

V1.0 2008.01.28

# **RemoDAQ-8037 Module**

## **User's Manual**



**Beijing DSLC Technology Co. , Ltd**

## Copyright Notice

This document is copyrighted, 2008, by **Beijing DSLC Technology Co. , Ltd.**

All rights are reserved. **Beijing DSLC Technology Co. , Ltd** reserves the right to make improvements to the products described in this manual at any time without notice.

No part of this manual may be reproduced, copied, translated or transmitted in any form or by any means without the prior written permission of **Beijing DSLC Technology Co. , Ltd**. Information provided in this manual is intended to be accurate and reliable. However, **Beijing DSLC Technology Co. , Ltd** assumes no responsibility for its use, or for any infringements upon the rights of third parties, which may result from its use.

## Acknowledgments

RemoDAQ is a trademark of **Beijing DSLC Technology Co. , Ltd.**

**Edition 1.0**

**Jan. 2008**

## **Additional Information and Assistance**

1. Visit the DSLC websites at [www.DSLC.com.cn](http://www.DSLC.com.cn) in which you can find the latest information about the product.
2. Contact your distributor, sales representative, or DSLC's customer service center for technical support if you need additional assistance. Please have the following information ready before you call :
  - Product name and serial number
  - Description of your peripheral attachments
  - Description of your software (operating system, version , application software, etc.)
  - A complete description of the problem
  - The exact wording of any error messages

## Table of Contents

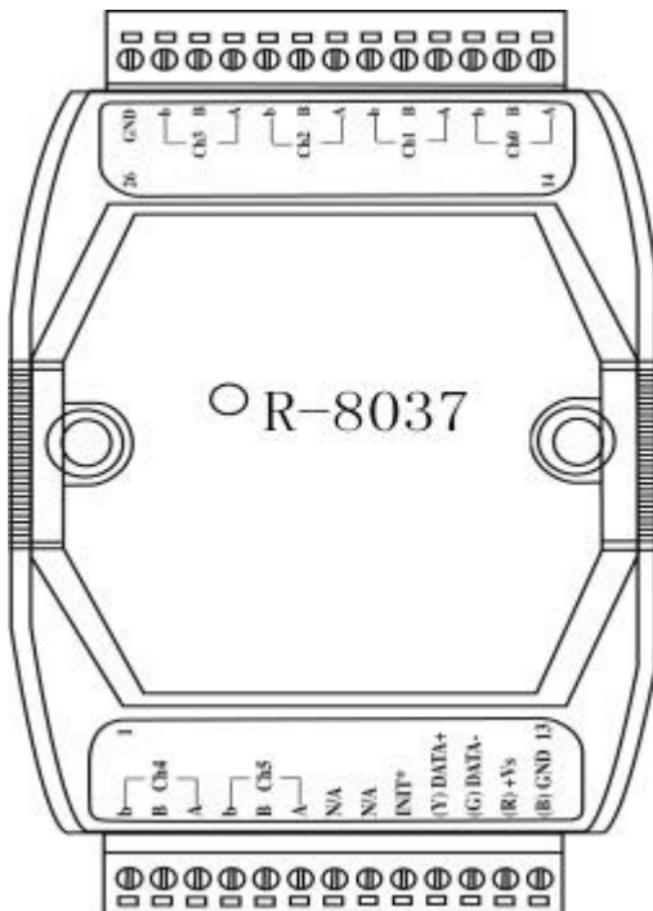
1	Introduction.....	4
1.1	Pin Assignment & Specifications .....	5
1.2	Block Diagram .....	7
1.3	Application Wiring .....	7
1.4	Default Setting.....	7
1.5	Calibration.....	8
1.6	Install List.....	8
2	Initialization & Installation.....	10
2.1	Installation Guideline.....	10
2.2	Software Installation .....	11
2.3	Basic configuration and hook-up.....	11
2.4	Baudrate and Checksum.....	14
3	Command Set .....	15
3.1	%AANNTTCCFF.....	19
3.2	#AA.....	21
3.3	#AAN .....	22
3.4	\$AA0Ci.....	23
3.5	\$AA1Ci.....	24
3.6	\$AA2 .....	25
3.7	\$AA5VV .....	26
3.8	\$AA6.....	27
3.9	#** .....	28
3.10	\$AA4.....	29
3.11	\$AAF .....	30
3.12	\$AA7CiRr.....	31
3.13	\$AA8Ci.....	32
3.14	\$AAXnnnn.....	33
3.15	\$AAY .....	34
3.16	\$AAB .....	35
3.17	\$AAS0.....	36
3.18	\$AAS1.....	37

## 1 Introduction

The RemoDAQ-8000 Series is a set of intelligent sensor to computer interface modules containing built in microprocessor. They are remotely controlled through a simple set of commands issued in ASCII format and transmitted in RS-485 protocol. They provide signal conditioning, isolation, ranging, A/D and D/A conversion, data comparison, digital communication, timer/counter, wireless communication, collection AC and other functions.

RemoDAQ-8037 RTD Input Module provides 6 RTD input channels for different types of RTD signal such as Pt, Ni, Balco. It is an effective solution in industrial & building automation. Normally, broken external wire will lead to an inaccurate current value; however, RemoDAQ-8037 provides a broken wire detecting function. Therefore, users can easily fix the broken wire problems. This module can accept RTD sensors that have two or three wires.

## 1.1 Pin Assignment & Specifications



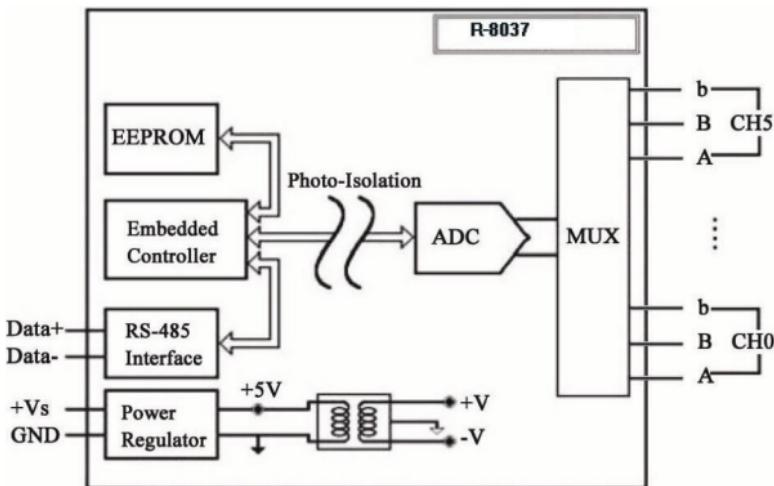
## RemoDAQ-8037 User's Manual

---

### RemoDAQ-8037 Specifications :

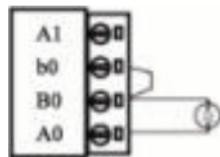
Input channel	6
Input type	2, 3 wire
RTD type	Pt100、 Pt1000、 BALCO500、 NI
Temperature range	Pt100 : -50-150 ,0-100 ,0-200 ,0-400 ,-200-200 BALCO500 : -30-120 NI50RTD : -80-100 NI508RTD : 0-100
Speed (in bps)	1200,2400,4800,9600,19.2K, 38.4K, 57.6K, 115.2K
Maximum distance	4000ft. (1200m.)
Conversion rate	12 samples/sec (60/50Hz)
Bandwidth	5.24Hz
Accuracy	$\pm 0.1\%$ or better
Zero drift	3 $\mu$ V $^{\circ}$ C
Span drift	25ppm/
CMR@50/60Hz	150dB
NMR@50/60Hz	100dB
Isolation voltage	3000VDC
Power supply	10~30VDC
Power consumption	1.2W
I/O connector type	13-pin plug-terminal
Environment	Operating Temperature : -20 ~ 70 $^{\circ}$ C
	Humidity : 5 ~ 95%, non-condensing

## 1.2 Block Diagram

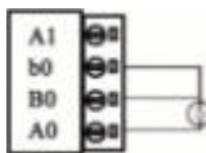


## 1.3 Application Wiring

2-wire RTD connection



3-wire RTD connection



## 1.4 Default Setting

- Address : 01
- RTD type : Type 20 , Pt100,-100 ~+100
- Baudrate : 9600 bps
- Checksum disable,60Hz rejection ,engineer unit format

## 1.5 Calibration

### Calibrate Order :

1. Apply power to the module and let it warm up for about 30 minutes
2. Setting type, join resistance on 0 channel
3. Enable calibration
4. Apply zero calibration resistance
5. Perform zero calibration command
6. Apply span calibration resistance
7. Perform span calibration command
8. Repeat step 3 to step 7 three times.

## 1.6 Install List

### Baudrate Setting (CC)

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

### Data format setting ( FF )

7	6	5	4	3	2	1	0
*1	*2			0			*3

\*1 : 0=60Hz Restrain      1=50Hz Restrain

\*2 : Checksum : 0=Disabled; 1=Enable

\*3 : 00 = Engineering Unit Format

  01 = Percentage Format

  10 = 2's Complement HEX Format

  11 = Ohms

### RTD Input Type Table

Code	Input span	Data format	+F.S.	Zero	-F.S
20	100 RTD	Project Unit	+100.00	-100.000	0.01

# RemoDAQ-8037 User's Manual

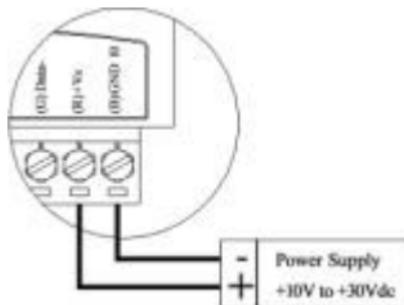
---

	a=0.00385 -100 ~100	% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	8000	1LSB
		Ohm	138.50	60.60	10m
21	100 RTD a=0.00385 0 ~100	Project Unit	+100.00	00.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	0000	1LSB
		Ohm	138.50	100.00	10m
22	100 RTD a=0.00385 0 ~200	Project Unit	+200.00	00.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	0000	1LSB
		Ohm	175.84	100.00	10m
23	100 RTD a=0.00385 0 ~600	Project Unit	+600.00	00.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	0000	1LSB
		Ohm	313.59	100.00	10m
24	100 RTD a=0.00392 -100 ~100	Project Unit	+100.00	-100.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	8000	1LSB
		Ohm	139.16	60.60	10m
25	100 RTD a=0.00392 0 ~100	Project Unit	+100.00	00.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	0000	1LSB
		Ohm	139.16	100.00	10m
26	100 RTD a=0.00392 0 ~200	Project Unit	+200.00	00.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	0000	1LSB

		Ohm	117.13	100.00	10m
27	100 RTD a=0.00392 0 ~600	Project Unit	+600.00	00.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	0000	1LSB
		Ohm	317.28	100.00	10m
28	120 NickelRTD a=0.00392 -80 ~100	Project Unit	+100.00	-80.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	8000	1LSB
		Ohm	200.64	66.60	10m
29	100 NickelRTD a=0.00392 0 ~100	Project Unit	+100.00	00.000	0.01
		% ( FSR )	+100.000	+000.00	0.01%
		2's Complement HEX	7FFF	0000	1LSB
		Ohm	200.64	120.00	10m

## 2 Initialization & Installation

### 2.1 Installation Guideline



### **Figure 2-1 Power Supply Connections**

We advise that the following standard colors (as indicated on the modules) be used for power lines :

+Vs (R)	Red
GND (B)	Black

We advise that the following standard colors (as indicated on the modules) be used for the communication lines :

DATA+ (Y)	Yellow
DATA- (G)	Green

## **2.2 Software Installation**

1. If you have already installed “RemoDAQ-8000 Utility”, then skip other steps.
2. Backup your software diskette.
3. Insert “RemoDAQ-8000 Utility” disc into CD-ROM :
4. Change drive to the path of CD-ROM. For example, your drive of CD-ROM is F :then change the drive to F :
5. Find the setup of “RemoDAQ-8000 Utility” and run it.
6. Please follow the steps of setup program then you can successfully install the RemoDAQ-8000 Utility.

## **2.3 Basic configuration and hook-up**

Before placing a module in an existing network, the module should be configured. Though all modules are initially configured at the factory, it is recommended to check that the baud rate is set correctly.

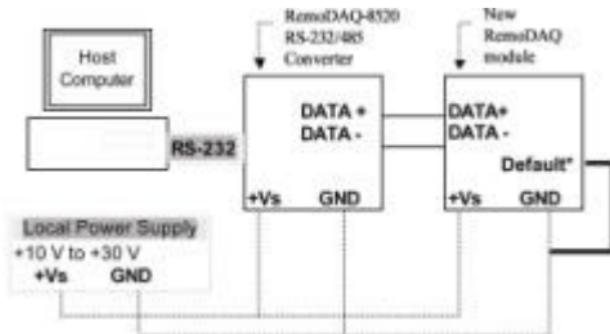
### **Default Factory Settings**

Baud rate : 9600 Bit/sec.

Address : 01 (hexadecimal)

Checksum : disable

The basic hook-up for module configuration is shown below.



**Figure 2-2** Layout for Initialization of the RemoDAQ module

The following items are required to configure a module : a RemoDAQ converter module, a personal computer with RS-232 port (baudrate set to 9600) and the RemoDAQ utility software.

## Configuration with the RemoDAQ Utility Software

The easiest way to configure the RemoDAQ module is by using the RemoDAQ utility software : an easy-to-use menu-structured program will guide you through every step of the configuration.

## Configuration with the RemoDAQ command set

RemoDAQ modules can also be configured by issuing direct commands from a terminal emulation program within what is part of the RemoDAQ utility software.

The following example guides you through the setup of an analog input module. Assume that RemoDAQ-8031 still has its default settings (baud rate 9600 and address 01h). Before the module is reconfigured, it is first requested to send its default settings.

To change the configuration of the analog input module, the following command is issued :

%0107200600(cr)

% = change configuration

01 = target module at address 00 to :

07 = change address to 07 hexadecimal

20 = set input range to Type 20

06 = set baud rate to 9600

00 = set integration time to 50 ms (60 Hz)

disable checksum

set data format to engineering units

(See Chapter 3, Command Set for a full description of the syntax of the configuration command for module)

When the module received the configuration command it will respond with its new address : !07(cr)

**NOTICE :** *All reconfiguration except changing of baud rate and checksum values can be done dynamically, i.e. the modules need not to be reset. When changing the baud rate or checksum, these changes should be made for all connected devices. After reconfiguration, all modules should be powered down and powered up to force a reboot and let the changes take effect.*

## 2.4 Baudrate and Checksum

RemoDAQ modules contain EEPROMs to store configuration information and calibration constants. The EEPROM replaces the usual array of switches and ports required to specify baudrate, input/output range etc.

All of the RemoDAQ modules can be configured remotely through their communication ports, without having to physically alter port or switch settings.

Forcing the module in the INIT\* state does not change any parameters in the module's EEPROM. When the module is in the INIT\* state with its INIT\* and GND terminals shorted, all configuration settings can be changed and the module will respond to all other commands normally.

### Changing Baud rate and Checksum

Baud rate and checksum settings have several things in common :

- They should be the same for all modules and host computer.
- Their setting can only be changed by putting a module in the INIT\* state.
- Changed settings can only take effect after a module is rebooted

To alter baudrate or checksum settings you must perform the following steps :

- Power on all components except the RemoDAQ Module.
- Power the RemoDAQ module on while shorting the

INIT\* and GND terminals

- Wait at least 7 seconds to let self calibration and ranging take effect.
- Configure the checksum status and/or the baud rate.
- Switch the power to the RemoDAQ Module OFF.
- Remove the grounding of the INIT\* terminal and power the module on.
- Wait at least 7 seconds to let self calibration and ranging take effect.
- Check the settings (If the baud rate has changed, the settings on the host computer should be changed accordingly.)

## 3 Command Set

### Introduction

To avoid communication conflicts when several devices try to send data at the same time, all actions are instigated by the host computer. The basic form is a command/response protocol with the host initiating the sequence.

When modules are not transmitting they are in listen mode. The host issues a command to a module with a specified address and waits a certain amount of time for the module to respond. If no response arrives, a timeout aborts the sequence and returns control to the host.

Changing RemoDAQ's configuration might require the

module to perform auto calibration before changes can take effect. Especially when changing the range, the module has to perform all stages of auto calibration that it also performs when booted. When this process is under way, the module does not respond to any other commands.

The command set includes the exact delays that might occur when modules are reconfigured.

### **Syntax**

[delimiter character][address][command][data][checksum]  
[carriage return]

Every command begins with a delimiter character. There are four valid characters : a dollar sign \$, a pound sign #, a percentage sign % and an at sign @.

The delimiter character is followed by a two-character address (hexadecimal) that specifies the target module. The actual two-character command follows the address.

Depending on the command, an optional data segment follows the command string. An optional two character checksum may be appended to the total string. Every command is terminated by a carriage return (cr).

### **Calculate Checksum :**

1. Calculate ASCII sum of all characters of command (or response) string except the character return(cr).
2. Mask the sum of string with Offh.

### **Example :**

Command string : \$012(cr)

Sum of string=‘\$’+‘0’+‘1’+‘2’=24h+30h+31h+32h=B7h

The checksum is B7h , and [CHK] = “ B7”

Command string with checksum : \$012B7(cr)

Response string : !01200600 (cr)

Sum of string : ‘!’+‘0’+‘1’+‘2’+‘0’+‘0’+‘6’+‘0’+‘0’

=1h+30h+31h+32h+30h+30h+36h+30h+30h=1AAh

The checksum is AAh , and [CHK] = “ AA”

Response string with checksum : !01200600AA (cr)

**General Command Sets**

Command	Name	Command Description	Sec.
%AANNTTCCFF	Configuration	Sets the address, input range, baudrate, dataformat, checksum status	3.1
#AA	Analog data in	Return the input value from the module in the currently configured data format	3.2
#AAN	Analog data in	Return the input value from the module channels N in the currently configured data format	3.3
\$AA0i	Span calibration	Calibrates an AI module to correct for gain errors	3.4
\$AA1i	Zero Calibration	Calibrates an AI module to correct for gain errors	3.5
\$AA2	Configuration status	Return the configuration parameters for the module	3.6

## RemoDAQ-8037 User's Manual

---

\$AA5VV	Enable/Disable channels for multiplexing	Enable or disable the individual channels in the module	3.7
\$AA6	Read channel status	Get the enable/disable status of all channels in the module	3.8
#**	Synchronized Sampling	Orders all analog input modules to sample their input values and store them in special registers	3.9
\$AA4	Read synchronized data	Returns the value that was stored in the specified module's register after the #** command	3.10
\$AAF	Read firmware version	Return the firmware version code	3.11
\$AA7CiRrr	Single channel range configuration	Configure the input type and range of the specified channel	3.12
\$AA8Ci	Read single channel range configuration	Get the input type and range of the specified channel	3.13
\$AAXnnnn	Watchdog timer setting	Set communication WDT cycle time from 0000~9999	3.14
\$AAY	Read Watchdog timer setting	Read the cycle time setting of communication WDT	3.15
\$AAB	Channel Diagnose	Diagnose channel status in over range, under range, and wire opening	3.16
\$AAS0	Internal Calibration	Internal selfcalibration for offset and gain errors	3.17
\$AAS1	Reload default parameter	Reload factory default calibrating parameter to overwrite current calibrating parameter	3.18

### 3.1 %AANNTTCCFF

**Name :** Configuration

**Description :** Sets address, type code, baudrate, data format

**Syntax :** %AANNTTCCFF (cr)

% delimiter character.

AA address of setting module (00-FF)

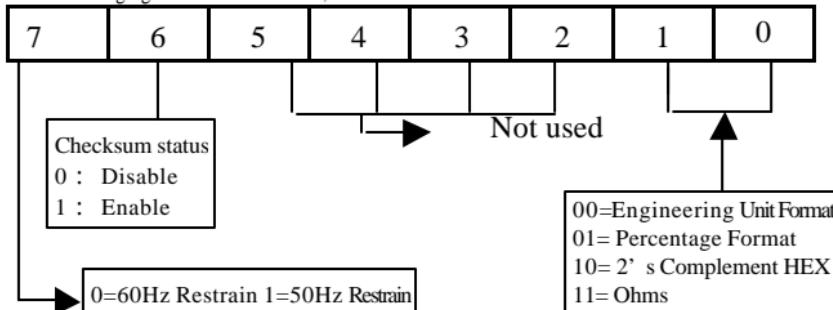
NN New address (00-FF)

TT New type

CC New baudrate

FF New data format

When changing baudrate or checksum, we should INIT\* termination land.



**Figure 3-1 Data format setting of AI modules**

**Response :** !AA(cr) if the command was valid.

?AA(cr) if an invalid operation was entered. If the INIT\* terminal was not grounded when attempting to change baud rate or checksum settings.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of setting module (00-FF)

(cr) is the terminating character, carriage return (0Dh)

## Example :

Command : %0102200600(cr) Response : !02(cr)

Change address from 01 to 02,an input type 20, baud rate 9600, integration time 50 ms (60 Hz), engineering units data format and no checksum checking or generation.

The response indicates that the command was received.

**Table 3-1 Input Rang Codes (Type Code)**

Code	RTD type	Temperature Range
20	Platinum100,a=0.00385	-100 ~ 100
21	Platinum100,a=0.00385	0 ~ 100
22	Platinum100,a=0.00385	0 ~ 200
23	Platinum100,a=0.00385	0 ~ 600
24	Platinum100,a=0.00392	-100 ~ 100
25	Platinum100,a=0.00392	0 ~ 100
26	Platinum100,a=0.00392	0 ~ 200
27	Platinum100,a=0.00392	0 ~ 600
28	NickelRTD,a=0.00392	-80~100
29	NickelRTD,a=0.00392	0 ~ 100

**Table 3-2 Baudrate Code**

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

## 3.2 #AA

**Name :** Analog Data In

**Description :** Command will return the input value from module in the currently configured data format.

**Syntax** #AA (cr)

# delimiter character.

AA address of reading module (00~FF)

(cr) is the terminating character, carriage return (0Dh).

**Response :** >(data)(cr)

Syntax error or communication error may get no response.

> delimiter character.

data AI input value, the data is the combination for each channel respectively

**Example :**

Command : #01 Response : >+02.555

Read analog input value at address 01, return with +02.555

### 3.3 #AAN

**Name :** Analog Data From channel N

**Description :** The command will return the input value from one of the 8 channels of a specified (AA) module in the currently configured data format.

**Syntax :** #AAN (cr)

# delimiter character.

AA address of reading module (00~FF)

N channel

(cr) is the terminating character, carriage return (0Dh).

**Response :** >(data)(cr)

Syntax error or communication error may get no response.

> delimiter character.

data AI input value.(For data formats, refer sec.1.6).

**Example :**

Command : #32 Response : >+02.455

Read address 03 and channel 2, return +02.455

Command : #28 Response : ?02

Read address 02 and channel 8, return error channel number

### 3.4 \$AA0Ci

**Name :** Span Calibration

**Description :** The command calibrates a specified channel to correct for gain errors.

**Syntax :** \$AA0Ci (cr)

\$ delimiter character.

AA address of the module that is to be calibrated (00~FF)

0 span calibration command.

Ci the specified input channel you want to calibrate.

(cr) the terminating character, carriage return (0Dh).

**Response :** !AA(cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

In order to successfully calibrate an analog input module's input range, a proper calibration input signal should be connected to the analog input module before and during the calibration.

**Example :**

Command : \$010C3 Response : !01

Span calibration of channel 3 of the modulue at address 01, and return success.

### 3.5 \$AA1Ci

**Name :** Zero Calibration

**Description :** Calibrates module to correct for gain errors.

**Syntax :** \$AA1Ci (cr)

\$ delimiter character.

AA address of the module that is to be calibrated (00~FF)

1 zero calibration command.

Ci the specified input channel you want to calibrate.

(cr) the terminating character, carriage return (0Dh).

**Response :** !AA(cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

In order to successfully calibrate an analog input module's input range, a proper calibration input signal should be connected to the analog input module before and during the calibration.

**Example :**

Command : \$011C2      Response : !01

Zero calibration of channel 2 of the modulue at address 01, and return success.

## 3.6 \$AA2

**Name :** Configuration Status

**Description :** The command requests the return of the configuration data from the analog input module at address AA.

**Syntax :** \$AA2 (cr)

\$ delimiter character.

AA address of reading module (00~FF)

2 the Configuration Status command.

(cr) the terminating character, carriage return (0Dh).

**Response :** !AATTCCFF(cr) if the command is valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of module(00~FF)

TT represents the type code.

CC represents the baud rate code.

FF data format

(Also see the %AANNTTCCFF configuration command)

**Example :**

Command : \$012 Response : !01200600

Read address 01 configuration,return success

### 3.7 \$AA5VV

**Name :** Enable/disable channels for multiplexing

**Description :** Enables/disables the individual channel

**Syntax :** \$AA5VV (cr)

\$ delimiter character.

AA address of setting module (00~FF)

5 the enable/disable channels command.

VV channels enable/disable, 00=disable,FF=enable

(cr) is the terminating character, carriage return (0Dh).

**Response :** !AA(cr) if the command is valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of response module(00~FF)

**Example :**

Command : \$0155A      Response : !01

Set address 01 enable channel 1,3,4,6 and disable channel 0,2,5,7,return success

Command : \$016      Receive : !015A

Read address 01 channel status , return channel 1,3,4,6 are enabled and channel 0,2,5,7 are disabled.

### 3.8 \$AA6

**Name :** Read Channel Status

**Description :** Asks a specified input module to return the status of all channels

**Syntax :** \$AA6 (cr)

\$ delimiter character.

AA address of reading module (00~FF)

6 represents the read channel statuses command.

(cr) is the terminating character, carriage return (0Dh).

**Response :** !AAVV(cr) if the command is valid.

?AA(cr)if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of response module (00~FF)

VV channels enable/disable, 00=disable,FF=enable

**Example :**

Command : \$015A5      Response : !01

Set address 01 to enable channel 0,2,5,7 and disable channel 1,3,4,6,return success.

Command : \$016      Receive : !01A5

Read address 01 channel status, return channel 0,2,5,7 are enabled and channel 1,3,4,6 are disabled.

### **3.9 #\*\***

**Name :** Synchronized Sampling command

**Description :** Orders all analog input modules to sample their input values and store the values in special registers.

**Syntax :** #\*\* (cr)

# delimiter character.

AA address of reading module (00~FF)

\*\* the Synchronized Sampling command

(cr) is the terminating character, carriage return (0Dh).

**Response :** No responses

### 3.10 \$AA4

**Name :** Read Synchronized Data command

**Description :** Returns the input value that was stored in the addressed module's register, after command #\*\* was issued.

**Syntax :** \$AA4 (cr)

\$ delimiter character.

AA address of reading module (00~FF)

4 Read synchronized data command.

(cr) is the terminating character, carriage return (0Dh).

**Response :** !AA(status)(data)(cr) if the command is valid.

?AA(cr)if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of response module (00~FF)

VV channels enable/disable, 00=disable,FF=enable

status If status = 1, then the data has been sent for the first time since #\*\* command was issued.

If status = 0, then the data has been sent at least once before.

data Synchronized data value

**Example :**

Command : \$014 Response : !011+3.5266

Read address 01 synchronized data value, return 3.5266,

Status=1, the data has been sent for the first time.

### 3.11 \$AAF

**Name :** Read Firmware Version

**Description :** The command requests the module at address AA to return the version code of its firmware.

**Syntax :** \$AAF (cr)

\$ delimiter character.

AA address of reading module (00~FF)

F identifies the version command.

(cr) is the terminating character, carriage return (ODh)

**Response :** !AA(data)(cr) if the command is valid.

?AA (cr) if an invalid command was issued.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of response module (00~FF)

Data is the version code of the module's firmware at address AA.

**Example :**

Command : \$01F Receive : !0120050412

Read address 01 firmware version , return version 20050412

Command : \$02F Receive : !0120040101

Read address 02 firmware version , return version 20040101

### 3.12 \$AA7CiRrR

**Name** : Channel Range Configuration command

**Syntax** : \$AA7CiRrr (cr)

\$ delimiter character.

AA address of setting module (00~FF)

7 channel range configuration command.

Ci the specified input channel you want to configure.

Rrr represents the type and range you want to set

(cr) the terminating character, carriage return (0Dh).

**Response** : !AA(cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

**Example** :

Command : \$017C5R21      Response : !01

The command configures the range of channel 5 in the module at address 01 as Pt100(IEC) 0 ~ 100

### 3.13 \$AA8Ci

**Name :** Read Channel Range Configuration command

**Syntax :** \$AA8Ci (cr)

\$ delimiter character.

AA address of reading module (00~FF)

8 read channel range configuration command.

Ci the specified input channel you want to read

(cr) the terminating character, carriage return (0Dh).

**Response :** !AACiRrr (cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

Ci the specified input channel you want to read.

Rrr represents the type and range you want to set

**Example :**

Command : \$018C5      Response : !01C5R21

The command read the range of channel 5 in the module at address 01. The response 'R21' means Pt100(IEC)0~100

### 3.14 \$AAxnnnn

**Name :** Watchdog Timer Setting command

**Description** This command set the communication watchdog timer (WDT) cycle time

**Syntax :** \$AAxnnnn (cr)

\$ delimiter character.

AA address of setting module (00~FF)

X the watchdog timer setting command

nnnn watchdog timer value 0000~9999

(cr) the terminating character, carriage return (0Dh).

**Response :** !AA (cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

**Example :**

Command : \$01X1234      Response : !01

The command set the WDT cycle as 1234 in the input module at address 01.

### 3.15 \$AAY

**Name :** Read Watchdog Timer Setting command

**Description :** This command read the setting of communication watchdog timer (WDT) cycle time

**Syntax :** \$AAY (cr)

\$ delimiter character.

AA address of reading module (00~FF)

Y the read watchdog timer setting command

(cr) the terminating character, carriage return (0Dh).

**Response :** !AAnnnn (cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

nnnn watchdog timer value 0000~9999

**Example :**

Command : \$01Y      Response : !011234

Read the WDT in the module at address 01,return 1234

### 3.16 \$AAB

**Name :** Channel Diagnose command

**Description :** Diagnose channel status in over range, under range, and wire opening.

**Syntax :** \$AAB (cr)

\$ delimiter character.

AA address of setting module (00~FF)

B the channel diagnose command

(cr) the terminating character, carriage return (0Dh).

**Response :** !AAN (cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

N Bit value 0 means normal status and bit value 1 means channel over range, under range, or open wiring.

### 3.17 \$AAS0

**Name :** Internal Calibration command

**Description :** This command executes internal self-calibration  
for offset and gain errors

**Syntax :** \$AAS0 (cr)

\$ delimiter character.

AA address of the module that is to be calibrated (00~FF)

S0 the internal calibration command

(cr) the terminating character, carriage return (0Dh).

**Response :** !AA (cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

**Example :**

Command : \$01S0      Response : !01

Calibrate address 01 offset value, and return success.

### 3.18 \$AAS1

**Name :** Reload Default Calibrating Parameter command

**Description :** Reload factory default calibrating parameter to  
overwrite current calibrating parameter.

**Syntax :** \$AAS1 (cr)

\$ delimiter character.

AA address of the module that is to be calibrated (00~FF)

S1 the reload calibrating parameter command

(cr) the terminating character, carriage return (0Dh).

**Response :** !AA (cr) if the command was valid.

?AA(cr) if an invalid operation was entered.

Syntax error or communication error may get no response.

! command is valid.

? command is invalid.

AA address of the module (00~FF)

**Example :**

Command : \$01S1      Response : !01

Address 01 reload default calibrating parameter:offset  
value, and return success.