmware version information for a specified module.

Syntax:

$AAF[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
F	The command to read the firmware version information

Response:

Valid Response: **!AA(Data)[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A string indicating the firmware version information for the module

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: $01F Response: !01A2.0

Reads the firmware version information for module 01 and shows that it is version A2.0.

2.31 $AAI

Description:

This command is used to read the status of the INIT switch on a specified module.

Syntax:

$AAI[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
I	The command to read the status of the INIT switch on the module

Response:

Valid Response: **!AAS[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
S	The status of the INIT switch on the module 0: The INIT switch is currently in the INIT position 1: The INIT switch is currently in the Normal position

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: $01I Response: !010

Reads the status of the INIT switch on module 01 and shows that it is currently in the INIT position.

2.32 $AAM

Description:

This command is used to read the name of a specified module.

Syntax:

$AAM[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
M	The command to read the name of the module

Response:

Valid Response: **!AA(Name)[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Name)	A string showing the name of the module

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: $01M Response: !0187026P

Reads module 01 and returns the name “87026P”.

Related Command:

Section 2.50 ~AAO(Name)

2.33 $AALS

Description:
This command is used to read the status of the digital input/output latch for each channel of a specified module.

Syntax:
$AALS[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
L	The command to read the status of the latches for each channel
S	0 = Latch low 1 = Latch high

Response:

Valid Response:	**!(Data)[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The status of the latched digital output/input channels (a four-digit hexadecimal value followed by 00)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $01L1 Response: !030100

Reads module 01 and returns a valid response showing that the latches are high on digital output channels 0 and 1 and digital input channel 0.

Command: $01C Response: !01

Clears the digital input and output latch status for module 01 and returns a valid response.

Command: $01L1 Response: !000000

Reads the module 01 and returns a valid response showing that high latches have not occurred on any digital input or output channels.

Command: $01L2 Response: ?01

Attempts to read the module 01 and returns an invalid response indicating that “2” is over our definitions.

Related Commands:

Section 2.25 $AAC, Section 2.66 @AADODD

2.34 $AAS1

Description:
This command is used to reload the factory default calibration parameters for a specified module, including the internal calibration parameters.

Syntax:
$AAS1[CHKSUM](CR)

$	Delimiter character
AA	The address of the module where the default parameters are to be reloaded in hexadecimal format (00 to FF)
S1	The command to reload the factory default calibration parameters

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: $01S1 Response: !01

Sends a command to module 01 to reload the factory default calibration parameters and returns a valid response.

Related Commands:

Section 2.6 $AA0, Section 2.7 $AA1, Section 2.48 ~AAEV

Related Topics:

Section 1.2.1 Calibration

2.35 ~**

Description:

This command is used to inform all the modules on the network that the host is OK

Syntax:

~[CHKSUM](CR)**

~ Delimiter character

****** The Host OK command

Response:

There is no response to this command.

Example:

Command: ~** No response

Sends a “Host OK” command to all modules on the network.

Related Commands:

Section 2.36 ~AA0, Section 2.37 ~AA1, Section 2.38 ~AA2, Section 2.39 ~AA3EVV, Section 2.40 ~AA4N, Section 2.43 ~AA5PPSS

2.36 ~AA0

Description:

This command is used to read the status of the Host Watchdog for a specified module.

Syntax:

~AA0[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
0	The command to read the status of the module's Host Watchdog

Response:

Valid Response: **!AASS[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
SS	A two-digit hexadecimal value that represents the status of the Host Watchdog, where: Bit 2: 0 indicates that no Host Watchdog timeout has occurred, and 1 indicates that a Host Watchdog timeout has occurred. The status information for the Host Watchdog is stored in EEPROM and can only be reset using the ~AA1 command. Bit 7: 0 indicates that the Host Watchdog is disabled, and 1 indicates the Host Watchdog is enabled.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~010 Response: !0100

Reads the status of the Host Watchdog for module 01 and returns a valid response with a value of 00, meaning that the Host Watchdog is disabled and no Host Watchdog timeout has occurred.

Command: ~020 Response: !0204

Reads the status of the Host Watchdog for module 02 and returns a valid response with a value of 04, meaning that a Host Watchdog timeout has occurred.

Related Commands:

Section 2.35 ~**, Section 2.37 ~AA1, Section 2.38 ~AA2, Section 2.39 ~AA3EVV

2.37 ~AA1

Description:

This command is used to reset the timeout status of the Host Watchdog.

Syntax:

~AA1[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be reset in hexadecimal format (00 to FF)
1	The command to reset the timeout status of the Host Watchdog

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~010 Response: !0104

Reads the status of the Host Watchdog for module 01 and returns a valid response showing that a Host Watchdog timeout has occurred.

Command: ~011 Response: !01

Resets the Host Watchdog timeout for module 01 and returns a valid response.

Command: ~010 Response: !0100

Reads the status of the Host Watchdog for module 01 and shows that no Host Watchdog timeout has occurred.

Related Commands:

Section 2.35 ~**, Section 2.36 ~AA0, Section 2.38 ~AA2, Section 2.39 ~AA3EVV

2.38 ~AA2

Description:

This command is used to read the Host Watchdog timeout value for a specified module.

Syntax:

~AA2[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
2	The command to read the Host Watchdog timeout value

Response:

Valid Response: **!AAEVV[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
E	0: The Host Watchdog is disabled 1: The Host Watchdog is enabled
VV	Two hexadecimal digits to represent the Host Watchdog timeout value in tenths of a second, for example, 01 means 0.1 seconds and FF means 25.5 seconds.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: ~013164 Response: !01

Enables the Host Watchdog for module 01 and sets the Host Watchdog timeout value to 10.0 seconds. The module returns a valid response.

Command: ~012 Response: !011FF

Reads the Host Watchdog timeout value for module 01 and returns a valid response with a value of FF, meaning that the Host Watchdog is enabled and the Host Watchdog timeout value is 25.5 seconds.

Related Commands:

Section 2.35 ~**, Section 2.36 ~AA0, Section 2.37 ~AA1, Section 2.39 ~AA3EVV

2.39 ~AA3EVV

Description:

This command is used to enable/disable the Host Watchdog for a specified module and set the Host Watchdog timeout value.

Syntax:

~AA3EVV[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
3	The command to set the Host Watchdog
E	0: Disables the Host Watchdog 1: Enables the Host Watchdog
VV	Two hexadecimal digits to represent the Host Watchdog timeout value in tenths of a second, for example, 01 means 0.1 seconds and FF means 25.5 seconds.

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~013164 Response: !01

Enables the Host Watchdog for module 01 and sets the Host Watchdog timeout value to 10.0 seconds. The module returns a valid response.

Command: ~012 Response: !01164

Reads the Host Watchdog timeout value for module 01 and returns a valid response with a value of 164, meaning that the Host Watchdog is enabled and the Host Watchdog timeout value is 10.0 seconds.

Related Commands:

Section 2.35 ~**, Section 2.36 ~AA0, Section 2.37 ~AA1, Section 2.38 ~AA2

2.40 ~AA4N

Description:
This command is used to read the analog output safe value for channel N of a specified module.

Syntax:
~AA4N[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
4	The command to read the analog output safe value
N	The channel to be read, zero based

Response:

Valid Response:	**!AA(Data)[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**
!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The analog output safe value, see Section 1.3 for details of the data format.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #010+06.000 Response: >

Outputs the channel 0 of the module 01 to 6V and returns a valid response.

Command: ~0150 Response: !01

Sets the current analog output value for channel 0 of module 01 as the safe value and returns a valid response.

Command: ~0140 Response: !01+06.000

Reads the analog output safe value for channel 0 of module 01 and returns a valid response with a value of 6V.

Command: ~014F Response: ?01

Attempts to read the analog output safe value for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.5 #AAN(Data), Section 2.42 ~AA5N

Related Topics:

Section 1.3 Configuration Tables

2.41 ~AA4

Description:
This command is used to read the digital output power-on/safe value for a specified module.

Syntax:
~AA4[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
4	The command to read the digital output power-on/safe value

Response:
Valid Command: **!AAPPSS[CHKSUM](CR)**
Invalid Command: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid command
?	Delimiter character for an invalid command
AA	The address of the responding module in hexadecimal format (00 to FF)
PP	Two hexadecimal digits to represent the digital output power-on value
SS	Two hexadecimal digits to represent the digital output safe value.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address

Note:
Neither the power-on value nor the safe value have any effect on digital outputs those are associated with alarm outputs.

Examples:
Command: ~0150300 Response: !01

Sets the digital output power-on value to 03 and sets the digital output safe value to 00, and returns a valid response.

Command: ~014 Response: !010300

Reads the digital output power-on value and the digital output safe value for module 01 and returns a valid response indicating a power on value of 03 and safe value of 00.

Related Commands:
Section 2.35 ~**, Section 2.36 ~AA0, Section 2.37 ~AA1, Section 2.38 ~AA2, Section 2.39 ~AA3EVV, Section 2.43 ~AA5PPSS

2.42 ~AA5N

Description:
This command is used to set the analog output safe value for Channel N of a specified module.

Syntax:
~AA5N[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
5	The command to set the analog output safe value
N	The channel to be set, zero based

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**
!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: #010+06.000 Response: >

Outputs the channel 0 of the module 01 to 6V and returns a valid response.

Command: ~0150 Response: !01

Sets the current analog output value for channel 0 of module 01 as the safe value and returns a valid response.

Command: ~0140 Response: !01+06.000

Reads the analog output safe value for channel 0 of module 01 and returns a valid response with a value of 6V.

Command: ~015F Response: ?01

Attempts to set the current analog output value for channel 15 of module 01 as the safe value and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.5 #AAN(Data), Section 2.40 ~AA4N

Related Topics:

Section 1.3 Configuration Tables

2.43 ~AA5PPSS

Description:
This command is used to set the digital output power-on and safe values for a specified module.

Syntax:
~AA5PPSS[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
5	The command to set the digital output power-on and safe values
PP	Two hexadecimal digits to represent the digital output power-on value
SS	Two hexadecimal digits to represent the digital output safe value

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Note:
Neither the power-on value nor the safe value have any effect on digital outputs those are associated with alarm outputs.

Examples:
Command: ~0150102 Response: !01

Sets the digital output power-on value to 01 and sets the digital output safe value to 02, and returns a valid response.

Command: ~014 Response: !010102

Reads the digital output power-on value and the digital output safe value for module 01 and returns a valid response with a value of 0102, which denotes that the digital output power-on value is 01 and digital output safe value is 02.

Related Commands:
Section 2.35 ~**, Section 2.36 ~AA0, Section 2.37 ~AA1, Section 2.38 ~AA2, Section 2.39 ~AA3EVV, Section 2.41 ~AA4

2.44 ~AA6PN(Data)

Description:
This command is used to set the analog output power-on value for a specific channel of a specified module.

Syntax:
~AA6PN(Data) [CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
6P	The command to set the analog output power-on value
N	The channel to be set, zero based
(Data)	The analog output value, see Section 1.3 for details of the data format.

Response:
Valid Response: **!AA[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~016P1+05.000 Response: !01

Sets the analog output power-on value for channel 1 of module 01 to 5V and returns a valid response.

Command: ~016P0+25.000 Response: ?01

Attempts to set the analog output power-on value for channel 0 of module 01 to 25V and returns an invalid response, meaning that the output value is out of range.

Command: $0171 Response: !01+05.000

Reads the power-on value for channel 1 of module 01 and returns a valid response of +5V.

Related Commands:

Section 2.13 $AA4N, Section 2.18 $AA7N

Related Topics:

Section 1.3 Configuration Tables

2.45 ~AA6SN(Data)

Description:
This command is used to set the analog output safe value for a specific channel of a specified module.

Syntax:
~AA6PN(Data) [CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
6S	The command to set the analog output safe value
N	The channel to be set, zero based
(Data)	The analog output value, see Section 1.3 for details of the data format.

Response:
Valid Response: **!AA[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~016S1+05.000 Response: !01

Sets the analog output safe value for channel 1 of module 01 to 5V and returns a valid response.

Command: ~016S0+25.000 Response: ?01

Attempts to set the analog output safe value for channel 0 of module 01 to 25V and returns an invalid response, meaning that the output value is out of range.

Command: ~0141 Response: !01+05.000

Reads the safe value for channel 1 of module 01 and returns a valid response of +05.000 (5V).

Related Commands:

Section 2.40 ~AA4N, Section 2.42 ~AA5N

Related Topics:

Section 1.3 Configuration Tables

2.46 ~AAD

Description:

This command is used to read the miscellaneous settings for a specified module.

Syntax:

~AAD[CHKSUM](CR)

~ Delimiter character

AA The address of the module to be read in hexadecimal format (00 to FF)

D The command to read the miscellaneous settings

Response:

Valid Response: **!AAVV[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

! Delimiter character for a valid response

? Delimiter character for an invalid response

AA The address of the responding module in hexadecimal format (00 to FF)

VV A two-digit hexadecimal value that represents the miscellaneous settings as indicated in the following tables:

7	6	5	4	3	2	1	0
Reserved						OA	IA

Key	Description
OA	DO active state 0: Output value 1 indicates the relay is active Output value 0 indicates the relay is inactive 1: Output value 0 indicates the relay is active Output value 1 indicates the relay is inactive
IA	DI active state 0: Input value 1 indicates that there is no signal or the voltage is low Input value 0 indicates high voltage 1: Input value 0 indicates that there is no signal or the voltage is low Input value 1 indicates high voltage

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: ~$01D01 Response: !01

Sets the miscellaneous settings for module 01 to 01 meaning that the digital input will be inversed and returns a valid response.

Command: ~$01D Response: !0101

Reads the miscellaneous settings of module 01 and returns a value of 01 indicating that the digital input will be inversed.

Related Command:

Section 2.47 ~AADVV

2.47 ~AADVV

Description:

This command is used to set the miscellaneous settings for a specified module.

Syntax:

~AADVV[CHKSUM](CR)

~ Delimiter character

AA The address of the module to be set in hexadecimal format (00 to FF)

D The command to set the miscellaneous settings

VV A two-digit hexadecimal value that represents the miscellaneous settings as indicated in the following tables:

7	6	5	4	3	2	1	0
Reserved						OA	IA

Key	Description
OA	DO active state 0: Output value 1 indicates the relay is active Output value 0 indicates the relay is inactive 1: Output value 0 indicates the relay is active Output value 1 indicates the relay is inactive
IA	DI active state 0: Input value 1 indicates that there is no signal or the voltage is low Input value 0 indicates high voltage 1: Input value 0 indicates that there is no signal or the voltage is low Input value 1 indicates high voltage

Response:

Valid Response: **!AA [CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

! Delimiter character for a valid response
? Delimiter character for an invalid response
AA The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: ~$01D01 Response: !01

Sets the miscellaneous settings for module 01 to 01 meaning that the digital input will be inversed and returns a valid response.

Command: ~$01D Response: !0101

Reads the miscellaneous settings of module 01 and returns a value of 01 indicating that the digital input will be inversed.

Related Command:

Section 2.46 ~AAD

2.48 ~AAEV

Description:

This command is used to enable or disable calibration on a specified module.

Syntax:

~AAEV[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be calibrated in hexadecimal format (00 to FF)
E	The command to enable/disable calibration
V	0: Disables calibration 1: Enables calibration

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: $010 Response: ?01

Attempts to send the command to perform a span calibration on module 01 and returns an invalid response because the “enable calibration” command, ~AAEV, was not sent in advance.

Command: ~01E1 Response: !01

Enables calibration on module 01 and returns a valid response.

Command: $010 Response: !01

Sends the command to perform a span calibration on module 01 and returns a valid response.

Related Commands:

Section 2.6 $AA0, Section 2.7 $AA1

Related Topics:

Section 1.2.1 Calibration

2.49 ~AAI

Description:
This command is used to enable modification of the Baud Rate and checksum settings for a specified module using the software INIT function only.

Syntax:
~AAI[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
I	The command to set the software INIT function

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**
!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: ~01T10 Response: !01

Sets the timeout value of the software INIT function on module 01 to 16 seconds and returns a valid response..

Command: $01I Response: !01

Sets the software INIT function on module 01 to be enabled and returns a valid response.

Command: %0101000600 Response: !01

Sets the Baud Rate for module 01 to 9600bps and returns a valid response.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.53 ~AATnn

2.50 ~AAO(Data)

Description:

This command is used to set the name of a specified module

Syntax:

~AAO(Name)[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
O	The command to set the module name
(Name)	The new name of the module (max. 6 characters).

Response:

Valid Response: **!AA[CHKSUM](CR)**

Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~01O87026P Response: !01

Sets the name of module 01 to “87026P” and returns a valid response.

Command: $01M Response: !0187026P

Reads module 01 and returns the name “87026P”.

Related Commands:

Section 2.32 $AAM

2.51 ~AARD

Description:
This command is used to read the response delay time for a specified module.

Syntax:
~AARD[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RD	The command to read the response delay time

Response:

Valid Response: **!AATT[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
TT	Two hexadecimal digits to represent the response time value in milliseconds. For example, 01 denotes 1 millisecond and 1A denotes 26 milliseconds. The value must be less than or equal to 1E.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~01RD10 Response: !01

Sets the response delay time to 16 milliseconds and returns a valid response..

Command: ~01RD Response: !0110

Reads the response delay time and returns a value of 16 milliseconds. The response will be sent after 16 milliseconds have elapsed.

Related Command:

Section 2.52 ~AARDTT

2.52 ~AARDTT

Description:
This command is used to set the response delay time for a specified module.

Syntax:
~AARDTT[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
RD	The command to set the response delay time
TT	Two hexadecimal digits to represent the response time value in milliseconds. For example, 01 denotes 1 millisecond and 1A denotes 26 milliseconds. The value must be less than or equal to 1E.

Response:

Valid Response:	**!AA [CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~01RD10 Response: !01

Sets the response delay time to 16 milliseconds and returns a valid response..

Command: ~01RD Response: !0110

Reads the response delay time and returns a value of 10 indicating 16 milliseconds. The response will be sent after 16 milliseconds have elapsed.

Related Command:

Section 2.51 ~AARD

2.53 ~AATnn

Description:
This command is used to set the timeout value for the software INIT function on a specified module.

Syntax:
~AARDTT[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
T	The command to set the time out value for the software INIT function
nn	Two hexadecimal digits to represent the timeout value for the software INIT function in seconds. For example, 01 denotes 1 second and 1A denotes 26 seconds. The value must be less than or equal to 3C.

Response:

Valid Response:	**!AA [CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: ~01T10 Response: !01

Sets the timeout value of the software INIT function on module 01 to 16 seconds and returns a valid response.

Command: $01I Response: !01

Sets the software INIT function on module 01 to be enabled and returns a valid response.

Command: %0101000600 Response: !01

Sets the Baud Rate for module 01 to 9600bps and returns a valid response.

Command: ~01TFF Response: ?01

Attempts to set the timeout value of the software INIT function on module 01 to 255 seconds and returns an invalid response because it is over the limit.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.49 ~AAI

2.54 @AACECi

Description:
This command is used to reset the counter for a specific channel of a specified module.

Syntax:
@AACECi[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be reset in hexadecimal format (00 to FF)
CE	The command to reset the counter
Ci	i specifies the channel to be reset, zero based

Response:

Valid Response:	**!AA [CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01REC1 Response: !0100000008

Reads data from channel 1 of module 01 and returns a counter value of 00000008.

Command: @01CEC1 Response: !01

Resets the counter for channel 1 of module 01 to the preset value and returns a valid response.

Command: @01REC1 Response: !0100000000

Reads data from channel 1 of module 01 and returns a counter value of 00000000.

Command: @01CECF Response: !01

Attempts to reset the counter for channel 15 of module 01 to the preset value and returns an invalid response because channel 15 does not exist.

Related Command:

Section 2.68 @AARECi

2.55 @AACH

Description:
This command is used to clear the high latch values for all channels of a specified module.

Syntax:
@AACH [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be cleared in hexadecimal format (00 to FF)
CH	The command to clear the high latch values

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01RH0 Response: !01+05.000

Reads the high latch value for channel 0 of module 01 and returns a value of +05.000 (5V).

Command: @01CH Response: !01

Clears the high latch value for all channels of module 01 and returns a valid response.

Command: @01RH0 Response: !01+00.000

Reads the high latch value for channel 0 of module 01 and returns a value of +00.000 (0V).

Related Commands:

Section 2.56 @AACHi, Section 2.69 @AARH, Section 2.70 @AARHi

2.56 @AACHi

Description:
This command is used to clear the high latch value for a specific channel of a specified module.

Syntax:
@AACHi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be cleared in hexadecimal format (00 to FF)
CH	The command to clear the high latch value
i	The channel to be cleared, zero based

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**
!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01RH1 Response: !01+06.000

Reads the high latch value for channel 1 of module 01 and returns a value +06.000 (6V).

Command: @01CH1 Response: !01

Clears the high latch value for channel 1 of module 01 and returns a valid response.

Command: @01RH1 Response: !01+00.000

Reads the high latch value for channel 1 of module 01 and returns a value +00.000 (0V).

Command: @01CHF Response: !01

Attempts to clear the high latch value for channel 15 of module 01 and returns an invalid response because channel 15 does not exist.

Related Commands:

Section 2.55 @AACH, Section 2.69 @AARH, Section 2.70 @AARHi

2.57 @AACHCi

Description:
This command is used to clear the status of the high alarm for a specific channel of a specified module.

Syntax:
@AACHCi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be cleared in hexadecimal format (00 to FF)
CH	The command to clear the status of the high alarm
Ci	The channel to be cleared, zero based

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01CHC0 Response: !01

Clears the high alarm status for channel 0 of module 01 and returns a valid response.

Command: @01CHCF Response: !01

Attempts to clear the high alarm status for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.62 @AADHCi, Section 2.64 @AAHI(Data)CiTOj, Section 2.67 @AARAOj, Section 2.75 @AAROOj

2.58 @AACL

Description:
This command is used to clear the low latch values for all channels of a specified module.

Syntax:
@AACL [CHKSUM](CR)
@ Delimiter character
AA The address of the module to be cleared in hexadecimal format (00 to FF)
CL The command to clear the low latch values

Response:
Valid Response: **!AA[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**
! Delimiter character for a valid response
? Delimiter character for an invalid response
AA The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01RL0 Response: !01-05.000

Reads the low latch value for channel 0 of module 01 and returns a value of -05.000 (-5V).

Command: @01CL Response: !01

Clears the low latch value for all channels on module 01 and returns a valid response.

Command: @01RL0 Response: !01+00.000

Reads the low latch value for channel 0 of module 01 and returns a value of +00.000 (0V).

Related Commands:

Section 2.59 @AACLi, Section 2.72 @AARL, Section 2.73 @AARLi

2.59 @AACLi

Description:
This command is used to clear the low latch value for a specific channel of specified module.

Syntax:
@AACLi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be cleared in hexadecimal format (00 to FF)
CL	The command to clear the low latch value
i	The channel to be cleared, zero based

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**
!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01RL1 Response: !01-06.000

Reads the low latch value for channel 1 of module 01 and returns a value of -06.000 (-6V).

Command: @01CL1 Response: !01

Clears the low latch value for channel 1 of module 01 and returns a valid response.

Command: @01RL1 Response: !01+00.000

Reads the low latch value for channel 1 of module 01 and returns a value of +00.000 (0V).

Related Commands:

Section 2.58 @AACL, Section 2.72 @AARL, Section 2.73 @AARLi

2.60 @AACLCi

Description:
This command is used to clear the status of the low alarm for a specific channel of a specified module.

Syntax:
@AACLCi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be cleared in hexadecimal format (00 to FF)
CL	The command to clear the status of the low alarm
Ci	The channel to be cleared, zero based

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01CHC7 Response: !01

Clears the low alarm status for channel 7 of module 01 and returns a valid response.

Command: @01CHCF Response: !01

Clears the low alarm status for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.63 @AADLCi, Section 2.65 @AALO(Data)CiTOj, Section 2.67 @AARAOj, Section 2.75 @AAROOj

2.61 @AADI

Description:
This command is used to read the status of the digital input and digital output channels of a specified module.

Syntax:
@AADI [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
DI	The command to read the status of the DI/DOO channels

Response:

Valid Response:	**!AAHHLL[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
HH	A two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that the DO channel has no output, and 1 denotes that the DO channel has output.
LL	A two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that the DI channel has no input, and 1 denotes that the DI channel has input.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01DO01 Response: !01

Sets digital output channel 0 of module 01 to on, sets digital output channel 1 of module 01 to off and returns a valid response.

Command: @01DI Response: !010102

Reads the status of the DI/DO channels of module 01 and returns a response of 010102 indicating that DO exists on channel 0 of module 01 and that DI exists on channel 1 of module 01.

Related Commands:

Section 2.62 @AADHCi, Section 2.63 @AADLCi, Section 2.64 @AAHI(Data)CiT, Section 2.65 @AALO(Data)CiT, Section 2.66 @AADODD

2.62 @AADHCi

Description:
This command is used to disable the high alarm for a specific channel of a specified module.

Syntax:
@AADHCi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
DH	The command to disable the high alarm
Ci	The channel where the alarm is to be disabled, zero based

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01DHC0 Response: !01

Disables the high alarm for channel 0 of module 01 and returns a valid response.

Command: @01DHCF Response: ?01

Attempts to disable the high alarm for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.57 @AACHCi, Section 2.64 @AAHI(Data)CiTOj, Section 2.67 @AARAOj, Section 2.71 @AARHCi, Section 2.75 @AAROOj

2.63 @AADLCi

Description:
This command is used to disable the low alarm for a specific channel of a specified module.

Syntax:
@AADLCi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
DL	The command to disable the low alarm
Ci	The channel where the low alarm is to be disabled, zero based

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01DLC5 Response: !01

Disables the low alarm for channel 5 of module 01 and returns a valid response.

Command: @01DLCF Response: ?01

Attempts to disable the low alarm for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.60 @AACLCi, Section 2.65 @AALO(Data)CiTOj, Section 2.67 @AARAOj, Section 2.74 @AARLCi, Section 2.75 @AAROOj

2.64 @AAHI(Data)CiTOj

Description:
This command is used to set the high alarm limit for a specific channel of a specified module.

Syntax:
@AAHI(Data)CiTOj [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
HI	The command to set the high alarm
(Data)	The high alarm limit. This should be consistent with the data format. Refer to Section 1.3 for details.
Ci	The channel to be set, zero based
T	The alarm type: M: Momentary alarm L: Latched alarm
Oj	j specifies the digital output port to be used for the alarm ouput

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01HI+09.000C0MO1 Response: !01

Sets the high alarm limit for channel 0 of module 01 to +09.000 (9V). The alarm type is set to momentary and digital output channel 1 of module 01 is set as the high alarm output. A valid response is returned.

Command: @01RHC0 Response: !01+09.000MO1

Reads the status of the high alarm and returns a valid response indicating that the high alarm limit for channel 0 of module 01 is +09.000 (9V), the alarm type is set to momentary and digital output channel 1 of module 01 is set as the high alarm output.

Command: @01HI+09.000CFMO1 Response: ?01

Attempts to set the high alarm limit for channel 15 of module 01 to +09.000 (9V). The alarm type is set to momentary and digital output channel 1 of module 01 is set as the high alarm output. An invalid response is returned because channel 15 does not exist.

Related Commands:

Section 2.57 @AACHCi, Section 2.62 @AADHCi, Section 2.67 @AARAOj, Section 2.71 @AARHCi, Section 2.75 @AAROOj

Related Topics:

Section 1.3 Configuration Tables

2.65 @AALO(Data)CiTOj

Description:
This command is used to set the low alarm limit for a specific channel of a specified module.

Syntax:
@AALO(Data)CiTOj [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
LO	The command to set the low alarm
(Data)	The low alarm limit. This should be consistent with the data format. Refer to Section 1.3 for details.
Ci	The channel to be set, zero based
T	The alarm type: M: Momentary alarm L: Latched alarm
Oj	j specifies the digital output port to be used for the alarm output

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Examples:

Command: @01LO-03.000C1LO0 Response: !01

Sets the low alarm limit for channel 1 of module 01 to -03.000 (-3V). The alarm type is set to latched and digital output channel 0 of module 01 is set as the low alarm output. A valid response is returned.

Command: @01RLC1 Response: !010-03.000LO0

Reads the status of the low alarm and returns a valid response indicating that the low alarm limit for channel 1 of module 01 is -03.000 (-3V), the alarm type is set to latched and digital output channel 0 of module 01 is set as the low alarm output.

Command: @01LO-03.000CFLO0 Response: ?01

Attempts to set the low alarm limit for channel 15 of module 01 to -03.000 (-3V). The alarm type is set to latched and digital output channel 15 of module 01 is set as the low alarm output. An invalid response is returned because channel 15 does not exist.

Related Commands:

Section 2.60 @AACLCi, Section 2.63 @AADLCi, Section 2.67 @AARAOj, Section 2.74 @AARLCi, Section 2.75 @AAROOj

Related Topics:

Section 1.3 Configuration Tables

2.66 @AADODD

Description:
This command is used to set the status of the digital output for a specified module.

Syntax:
@AADODD[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
DO	The command to set the digital output ports
DD	A two-digit hexadecimal value, where bit 0 corresponds to DO0, bit 1 corresponds to DO1, etc. When the bit is 0, it denotes that the digital output port is off, and 1 denotes that the digital output port is on.

Response:

Valid Response:	**!AA[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Notes:

1. If the digital output port is already set as an alarm output port, the value written to the port will be ignored.

2. If a Host Watchdog timeout occurs, the module will return an invalid response for this command and the DO value that was sent will be ignored.

Example:

Command: @01DO01 Response: !01

Sets digital output channel 0 of module 01 to on, sets digital output channel 1 of module 01 to off and returns a valid response.

Command: @01DI Response: !010102

Reads the status of the DI/DO channels of module 01 and returns a response of 010102 indicating that DO exists on channel 0 of module 01 and that DI exists on channel 1 of module 01.

Related Command:

Section 2.61 @AADI

2.67 @AARAOj

Description:
This command is used to read which currently activated alarms are associated with a specific digital output port on a specified module.

Syntax:
@AARAOj[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RA	The command to read the currently activated alarms associated with a specific digital output port
Oj	j specifies the digital output port

Response:
Valid Response: **!AAHHLL[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
HH	A two-digit hexadecimal value to represent the currently activated high alarms associated with the digital output port, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that there are no activated high alarms associated with the channel. When the bit is 1, it denotes that there is an activated high alarm associated with the channel.

LL A two-digital hexadecimal value to represent the currently activated low alarms associated with the digital output port, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that there are no activated low alarms associated with the channel. When the bit is 1, it denotes that there is an activated low alarm associated with the channel.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01RAO1 Response: !011122

Reads the currently activated alarms associated with digital output channel 1 of module 01. The module returns a valid response with a value of 1122, which denotes that there is currently an activated high alarm associated with channels 0 and 4, and an activated low alarm associated with channels 1 and 5.

Command: @01RAOF Response: !011122

Reads the currently activated alarms associated with digital output channel 1 of module 01. The module returns an invalid response because channel 15 does not exist.

Related Commands:

Section 2.57 @AACHCi, Section 2.60 @AACLCi, Section 2.62 @AADHCi, Section 2.63 @AADLCi, Section 2.64 @AAHI(Data)CiTOj, Section 2.65 @AALO(Data)CiTOj, Section 2.75 @AAROOj

2.68 @AARECi

Description:
This command is used to read the digital input counter value for a specific channel of a specified module.

Syntax:
@AARECi[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RE	The command to read the digital input counter value
Ci	i specifies the channel to be read, zero based

Response:

Valid Response:	**!AA(Data)[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The digital input counter value for the specified channel

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01REC1 Response: !0100000008

Reads data from channel 1 of module 01 and returns a counter value of 00000008.

Command: @01CEC1 Response: !01

Resets the counter for channel 1 of module 01 to the preset value and returns a valid response.

Command: @01REC1 Response: !0100000000

Reads data from channel 1 of module 01 and returns a counter value of 00000000.

Related Command:

Section 2.54 @AACECi

2.69 @AARH

Description:
This command is used to read the high latch values for all channels of a specified module.

Syntax:
@AARH [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RH	The command to read the high latch values for all channels

Response:

Valid Response:	**!AA(Data)[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The high latch values for all channels. See Section 1.3 for details of the data format.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01RH Response:
!01+08.000+00.000+00.000+00.000+00.000+00.000

Reads the high latch values from module 01 and returns a valid response with the data in engineering format.

Related Commands:

Section 2.55 @AACH, Section 2.56 @AACHi, Section 2.70 @AARHi

Related Topics:

Section 1.3 Configuration Tables

2.70 @AARHi

Description:
This command is used to read the high latch value for a specific channel of a specified module.

Syntax:
@AARHi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RH	The command to read the high latch value
i	The channel to be read, zero based

Response:

Valid Response:	**!AA(Data)[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**
!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The high latch value for the specified channel. See Section 1.3 for details of the data format.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01RH0 Response: !01+08.000

Reads the high latch value for channel 0 of module 01 and returns a valid response with the value of +08.000 (8V) in engineering format.

Command: @01RHF Response: ?01

Attempts to read the high latch value for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.55 @AACH, Section 2.56 @AACHi, Section 2.69 @AARH

Related Topics:

Section 1.3 Configuration Tables

2.71 @AARHCi

Description:
This command is used to read the status of the high alarm for a specific channel of a specified module.

Syntax:
@AARHCi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RH	The command to read the status of the high alarm
Ci	The channel to be read, zero based

Response:
Valid Response: **!AA(Data)SOj[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The status of the high alarm for the specified channel. See Section 1.3 for details of the data format.
S	The alarm type: 0: Alarm disabled 1: Momentary alarm 2: Latched alarm
Oj	j specifies the digital output port

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01RHC0 Response: !01+08.0002O0

Reads the status of the high alarm for channel 0 of module 01 and returns a valid response indicating that the high alarm limit is +08.000 (8V), the alarm type is latched and the high alarm output channel is digital output channel 0.

Command: @01RHCF Response: ?01

Attempts to read the status of the high alarm for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.57 @AACHCi. Section 2.62 @AADHCi, Section 2.64 @AAHI(Data)CiTOj

Related Topics:

Section 1.3 Configuration Tables

2.72 @AARL

Description:
This command is used to read the low latch values for all channels of a specified module.

Syntax:
@AARL [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RL	The command to read the low latch values for all channels

Response:

Valid Response:	**!AA(Data)[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**
!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The low latch values for all channels. See Section 1.3 for details of the data format.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01RL Response:
!01-02.000+00.000+00.000+00.000+00.000+00.000

Reads the low latch values from module 01 and returns a valid response with the data in engineering format.

Related Commands:

Section 2.58 @AACL, Section 2.59 @AACLi, Section 2.73 @AARLi

Related Topics:

Section 1.3 Configuration Tables

2.73 @AARLi

Description:
This command is used to read the low latch value for a specific channel of a specified module.

Syntax:
@AARLi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RL	The command to read the low latch value
i	The channel to be read, zero based

Response:

Valid Response:	**!AA(Data)[CHKSUM](CR)**
Invalid Response:	**?AA[CHKSUM](CR)**
!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The high latch value for the specified channel. See Section 1.3 for details of the data format.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01RL0 Response: !01-02.000

Reads the low latch value for channel 0 of module 01 and returns a valid response with the value of -02.000 (-2V) in engineering format.

Command: @01RLF Response: ?01

Attempts to read the low latch value for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.58 @AACL, Section 2.59 @AACLi, Section 2.72 @AARL

Related Topics:

Section 1.3 Configuration Tables

2.74 @AARLCi

Description:
This command is used to read the status of the low alarm for a specific channel of a specified module.

Syntax:
@AARLCi [CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RL	The command to read the status of the low alarm
Ci	The channel to be read, zero based

Response:
Valid Response: **!AA(Data)SOj[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The status of the low alarm for the specified channel. See Section 1.3 for details of the data format.
S	The alarm type: 0: Alarm disabled 1: Momentary alarm 2: Latched alarm
Oj	j specifies the digital output port

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01RLC0 Response: !01-03.0001O1

Reads the status of the low alarm for channel 0 of module 01 and returns a valid response indicating that the low alarm limit is -03.000 (-3V), the type is momentary and the low alarm output channel is digital output channel 1.

Command: @01RLCF Response: ?01

Attempts to read the status of the low alarm for channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.60 @AACLCi, Section 2.63 @AADLCi, Section 2.65 @AALO(Data)CiTOj

Related Topics:

Section 1.3 Configuration Tables

2.75 @AAROOj

Description:
This command is used to read which alarms are associated with a specific digital output port on a specified module.

Syntax:
@AARAOj[CHKSUM](CR)

@	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
RO	The command to read the alarms associated with a specific digital output port
Oj	j specifies the digital output port

Response:
Valid Response: **!AAHHLL[CHKSUM](CR)**
Invalid Response: **?AA[CHKSUM](CR)**

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	The address of the responding module in hexadecimal format (00 to FF)
HH	A two-digit hexadecimal value to represent the high alarms associated with a specified digital output port, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 0, it denotes that the high alarm associated with the channel is disabled. When the bit is 1, it denotes that the high alarm associated with the channel is activated.
LL	A two-digit hexadecimal value to represent the low alarms associated with the digital output port, where bit 0 corresponds to channel 0, bit 1

corresponds to channel 1, etc. When the bit is 0, it denotes that the low alarm associated with the channel is disabled. When the bit is 1, it denotes that the low alarm associated with the channel is activated.

There will be no response if the command syntax is incorrect, there is a communication error, or there is no module with the specified address.

Example:

Command: @01ROO1 Response: !011122

Reads the alarms associated with digital output channel 1 of module 01. The module responds with a value of 1122, meaning that the high alarms on channels 0 and 4 and the low alarms on channels 1 and 5 are associated with digital output channel 1.

Command: @01ROOF Response: ?01

Attempts to read the alarms associated with digital output channel 15 of module 01 and returns an invalid response indicating that channel 15 does not exist.

Related Commands:

Section 2.57 @AACHCi, Section, 2.60 @AACLCi, Section 2.62 @AADHCi, Section 2.63 @AADLCi, Section 2.64 @AAHI(Data)CiTOj, Section 2.65 @AALO(Data)CiTOj, Section 2.67 @AARAOj