

8017, 8018, 8019, 8017, M-8018, M-8019 Series User's Manual

Warranty

All products manufactured by SuperLogics are under warranty regarding defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

SuperLogics assumes no liability for damages resulting from the use of this product. SuperLogics reserves the right to change this manual at any time without notification. The information furnished by SuperLogics is believed to be accurate and reliable. However, no responsibility is assumed by SuperLogics for its use, or for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Copyright 1999 - 2004. All rights reserved.

Trademark

The names used for identification only may be registered trademarks of their respective companies.

Date: 2004/6/1

Table of Contents

1. Introduction	5
1.1 More Information	7
1.2 Terminal Assignment	8
1.3 Specifications	13
1.4 Block Diagrams	16
1.4.1 Block diagram for the 8017, 8017F, 8017 and 8017F	16
1.4.2 Block diagram for the 8017C and 8017C	16
1.4.3 Block diagram for the 8017R and 8017R	17
1.4.4 Block diagram for the 8018, 8018P, 8018 and 8018P	17
1.4.5 Block diagram for the 8018BL and 8018BL	18
1.4.6 Block diagram for the 8018R and 8018R	18
1.4.7 Block diagram for the 8019 and 8019	19
1.4.8 Block diagram for the 8019R and 8019R	19
1.5 Dimensions	20
1.5.1 Modules without Frame Ground	20
1.5.2 Modules with Frame Ground	21
1.6 Wiring Diagrams	22
1.6.1 Wiring diagram for the 8017 and 8017F	22
1.6.2 Wiring diagram for the 8017C, 8017RC, 8017C and 8017RC	23
1.6.3 Wiring diagram for the 8017, 8017F, 8017R and 8017R	23
1.6.4 Wiring diagram for the 8018, 8018P, 8018BL, 8018R, 8018, 8018P, 8018BL and 8018R	24
1.6.5 Wiring diagram for the 8019, 8019R, 8019 and 8019R	24
1.7 Jumper Settings	25
1.7.1 8017 and 8017F Jumper Settings	25
1.7.2 8019, 8019R, 8019 and 8019R Jumper Settings	26
1.8 Quick Start	27
1.9 Default Settings	29
1.10 Calibration	30
1.11 Configuration Tables	32
1.12 8000 Notes	39
1.12.1 Protocol Switching	39
1.12.2 INIT Mode	40
1.13 Mounting	41
1.13.1 DIN Rail Mounting	41

1.13.2 Piggyback Mounting	43
1.13.3 Wall Mounting.....	44
1.14 Technical Support.....	45
2. DCON Protocol	46
2.1 %AANNTTCFF	50
2.2 #**	54
2.3 #AA	56
2.4 #AAN	58
2.5 \$AA0	60
2.6 \$AA1	62
2.7 \$AA0Ci.....	64
2.8 \$AA1Ci.....	66
2.9 \$AA2	68
2.10 \$AA3	70
2.11 \$AA4	72
2.12 \$AA5VV	74
2.13 \$AA6	76
2.14 \$AA7CiRrr	78
2.15 \$AA8Ci	80
2.16 \$AA9	82
2.17 \$AA9SNNNN	84
2.18 \$AAA	86
2.19 \$AAA	88
2.20 \$AAAi	90
2.21 \$AAB.....	92
2.22 \$AAF	94
2.23 \$AAM.....	95
2.24 \$AAP	97
2.25 \$AAPN	99
2.26 \$AAS0	101
2.27 \$AAS1	103
2.28 ~AAC	105
2.29 ~AACN.....	107
2.30 ~AAEV.....	109
2.31 ~AAI.....	111
2.32 ~AAO(Name).....	113
2.33 ~AATnn.....	115
2.34 ~**	118
2.35 ~AA0	119
2.36 ~AA1	121
2.37 ~AA2	123
2.38 ~AA3EVV.....	125
3. Modbus RTU Protocol	127

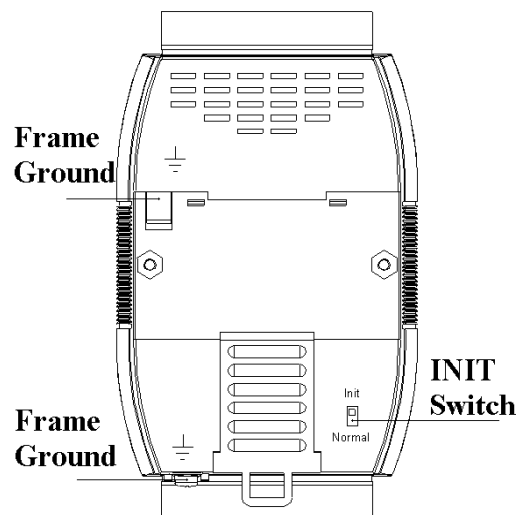
3.1	02 (0x02)	Read Input Status.....	128
3.2	04 (0x04)	Read Input Channels.....	129
3.3	70 (0x46)	Read/Write Module Settings	130
	3.3.1	Sub-function 00 (0x00) Read module name	131
	3.3.2	Sub-function 04 (0x04) Set module address	132
	3.3.3	Sub-function 05 (0x05) Read communication settings.....	133
	3.3.4	Sub-function 06 (0x06) Set communication settings.....	134
	3.3.5	Sub-function 07 (0x07) Read type code.....	135
	3.3.6	Sub-function 08 (0x08) Set type code.....	136
	3.3.7	Sub-function 32 (0x20) Read firmware version.....	137
	3.3.8	Sub-function 37 (0x25) Read channel enabled/disabled status	138
	3.3.9	Sub-function 38 (0x26) Set channel enable/disable.....	139
	3.3.10	Sub-function 41 (0x29) Read miscellaneous settings	140
	3.3.11	Sub-function 42 (0x2A) Write miscellaneous settings ...	141
	3.3.12	Sub-function 43 (0x2B) Read CJC offset	142
	3.3.13	Sub-function 44 (0x2C) Write CJC offset	143
	3.3.14	Sub-function 45 (0x2D) Read CJC enabled/disabled status	144
	3.3.15	Sub-function 46 (0x2E) Set CJC enable/disable.....	145
	3.3.16	Sub-function 47 (0x2F) Read CJC update setting.....	146
	3.3.17	Sub-function 48 (0x30) Write CJC update setting.....	147
4.		Troubleshooting	148
	4.1	Communicating with the module.....	149
	4.2	Reading Data	150
A.		Appendix	151
	A.1	INIT Mode.....	151
	A.2	Dual Watchdog Operation.....	153
	A.3	Thermocouple.....	154
	A.4	Frame Ground.....	155
	A.5	Node Information Area	157

1. Introduction

The 8000 series is a family of network data acquisition and control modules, providing analog-to-digital, digital-to-analog, digital input/output, timer/counter and other functions. The modules can be remotely controlled using a set of commands, which we call the DCON protocol. Communication between the module and the host is in ASCII format via an RS-485 bi-directional serial bus standard. Baud Rates are software programmable and transmission speeds of up to 115.2K baud can be selected.

The functionality of the 8000 series is the same as the 8000 series, with the exception that the 8000 series offers extended support for the Modbus RTU protocol.

Some 8000 and all 8000 modules feature a new design for the frame ground and INIT switch as shown in the figure (rear view). The frame ground provides enhanced static protection (ESD) abilities and ensures the module is more reliable. The INIT switch allows INIT mode to be accessed more easily. Refer to Sections A.1 and A.4 for more details.



The common features of the 8017, 8018 and 8019 are as follows:

1. 3000V DC isolated analog input

2. 24-bit sigma-delta ADC to provide excellent accuracy
3. Software calibration

The 8017 is an 8-channel voltage and current input module. The 8018 is an 8-channel voltage, current and thermocouple input module. The 8019 is an 8-channel voltage, current, and thermocouple input module, with the ability to connect various types of inputs to a single module.

The 8017 has the following variations:

- 8017F: added support for fast mode, 75 samples/second
- 8017C: used for current input only, with no external resistor required
- 8017R: added high voltage overload protection, 240Vrms.
- 8017RC: used for current input only, with no external resistor required. Added high voltage overload protection, 240Vrms

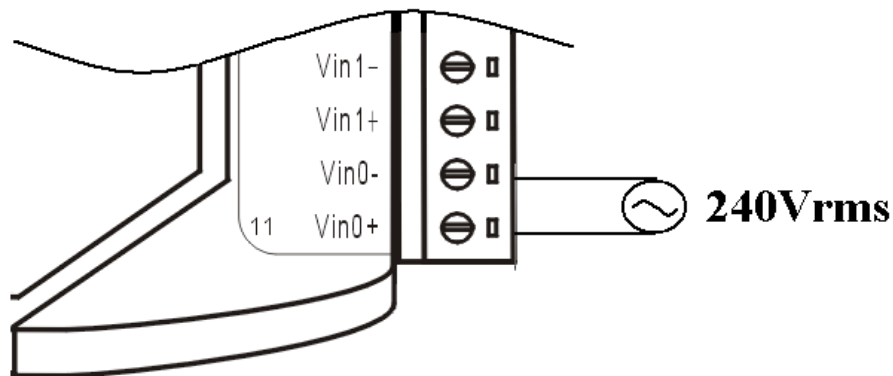
The 8018 has the following variations:

- 8018P: added support for two additional thermocouple types, L and M
- 8018BL: added thermocouple wire opening detection
- 8018R: added thermocouple wire opening detection and high voltage overload protection, 240Vrms

The 8019 has the following variation:

- 8019R: added high voltage overload protection, 240Vrms

The 8017R, 8017RC, 8018R and 8019R modules are designed for industrial plant environments and have special input circuits to provide 240Vrms continuous overload protection as shown in the figure.

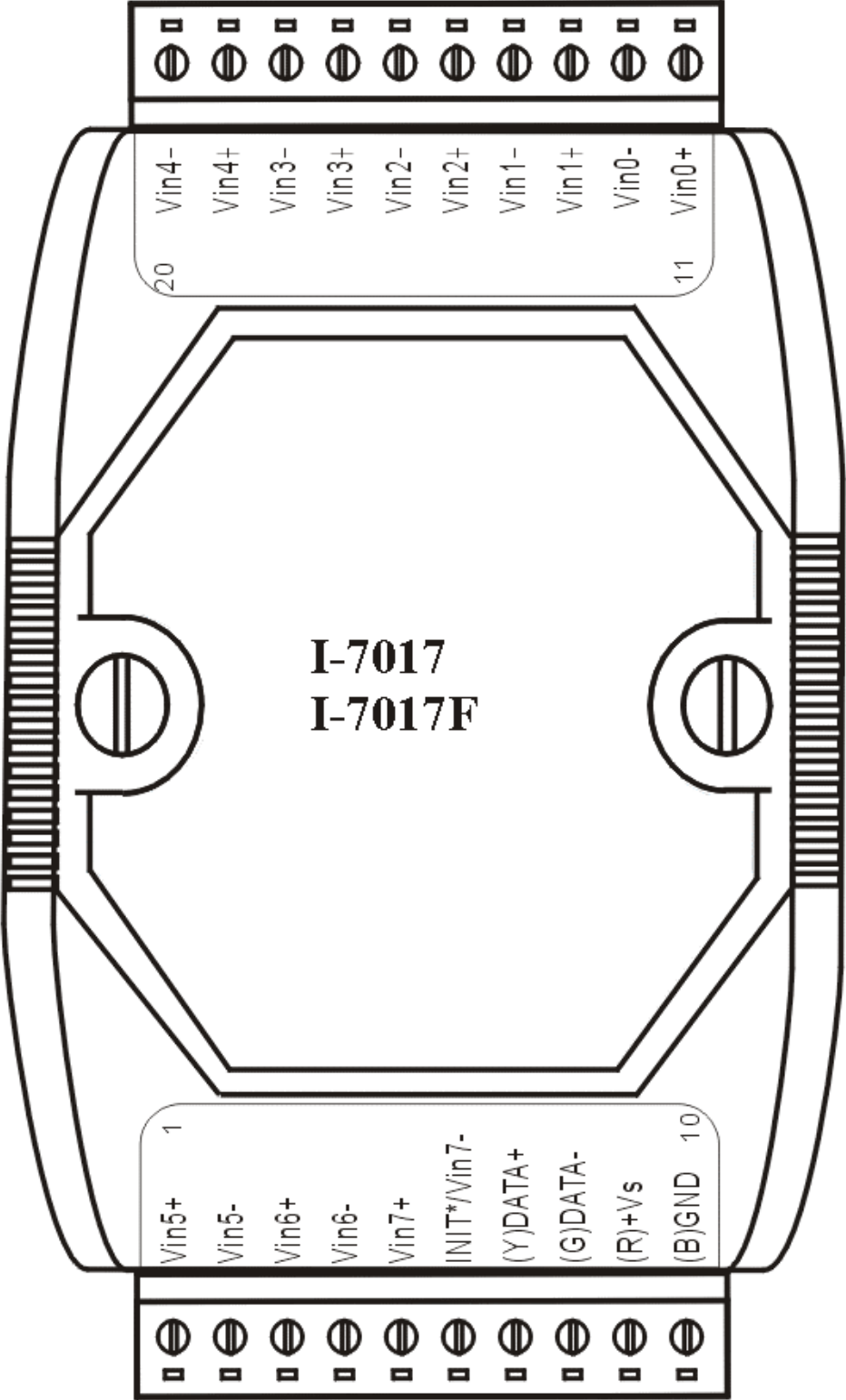


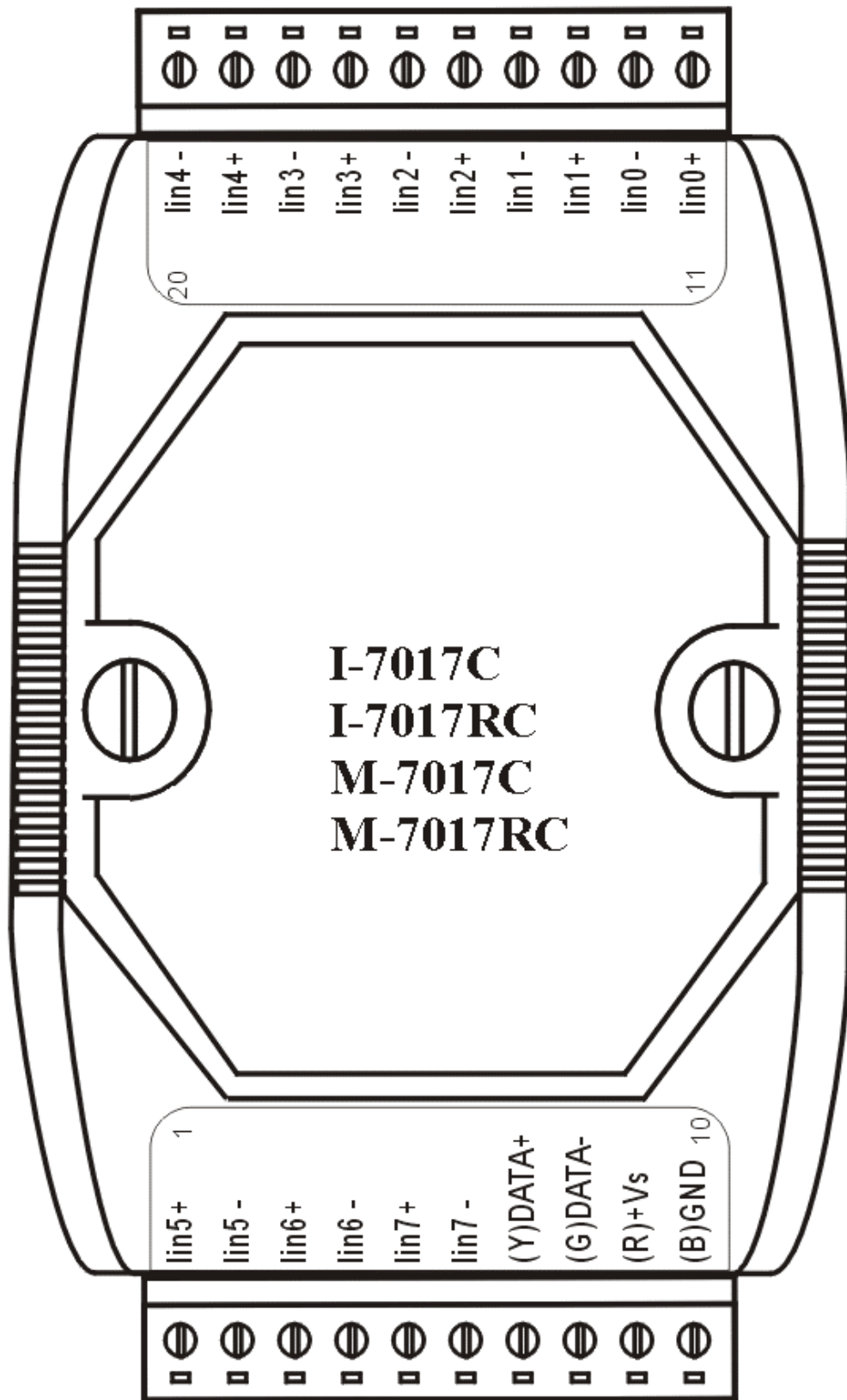
1.1 More Information

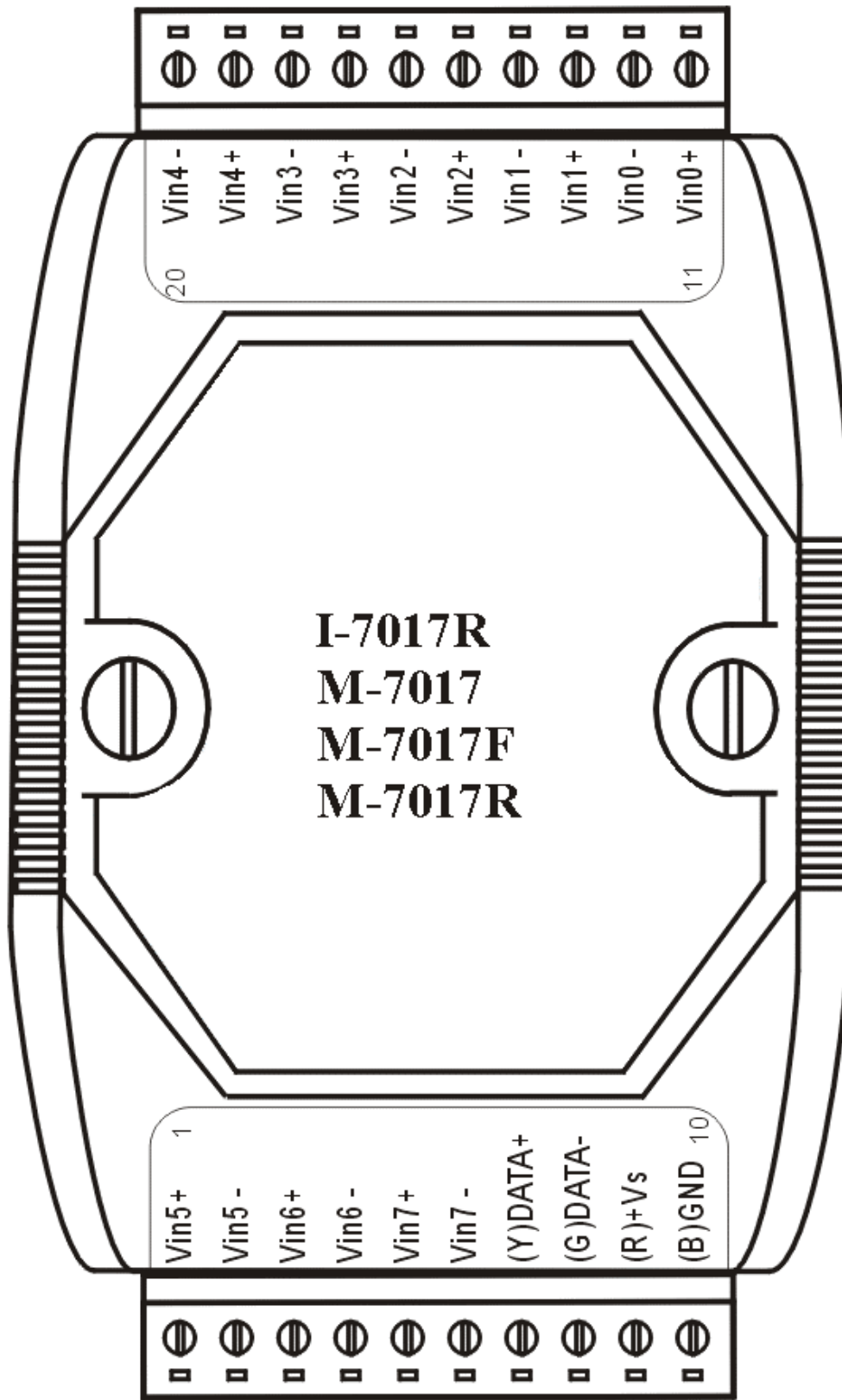
Refer to chapter 1 of the “8000 Bus Converter User’s Manual” as shown below or visit the SuperLogics' website www.superlogics.com for more information regarding the 8000 series.

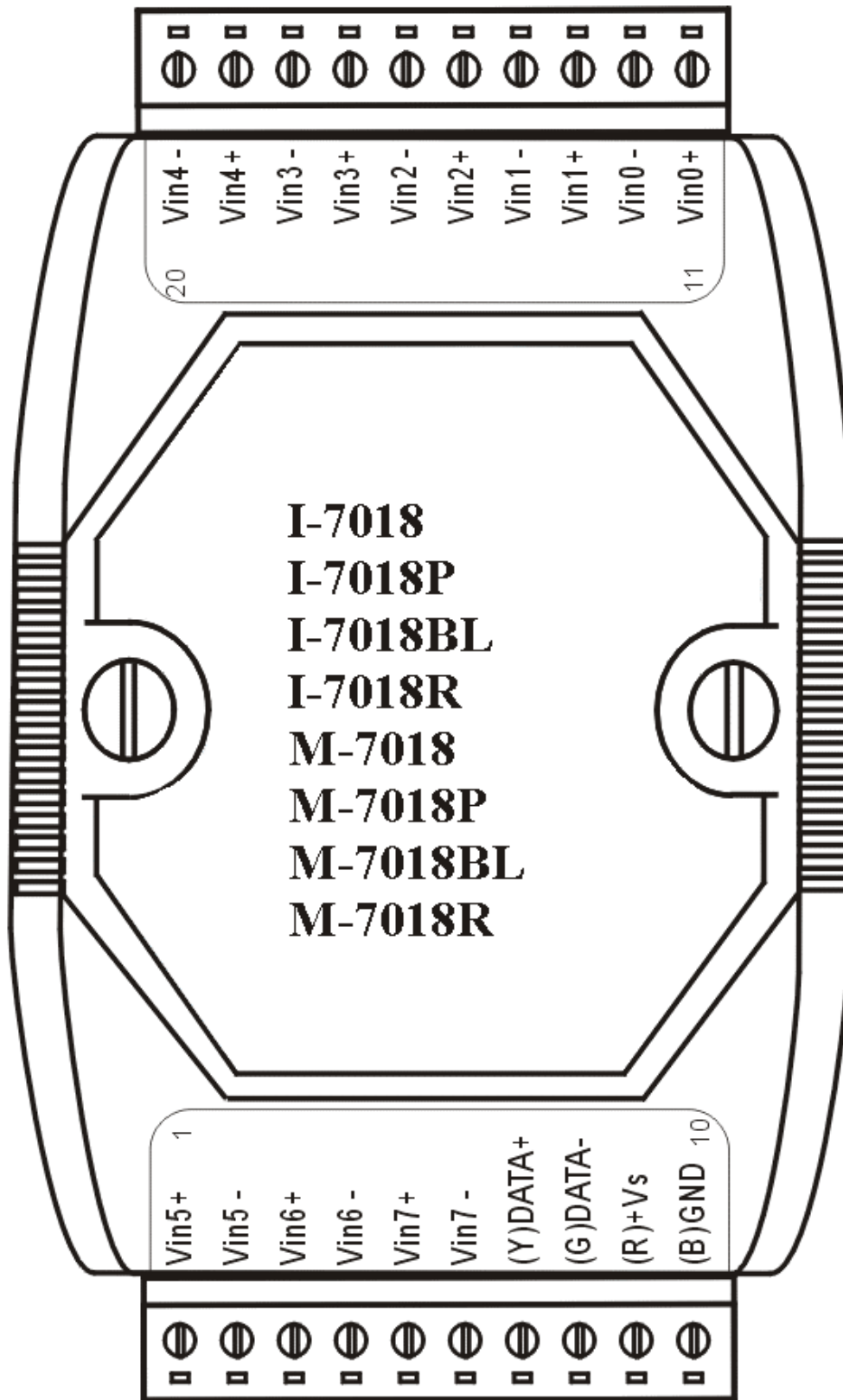
- 1.1 The 8000 Series Overview
- 1.2 Related Documentation for the 8000 Series
- 1.3 Common Features of the 8000 Series
- 1.4 The 8000 Series System Network Configuration
- 1.5 8000 Dimensions

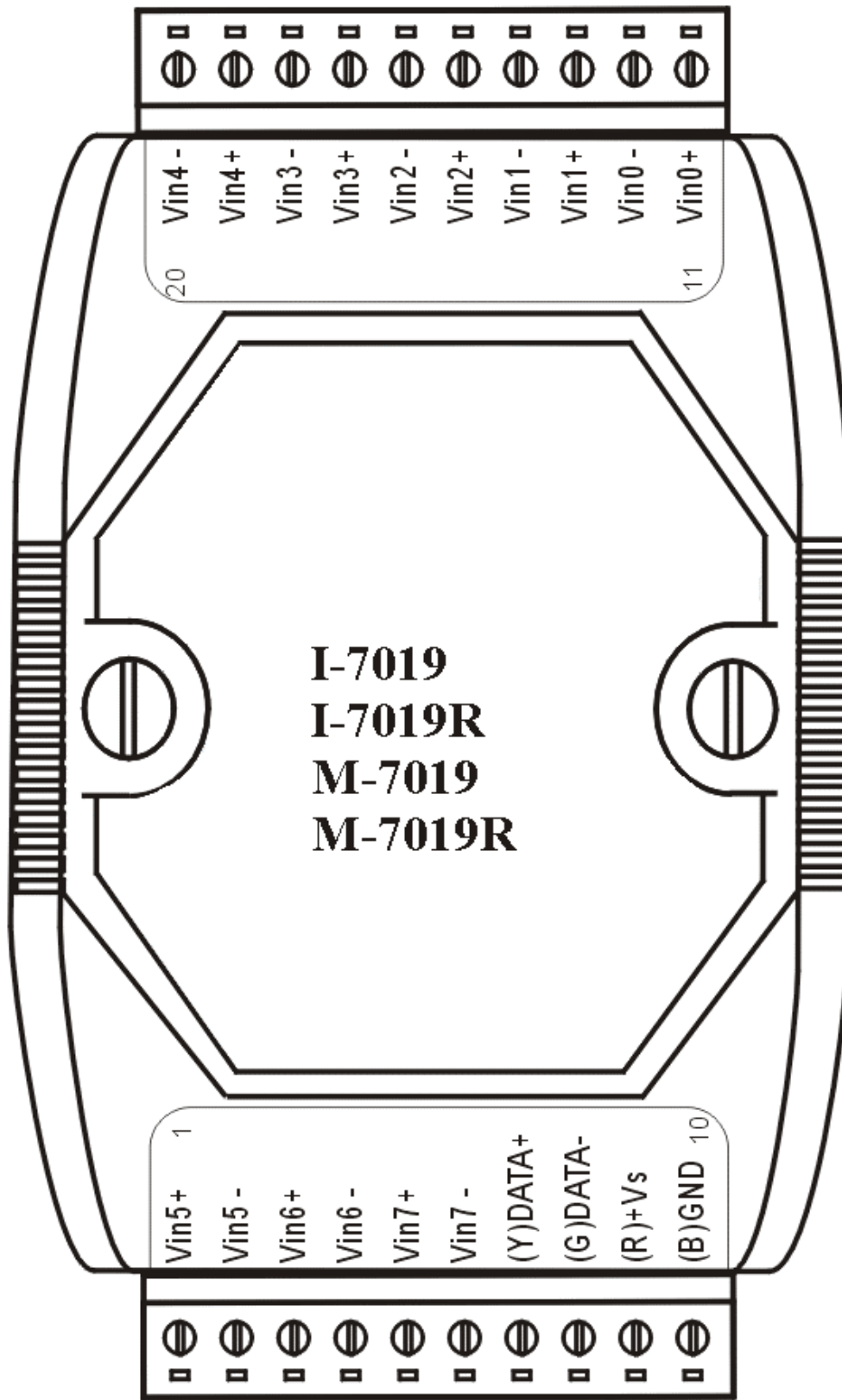
1.2 Terminal Assignment











1.3 Specifications

	8017/8017R 8017/8017R	8017F 8017F	8017C/8017RC 8017C/8017RC
Analog Input			
Input Channels	8017R/8017/ 8017R 8 differential 8017 8 differential or 6 differential and 2 single-ended	8017F 8 differential 8017F 8 differential or 6 differential and 2 single-ended	8 differential
Input Type	mV, V, mA (requires optional external 125 ohm resistor)	mV, V, mA (requires optional external 125 ohm resistor)	mA
Sampling Rate	10 samples/sec	10 samples/sec (normal) 60 samples/sec (fast)	10 samples/sec
Bandwidth	15.7Hz	15.7Hz (normal) 78.7Hz (fast)	15.7Hz
Accuracy	±0.1%	±0.1% (normal) ±0.5% (fast)	±0.1%
Zero Drift	20 μ V/°C	20 μ V/°C	20 μ V/°C
Span Drift	25ppm/°C	25ppm/°C	25ppm/°C
CMR@50/60Hz	86dB min	86dB min	86dB min
NMR@50/60Hz	100dB min	100dB min	100dB min
Input Impedance	8017R/8017R 1M Ω 8017/8017 20M Ω	20M Ω	8017RC/8017RC 1M Ω 8017C/8017C 20M Ω
Voltage overload Protection	8017R/8017R ±240V 8017/8017 ±35V	±35V	8017RC/8017RC ±240V 8017C/8017CR ±35V
Isolation	3000V DC	3000V DC	3000V DC
Modbus RTU	8017/8017R	8017F	8017C/8017RC
Power			
Requirement	+10 to +30V DC	+10 to +30V DC	+10 to +30V DC
Consumption	1.3W	1.3W	1.3W

Note: A warm up period of 30 minutes is recommended in order to achieve the complete performance results described in the specifications.

	8018 8018	8018P 8018P	8018BL/8018R 8018BL/8018R
Analog Input			
Input Channels	8 differential	8 differential	8 differential
Input Type	mV, V, mA (requires optional external 125 ohm resistor)	mV, V, mA (requires optional external 125 ohm resistor)	mV, V, mA (requires optional external 125 ohm resistor)
Thermocouple Type	J, K, T, E, R, S, B, N, C	J, K, T, E, R, S, B, N, C, L, M	J, K, T, E, R, S, B, N, C
Sampling Rate	10 samples/sec	10 samples/sec	10 samples/sec
Bandwidth	15.7Hz	15.7Hz	15.7Hz
Accuracy	±0.1%	±0.1%	8018R/8018R ±0.2% 8018BL/8018BL ±0.1%
Zero Drift	0.5 μ V/°C	0.5 μ V/°C	10 μ V/°C
Span Drift	25ppm/°C	25ppm/°C	25ppm/°C
CMR@50/60Hz	150dB min	150dB min	86dB min
NMR@50/60Hz	100dB min	100dB min	100dB min
Input Impedance	20M Ω	20M Ω	8018R/8018R 1M Ω 8018BL/8018BL 20M Ω
Voltage overload Protection	±35V	±35V	8018R/8018R ±240V 8018BL/8018BL ±35V
Isolation	3000V DC	3000V DC	3000V DC
Wire Opening Detection	No	No	Yes
Modbus RTU	8018	8018P	8018R/8018BL
Power			
Requirement	+10 to +30V DC	+10 to +30V DC	+10 to +30V DC
Consumption	1.0W	1.0W	1.0W

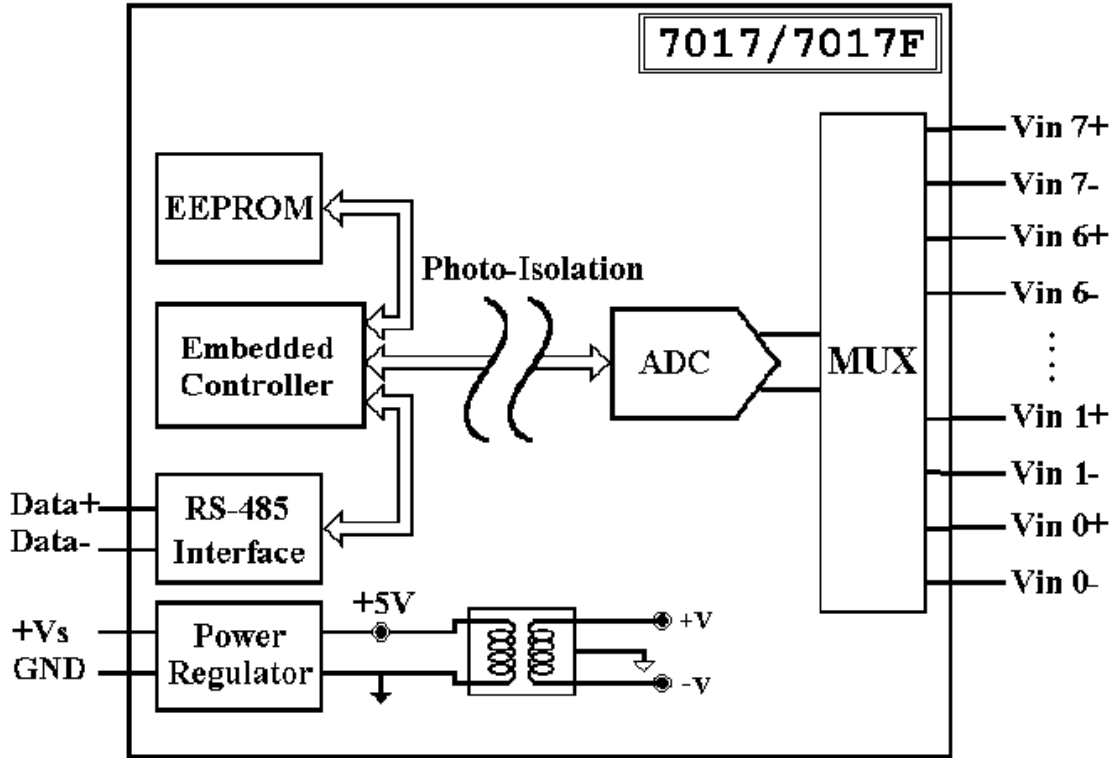
Note: A warm up period of 30 minutes is recommended in order to achieve the complete performance results described in the specifications.

	8019 8019	8019R 8019R
Analog Input		
Input Channels	8 differential	8 differential
Input Type	mV, V, mA (jumper selectable)	mV, V, mA (jumper selectable)
Thermocouple Type	J, K, T, E, R, S, B, N, C, L, M, LDIN43710	J, K, T, E, R, S, B, N, C, L, M, LDIN43710
Sampling Rate	8 samples/sec	8 samples/sec
Bandwidth	5.24Hz	5.24Hz
Accuracy	±0.1%	±0.15%
Zero Drift	0.5 μ V/°C	10 μ V/°C
Span Drift	25ppm/°C	25ppm/°C
CMR@50/60Hz	86dB min	86dB min
Input Impedance	20M Ω	2M Ω
Voltage overload Protection	±35V	±240V
Isolation	3000V DC	3000V DC
Wire Opening Detection	Yes	Yes
Modbus RTU	8019	8019R
Power		
Requirement	+10 to +30V DC	+10 to +30V DC
Consumption	1.2W	1.2W

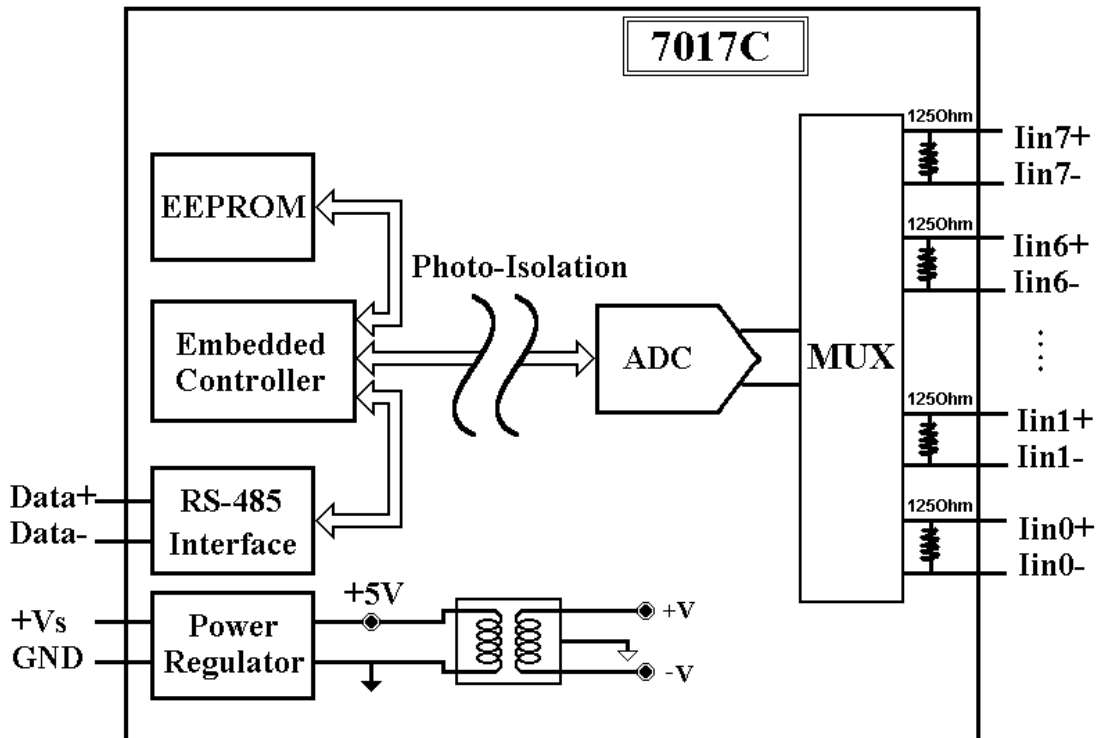
Note: A warm up period of 30 minutes is recommended in order to achieve the complete performance results described in the specifications.

1.4 Block Diagrams

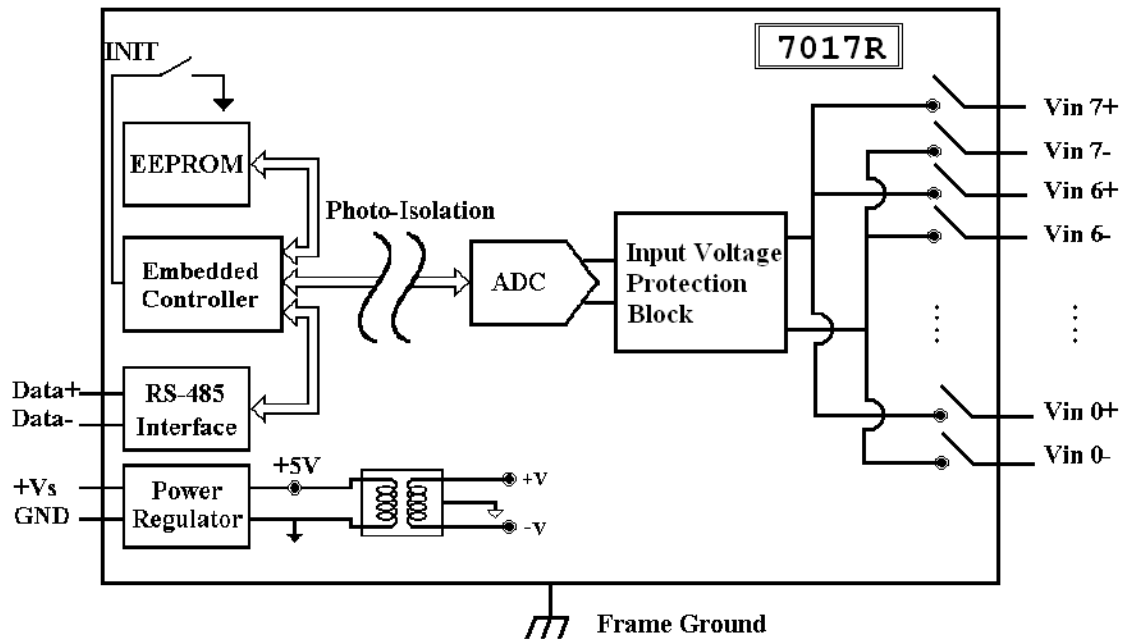
1.4.1 Block diagram for the 8017, 8017F, 8017 and 8017F



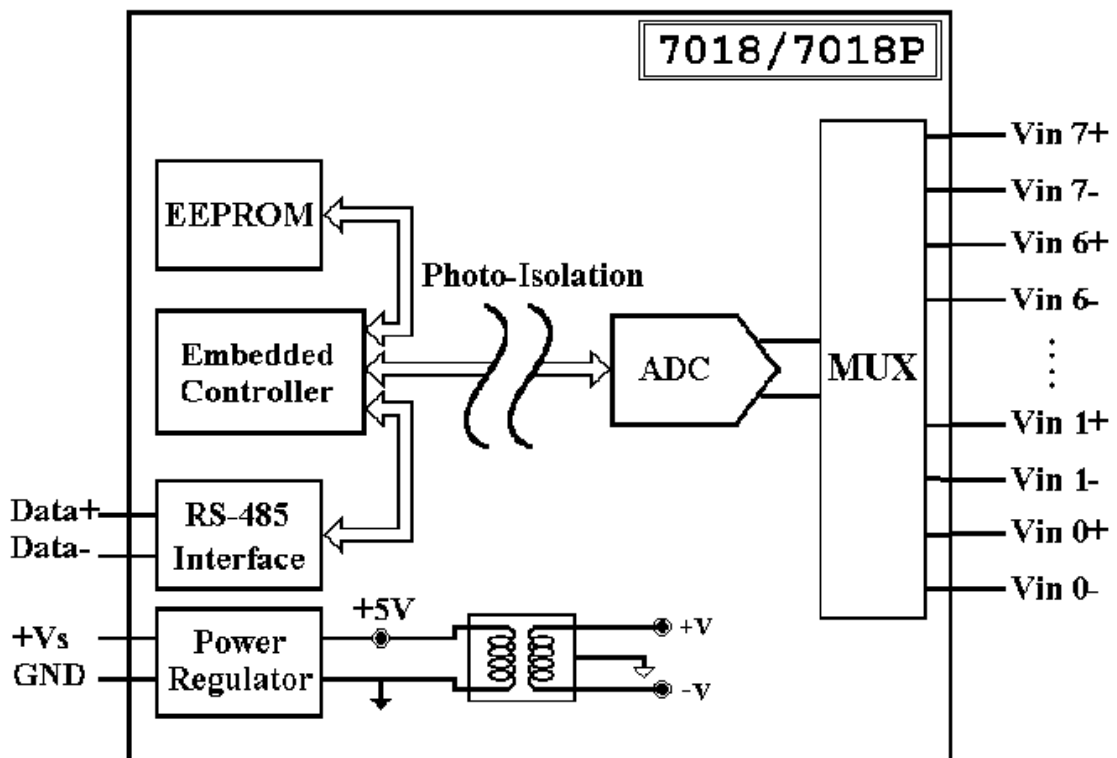
1.4.2 Block diagram for the 8017C and 8017C



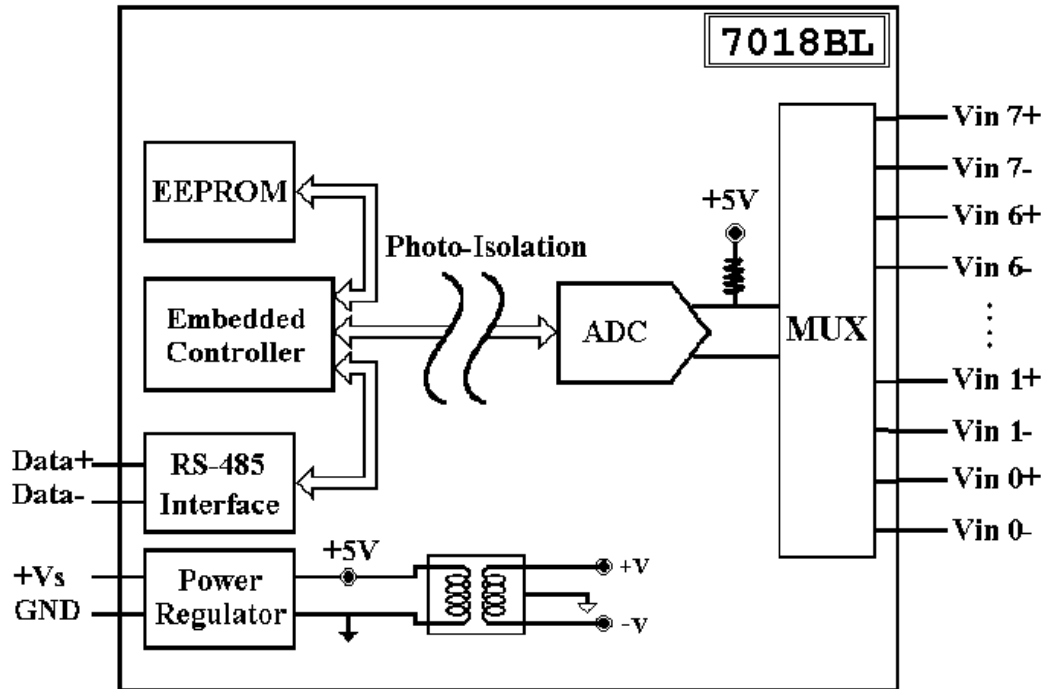
1.4.3 Block diagram for the 8017R and 8017R



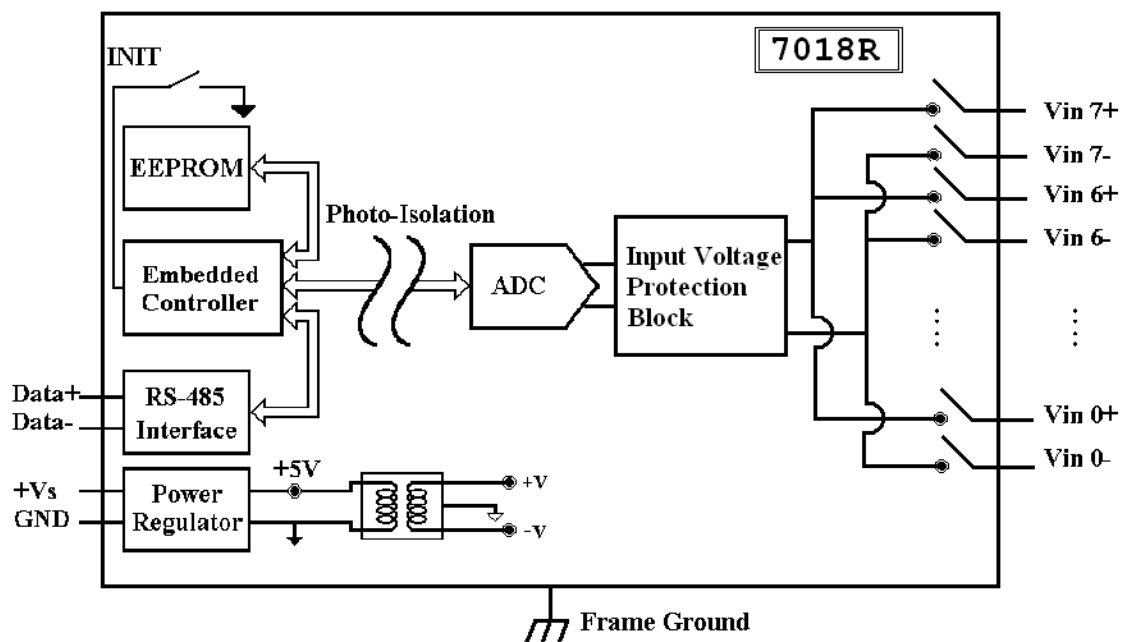
1.4.4 Block diagram for the 8018, 8018P, 8018 and 8018P



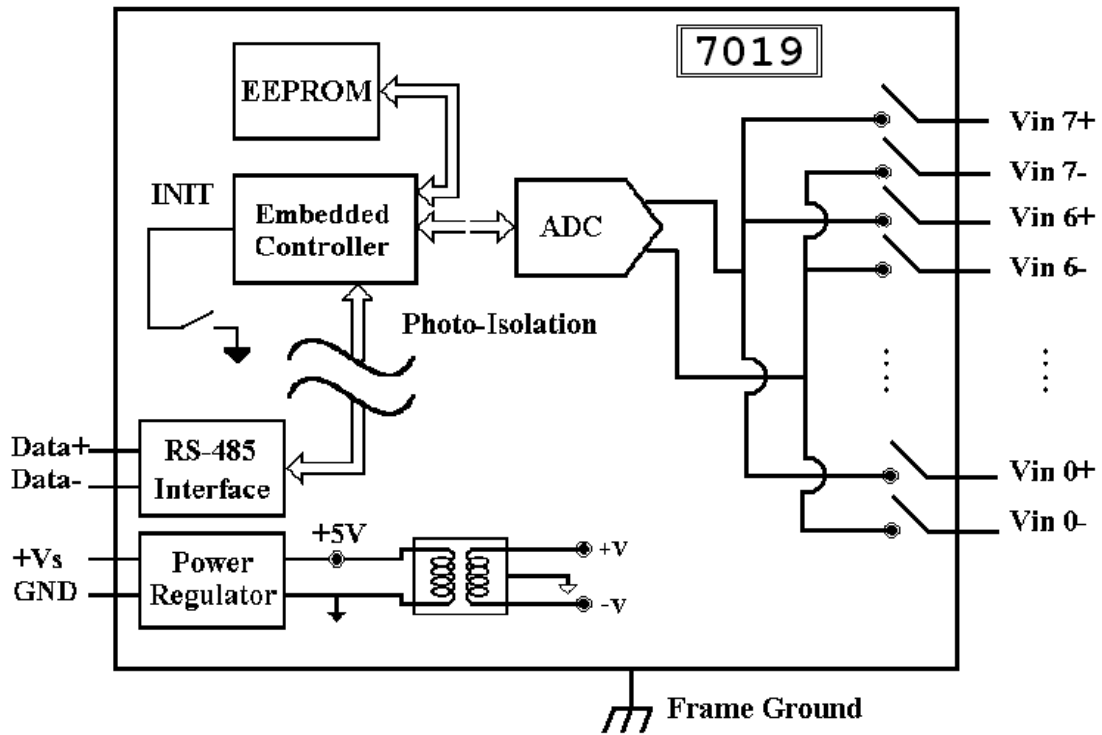
1.4.5 Block diagram for the 8018BL and 8018BL



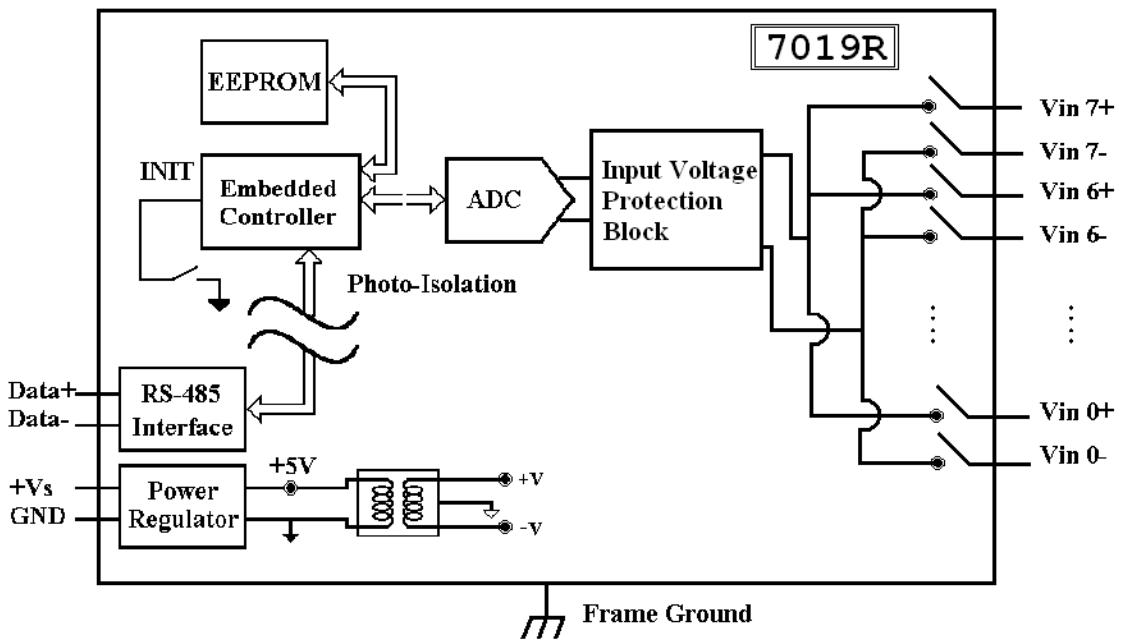
1.4.6 Block diagram for the 8018R and 8018R



1.4.7 Block diagram for the 8019 and 8019



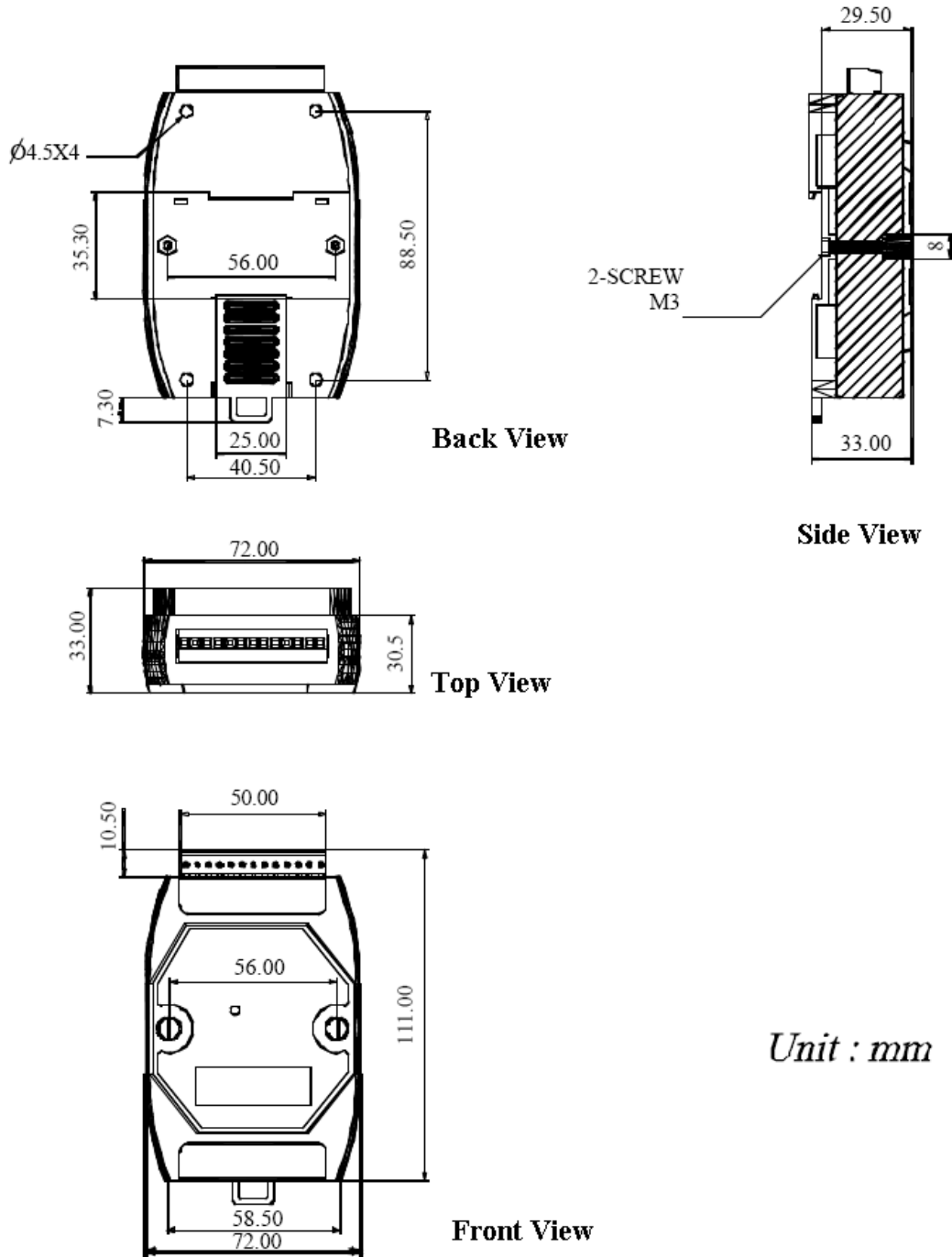
1.4.8 Block diagram for the 8019R and 8019R



1.5 Dimensions

1.5.1 Modules without Frame Ground

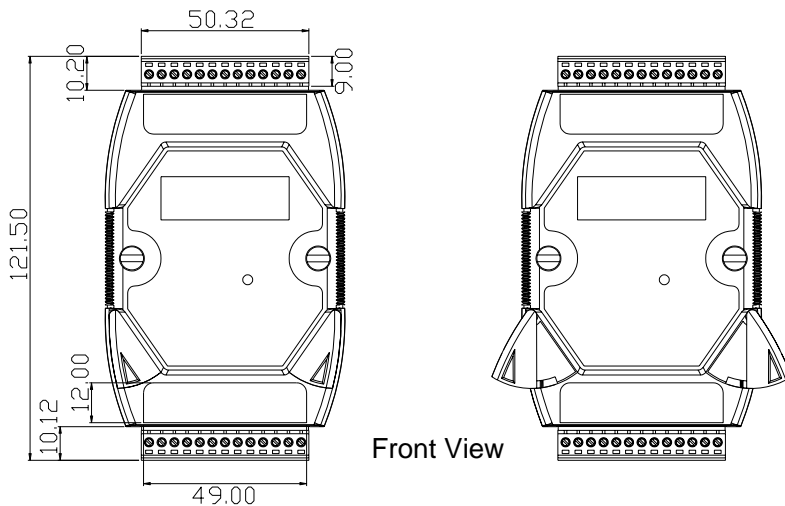
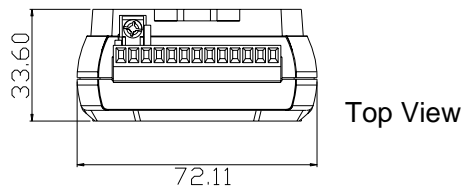
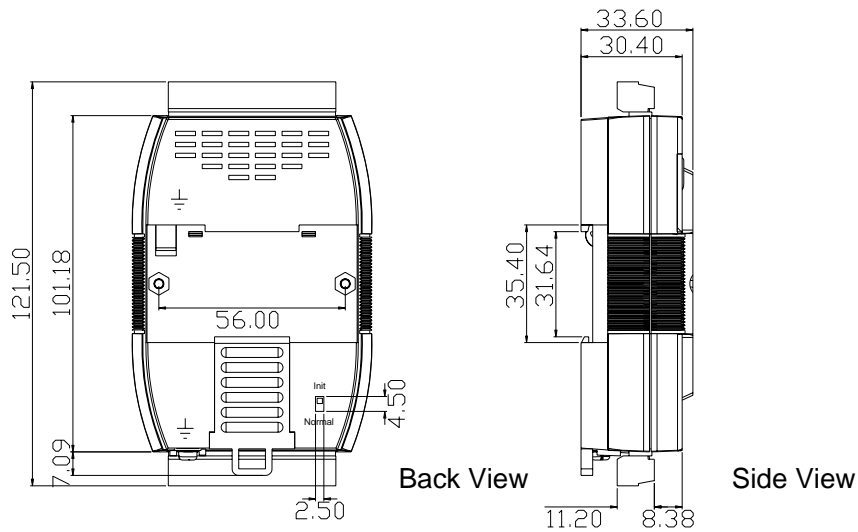
The dimensions of the 8017 and 8017F are as shown below.



Unit : mm

1.5.2 Modules with Frame Ground

The dimensions of the 8017, 8017F, 8017C, 8017C, 8017R, 8017R, 8017RC, 8017RC, 8018, 8018, 8018P, 8018P, 8018BL, 8018BL, 8018R, 8018R, 8019, 8019, 8019R and 8019R are as shown below.

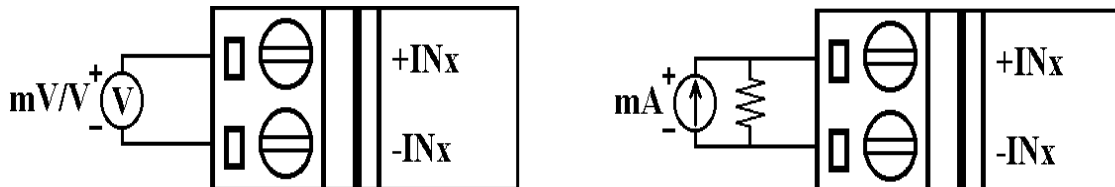


Unit : mm

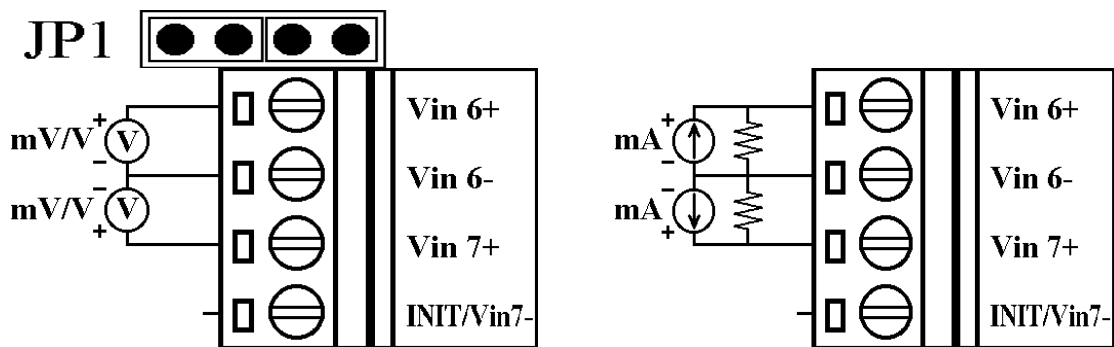
1.6 Wiring Diagrams

1.6.1 Wiring diagram for the 8017 and 8017F

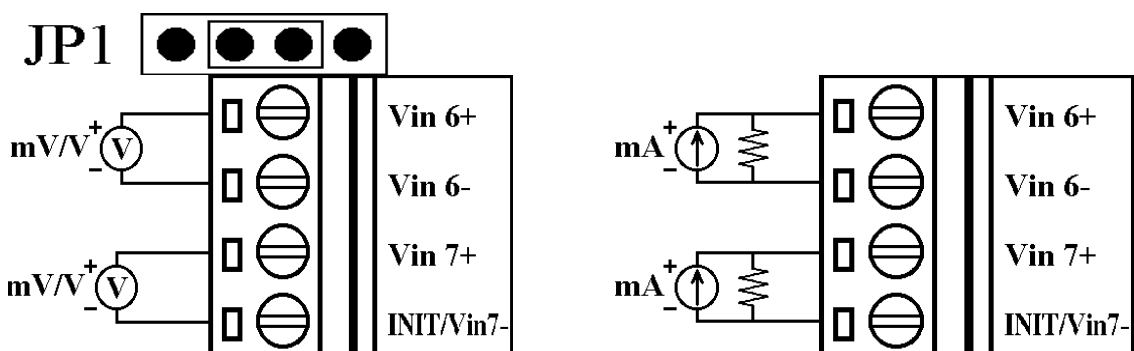
For channels 0 to 5



For channels 6 and 7, single-ended mode, INIT mode available (factory default)

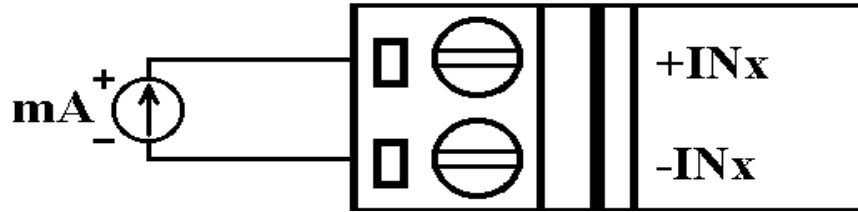


For channels 6 and 7, differential mode, INIT mode not available

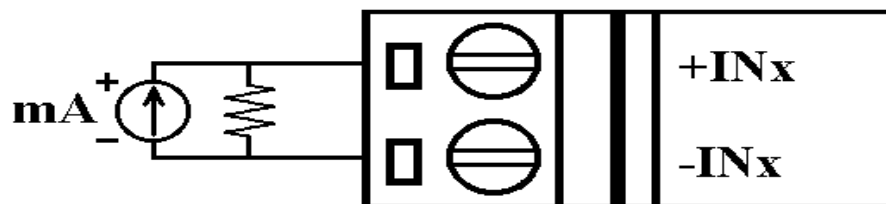
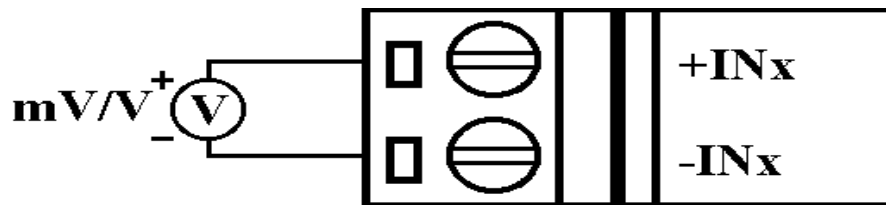


Note: When connecting to a current source, an optional external 125-ohm resistor is required.

1.6.2 Wiring diagram for the 8017C, 8017RC, 8017C and 8017RC

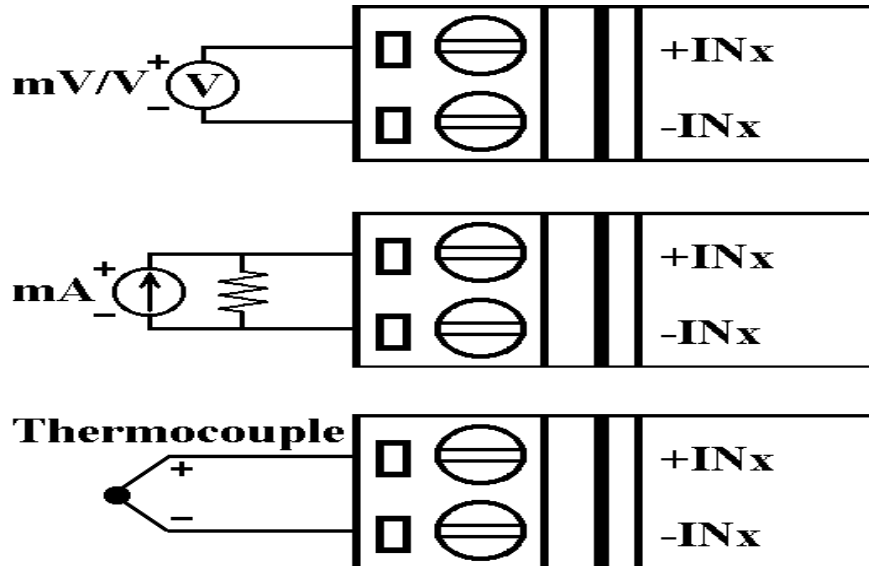


1.6.3 Wiring diagram for the 8017, 8017F, 8017R and 8017R



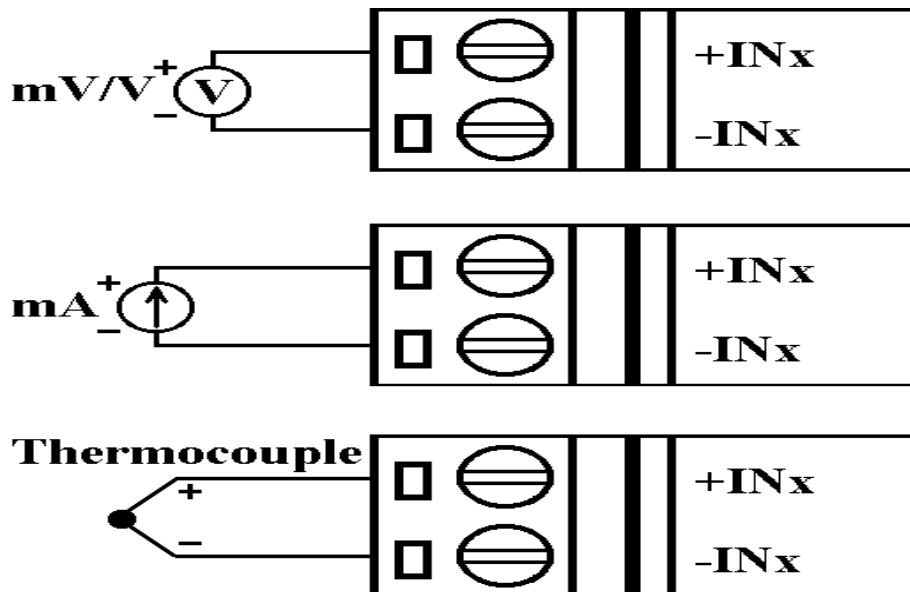
Note: When connecting to a current source, an optional external 125-ohm resistor is required.

1.6.4 Wiring diagram for the 8018, 8018P, 8018BL, 8018R, M-8018, 8018P, 8018BL and 8018R



Note: When connecting to a current source, an optional external 125-ohm resistor is required.

1.6.5 Wiring diagram for the 8019, 8019R, 8019 and 8019R



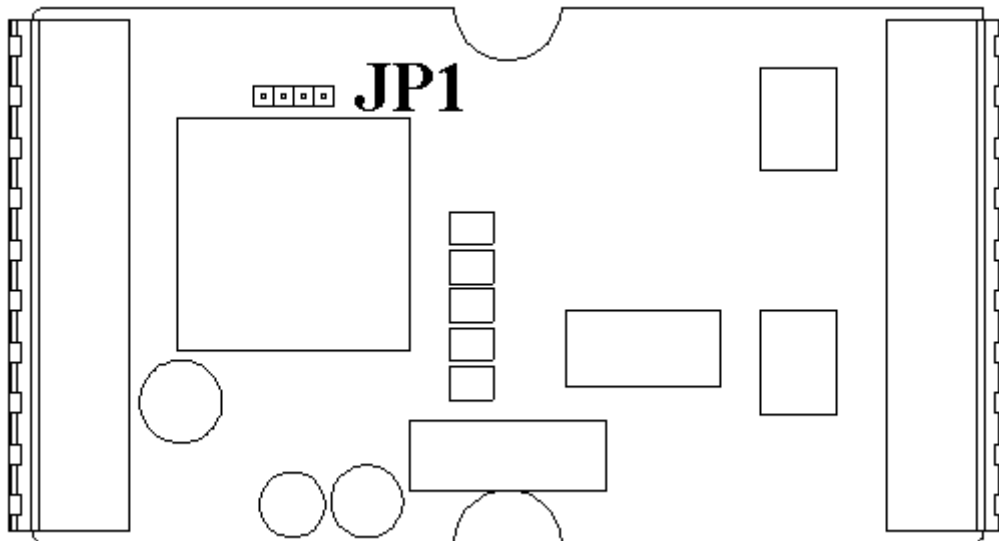
Note: When a channel is connected to a current source, the corresponding jumper should be shorted, see Section 1.7.2 for details.

1.7 Jumper Settings



Note: To access the jumpers, the cover must be opened.

1.7.1 8017 and 8017F Jumper Settings

For the 8017 and 8017F modules, the INIT* terminal is shared with the Vin7- terminal. The functionality of the terminal is set using the JP1 jumper. The position of the JP1 jumper is shown in the figure below.

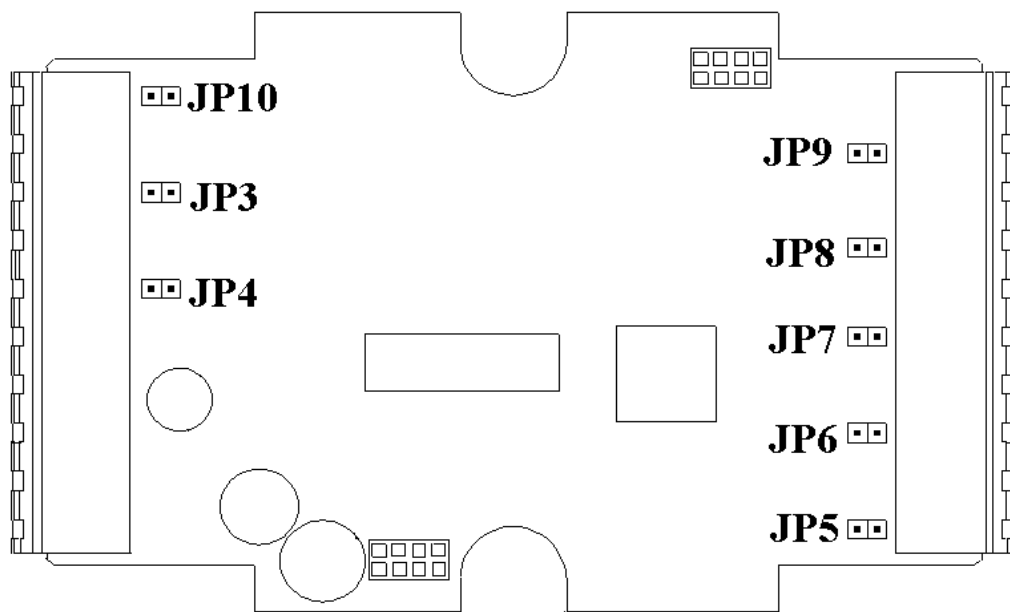


The settings for the JP1 jumper are as follows.

JP1 	Terminal INIT*/Vin7- is set to INIT*. Channels 6 and 7 are in single-ended mode and INIT mode is available. (factory default)
JP1 	Terminal INIT*/Vin7- is set to Vin7-. Channels 6 and 7 are in differential mode and INIT mode is not available.

1.7.2 8019, 8019R, 8019 and 8019R Jumper Settings

The 8019, 8019R, 8019 and 8019R modules can accept current inputs from -20mA to 20mA . No external resistor required but the corresponding jumper must be shorted. The following figure shows the jumper positions.



The corresponding jumpers and channels are as follows.

Channel	0	1	2	3	4	5	6	7
Jumper	JP5	JP6	JP7	JP8	JP9	JP10	JP3	JP4

When a channel is connected to a current input, the corresponding jumper should be shorted. For example, to connect a current input to channel 3, short the JP8 jumper.

1.8 Quick Start

To install the module, follow the steps below:

1. Connect the analog input. See Section 1.2 for the terminal assignment and Section 1.6 for the wiring diagram.
2. Connect the module to the RS-485 network using the DATA+ and DATA- terminals. See Section 1.2 for the terminal assignment. If the host is only equipped with an RS-232 interface, then an RS-232 to RS-485 converter will be required. Refer to the “**8000 Bus Converter User’s Manual**” for more information.
3. Connect the module to the power supply using the +Vs and GND terminals. See Section 1.2 for the terminal assignment. Note that the voltage supplied should be in the range of +10 to +30V DC.
4. For 8000 modules, configure the module by sending the %AANNTTCCFF command. See Section 2.1 for details. To configure the 8019, the \$AA7CiRrr command must also be sent. See Section 2.14 for details. For 8000 modules using the Modbus RTU protocol, configure the module using the following functions.
 - Sub-function 04h of Function 46h, see Section 3.3.2
 - Sub-function 06h of Function 46h, see Section 3.3.4
 - Sub-function 08h of Function 46h, see Section 3.3.6The default settings for the module can be found in Section 1.9.
5. For 8000 modules, in order to read data from the input channels, send either the #AA or #AAN command to the module. See Sections 2.3 and 2.4 for details. For 8000 modules using the Modbus RTU

protocol, use Function 04h to read the data from the input channels. See Section 3.2 for details.

6. If the host is a PC with a Windows operating system installed, the DCON Utility can be used to allow easy configuration and reading of data. The DCON Utility can be downloaded from the SuperLogics website (www.superlogics.com). The documentation for the DCON Utility can be found in the **”Getting Started For 8000 Series Modules”** manual.

Refer to the **”8000 Bus Converter User’s Manual”** and **”Getting Started For 8000 Series Modules”** manuals for more details. The **”Getting Started For 8000 Series Modules”** manual can be downloaded from the SuperLogics website (www.superlogics.com).

1.9 Default Settings

Default settings for the 8017, 8018 and 8019 series modules are:

- Module address: 01
- Analog input type:
 - Type 08, -10V to 10V, for the 8017 and 8019 series
 - Type 05, -2.5V to 2.5V, for the 8018 series
- Baud Rate: 9600 bps
- Checksum disabled
- Engineering unit format
- Filter set at 60Hz rejection (for the 8017 and 8018 series. Not used by the 8019 series)
- Fast mode for 8017F
- JP1 set to single-ended mode for the 8017 and 8017F.

Default settings for the 8017, 8018 and 8019 series modules are:

- Protocol: Modbus RTU
- Module address: 01
- Analog input type:
 - Type 08, -10V to 10V, for the 8017 and 8019 series
 - Type 05, -2.5V to 2.5V, for the 8018 series
- Baud Rate: 9600 bps
- Filter set at 60Hz rejection (for the 8017 and 8018 series. Not used by the 8019 series)
- Fast mode for 8017F

1.10 Calibration

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

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 Sections 2.1 and 2.14 for details.
3. Enable calibration. Refer to Section 2.30 for details.
4. Apply the zero calibration voltage/current.
5. Send the zero calibration command. Refer to Sections 2.6 and 2.7 for details.
6. Apply the span calibration voltage/current.
7. Send the span calibration command. Refer to Sections 2.5 and 2.8 for details.
8. Repeat steps 3 to 7 three times.

Notes:

1. For the 8017 and 8018 series, connect the calibration voltage/current to channel 0.
2. When calibrating type 06 for the 8018 series and type 0D for the 8017 series, a resistor of 125 ohms, 0.1% should be connected. Refer to Section 1.6 for details.
3. For the 8019 series, each channel should be calibrated separately. When calibrating type 06, the jumper of the corresponding channel should be short. Refer to Section 1.7.2 for details.
4. Calibration voltages and currents are shown on the next page.
5. The M-8000 series modules must be switched to the DCON protocol mode before calibrating. Refer to Section 3.3.4 for details of the switching protocol.

Calibration voltages/current used by the 8017 and M-8017 series:

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

Calibration voltages/current used by the 8018 and M-8018 series:

Type Code	00	01	02	03	04	05	06
Zero Input	0mV	0mV	0mV	0mV	0V	0V	0mA
Span Input	+15mV	+50mV	+100mV	+500mV	+1V	+2.5V	+20mA

Calibration voltages/current used by the 8019 and M-8019 series:

Type Code	00	01	02	03	04
Zero Input	0mV	0mV	0mV	0mV	0V
Span Input	+15mV	+50mV	+100mV	+500mV	+1V
Type Code	05	06	08	09	10
Zero Input	0V	0mA	0V	0V	0mV
Span Input	+2.5V	+20mA	+10V	+5V	+35mV

1.11 Configuration Tables

Baud Rate Setting (CC)

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

Note: The data bits are fixed at one start bit, eight data bits, no parity and one stop bit for both the 8000 and M-8000 series.

Analog Input Type Setting (TT)

Type Code	Analog Input Type	Range
00	+/-15mV	-15mV ~ 15mV
01	+/-50mV	-50mV ~ 50mV
02	+/-100mV	-100mV ~ 100mV
03	+/-500mV	-500mV ~ 500mV
04	+/-1V	-1V ~ 1V
05	+/-2.5V	-2.5V ~ 2.5V
06	+/-20mA	-20mA ~ 20mA
08	+/-10V	-10V ~ 10V
09	+/-5V	-5V ~ 5V
0A	+/-1V	-1V ~ 1V
0B	+/-500mV	-500mV ~ 500mV
0C	+/-150mV	-150mV ~ 150mV
0D	+/-20mA	-20mA ~ 20mA
0E	Type J Thermocouple	-210°C ~ 760°C
0F	Type K Thermocouple	-270°C ~ 1372°C
10	Type T Thermocouple	-270°C ~ 400°C
11	Type E Thermocouple	-270°C ~ 1000°C
12	Type R Thermocouple	0°C ~ 1768°C
13	Type S Thermocouple	0°C ~ 1768°C
14	Type B Thermocouple	0°C ~ 1820°C
15	Type N Thermocouple	-270°C ~ 1300°C
16	Type C Thermocouple	0°C ~ 2320°C
17	Type L Thermocouple	-200°C ~ 800°C
18	Type M Thermocouple	-200°C ~ 100°C
19	Type L DIN43710 Thermocouple	-200°C ~ 900°C

Note:

1. Types 00, 01, 02, 03, 04, 05, 06, 0E, 0F, 10, 11, 12, 13, 14, 15 and 16 are only used in 8018, M-8018, 8019 and M-8019 series modules.
2. Types 17 and 18 are only used in the 8018P, M-8018P, 8019, M-8019, 8019R and M-8019R.
3. Type 19 is only used in 8019 and M-8019 series modules.
4. Types 08, 09, 0A, 0B, 0C and 0D are only used in 8017, M-8017, 8019 and M-8019 series modules.
5. When type 0D is selected in 8017, 8017R, 8017F, M-8017, M-8017R and M-8017F modules, an optional external 125-ohm resistor is required for each channel, as shown in Sections 1.6.1 and 1.6.3.
6. When type 06 is selected in 8018 and M-8018 series modules, an optional external 125-ohm resistor is required for each channel, as shown in Section 1.6.4.
7. When type 06 or 0D is selected in 8019 and M-8019 series modules, the jumper of the corresponding channel should be shorted, as shown in Section 1.7.2.

Data Format Setting (FF)

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

Key	Description
DF	Data format 00: Engineering unit 01: % of FSR (full scale range) 10: 2's complement hexadecimal
MS	Mode settings for 8017F and M-8017F 0: Normal mode 1: Fast mode Reserved for other modules and should be zero.
CS	Checksum settings 0: Disabled 1: Enabled
FS	Filter settings for the 8017, M-8017, 8018 and M-8018 series 0: 60Hz rejection 1: 50Hz rejection Reserved for the 8019 and M-8019 series and should be zero.

Note: The reserved bits should be zero.

Analog Input Type and Data Format Table

Type code	Input Type	Data Format	+F.S	-F.S.
00* ¹	-15 to +15 mV	Engineering unit	+15.000	-15.000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
01* ¹	-50 to +50 mV	Engineering unit	+50.000	-50.000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
02* ¹	-100 to +100 mV	Engineering unit	+100.00	-100.00
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
03* ¹	-500 to +500 mV	Engineering unit	+500.00	-500.00
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
04* ¹	-1 to +1 V	Engineering unit	+1.0000	-1.0000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
05* ¹	-2.5 to +2.5 V	Engineering unit	+2.5000	-2.5000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
06* ¹	-20 to +20 mA	Engineering unit	+20.000	-20.000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
08* ²	-10 to +10 V	Engineering unit	+10.000	-10.000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
09* ²	-5 to +5 V	Engineering unit	+5.0000	-5.0000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
0A* ²	-1 to +1 V	Engineering unit	+1.0000	-1.0000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000

Type code	Input Type	Data Format	+F.S	-F.S.
0B*2	-500 to +500 mV	Engineering unit	+500.00	-500.00
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
0C*2	-150 to +150 mV	Engineering unit	+150.00	-150.00
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
0D*2	-20 to +20 mA	Engineering unit	+20.000	-20.000
		% of FSR	+100.00	-100.00
		2's comp HEX	7FFF	8000
0E*1	Type J Thermocouple -210 ~ 760°C	Engineering unit	+760.00	-210.00
		% of FSR	+100.00	-027.63
		2's comp HEX	7FFF	DCA2
0F*1	Type K Thermocouple -270 ~ 1372°C	Engineering unit	+1372.0	-0270.0
		% of FSR	+100.00	-019.68
		2's comp HEX	7FFF	E6D0
10*1	Type T Thermocouple -270 ~ 400°C	Engineering unit	+400.00	-270.00
		% of FSR	+100.00	-067.50
		2's comp HEX	7FFF	DCA2
11*1	Type E Thermocouple -270 ~ 1000°C	Engineering unit	+1000.0	-0270.0
		% of FSR	+100.00	-027.00
		2's comp HEX	7FFF	DD71
12*1	Type R Thermocouple 0 ~ 1768°C	Engineering unit	+1768.0	-0000.0
		% of FSR	+100.00	-000.00
		2's comp HEX	7FFF	0000
13*1	Type S Thermocouple 0 ~ 1768°C	Engineering unit	+1768.0	-0000.0
		% of FSR	+100.00	-000.00
		2's comp HEX	7FFF	0000
14*1	Type B Thermocouple 0 ~ 1820°C	Engineering unit	+1820.0	-0000.0
		% of FSR	+100.00	-000.00
		2's comp HEX	7FFF	0000

Type code	Input Type	Data Format	+F.S	-F.S.
15 ^{*1}	Type N Thermocouple -270 ~ 1300°C	Engineering unit	+1300.0	-0270.0
		% of FSR	+100.00	-020.77
		2's comp HEX	7FFF	E56B
16 ^{*1}	Type C Thermocouple 0 ~ 2320°C	Engineering unit	+2320.0	-0000.0
		% of FSR	+100.00	-000.00
		2's comp HEX	7FFF	0000
17 ^{*3}	Type L Thermocouple -200 ~ 800°C	Engineering unit	+800.00	-200.00
		% of FSR	+100.00	-025.00
		2's comp HEX	7FFF	E000
18 ^{*3}	Type M Thermocouple -200 ~ 100°C	Engineering unit	+100.00	-200.00
		% of FSR	+050.00	-100.00
		2's comp HEX	4000	8000
19 ^{*4}	Type L <small>DIN43710</small> Thermocouple -200 ~ 800°C	Engineering unit	+900.00	-200.00
		% of FSR	+100.00	-022.22
		2's comp HEX	7FFF	E38E
<p>*1: only available with the 8018, M-8018, 8019 and M-8019 series</p> <p>*2: only available with the 8017, M-8017, 8019 and M-8019 series</p> <p>*3: only available with the 8018P, M-8018P, 8019, M-8019, 8019R and M-8019R.</p> <p>*4: only available with the 8019 and M-8019 series.</p>				

Thermocouple Over Range/Under Range Reading for the 8018 series, version B1.4 and earlier.

	Over Range	Under Range
Engineering Unit	+9999	-0000
% of FSR	+9999	-0000
2's Complement HEX	7FFF	8000

Thermocouple Over Range/Under Range Reading for the 8018 series, version B1.5 and later, M-8018 series, 8019 series and the M-8019 series

	Over Range	Under Range
Engineering Unit	+9999.9	-9999.9
% of FSR	+999.99	-999.99
2's Complement HEX	7FFF	8000

Thermocouple Over Range/Under Range Reading for the M-8018 series and the M-8019 series with Modbus RTU protocol

Over Range	Under Range
7FFFh	8000h

1.12 M-8000 Notes

The main difference between the 8000 and M-8000 series is that the M-8000 series has additional support for the Modbus RTU communication protocol, which is the default protocol of the M-8000 series. The communication Baud Rates for the Modbus RTU protocol can be in the range of 1200 bps to 115200 bps, and the parity, data and stop bits are fixed as no parity, 8 data bits and 1 stop bit.

Modbus functions supported by the module are described in Chapter 3.

1.12.1 Protocol Switching

To switch to the DCON protocol, use sub-function 06h of the function 46h and set byte 8 to a value of 1. After a power-on reset, the communication protocol will be changed to DCON. See Section 3.3.4 for additional details.

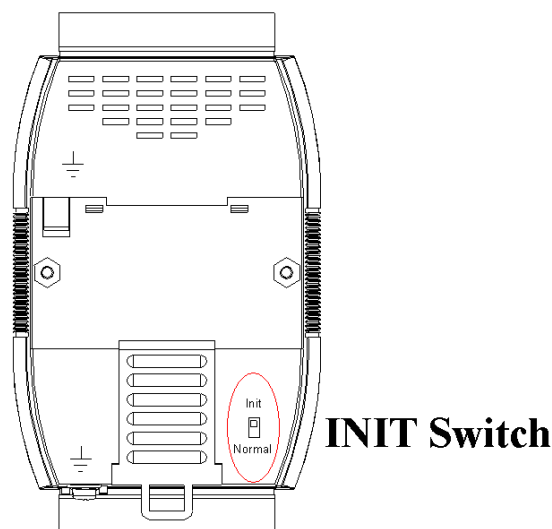
To change back to the Modbus RTU protocol, send an \$AAPN command and set N to a value of 1. Note that the slide switch on the rear side of the module should be set to the INIT position, see the figure on the next page. After a power-on reset, the communication protocol will be changed to the Modbus RTU protocol. See Section 2.25 for additional details.

1.12.2 INIT Mode

When the module is powered on, with the rear slide switch set to the INIT position as shown in the figure below, the module is in INIT mode, (see Section A.1 for details), and the communication settings are as follows:

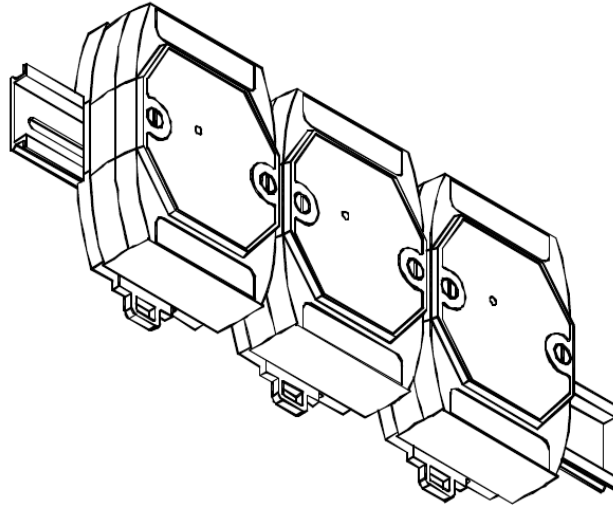
1. Address: 00
2. Baud Rate: 9600 bps
3. No checksum
4. Protocol: DCON

If communication with the module is not possible, set the module to the INIT mode and use the above settings to communicate with the module. To read the current settings, send the commands \$AA2, (see Section 2.9), and \$AAP, (see Section 2.24). To set new settings, send the commands %AANNTTCFF, (see Section 2.1) and \$AAPN, (see Section 2.25). The new communication settings will be effective after the next power-on reset.



1.13 Mounting

1.13.1 DIN Rail Mounting

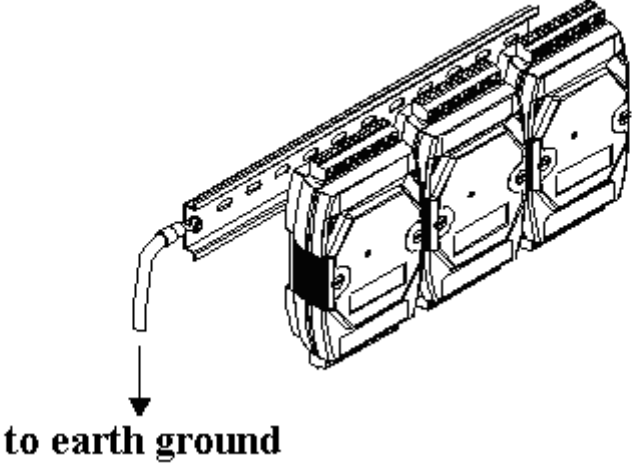


There are three new DIN rail models available. Each are made of stainless steel, which is stronger than those made of aluminum. There is a screw at one end and a ring terminal is included so that it can be easily connected to the earth ground. The three new DIN rail models are as follows.

Part number	Max number of modules	Dimensions
DRS-360	5	360mm x 35mm

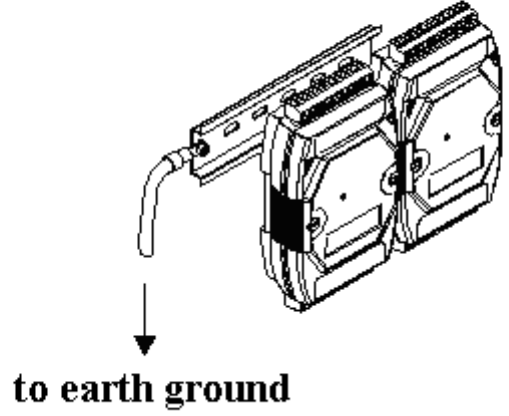
to earth ground

Part number	Max number of modules	Dimensions
DRS-240	3	240mm x 35mm



The diagram shows three DRS-240 modules mounted on a common backplane. A wire is connected to the side of the modules, and an arrow points from the wire to the text "to earth ground".

Part number	Max number of modules	Dimensions
DRS-125	2	125mm x 35mm

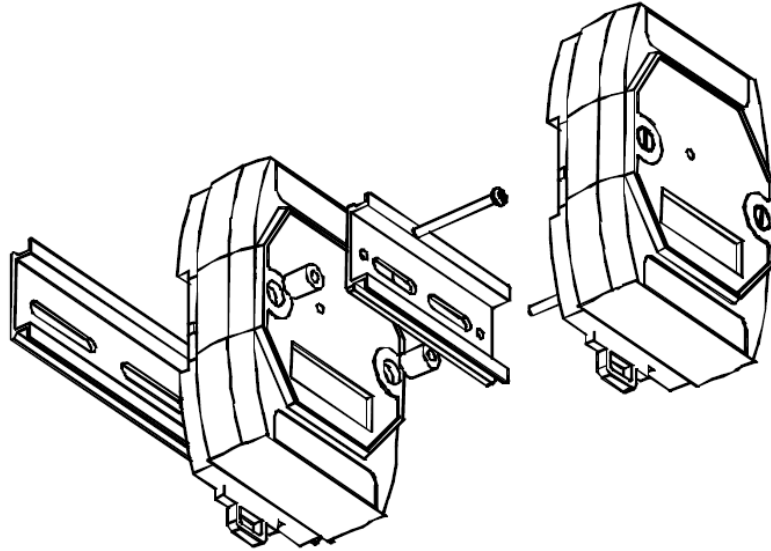


The diagram shows two DRS-125 modules mounted on a common backplane. A wire is connected to the side of the modules, and an arrow points from the wire to the text "to earth ground".

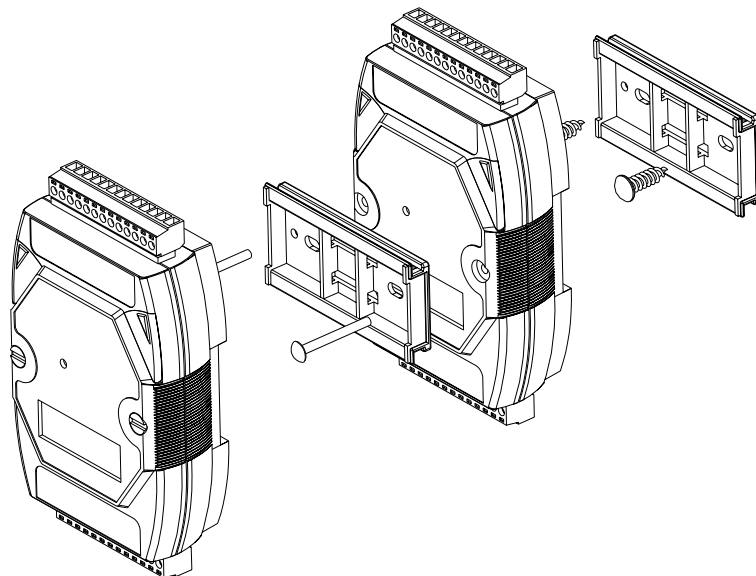
Note: The recommended wire to connect to the earth ground is 16 – 14 AWG wire.

1.13.2 Piggyback Mounting

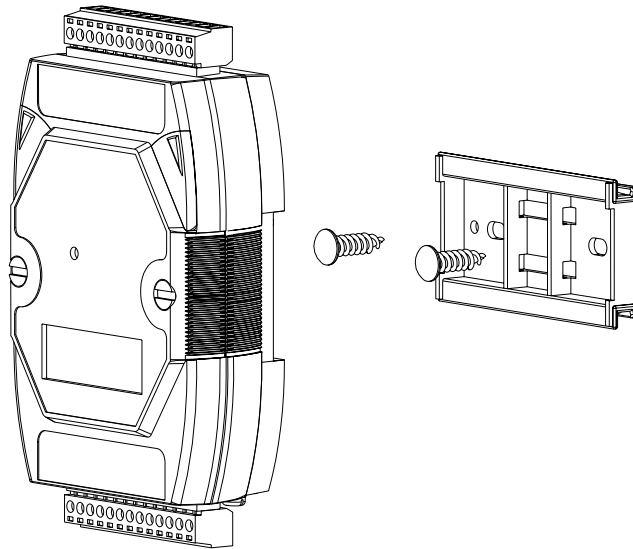
For 8017 and 8017F modules



For other modules



1.13.3 Wall Mounting



1.14 Technical Support

Should you encounter problems while using the 8000 and M-8000 module, and are unable to find the help you need in this manual or on our website, please contact SuperLogics Product Support.

Email: support@superlogics.com

Website: www.superlogics.com

When requesting technical support, be prepared to provide the following information about your system:

1. Module name and serial number: The serial number can be found printed on the barcode label attached to the cover of the module.
2. Firmware version: See Section 2.22 and 3.3.7 for information regarding the command used to identify the firmware version.
3. Host configuration (type and operating system)
4. If the problem is reproducible, please give full details describing the procedure used to reproduce the problem.
5. Specific error messages displayed. If a dialog box with an error message is displayed, please include the full text of the dialog box, including the text in the title bar.
6. If the problem involves other programs or hardware devices, please describe the details of the problem in full.
7. Any comments and suggestions related to the problem are welcome.

SuperLogics will reply to your request by email within three business days.

2. DCON Protocol

All communication with 8000 modules consists of commands generated by the host and responses transmitted by the 8000 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 modules contain the ID address, meaning that only the addressed module will respond. The only exception to this is commands #** (Section 2.2) and ~** (Section 2.34), which are sent to all modules, but in these cases, 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 Section 1.11 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.

General Command Sets			
Command	Response	Description	Section
%AANNTTCCFF	!AA	Set Module Configuration	2.1
#**	No Response	Synchronized Sampling	2.2
#AA	>(Data)	Reads the Analog Inputs of All Channels	2.3
#AAN	>(Data)	Reads the Analog Input of the Specified Channel	2.4
\$AA0	!AA	Performs a Span Calibration	2.5
\$AA1	!AA	Performs a Zero Calibration	2.6
\$AA0Ci	!AA	Performs a Single Channel Zero calibration	2.7
\$AA1Ci	!AA	Performs a Single Channel Span calibration	2.8
\$AA2	!AANNTTCCFF	Reads the Module Configuration	2.9
\$AA3	>(Data)	Reads the CJC Temperature	2.10
\$AA4	>AAS(Data)	Reads the Synchronized Data	2.11
\$AA5VV	!AA	Enables/Disables the Channel	2.12
\$AA6	!AAVV	Reads the Channel Enable/Disable Status	2.13
\$AA7CiRrr	!AA	Sets the Single Channel Range Configuration	2.14
\$AA8Ci	!AACiRrr	Reads the Single Channel Range Configuration	2.15
\$AA9	!AA(Data)	Reads the CJC Offset	2.16
\$AA9SNNNN	!AA	Sets the CJC Offset	2.17
\$AAA	!(data)	Reads the Analog Inputs of All Channels	2.18
\$AAA	!AAi	Reads the CJC Update Setting	2.19
\$AAAi	!AA	Sets the CJC Update Setting	2.20
\$AAB	!AANN	Reads the Channel Diagnostic Status	2.21
\$AAF	!AA(Data)	Reads the Firmware Version	2.22
\$AAM	!AA(Data)	Reads the Module Name	2.23
\$AAP	!AASC	Reads the Protocol	2.24
\$AAPN	!AA	Sets the Protocol	2.25

Command	Response	Description	Section
\$AAS0	!AA	Internal Calibration	2.26
\$AAS1	!AA	Reloads the Default Calibration Parameters	2.27
~AAC	!AAN	Reads the CJC Enable/Disable	2.28
~AACN	!AA	Enables/Disables the CJC	2.29
~AAEV	!AA	Enables/Disables the Calibration	2.30
~AAI	!AA	Soft INIT	2.31
~AAO(Name)	!AA	Sets the Module Name	2.32
~AATnn	!AA	Sets the Soft INIT Timeout	2.33

Host Watchdog Command Sets			
Command	Response	Description	Section
~**	No Response	Host OK	2.34
~AA0	!AASS	Reads the Host Watchdog Status	2.35
~AA1	!AA	Resets the Host Watchdog Status	2.36
~AA2	!AAETT	Reads the Host Watchdog Timeout Settings	2.37
~AA3ETT	!AA	Sets the Host Watchdog Timeout Settings	2.38

2.1 %AANNTTCCFF

Description:

Sets the configuration of an analog input module.

Syntax:

%AANNTTCCFF[CHKSUM](CR)

- %** Delimiter character
- AA** Address of the module to be configured in hexadecimal format (00 to FF)
- NN** New address of the module in hexadecimal format (00 to FF)
- TT** New type code, see Section 1.11 for details. Not used by the 8019 and M-8019 series. For the 8019 and M-8019 series, use the \$AA7CiRrr command to set the type of each channel, see Section 2.14 for details.
- CC** New Baud Rate code, see Section 1.11 for details. To change the Baud Rate, the INIT* terminal must be connected to ground or the rear slide switch must be set to the INIT position. See Section A.1 for details.
- FF** Used to set the data format, checksum, and filter settings (Section 1.11). To change the checksum setting, the INIT* terminal must be connected to ground or the rear slide switch must be set to the INIT position. See Section A.1 for details.

Note: The filter setting is not available in the 8019 and M-8019 series.

Response:

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

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

! Delimiter character for a valid response

? Delimiter character for an invalid response. If changing the **Baud Rate** or **checksum** settings without connecting the INIT* pin to ground or switching the rear slide switch to the INIT position, the module will return an invalid command.

AA Address of the 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: %0102000600 Response: !02
Change the address of module 01 to 02. The module returns a valid response.

Command: %0202000602 Response: !02
Set the data format of module 02 to be 2 (2's complement hexadecimal). The module returns a valid response.

Command: %0101000A00 Response: ?01
Change the Baud Rate of module 01 to 115200bps. The module returns an invalid command, because it is not in INIT* mode.

Command: %0101000A00 Response: !01
Change the Baud Rate of module 01 to 115200bps and the module is in INIT* mode. The module returns a valid response.

Related Commands:

Section 2.9 \$AA2, Section 2.31 ~AAI, Section 2.33 ~AATnn

Related Topics:

Section 1.11 Configuration Tables, Section A.1 INIT Mode

Notes:

1. 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 on the next power on reset.
2. For the 8019 and M-8019 series modules, changing the Baud Rate and checksum settings can be achieved using software only and is performed using the following commands:
 - I. Send a ~AATnn command. See Section 2.33 for details.
 - II. Send a ~AAI command. See Section 2.31 for details.
 - III. Send a %AANNTTCCFF command.
If the command is valid, the Baud Rate and checksum settings will be changed after the module responds with !AA.

2.2 #**

Description:

This command allows every analog input module to read data from every input channels when the command is received and will store the data for later retrieval.

Syntax:

#**[CHKSUM](CR)

Delimiter character

** Synchronized sampling command

Response:

There is no response with this command. To access the data, another command, \$AA4, must be sent, see Section 2.11 for details.

2.3 #AA

Description:

Reads the data from every analog input channel.

Syntax:

#AA[CHKSUM](CR)

Delimiter character

AA Address of the module to be read (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) Data from every analog input channels, see Section 1.11 for the details of data format. For the 8019 and M-8019 series, data from disabled channels will be filled with space characters.

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.

8017, 8018, 8019 series
M-8017, M-8018, M-8019 series

Examples:

Command: #01 Response:
>+025.12+020.45+012.78+018.97+003.24+015.35+008.0
7+014.79

 Reads module 01 and receives the data in engineering format.

Command: #02 Response:
>4C532628E2D683A20F2ADBA16284BA71

 Reads module 02 and receives the data in hexadecimal format.

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

 Reads module 03 and the data is under range.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.9 \$AA2

Related Topics:

Section 1.11 Configuration Tables

2.4 #AAN

Description:

Reads the analog input of channel N.

Syntax:

#AAN[CHKSUM](CR)

- # Delimiter character
- AA Address of the module to be read (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. An invalid command is returned if the specified channel is incorrect.
- (Data) Analog input data of the specified channel, see Section 1.11 for details of the data format. For 8019 and M-8019 series modules, if the specified channel is disabled then the data field will be filled with space characters.
- AA Address of the responding module (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.

2.5 \$AA0

Description:

Performs a span calibration.

Syntax:

\$AA0[CHKSUM](CR)

\$ Delimiter character

AA Address of the module to be calibrated (00 to FF)

0 Command for the 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 Address of the responding module (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

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

Command: \$020

Response: ?02

Performs a span calibration on module 02. An invalid command is returned because the “enable calibration” command was not sent in advance.

Related Commands:

Section 2.6 \$AA1, Section 2.30 ~AAEV

Related Topics:

Section 1.10 Calibration

Notes:

1. This command is only applicable to 8017, 8018, M-8017 and 8018 series modules.
2. The “enable calibration” command, ~AAEV, must be sent before this command is used, see Section 1.10 for details.

2.6 \$AA1

Description:

Performs a zero calibration.

Syntax:

\$AA1[CHKSUM](CR)

\$ Delimiter character

AA Address of the module to be set (00 to FF)

1 Command for the 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 Address of the responding module (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.

2.7 \$AA0Ci

Description:

Performs a zero calibration on the specified channel.

Syntax:

\$AA0Ci[CHKSUM](CR)

\$ Delimiter character

AA Address of the module to be calibrated (00 to FF)

0 Command for the zero calibration

Ci Specifies the channel to be calibrated

Response:

Valid Response: !AA[CHKSUM](CR)

Invalid Response: ?AA[CHKSUM](CR)

! Delimiter character for a valid response

? Delimiter character for an invalid response

AA Address of the responding module (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: \$010C0 Response: !01
Performs a zero calibration on channel 0 of module 01 and returns a valid response.
- Command: \$020C5 Response: !02
Performs a zero calibration on channel 5 of module 02 and returns a valid response.
- Command: \$030C1 Response: ?03
Performs a zero calibration on channel 1 of module 03. An invalid command is returned because the “enable calibration” command was not sent in advance.

Related Commands:

Section 2.8 \$AA1Ci, Section 2.30~AAEV

Related Topics:

Section 1.10 Calibration

Notes:

- 1 This command is only applicable to 8019 and M-8019 series modules.
- 2 The “enable calibration” command, ~AAEV, must be sent before this command is used, see Section 1.10 for details.

2.8 \$AA1Ci

Description:

Performs a span calibration on the specified channel.

Syntax:

\$AA1Ci[CHKSUM](CR)

\$ Delimiter character

AA Address of the module to be calibrated (00 to FF)

1 Command for the span calibration

Ci Specifies the channel to be calibrated

Response:

Valid Response: !AA[CHKSUM](CR)

Invalid Response: ?AA[CHKSUM](CR)

! Delimiter character for a valid response

? Delimiter character for an invalid response

AA Address of the responding module (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: \$011C0 Response: !01
Performs a span calibration on channel 0 of module 01 and returns a valid response.

Command: \$021C5 Response: !02
Performs a span calibration on channel 5 of module 02 and returns a valid response.

Command: \$031C1 Response: ?03
Performs a span calibration on channel 1 of module 03. An invalid command is returned because the “enable calibration” command was not sent in advance.

Related Commands:

Section 2.7 \$AA0Ci, Section 2.30 ~AAEV

Related Topics:

Section 1.10 Calibration

Notes:

1. This command is only applicable to 8019 and M-8019 series modules
2. The “enable calibration” command, ~AAEV, must be sent before this command is used, see Section 1.10 for details.

2.9 \$AA2

Description:

Reads the module configuration.

Syntax:

\$AA2[CHKSUM](CR)

\$ Delimiter character
AA Address of the module to be read (00 to FF)
2 Command to read the module configuration

Response:

Valid Response: !AATTCCFF[CHKSUM](CR)

Invalid Response: ?AA[CHKSUM](CR)

! Delimiter character for a valid response
? Delimiter character for an invalid response
AA Address of the responding module (00 to FF)
TT Type code of the module, see Section 1.11 for
 details.
CC Baud Rate code of the module, see Section 1.11
 for details.
FF Data format, checksum settings and filter
 settings of the module, see Section 1.11 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.

2.10 \$AA3

Description:

Reads the CJC (cold junction compensation) temperature.

Syntax:

\$AA3[CHKSUM](CR)

\$ Delimiter character
AA Address of the module to be read (00 to FF)
3 Command to read the CJC temperature

Response:

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

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

> Delimiter character for a valid response
? Delimiter character for an invalid response
AA Address of the responding module (00 to FF)
(Data) CJC temperature in degrees Celsius, consisting
 of a sign byte, '+' or '-', and followed by 5
 decimal digits with a fixed decimal point in
 tenths of a degree.

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.

2.11 \$AA4

Description:

Reads the synchronized data that was obtained from the last #** command.

Syntax:

\$AA4[CHKSUM](CR)

- \$ Delimiter character
- AA Address of the module to be read (00 to FF)
- 4 Command to read the synchronized 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 Address of the responding module (00 to FF)
- S Status of the synchronized data,
 - 1: first reading
 - 0: not the first reading
- (Data) Synchronized data. See Section 1.11 for details of the data format. For 8019 and M-8019 series modules, data from disabled channels will be filled with space characters.

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: \$014 Response: ?01
Reads the synchronized data for module 01. An invalid command is returned because the #** command has not been sent in advance.
- Command: #** No response
Sends the synchronized sampling command.
- Command: \$014 Response:
>011+025.56+002.34+015.79+022.66+008.53+021.43+007.06+022.88
Reads the synchronized data for module 01. The module returns the synchronized data and sets the status byte to 1 to indicate that it is the first time the synchronized data has been read.
- Command: \$014 Response:
>010+25.56+002.34+015.79+022.66+008.53+021.43+007.06+022.88
Reads the synchronized data for module 01. The module returns the synchronized data and sets the status byte to 0 to indicate that the synchronized data has been read.

Related Commands:

Section 2.2 #**

Note:

This command is only applicable to 8019 and M-8019 series modules.

2.12 \$AA5VV

Description:

Specifies the channel(s) to be enabled.

Syntax:

\$AA5VV[CHKSUM](CR)

\$ Delimiter character
AA Address of the module to be set (00 to FF)
5 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 1 it means that the channel is enabled and 0 means that the channel is disabled.

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 command is returned if an attempt is made to enable a channel that is not present.
AA Address of the responding module (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 and disables all other channels of module 01. The module returns a valid response.

Command: \$016

Response: !013A

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

Related Commands:

Section 2.13 \$AA6

Note:

It is recommended that only the channels that will be used are enabled.

2.13 \$AA6

Description:

Reads the enabled/disabled status of each channel.

Syntax:

\$AA6[CHKSUM](CR)

\$ Delimiter character
AA Address of the module to be read (00 to FF)
6 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 Address of the responding module (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 1 it means that
 the channel is enabled and 0 means that the
 channel is disabled.

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 and disables all other channels of module 01. The module returns a valid response.

Command: \$016 Response: !013A

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

Related Commands:

Section 2.12 \$AA5VV

2.14 \$AA7CiRrr

Description:

Sets the type code of a channel.

Syntax:

\$AA7CiRrr[CHKSUM](CR)

\$ Delimiter character
AA Address of the module to be set (00 to FF)
7 Command to set the channel range code
Ci i specifies the input channel to be set (0-7)
Rrr rr represents the type code of the channel to be set. Refer to the Analog Input Type Setting table in Section 1.11 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 invalid type code

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: \$017C0R03 Response: !01
Sets the type code for channel 0 of module 01 to be 03 (-500~+500mV) and the module returns a valid response.

Command: \$027C5R16 Response: !02
Sets the type code for channel 5 of module 02 to be 16 (Type C thermocouple, +0~+2320°C) and the module returns a valid response.

Command: \$037C1R30 Response: ?03
Sets the type code for channel 1 of module 03 to be 30. The module returns an invalid response because the type code is invalid.

Related Commands:

Section 2.15 \$AA8Ci

Related Topics:

Section 1.11 Configuration Tables

Note:

This command is only applicable to 8019 and M-8019 series modules.

2.15 \$AA8Ci

Description:

Reads the type code information of a channel.

Syntax:

\$AA8Ci[CHKSUM](CR)

\$	Delimiter character
AA	Address of the module to be read (00 to FF)
8	Command to read the type code of a channel
Ci	Specifies which channel to access for the type code information (i=0-7)

Response:

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

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

!	Delimiter character for a valid response
?	Delimiter character for an invalid response or invalid channel
AA	Address of the responding module (00 to FF)
Ci	Specifies which input channel to access to retrieve the type code information.
Rrr	Represents the type code of the specified input 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.

Examples:

Command: \$018C0

Response: !01C0R02

Reads the channel 0 input range of module 01 and returns 02 (-100~+100mV).

Related Commands:

Section 2.14 \$AA7CiRrr

Related Topics:

Section 1.11 Configuration Tables

Note:

This command is only applicable to 8019 and M-8019 series modules.

2.16 \$AA9

Description:

Reads the CJC (cold junction compensation) offset value that is set by the \$AA9SNNNN command (Section 2.17).

Syntax:

\$AA9[CHKSUM](CR)

\$ Delimiter character
AA Address of the module to be read (00 to FF)
9 Command to read the CJC offset value

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 Address of the responding module (00 to FF)
(Data) CJC offset value consisting of a sign byte, '+' or
 '-', followed by 4 hexadecimal digits. Each
 count is equal to 0.01°C.

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.

2.17 \$AA9SNNNN

Description:

Sets the CJC (cold junction compensation) offset value to adjust the error produced by the CJC sensor.

Syntax:

\$AA9SNNNN[CHKSUM](CR)

\$	Delimiter character
AA	Address of the module to be read (00 to FF)
9	Command to set the CJC offset value
S	Sign byte, '+' or '-', of the offset value
NNNN	The absolute value of the offset in four hexadecimal digits, which must be less than or equal to 1000h. Each count is equal to 0.01°C.

Response:

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

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

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	Address of the responding module (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: \$019+0010

Response: !01

Sets the CJC offset of module 01 to +0.16°C and returns a valid response.

Related Commands:

Section 2.10 \$AA3, Section 2.16 \$AA9, Section 2.19 \$AAA, Section 2.20 \$AAAi, Section 2.29 ~AACN

Note:

This command is only applicable to 8018, M-8018, 8019 and M-8019 series modules.

2.18 \$AAA

Description:

Reads the data from every analog input channel.

Syntax:

\$AAA[CHKSUM](CR)

\$ Delimiter character

AA Address of the module to be read (00 to FF)

A Command to read every analog input

Response:

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

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

> Delimiter character for a valid response

? Delimiter character for an invalid response

(Data) Data from every analog input channels in hex 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.

8017, M-8017 series

Examples:

Command: \$01A

Response: >0000012301257FFF1802744F98238124

Reads module 01 and receives the data in hex format.

Related Commands:

Section 2.3 #AA

Note:

This command is only applicable to 8017 and M-8017 series modules.

2.19 \$AAA

Description:

Reads the CJC temperature update setting.

Syntax:

\$AAA[CHKSUM](CR)

\$ Delimiter character
AA Address of the module to be read (00 to FF)
A Command to read the CJC temperature update
 setting

Response:

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

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

! Delimiter character for a valid response
? Delimiter character for an invalid response
AA Address of the responding module (00 to FF)
i CJC temperature update setting:
 0= CJC temperature update has been stopped.
 1= CJC temperature update has been started.
 2= Update CJC temperature once only.

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: \$01A

Response: !011

Reads the CJC temperature update setting of module 01 and the module responds with 1, meaning that the CJC temperature update has been started.

Related Commands:

Section 2.10 \$AA3, Section 2.16 \$AA9, Section 2.17 \$AA9SNNNN, Section 2.20 \$AAAi, Section 2.29 ~AACN

Note:

This command is only applicable to 8019 and M-8019 series modules.

2.20 \$AAAi

Description:

Sets the CJC temperature update setting.

Syntax:

\$AAAi[CHKSUM](CR)

\$	Delimiter character
AA	Address of the module to be read (00 to FF)
A	Command to set the CJC temperature update setting
i	CJC temperature update setting: 0= stop the CJC temperature update 1= start the CJC temperature update (default value) 2= update the CJC temperature only once, after the command is received.

Response:

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

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

!	Delimiter character for a valid response
?	Delimiter character for an invalid response
AA	Address of the responding module (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: \$01A1

Response: !01

Sends a command to start the CJC temperature update of module 01 and returns a valid response.

Related Commands:

Section 2.10 \$AA3, Section 2.16 \$AA9, Section 2.17 \$AA9SNNNN, Section 2.19 \$AAA, Section 2.29 ~AACN

Note:

This command is only applicable to 8019 and M-8019 series modules.

2.21 \$AAB

Description:

Diagnoses the over-range, under-range, and wire opening status of the analog inputs.

Syntax:

\$AAB[CHKSUM](CR)

- \$ Delimiter character
- AA Address of the module to be diagnosed (00 to FF)
- B Command to diagnose the analog inputs

Response:

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

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

- ! Delimiter character for a valid response
- ? Delimiter character for an invalid response
- AA Address of the responding module (00 to FF)
- NN Represents the diagnostic results of every analog input channel (00 to FF), where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 1, it means that the channel is enabled and it is either over-range, under-range or wire opening. If the bit is 0 it means that the channel is disabled or normal.

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: \$01B

Response: !0101

Diagnoses the analog inputs of module 01. The module returns a valid response and channel 1 is either over-range, under-range or wire opening.

Note:

This command is only applicable to 8019 and M-8019 series modules.

2.22 \$AAF

Description:

Reads the firmware version of a module.

Syntax:

\$AAF[CHKSUM](CR)

- \$ Delimiter character
- AA Address of the module to be read (00 to FF)
- F Command to read the firmware version

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 Address of the responding module (00 to FF)
- (Data) A string indicating the firmware version 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.

Examples:

Command: \$01F

Response: !01A2.0

Reads the firmware version of module 01, and shows that it is version A2.0.

2.23 \$AAM

Description:

Reads the name of a module.

Syntax:

\$AAM[CHKSUM](CR)

\$ Delimiter character

AA Address of the module to be read (00 to FF)

M Command to read the module name

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 Address of the responding module (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.

2.24 \$AAP

Description:

Reads the communication protocol information.

Syntax:

\$AAP[CHKSUM](CR)

\$ Delimiter character

AA Address of the module to be read (00 to FF)

P Command to read the communication protocol

Response:

Valid Response: !AASC[CHKSUM](CR)

Invalid Response: ?AA[CHKSUM](CR)

! Delimiter character for a valid response

? Delimiter character for an invalid response

AA Address of the responding module (00 to FF)

S The protocols supported by the module

0: only DCON protocol is supported

1: both the DCON and Modbus RTU protocols
are supported

C Current protocol saved in EEPROM that will be
used at the next power on reset

0: the protocol set in EEPROM is DCON

1: the protocol set in EEPROM is Modbus RTU

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: \$01P

Response: !0110

Reads the communication protocol of module 01 and returns a response of 10 meaning that it supports both the DCON and Modbus RTU protocols and the protocol that will be used at the next power on reset is DCON.

Related Commands:

Section 2.25 \$AAPN

2.25 \$AAPN

Description:

Sets the communication protocol.

Syntax:

\$AAPN[CHKSUM](CR)

\$ Delimiter character

AA Address of the module to be read (00 to FF)

P Command to set the communication protocol

N 0: DCON protocol

1: Modbus RTU protocol

Before using this command, the rear slide switch must be in the INIT position, see Section A.1 for details. The new protocol is saved in the EEPROM and will be effective after the next power on reset.

Response:

Valid Response: !AA[CHKSUM](CR)

Invalid Response: ?AA[CHKSUM](CR)

! Delimiter character for a valid response

? Delimiter character for an invalid response

AA Address of the responding module (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: \$01P1

Response: ?01

Sets the communication protocol of module 01 to Modbus RTU and returns an invalid response because the module is not in INIT mode.

Command: \$01P1

Response: !01

Sets the communication protocol of module 01 to Modbus RTU and returns a valid response.

Related Commands:

Section 2.24 \$AAP

Related Topics:

Section A.1 INIT Mode

Note:

This command is only applicable to M-8017, M-8018 and M-8019 series modules.

,

2.26 \$AAS0

Description:

Performs an internal calibration on a module.

Syntax:

\$AAS0[CHKSUM](CR)

\$ Delimiter character

AA Address of module to be calibrated (00 to FF)

S0 Command to perform the internal 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 Address of the responding module (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: \$01S0

Response: !01

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

Related Commands:

Section 2.27 \$AAS1

Note:

This command is only applicable to 8019 and M-8019 series modules.

Related Commands:

Section 2.26 \$AAS0

Related Topics:

Section 1.10 Calibration

Note:

This command is only applicable to 8019 and M-8019 series modules.

2.28 ~AAC

Description:

Reads the CJC (cold junction compensation) enabled/disabled status.

Syntax:

~AAC[CHKSUM](CR)

~ Delimiter character
AA Address of the module to be read (00 to FF)
C Command to read the CJC enabled/disabled status

Response:

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

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

! Delimiter character for a valid response
? Delimiter character for an invalid response
AA Address of the responding module (00 to FF)
N 0: CJC disabled
 1: CJC 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: ~01C

Response: !011

Sends a command to read the CJC enabled/disabled status of module 01 and shows that the CJC is enabled.

Related Commands:

Section 2.10 \$AA3, Section 2.16 \$AA9, Section 2.17 \$AA9SNNNN, Section 2.19 \$AAA, Section 2.20 \$AAAi, Section 2.29 ~AACN

Note:

This command is only applicable to 8018, M-8018, 8019 and M-8019 series modules.

2.29 ~AACN

Description:

Enable/disable CJC (cold junction compensation).

Syntax:

~AACN[CHKSUM](CR)

- ~ Delimiter character
- AA Address of the module to be read (00 to FF)
- C Command to enable/disable CJC
- N 0: disable CJC
1: enable CJC

Response:

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

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

- ! Delimiter character for a valid response
- ? Delimiter character for an invalid response
- AA Address of the responding module (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: ~01C0

Response: !01

Sends a command to disable CJC of module 01 and returns a valid response.

Related Commands:

Section 2.10 \$AA3, Section 2.16 \$AA9, Section 2.17 \$AA9SNNNN, Section 2.19 \$AAA, Section 2.20 \$AAAi, Section 2.28 ~AAC

Notes:

This command is only applicable to 8018, M-8018, 8019 and M-8019 series modules.

2.30 ~AAEV

Description:

Enable/Disable module calibration.

Syntax:

~AAEV[CHKSUM](CR)

- ~ Delimiter character
- AA Address of the module to be set (00 to FF)
- E Command to enable/disable calibration
- V 1: enable calibration
0: disable 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 Address of the responding module (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

Sends the command to perform a span calibration on module 01. It returns an invalid response because the “enable calibration” command 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 2.5 \$AA0, Section 2.6 \$AA1, Section 2.7 \$AA0Ci,
Section 2.8 \$AA1Ci

Related Topics:

Section 1.10 Calibration

Related Topics:

Section A.1 INIT Mode

Notes:

1. This command is only applicable to 8019 and M-8019 series modules.
2. The ~AATnn command should be sent prior to sending this command, see Section 2.33 for details.

2.32 ~AAO(Name)

Description:

Sets the name of a module.

Syntax:

~AAO(Name)[CHKSUM](CR)

~ Delimiter character

AA Address of the module to be set (00 to FF)

O Command to set the module name

(Name) 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 Address of the responding module (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.

8017, 8018, 8019 series
M-8017, M-8018, M-8019 series

Examples:

Command: ~01O8019A Response: !01
Sets the name of module 01 to be “8019A” and
returns a valid response.

Command: \$01M Response: !018019A
Reads the name of module 01 and returns “8019A”.

Related Commands:

Section 2.23 \$AAM

2.33 ~AATnn

Description:

Sets the soft INIT time out value.

Syntax:

~AATnn[CHKSUM](CR)

~	Delimiter character
AA	Address of the module to be set (00 to FF)
T	Command to set the soft INIT time out value
nn	Two hexadecimal digits representing the time out value in seconds. The max time out value is 60 seconds. When changing the Baud Rate or checksum settings without altering the INIT* pin, the ~AAI and %AANNTTCCFF commands should be sent consecutively and the time interval between the two commands should be less than the soft INIT time out. If the soft INIT time out is 0, then the Baud Rate and checksum settings cannot be changed using software only. The power on reset value of the soft INIT time out is 0.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.31 ~AAI

Related Topics:

Section A.1 INIT Mode

Notes:

1. This command is only applicable to 8019 and M-8019 series modules.
2. It is recommended that the soft INIT time out value is reset to 0 once any changes to Baud Rate and checksum settings have been completed.

2.35 ~AA0

Description:

Reads the host watchdog status of a module.

Syntax:

~AA0[CHKSUM](CR)

- ~ Delimiter character
- AA Address of the module to be read (00 to FF)
- 0 Command to read the module status

Response:

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

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

- ! Delimiter character for a valid response
- ? Delimiter character for an invalid response
- AA Address of the responding module (00 to FF)
- SS Two hexadecimal digits that represent the host watchdog status, where:
 - Bit 7: 0 indicates that the host watchdog is disabled and 1 indicates the host watchdog is enabled,
 - Bit 2: 1 indicates that a host watchdog time out has occurred and 0 indicates that no host watchdog time out has occurred.The host watchdog status is stored in EEPROM and can only be reset using the ~AA1 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: ~010	Response: !0100
Reads the host watchdog status of module 01 and returns 00, meaning that the host watchdog is disabled and no host watchdog time out has occurred.	
Command: ~020	Response: !0284
Reads the host watchdog status of module 02 and returns 84, meaning that the host watchdog is enabled and a host watchdog timeout has occurred.	

Related Commands:

Section 2.34 ~**, Section 2.36 ~AA1, Section 2.37 ~AA2, Section 2.38 ~AA3Evv

Related Topics:

Section A.2 Dual Watchdog Operation

2.36 ~AA1

Description:

Resets the host watchdog time out status of a module.

Syntax:

~AA1[CHKSUM](CR)

- ~ Delimiter character
- AA Address of the module to be set (00 to FF)
- 1 Command to reset the host watchdog time out status

Response:

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

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

- ! Delimiter character for a valid response
- ? Delimiter character for an invalid response
- AA Address of the responding module (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.

2.37 ~AA2

Description:

Reads the host watchdog time out value of a module.

Syntax:

~AA2[CHKSUM](CR)

- ~ Delimiter character
- AA Address of the module to be read (00 to FF)
- 2 Command to read the host watchdog time out 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 Address of the responding module (00 to FF)
- VV Two hexadecimal digits to represent the time out 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.

8017, 8018, 8019 series
M-8017, M-8018, M-8019 series

Examples:

Command: ~012

Response: !01FF

Reads the host watchdog time out value of module 01 and returns FF, meaning that the host watchdog time out value is 25.5 seconds.

Related Commands:

Section 2.34 ~**, Section 2.35 ~AA0, Section 2.36 ~AA1,
Section 2.38 ~AA3Evv

Related Topics:

Section A.2 Dual Watchdog Operation

2.38 ~AA3E VV

Description:

Enable/disable the host watchdog and set the host watchdog time out value of a module.

Syntax:

~AA3E VV[CHKSUM](CR)

- ~ Delimiter character
- AA Address of the module to be set (00 to FF)
- 3 Command to set the host watchdog
- E 1: enable the host watchdog
0: disable the host watchdog
- VV Two hexadecimal digits to represent the time out 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 Address of the responding module (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 of module 01 and sets the host watchdog time out value to 10.0 seconds. The module returns a valid response.

Command: ~012

Response: !0164

Reads the host watchdog time out value of module 01. The module returns 64, meaning that the host watchdog time out value is 10.0 seconds.

Related Commands:

Section 2.34 ~**, Section 2.35 ~AA0, Section 2.36 ~AA1, Section 2.37 ~AA2

Related Topics:

Section A.2 Dual Watchdog Operation

3. Modbus RTU Protocol

The Modbus protocol is developed by Modicon Inc., originally developed for Modicon controllers. Detailed information can be found at <http://www.modicon.com/techpubs/toc7.html>. You can also visit <http://www.modbus.org> to find more valuable information.

M-8000 series modules support the Modbus RTU protocol. The communication Baud Rates range from 1200bps to 115200bps. The parity, data bits and stop bits are fixed as no parity, 8 data bits and 1 stop bit. The following Modbus functions are supported.

Function Code	Description	Section
02 (0x02)	Read input status	3.1
04 (0x04)	Read input channels	3.2
70 (0x46)	Read/write module settings	3.3

Note: Function 02 is used to read the thermocouple wire opening status and is only supported by M-8019 series modules.

If the function specified in the message is not supported, then the module responds as follows.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	Function code 0x80
02	Exception code	1 Byte	01

If a CRC mismatch occurs, the module will not respond.

3.1 02 (0x02) Read Input Status

This function code is used to read the thermocouple wire opening status of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x02
02 ~ 03	Starting channel	2 Bytes	0x80 to 0x87, where 0x80 corresponds to channel 0, 0x81 corresponds to channel 1, etc
04 ~ 05	Number of input channels	2 Bytes	N, 1 to 8; (Starting channel + N) should be less than or equal to 0x88

Note: This function is only available on M-8019 series modules.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x02
02	Byte count	1 Byte	1
03	Data of input channels	1 Byte	A bit corresponds to a channel. When the bit is 1 it denotes that the channel is enabled and is either over-range, under-range or wire opening. If the bit is 0 it denotes that the channel is disabled or normal.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x82
02	Exception code	1 Byte	02: starting channel out of range 03: (starting channel + number of input channels) out of range, incorrect number of bytes received

3.2 04 (0x04) Read Input Channels

This function code is used to read from 1 to 8 contiguous analog input channels or the CJC temperature.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x04
02 ~ 03	Starting channel	2 Bytes	0 to 7 for reading analog inputs 0x80 for reading CJC temperature
04 ~ 05	Number of input channels (N)	2 Bytes	1 to 8; (Starting channel + N) <= 8 for reading analog inputs. 1 for reading CJC temperature.

Note: The CJC temperature is only available on M-8018 and M-8019 series modules.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x04
02	Byte count	1 Byte	2 x N
03 ~	Data of input channels	2 x N Bytes	When used for the CJC temperature, this is a 2's complement hex value in 0.01°C increments.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x84
02	Exception code	1 Byte	02: starting channel out of range 03: (starting channel + number of input channels) out of range, incorrect number of bytes received

3.3 70 (0x46) Read/Write Module Settings

This function code is used to read the settings of the module or change the settings of the module. The following sub-function codes are supported.

Sub-function Code	Description	Section
00 (0x00)	Read the module name	3.3.1
04 (0x04)	Set the module address	3.3.2
05 (0x05)	Read the communication settings	3.3.3
06 (0x06)	Set the communication settings	3.3.4
07 (0x07)	Read the type code	3.3.5
08 (0x08)	Set the type code	3.3.6
32 (0x20)	Read the firmware version	3.3.7
37 (0x25)	Read the channel enable/disable status	3.3.8
38 (0x26)	Set the channel enable/disable	3.3.9
41 (0x29)	Read the miscellaneous settings	3.3.10
42 (0x2A)	Write the miscellaneous settings	3.3.11
43 (0x2B)	Read the CJC offset	3.3.12
44 (0x2C)	Write the CJC offset	3.3.13
45 (0x2D)	Read the CJC enable/disable status	3.3.14
46 (0x2E)	Set the CJC enable/disable	3.3.15
47 (0x2F)	Read the CJC update setting	3.3.16
48 (0x30)	Write the CJC update setting	3.3.17

If the module does not support the sub-function code specified in the message, then it responds as follows.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	02: invalid sub-function code

3.3.1 Sub-function 00 (0x00) Read module name

This sub-function code is used to read the name of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x00

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x00
03 ~ 06	Module name	4 Bytes	0x00 0x70 0x17 0x00 for M-8017 series modules 0x00 0x70 0x18 0x00 for M-8018 series modules 0x00 0x70 0x19 0x00 for M-8019 series modules

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: incorrect number of bytes received

3.3.2 Sub-function 04 (0x04) Set module address

This sub-function code is used to set the address of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x04
03	New address	1 Byte	1 to 247
04 ~ 06	Reserved	3 Bytes	0x00 0x00 0x00

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x04
03	Set address result	1 Byte	0: OK, others: error
04 ~ 06	Reserved	3 Bytes	0x00 0x00 0x00

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: new address out of range, reserved bytes should be filled with zero, incorrect number of bytes received

3.3.3 Sub-function 05 (0x05) Read communication settings

This sub-function code is used to read the communication protocol settings of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x05
03	Reserved	1 Byte	0x00

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x05
03	Reserved	1 Byte	0x00
04	Baud Rate	1 Byte	Baud Rate code, see Section 1.11 for details.
05 ~ 07	Reserved	3 Bytes	0x00 0x00 0x00
08	Mode	1 Byte	0: DCON protocol 1: Modubs RTU protocol
09 ~ 10	Reserved	2 Bytes	0x00 0x00

Note: This information is the data saved in the EEPROM and will be used for the next power-on reset. It is not the currently used settings.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: reserved byte should be filled with zero, incorrect number of bytes received

3.3.4 Sub-function 06 (0x06) Set communication settings

This sub-function code is used to set the communication protocol of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x06
03	Reserved	1 Byte	0x00
04	Baud Rate	1 Byte	Baud Rate code, see Section 1.11 for details.
05 ~ 07	Reserved	3 Bytes	0x00 0x00 0x00
08	Mode	1 Byte	0: DCON protocol 1: Modubs RTU protocol
09 ~ 10	Reserved	2 Bytes	0x00 0x00

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x06
03	Reserved	1 Byte	0x00
04	Baud Rate	1 Byte	0: OK, others: error
05 ~ 07	Reserved	3 Bytes	0x00 0x00 0x00
08	Mode	1 Byte	0: OK, others: error
09 ~ 10	Reserved	2 Bytes	0x00 0x00

Note: The new Baud Rate and protocol will be effective after the next power-on reset.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: Baud Rate or mode out of range, reserved bytes should be filled with zero, incorrect number of bytes received

3.3.5 Sub-function 07 (0x07) Read type code

This sub-function code is used to read the type code information of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x07
03 ~ 04	Reserved	1 Bytes	0x00
04	Channel	1 Byte	0x00 ~ 0x07 for M-8019 series modules 0x00 for M-8017 and M-8018 series modules

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x07
03	Type code	1 Byte	Type code, see Section 1.11 for details.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: reserved bytes should be filled with zero, channel out of range for M-8019 series modules, incorrect number of bytes received

3.3.6 Sub-function 08 (0x08) Set type code

This sub-function code is used to set the type code of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x08
03	Reserved	1 Byte	0x00
04	Channel	1 Byte	0x00 ~ 0x07 for M-8019 series modules 0x00 for M-8017 and M-8018 series modules
05	Type code	1 Byte	Type code, see Section 1.11 for details.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x08
03	Type code	1 Byte	0: OK others: error

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: type code out of range, channel out of range for M-8019 series modules, reserved bytes should be filled with zero, incorrect number of bytes received

3.3.7 Sub-function 32 (0x20) Read firmware version

This sub-function code is used to read the firmware version information of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x20

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x20
03	Major version	1 Byte	0x00 ~ 0xFF
04	Minor version	1 Byte	0x00 ~ 0xFF
05	Build version	1 Byte	0x00 ~ 0xFF

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: incorrect number of bytes received

3.3.8 Sub-function 37 (0x25) Read channel enabled/disabled status

This sub-function code is used to read the enabled/disabled status of each channel in a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x25

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x25
03	Enabled/disabled status	1 Byte	0x00 ~ 0xFF, enabled/disabled status of each channel, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 1 it denotes that the channel is enabled and 0 denotes that the channel is disabled.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: incorrect number of bytes received

3.3.9 Sub-function 38 (0x26) Set channel enable/disable

This sub-function code is used to specify the channels to be enabled in a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x26
03	Enable/disable setting	1 Byte	0x00 ~ 0xFF, enable/disable setting of each channel, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 1 it denotes that the channel is enabled and 0 denotes that the channel is disabled.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x26
03	Enable/disable setting	1 Byte	0: OK others: error.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: enable/disable setting out of range, incorrect number of bytes received

3.3.10 Sub-function 41 (0x29) Read miscellaneous settings

This sub-function code is used to read the miscellaneous settings of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x29

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x29
03	Miscellaneous settings	1 Byte	Bit 7: filter setting for M-8017 and M-8018 series modules, reserved for M-8019 series modules 0: 60Hz rejection 1: 50Hz rejection Bit 6: reserved Bit 5: for M-8017F only, reserved for other modules 0: normal mode 1: fast mode Bit 4~0: reserved

Note: The reserved fields are filled with zeros.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: incorrect number of bytes received

3.3.11 Sub-function 42 (0x2A) Write miscellaneous settings

This sub-function code is used to set the miscellaneous settings of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2A
03	Miscellaneous settings	1 Byte	Bit 7: filter setting for M-8017 and M-8018 series modules, reserved for M-8019 series modules 0: 60Hz rejection 1: 50Hz rejection Bit 6: reserved Bit 5: for M-8017F only, reserved for other modules 0: normal mode 1: fast mode Bit 4~0: reserved

Note: The reserved fields are filled with zeros.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2A
03	Miscellaneous settings	1 Byte	0: OK others: error

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: reserved bits should be filled with zero, incorrect number of bytes received

3.3.12 Sub-function 43 (0x2B) Read CJC offset

This sub-function code is used to read the CJC offset setting of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2B
03	Reserved	1 Byte	0x00

Note: This sub-function is only available on M-8018 and M-8019 series modules.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2B
03 ~ 04	CJC offset	2 Bytes	This is a 2's complement hex CJC offset value in 0.01°C increments

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: incorrect number of bytes received, reserved byte is not zero

3.3.13 Sub-function 44 (0x2C) Write CJC offset

This sub-function code is used to set the CJC offset setting of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2C
03	Reserved	1 Byte	0x00
04 ~ 05	CJC offset	2 Bytes	This is a 2's complement hex CJC offset value in 0.01°C increments, the absolute value should be less than or equal to 0x1000

Note: This sub-function is only available on M-8018 and M-8019 series modules.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2C
03	Miscellaneous settings	1 Byte	0: OK others: error

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: reserved byte should be zero, CJC offset value out of range, incorrect number of bytes received

3.3.14 Sub-function 45 (0x2D) Read CJC enabled/disabled status

This sub-function code is used to read the CJC enabled/disabled status of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2D
03	Reserved	1 Byte	0x00

Note: This sub-function is only available on M-8018 and M-8019 series modules.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2D
03	CJC enabled/disabled status	1 Byte	0: CJC disabled 1: CJC enabled

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: incorrect number of bytes received

3.3.15 Sub-function 46 (0x2E) Set CJC enable/disable

This sub-function code is used to enable/disable the CJC of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2E
03	Reserved	1 Byte	0x00
04	Enable/disable CJC	1 Byte	0: disable CJC 1: enable CJC

Note: This sub-function is only available on M-8018 and M-8019 series modules.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2E
03	Miscellaneous settings	1 Byte	0: OK others: error.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: reserved byte is not zero, setting byte incorrect, incorrect number of bytes received

3.3.16 Sub-function 47 (0x2F) Read CJC update setting

This sub-function code is used to read the CJC update settings of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2F

Note: This sub-function is only available on M-8019 series modules.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x2F
03	CJC temperature update setting	1 Byte	0: CJC temperature update has been stopped 1: CJC temperature update has been started 2: Update CJC temperature once only

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: incorrect number of bytes received

3.3.17 Sub-function 48 (0x30) Write CJC update setting

This sub-function code is used to set the CJC update setting of a module.

Request

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x30
03	CJC temperature update setting	1 Byte	0: stop CJC temperature update 1: start CJC temperature update 2: Update CJC temperature once only, after the command is received

Note: This sub-function is only available on M-8019 series modules.

Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0x46
02	Sub function code	1 Byte	0x30
03	Miscellaneous settings	1 Byte	0: OK, others: error.

Error Response

00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	03: setting byte incorrect, incorrect number of bytes received

4. Troubleshooting

If you are having difficulty using the I-8000 or M-8000 module, here are some suggestions that may help. If you cannot find the answers you need in these guides, contact SuperLogics Product Support. Contact information is located in Section 1.14.

4.1 Communicating with the module

If you attempt to communicate with the module and receive no response, first check the following:

- Make sure the supplied power is within the range of +10 to +30 V DC. If the supplied power is OK, then the power LED should be on.
- When the module receives a command, the power LED is set to “off”. The power LED is shown as “on” after the module responds. This method can be used to check whether the module has received a command sent from the host.
- If possible, use another device to check whether the host can communicate with the device through the same RS-485 network.
- If the host is a PC installed with a Windows operating system, then execute the DCON Utility to determine whether the module can be found. The DCON Utility can be downloaded from the SuperLogics website www.superlogics.com . The DCON Utility documentation can be found in the “**Getting Started For I-8000 Series Modules**” manual.
- Set the module to “INIT mode” and communicate with the module using the following settings: address 00, Baud Rate 9600bps, no checksum and DCON protocol. See Section A.1 for details.

4.2 Reading Data

If the data read from the input channel is not correct, first check the following:

- Make sure the type code and data format settings are correct. For 8017, 8018, M-8017 and M-8018 series modules, the type code is set by using the %AANNTTCFF command, see Section 2.1 for details. For 8019 and M-8019 series modules, the type code is set by using the \$AA7CiRrr command, see Section 2.17 for details. The data format is set by using the %AANNTTCFF command. For the Modbus RTU protocol, the type code is set by using sub-function 08h of the function 46h.
- If the voltage read by the module is incorrect, then it may be because the calibration parameters stored in the non-volatile memory are corrupted. You can calibrate the module by yourself. Be sure to read Section 1.10 in detail before doing any calibration. For 8019 and M-8019 series modules, use the \$AAS1 command to reload the factory calibration parameters, see Section 2.27 for details.

A. Appendix

A.1 INIT Mode

Each 8000 and M-8000 module has a built-in EEPROM to store configuration information such as module address, type code, Baud Rate, etc. Occasionally, the configuration of a module may be forgotten and there are no visual indications of the configuration of the module. It is difficult to communicate with the module when the configuration of the module is unknown. To help avoid this problem, the 8000 and M-8000 series has a special mode called “**INIT mode**”. When the module is powered on in “**INIT mode**” the configuration of the module is reset as follows, allowing it to be operated as normal.

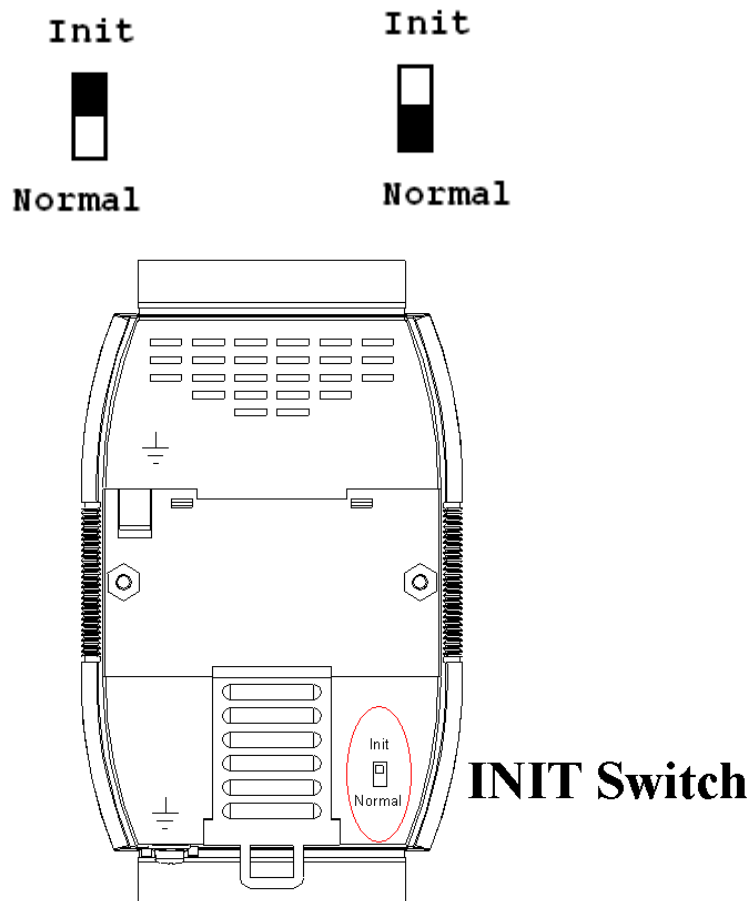
1. Address: 00
2. Baud Rate: 9600 bps
3. No checksum
4. Protocol: DCON

The configuration information stored in the EEPROM is not changed and they can be read by sending the \$002(CR) command at 9600bps.

There are commands that require the module to be in INIT mode. They are:

1. %AANNTTCCFF when changing Baud Rate and checksum settings. See Section 2.1 for details.
2. \$AAPN, see Section 2.25 for details.

Originally, the INIT mode is accessed by connecting the INIT* terminal to the GND terminal. For the 8017 and 8017F modules, the INIT* terminal is shared with the Vin7- terminal. The JP1 jumper should be set correctly as shown in Section 1.7.1. New 8000 and M-8000 modules have the INIT switch located on the rear side of the module allow easier access to INIT mode. For these modules, INIT mode is accessed by sliding the INIT switch to the Init position as shown below.



A.2 Dual Watchdog Operation

Dual Watchdog = Module Watchdog + Host Watchdog

The Module Watchdog is a hardware reset circuit that monitors the operating status of the module. While working in harsh or noisy environments, the module may be shut down by external signals. The circuit allows the module to work continuously without disruption.

The Host Watchdog is a software function that monitors the operating status of the host. Its purpose is to prevent problems due to network/communication errors or host malfunctions. When a host watchdog time out occurs, the module will reset all outputs to a safe state in order to prevent any erroneous operations of the controlled target.

8000 series modules include an internal Dual Watchdog, making the control system more reliable and stable.

For more information regarding the Dual Watchdog, please refer to Chapter 5 of the “**Getting Started For I-8000 Series Modules**” manual that can be downloaded from the SuperLogics website www.superlogics.com

A.3 Thermocouple

When two wires composed of dissimilar homogeneous metals are joined at one end, a thermoelectric electromotive force (emf) appears that depends only on the metals and the junction temperature. This is called the Seebeck effect. A pair of different metals with a fixed junction at one end constitutes a **thermocouple**. For small changes in temperature, the emf is linearly proportional to the temperature. This implies that the temperature reading can be obtained by measuring the emf.

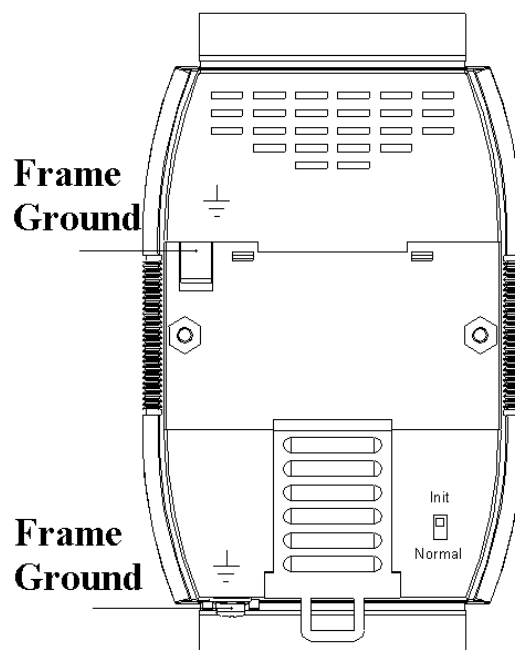
We cannot measure the emf, V_1 , directly because when a voltmeter is connected to the thermocouple, another emf, V_2 , is created at the (cold) junction of the thermocouple and the voltmeter. The cold junction compensation method is used to resolve the problem. Using another sensor, e.g. a thermistor, to measure the cold junction temperature, T_2 , we can calculate the emf, V_2 , which corresponds to T_2 . The thermocouple emf, V_1 , can be obtained by adding V_2 to that measured by the voltmeter and then the temperature.

A.4 Frame Ground

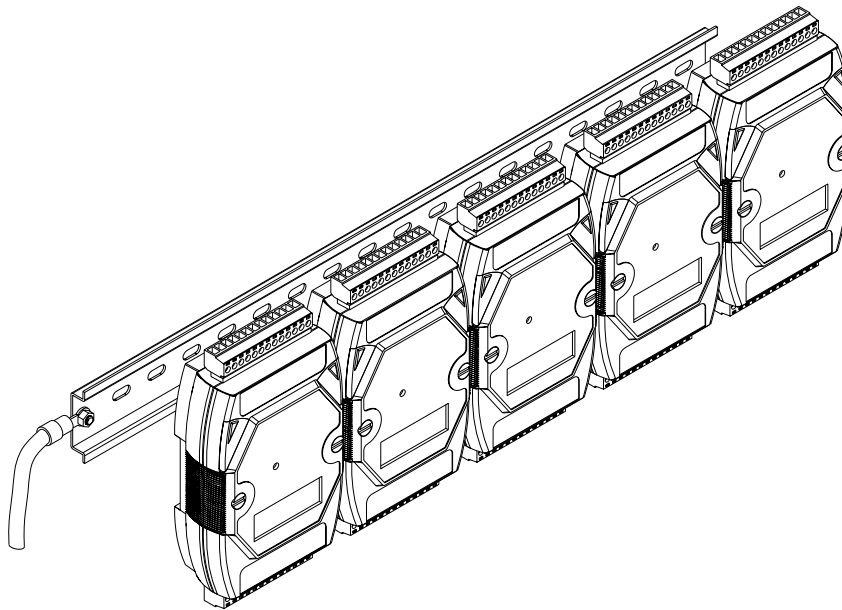
Electronic circuits are constantly vulnerable to ESD which become worse in a continental climate area. Some 8000, and all M-8000, modules feature a new design for the frame ground. The frame ground provides a path for bypassing ESD, which provides enhanced static protection (ESD) abilities and ensures the module is more reliable.

Either of the following options will provide a better protection for the module:

1. If the module is DIN rail mounted, connect the DIN rail to the earth ground because the DIN rail is contacted with the upper frame ground as shown in the figure below.
2. Alternatively, connect the lower frame ground terminal, as shown in the figure below, to a wire and connected the wire to the earth ground.



New DIN rail models are available that can easily be connected to the earth ground. Each are made of stainless steel, which is stronger than those made of aluminum. There is a screw at one end and a ring terminal is included as shown in the figure below. Refer to Section 1.13.1 for more information about the new DIN rail models.



A.5 Node Information Area

Each 8000 and M-8000 module has a built-in EEPROM to store configuration information such as module address, type code, Baud Rate, etc. One minor drawback is that there are no visual indications of the configuration of the module. New 8000 and M-8s000 modules include node information areas that are protected by a cover, as shown below, and can be used to make a written record of the node information, such as module address, Baud Rate, etc. To access the node information areas, first slide the covers outward, as shown in the figure below.

