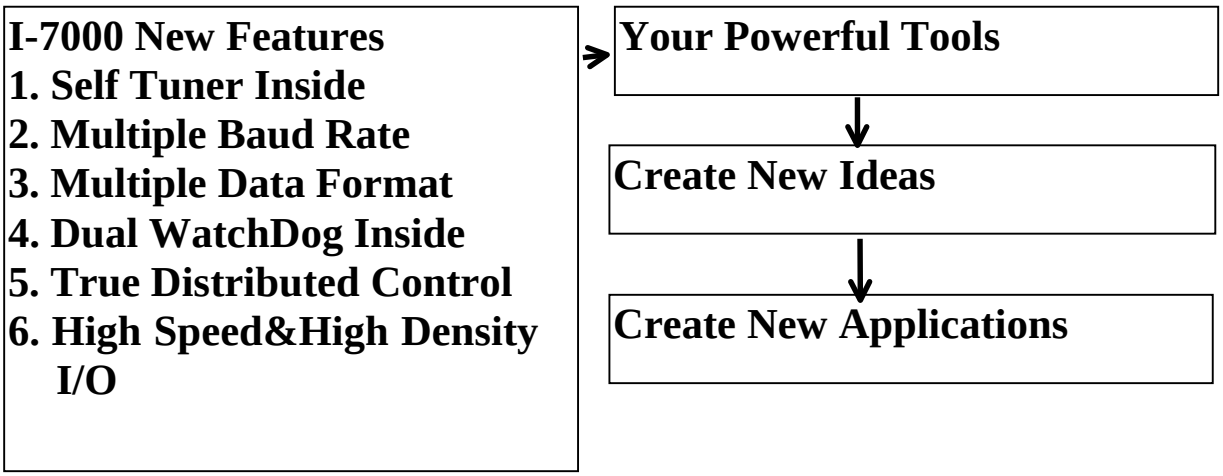
I-7021/I-7024

User Manual



Warranty

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

ICP DAS assume no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Trademark

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

License

The user can use, modify and backup this software on a single machine. The user may not reproduce, transfer or distribute this software, or any copy, in whole or in part.

Table of Contents

1. Introduction

I-7000 series are a family of remote controllable data acquisition modules. They provide A/D, D/A, DI/O, Timer/Counter and other functions. These modules can be remote controlled by a set of commands.

The I-7021 is a voltage or current analog output module. The special features of I-7021 are giving as following:

- one channel analog output.
- 12 bits D/A converter.
- Current read back from the external current loop path.
- Programmable slew rate control.
- 0-10V/0-20mA/4-20mA analog output programmable
- Power-on value and safe value programmable.

The I-7024 is a multi-channels voltage or current analog output module. The special features of I-7024 are giving as following:

- Four channels analog output.
- 14 bits D/A converter.
- Programmable slew rate control.
- 0 to 20mA, 4 to 20mA, 0 to 10V, -10 to 10V, 0 to 5V and –5 to 5V analog output programmable
- Power-on value and safe value programmable.

1.1 More Information

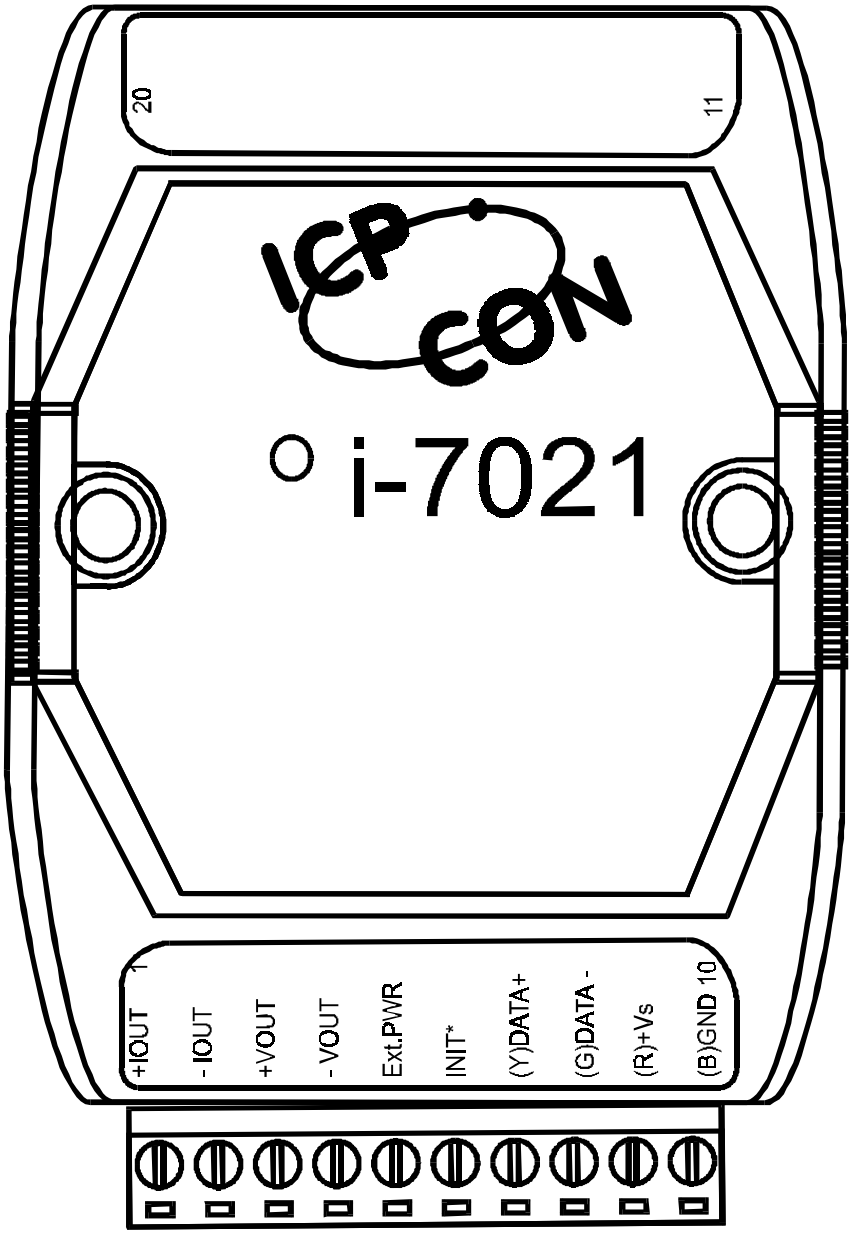
- Refer to chapter 1 of ***I-7000 Bus Converter User Manual*** for more information as following:

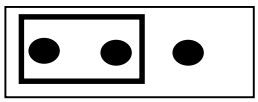
  1.1 I-7000 OVERVIEW
  1.2 I-7000 RELATED DOCUMENTATION
  1.3 I-7000 COMMON FEATURES
  1.4 I-7000 SYSTEM NETWORK CONFIGURATION
  1.5 I-7000 Dimension

- For DOS user, please refer to ***NAP7000S User Manual*** for the diagnostic program **TEST.EXE**.

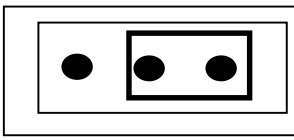
- For WIN 9X/NT user, please refer to on-line help of ***7000 Utility*** for diagnostic and configuration setting.

1.2 Pin Assignment

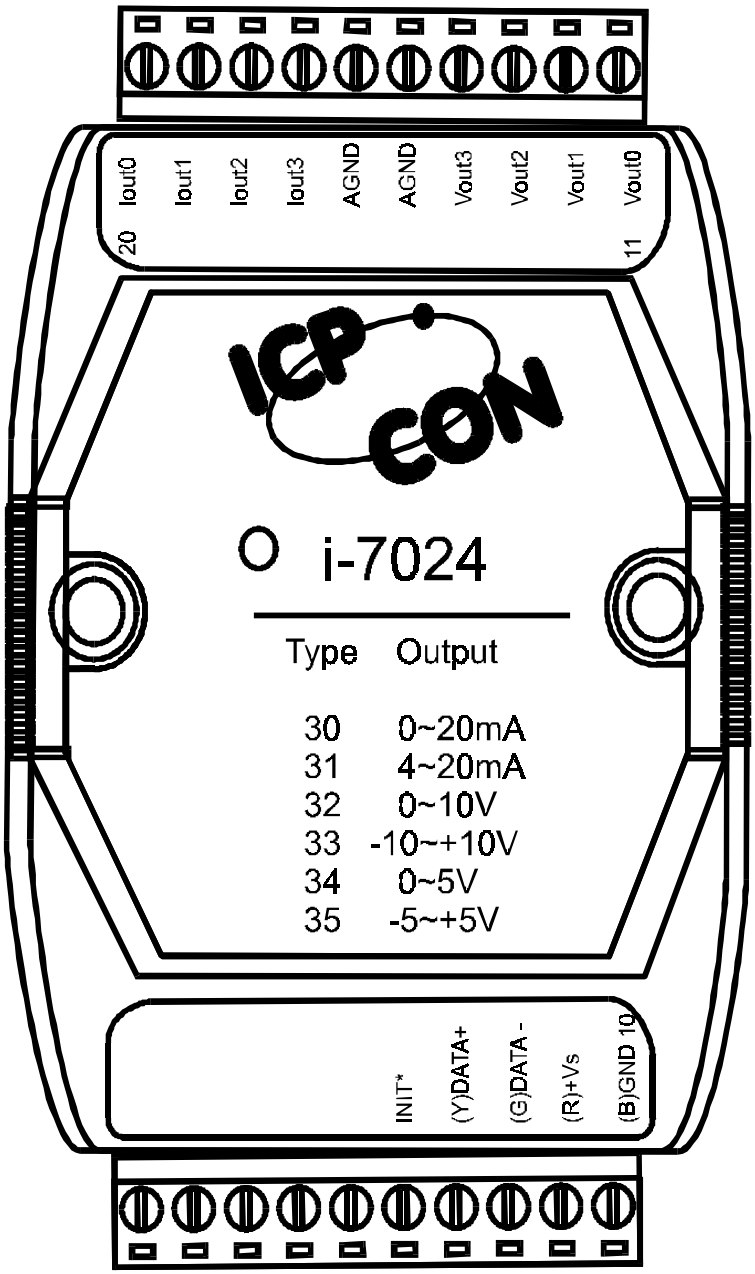




JP1 = internal power



JP1 = external power


Iout0
Iout1
Iout2
Iout3
AGND
AGND
Vout3
Vout2
Vout1
Vout0
20
11
i-7024
Type Output
30 0~20mA
31 4~20mA
32 0~10V
33 -10~+10V
34 0~5V
35 -5~+5V
INIT*
(Y)DATA+
(G)DATA -
(R)+Vs
(B)GND 10

1.3 Specifications

Analog output for I-7021

- Channels: 1
- Output type: mA, V
- Output range: 0 to 20mA , 4 to 20mA and 0 to 10V
- Accuracy: ±0.1% of FSR
- Read back accuracy: ±1% of FSR
- Resolution: ±0.02% of FSR
- Zero drift:
  Voltage output: ±30uV/°C
  Current output: ±0.2uA/°C
- Span temperature coefficient: ±25ppm/°C
- Programmable output slope:
  0.125 to 1024 mA/sec
  0.0625 to 512 V/sec
- Current load resistor:
  500Ω for internal power
  1050Ω for external 24V power

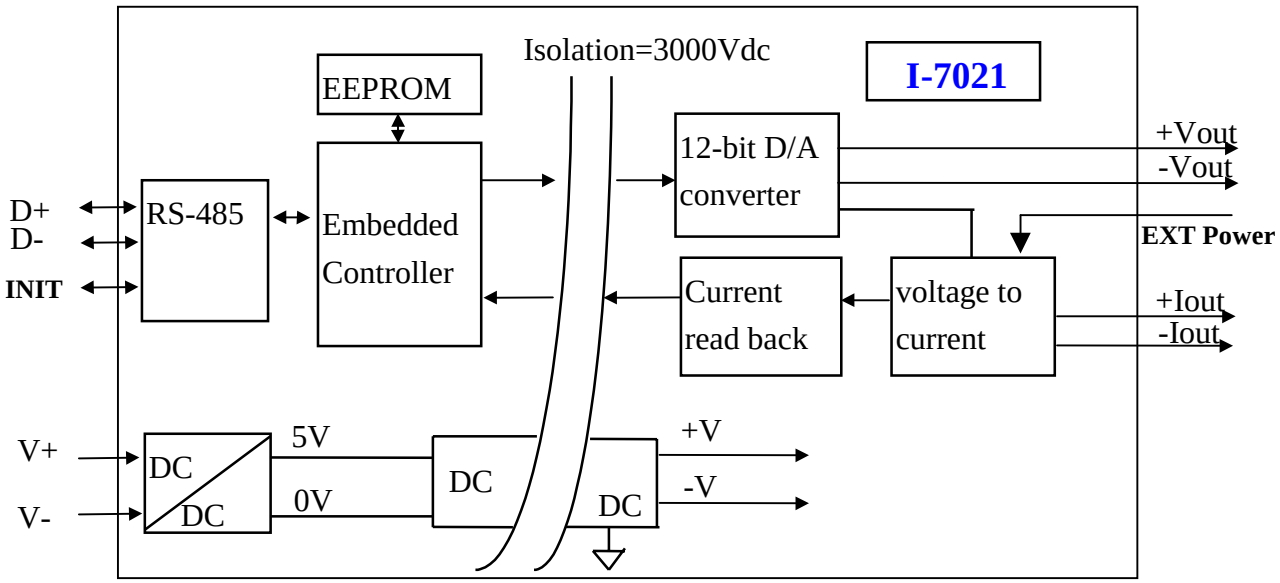
Analog output for I-7024

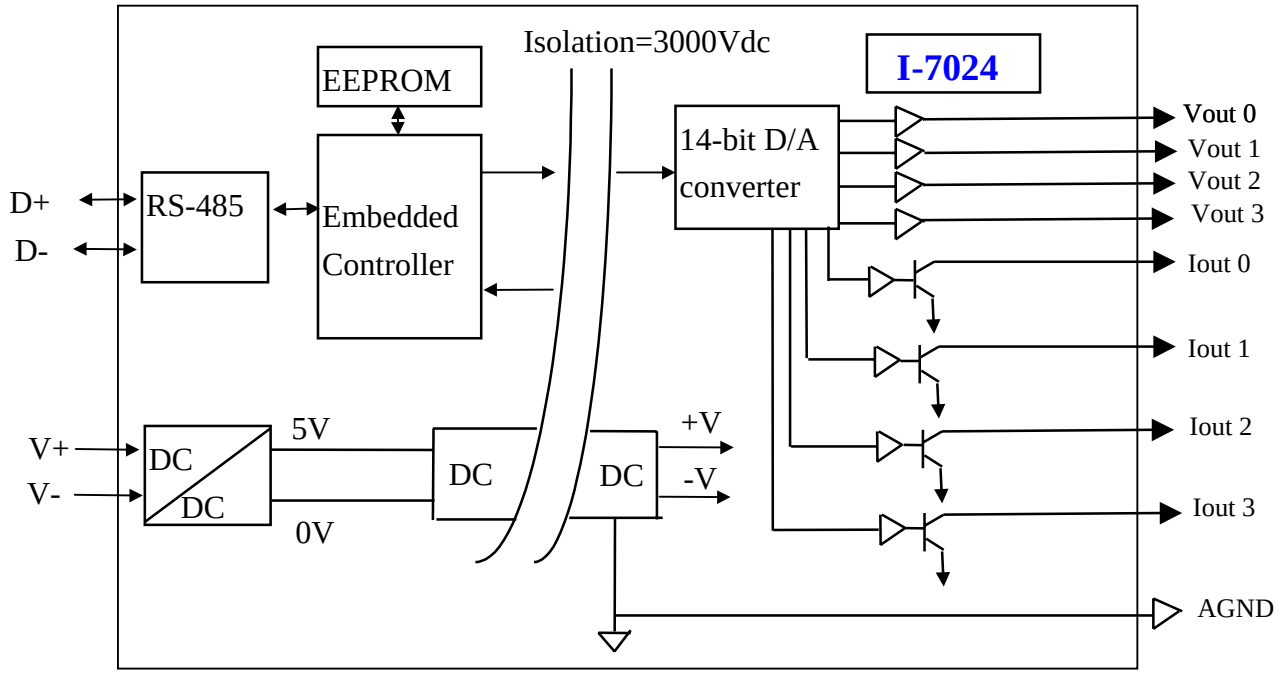
- Channels: 4
- Output type: mA, V
- Output range: 0 to 20mA , 4 to 20mA, 0 to 10V, -10 to 10V,
  0 to 5V, -5 to 5V.
- Accuracy: ±0.1% of FSR
- Resolution: ±0.02% of FSR
- Zero drift:
  Voltage output: ±30uV/°C
  Current output: ±0.2uA/°C
- Span temperature coefficient: ±25ppm/°C
- Programmable output slope:
  0.125 to 2048 mA/sec
  0.0625 to 1024 V/sec
- Current load resistor:
  1050Ω for external 24V power

Power (I-7021/I-7024)

- Power consumption : 2W

1.4 Block Diagram

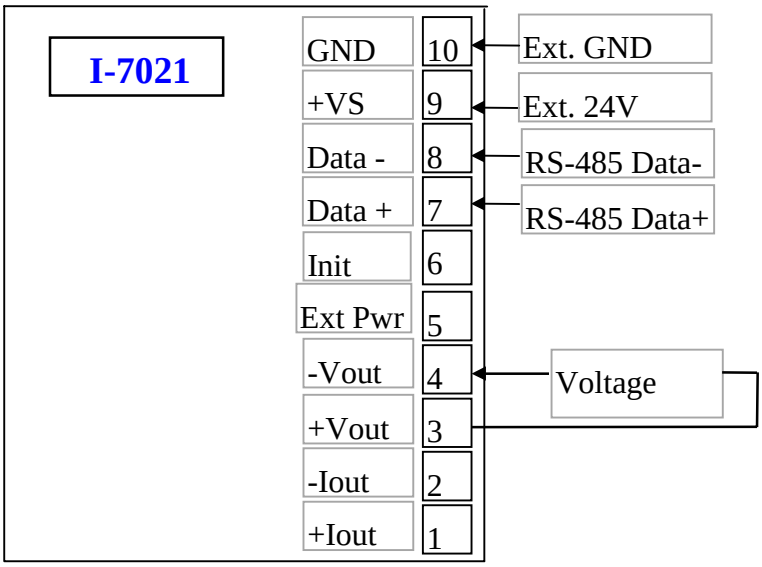




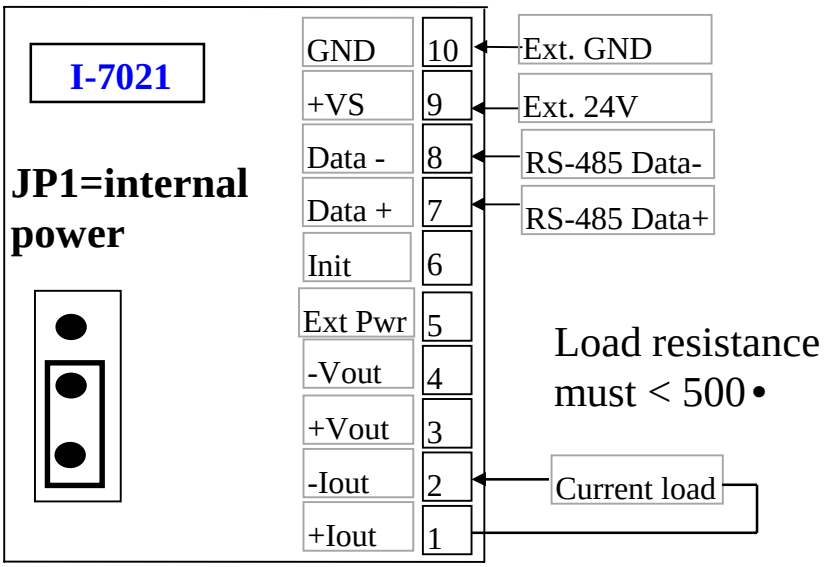
1.5 Application Wiring

1.5.1 Application Wiring for I-7021

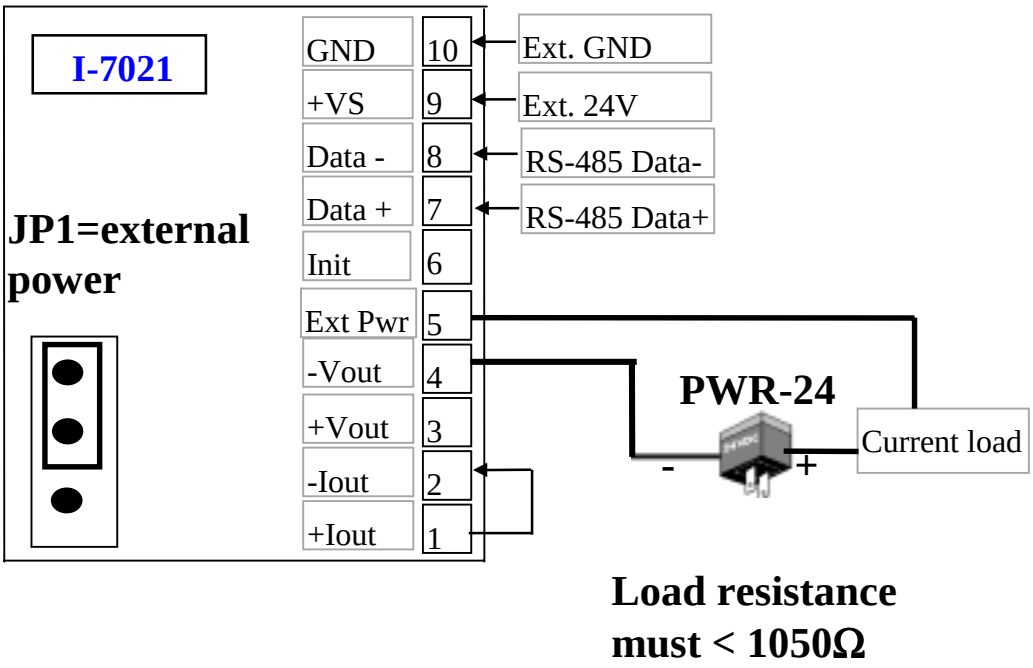
1.5.1.1 Voltage Output Diagram



1.5.1.2 Current Output Diagram (Internal power)

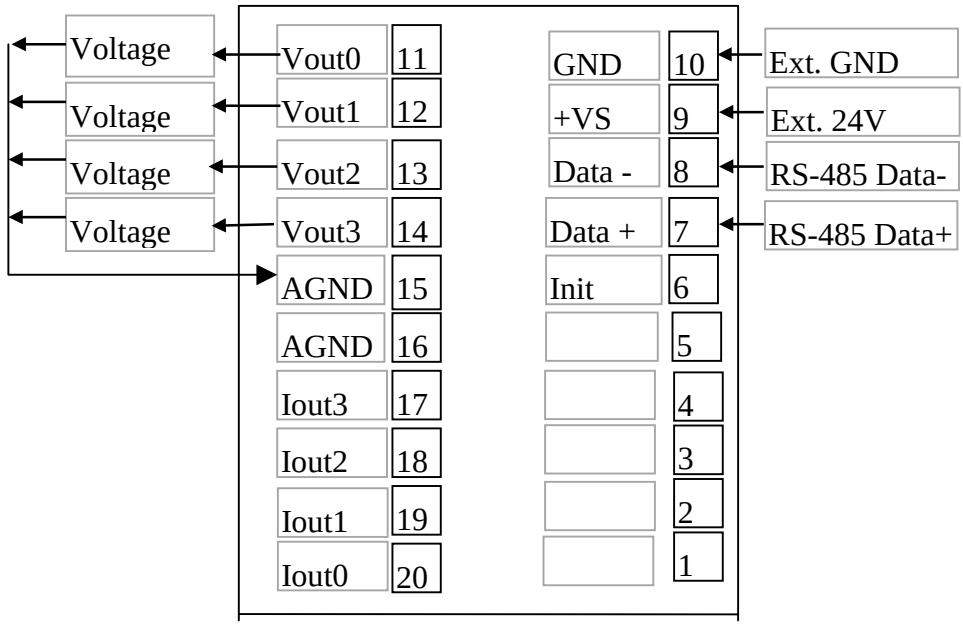


1.5.1.3 Current Output Diagram (External power)

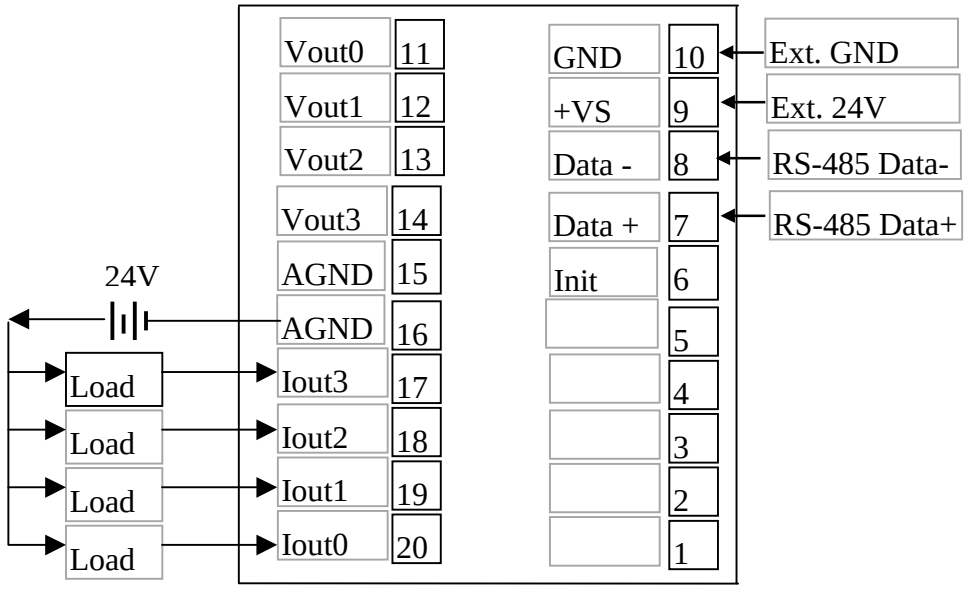


1.5.2 Application Wiring for I-7024

- **Voltage Output Diagram**



- **Current Output Diagram (External power)**



Note: Load Resistance must below 1K Ω

1.6 Quick Start

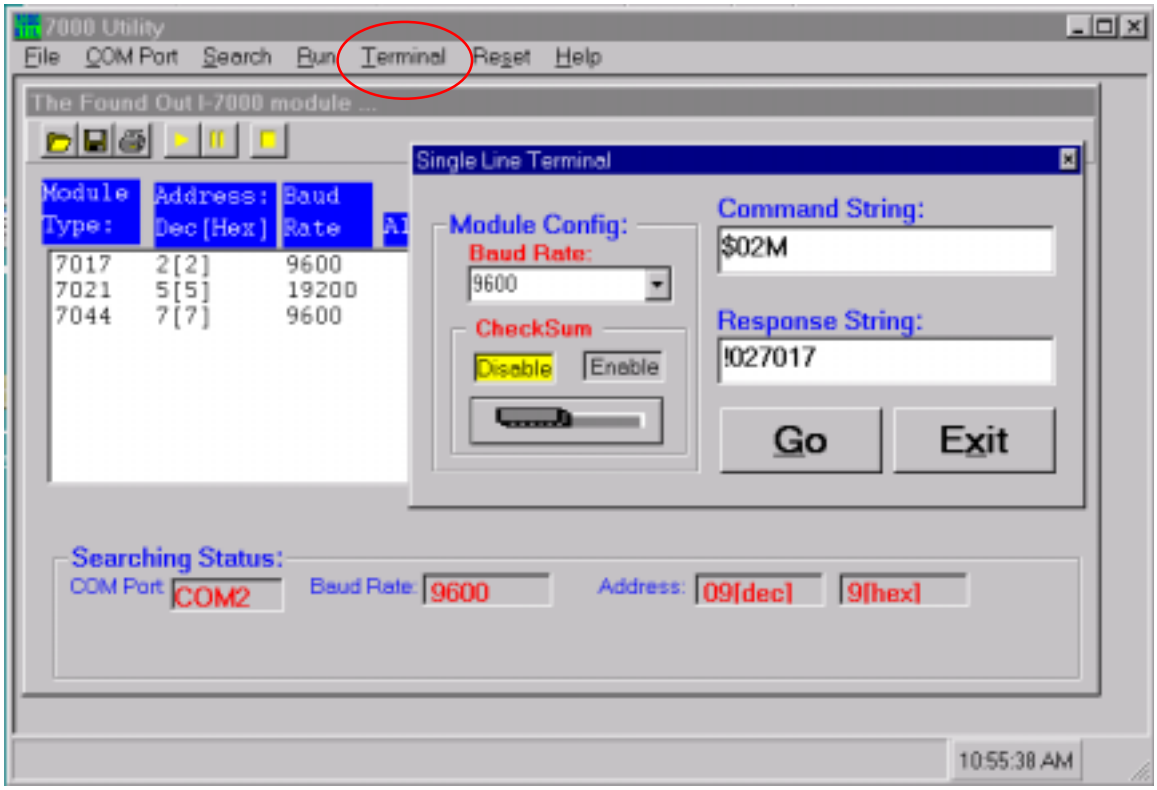
Refer to I-7000 Bus Converter User Manual. chapter-5 for the following functions:

- **module status unknown**(Sec. 5.1), **change address**(Sec. 5.2)
- **change baud rate**(Sec. 5.3), **checksum enable/disable**(Sec. 5.4)
- Diagnostic program

  - For DOS user, **TEST.EXE**.(refer to NAP7000S User Manual).

```
*************** Diagnostic Program ********************
* STATUS : COM=1,Baud_Rate= 9600,Checksum=DISABLE    *
*------------------------------------------------------*
*       0 : init                    (for all module) *
*       1 : search (1200 to 115200)(for all module) *
*       2 : send command            (for all module) *
*       3 : demo ($012,$01M,$01F)   (for all module) *
*       4 : Host Watchdog test      (for       7021) *
*       5 : RS-485 network testing (for all module) *
*       Q : quit                                     *
*************** Press Keyword ************************
```

- For WIN 9X/NT user, performing the ***Terminal/Signal Line*** menu in ***'7000 Utility'*** software(refer to on-line help of 7000 Utility).



1.6.1 Use I-7021 to output analog voltage

Refer to Sec 1.5.1 for wire connection, power on and run **test.exe** or ***7000 Utility*.**

1. press **2** **(NOTE: This step is not necessary for '7000 Utility' user)**
2. press **$012[Enter]** → Receive=**!01320600**
3. press **2**
4. press **#0105.000[Enter]** → Receive=>
5. press **2**
6. press **$016[Enter]** → Receive=**!0105.000**
7. press **2**
8. press **#0101.234[Enter]** → Receive=>
9. press **2**
10. press **$016[Enter]** → Receive=**!0101.234**
11. press **2**
12. press **$018[Enter]** → Receive=**!0100.000**

- step 2: the status of I-7021 is 0-10V voltage output, engineering unit format
- step 4: set voltage output=5.000 volt
- step 6: last command value read back = 05.000 ***(no measurement)***
- step 8: set voltage output=1.234 volt
- step 10: last command value read back = 1.234 ***(no measurement)***
- step 12: voltage output read back =00.000 **(The pin_1 & pin_2 is opened, so the voltage read back=0 volt, if the pin_1 & pin_2 is short together, the voltage output read back will be about 1.234 volt)**

1.6.2 Use I-7021 Current Output and Read Back

Refer to Sec. 1.5.1 for wire connection, power on and run **test.exe** or ***7000 Utility*.**

1. press **2**(NOTE: This step is not necessary for '7000 Utility' user)
2. press **%0101300600[Enter]** → Receive=**!01**
3. press **2**
4. press **#0110.000[Enter]** → Receive=>
5. press **2**
6. press **$016[Enter]** → Receive=**!0110.000**
7. press **2**
8. press **#0112.345[Enter]** → Receive=>
9. press **2**
10. press **$016[Enter]** → Receive=**!0112.345**
11. press **2**
12. press **$018[Enter]** → Receive=**!0112.34?**

- step 2: change the status of I-7021 to 0-20mA current output, engineering unit format
- step 4: set current output=10.000 mA
- step 6: last command value read back = 10.000 ***(no measurement)***
- step 8: set current output=12.345 mA
- step 10: last command value read back = 12.345 ***(no measurement)***
- step 12: current output read back =12.34? **(the current flow in the current output path, this value is measured by built-in A/D converter)**

1.6.3 Use I-7024 to output analog voltage

Refer to Sec 1.5.2 for wire connection, power on and run ***test.exe*** or ***7000 Utility*****.**

1. press **2**(NOTE: This step is not necessary for '7000 Utility' user)
2. press **$012[Enter]** → Receive=**!01320600**
3. press **2**
4. press **#010+05.000[Enter]** → Receive=>
5. press **2**
6. press **$0160[Enter]** → Receive=**!01+05.000**
7. press **2**
8. press **#011+01.234[Enter]** → Receive=>
9. press **2**
10. press **$0161[Enter]** → Receive=**!01+01.234**

- step 2: the status of I-7024 is 0-10V voltage output, engineering unit format
- step 4: set channel:0 voltage output=5.000 volt
- step 6: last command value read back = 05.000 ***(no measurement)***
- step 8: set channel:1 voltage output=1.234 volt
- step 10: last command value read back = 1.234 ***(no measurement)***

1.6.4 Current Output and Read Back

Refer to Sec. 1.5.2 for wire connection, power on and run **test.exe** or ***7000 Utility*.**

1. press **2**(NOTE: This step is not necessary for '7000 Utility' user)
2. press **%0101300600[Enter]** → Receive=**!01**
3. press **2**
4. press **#010+10.000[Enter]** → Receive=>
5. press **2**
6. press **$0160[Enter]** → Receive=**!01+10.000**
7. press **2**
8. press **#011+12.345[Enter]** → Receive=>
9. press **2**
10. press **$0161[Enter]** → Receive=**!01+12.345**

- step 2: change the status of I-7024 to 0-20mA current output, engineering unit format
- step 4: set channel:0 current output=10.000 mA
- step 6: last command value read back = 10.000 ***(no measurement)***
- step 8: set channel:1 current output=12.345 mA
- step 10: last command value read back = 12.345 ***(no measurement)***

1.7 Default Setting

The default setting for 7021

- address=01, baud rate=9600, checksum disable
- type=32, 0 to 10V voltage output, JP1=internal power
- data=1 start+8 data+1 stop(no parity)

The default setting for 7024

- address=01, baud rate=9600, checksum disable
- type=32, 0 to 10V voltage output for these 4 channels.
- data format = 1 start+8 data+1 stop(no parity)

1.8 Calibration

1.8.1 Calibration for 7021

There are three calibrations process listed as follows:

- 4-mA calibration(Sec. 1.8.1.1)
- 20-mA calibration(Sec. 1.8.1.2)
- 10V calibration(Sec. 1.8.1.3)

The total calibration steps are given as following:

Step 1: Wire connection. Power-on & warm-up about 30 minutes.

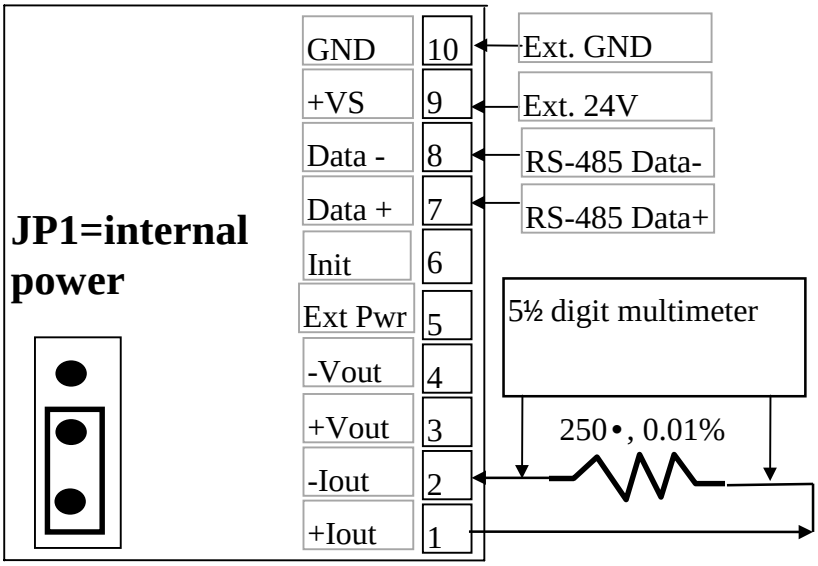
Step 2: Perform 4-mA calibration

Step 3: Perform 20-mA calibration

Step 4: Perform 10V calibration

1.8.1.1 4-mA calibration for I-7021

- **Wire connection for 4-mA calibration:**



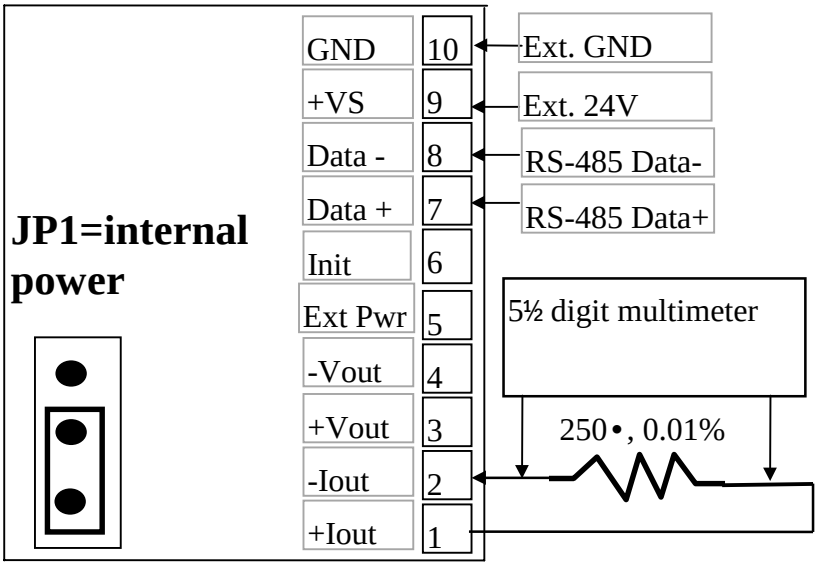
- **Command sequence for calibration:**

1. press **2** **(NOTE: This step is not necessary for '7000 Utility' user)**
2. press **%0101300600[Enter]** → Receive=**!01**
3. press **2**
4. press **#0104.000[Enter]** → Receive=>
5. repeat **6-7** until multimeter=1V (250*4=1000mV=1V)
6. press **2**
7. press **$013??[Enter]** **(NOTE)** → Receive=**!01**
8. press **2**
9. press **$010[Enter]** → Receive=!01

NOTE: for more detail information about this command, please refer to sec 2.9

1.8.1.2 20 mA calibration for 7021

- **Wire connection for 20-mA calibration:**



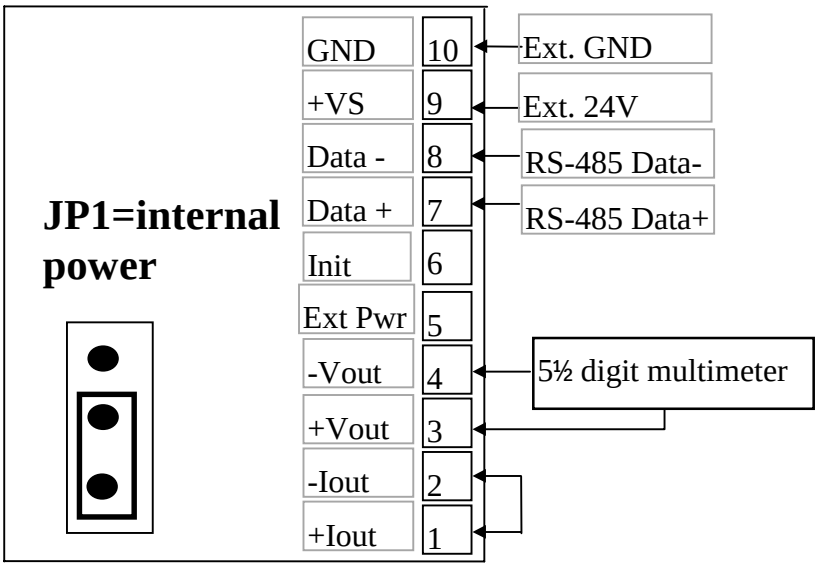
- **Command sequence for calibration:**

1. press **2** **(NOTE: This step is not necessary for '7000 Utility' user)**
2. press **%0101300600[Enter]** → Receive=**!01**
3. press **2**
4. press **#0120.000[Enter]** → Receive=>
5. repeat **6-7** until multimuter=5V(250*20=5000mV=5V)
6. press **2**
7. press **$013??[Enter]** **(NOTE)** → Receive=**!01**
8. press **2**
9. press **$011[Enter]** → Receive=**!01**

NOTE: for more detail information about this command, please refer to sec 2.9

1.8.1.3 10V calibration for 7021

- **Wire connection for 10V calibration:**



- **Command sequence for calibration:**

1. press **2** **(NOTE: This step is not necessary for '7000 Utility' user)**
2. press **%0101320600[Enter]** → Receive=**!01**
3. press **2**
4. press **#0110.000[Enter]** → Receive=>
5. repeat **6-7** until multimuter=10V
6. press **2**
7. press **$013??[Enter]** **(NOTE)** → Receive=**!01**
8. press **2**
9. press **$017[Enter]** → Receive=**!01**

NOTE: for more detail information about this command, please refer to sec 2.9

1.8.2 Calibration for 7024

There are four calibrations process listed as follows:

- 0-mA calibration(Sec. 1.8.2.1)
- 20-mA calibration(Sec. 1.8.2.2)
- -10V calibration(Sec. 1.8.2.3)
- 10V calibration(Sec. 1.8.2.4)

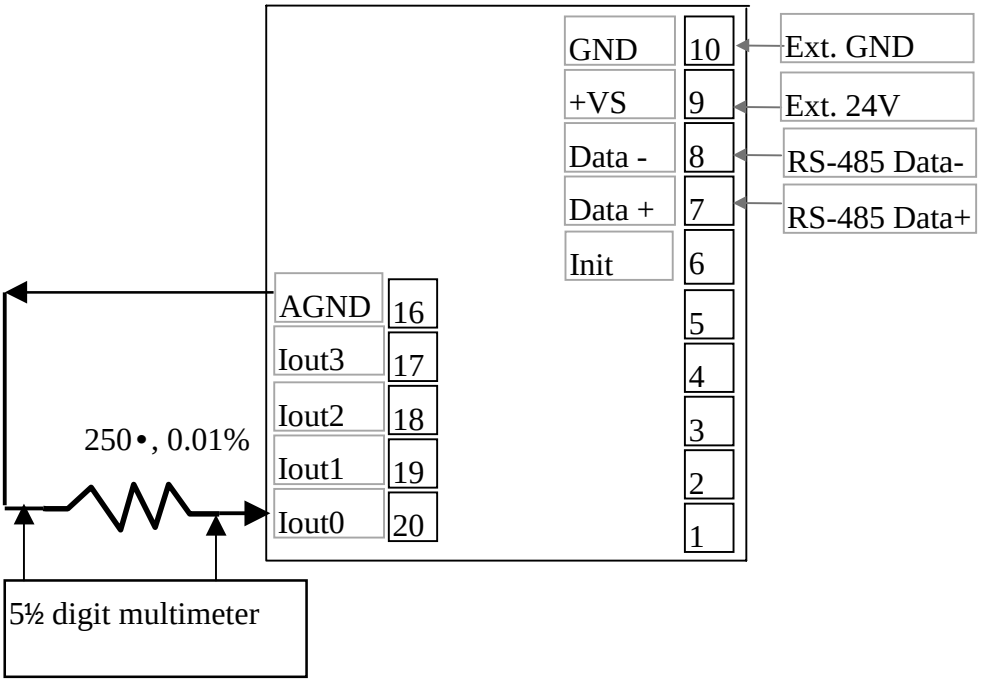
The total calibration steps are given as following:

Step 1: Wire connection. Power-on & warm-up about 30 minutes.
Step 2: Perform 0-mA calibration
Step 3: Perform 20-mA calibration
Step 4: Perform -10V calibration
Step 5: Perform 10V calibration

Note: Each channel has to calibrate seperately.

1.8.2.1 0-mA calibration for 7024

- **Wire connection for 0-mA calibration:(Ex. Ch:0)**



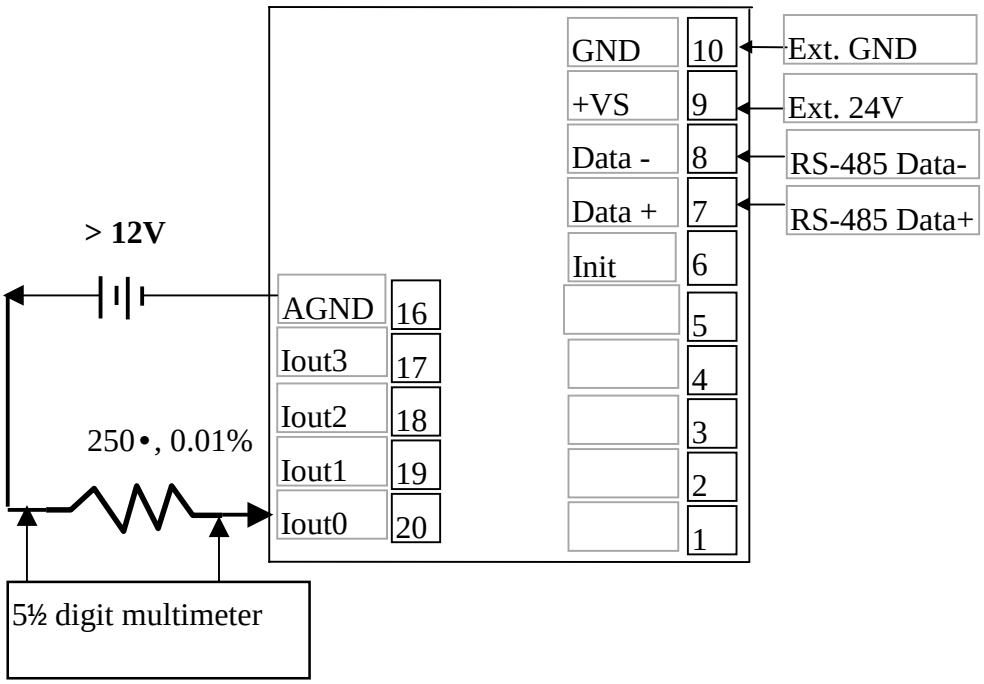
- **Command sequence for calibration:**

1. press **2** **(NOTE: This step is not necessary for '7000 Utility' user)**
2. press **%0101300600[Enter]** → Receive=**!01**
3. press **2**
4. press **#010+00.000[Enter]** → Receive=>
5. repeat **6-7** until multimuter=0V (250*0=0mV=0V)
6. press **2**
7. press **$0130??[Enter]** **(NOTE)** → Receive=**!01**
8. press **2**
9. press **$0100[Enter]** → Receive=**!01**

NOTE: for more detail information about this command, please refer to sec 2.10

1.8.2.2 20 mA calibration for 7024

- **Wire connection for 20-mA calibration: (Ex. Ch:0)**



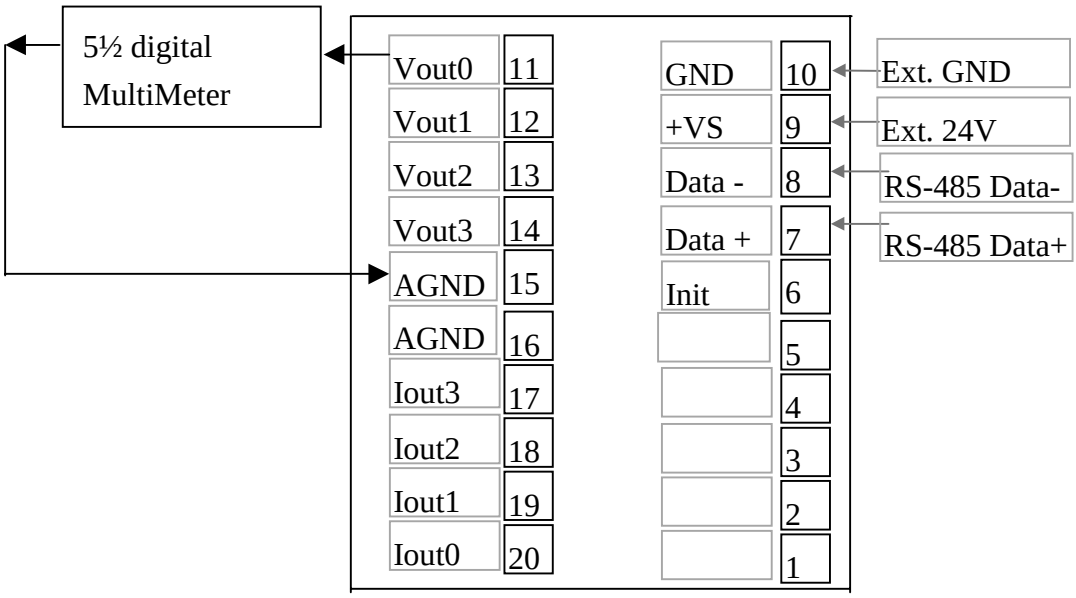
- **Command sequence for calibration:**

1. press **2**(NOTE: This step is not necessary for '7000 Utility' user)
2. press **%0101300600[Enter]** → Receive=**!01**
3. press **2**
4. press **#010+20.000[Enter]** → Receive=>
5. repeat **6-7** until multimuter=5V(250*20=5000mV=5V)
6. press **2**
7. press **$0130??[Enter]** (NOTE) → Receive=**!01**
8. press **2**
9. press **$0110[Enter]** → Receive=**!01**

NOTE: for more detail information about this command, please refer to sec 2.10

1.8.2.3 -10V calibration for 7024

- **Wire connection for -10V calibration:(Ex. Ch:0)**



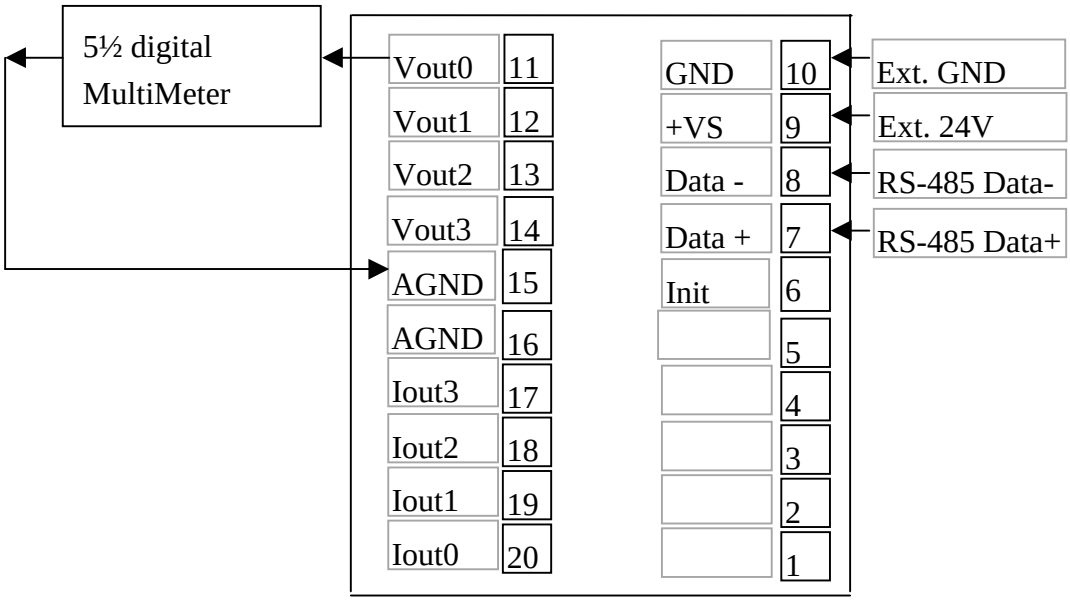
- **Command sequence for calibration:**

1. press **2**(NOTE: This step is not necessary for '7000 Utility' user)
2. press **%0101330600[Enter]** → Receive=**!01**
3. press **2**
4. press **#010-10.000[Enter]** → Receive=>
5. repeat **6-7** until multimuter=-10V
6. press **2**
7. press **$0130??[Enter]**(NOTE) → Receive=**!01**
8. press **2**
9. press **$0100[Enter]** → Receive=!01

NOTE: for more detail information about this command, please refer to sec 2.10

1.8.2.4 10V Calibration for 7024

- **Wire connection for 10V calibration: (Ex: Ch:0)**



- **Command sequence for calibration:**

1. press **2** **(NOTE: This step is not necessary for '7000 Utility' user)**
2. press **%0101330600[Enter]** → Receive=**!01**
3. press **2**
4. press **#010+10.000[Enter]** → Receive=>
5. repeat **6-7** until multimuter=10V
6. press **2**
7. press **$0130??[Enter]** **(NOTE)** → Receive=**!01**
8. press **2**
9. press **$0110[Enter]** → Receive=**!01**

NOTE: for more detail information about this command, please refer to sec 2.10

1.9 Tables

1.9.1 Configuration Table for I-7021

Configuration Code Table : TT , CC

TT	Output Range
30	0 to 20 mA
31	4 to 20 mA
32	0 to 10V

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

Configuration Code Table : FF

7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
set to 0	checksum 0=disable 1=enable	**Slew Rate Control** **code voltage current** 0000 : immediate change 0001 : 0.0625 V/sec 0.125 mA/sec 0010 : 0.125 V/sec 0.250 mA/sec 0011 : 0.250 V/sec 0.500 mA/sec 0100 : 0.500 V/sec 1.000 mA/sec 0101 : 1.000 V/sec 2.000 mA/sec 0110 : 2.000 V/sec 4.000 mA/sec 0111 : 4.000 V/sec 8.000 mA/sec 1000 : 8.000 V/sec 16.000 mA/sec 1001 :16.00 V/sec 32.00 mA/sec 1010 :32.00 V/sec 64.00 mA/sec 1011 :64.00 V/sec 128.00 mA/sec 1100 :128.0 V/sec 256.00 mA/sec 1101 :256.0 V/sec 512.00 mA/sec 1110 :512.0 V/sec 1024.0 mA/sec				00→engineering unit 01→% of FSR 10→hexadecimal	

Data Format Table

TT	Output Range	Format	MAX	MIN
30	0 to 20 mA	Engineering Unit	20.000	00.000
		% of FSR	+100.00	+000.00
		2's complement	FFF	000
31	4 to 20 mA	Engineering Unit	20.000	04.000
		% of FSR	+100.00	+000.00
		2's complement	FFF	000
32	0 to 10V	Engineering Unit	10.000	00.000
		% of FSR	+100.00	+000.00
		2's complement	FFF	000

1.9.2 Configuration Table for I-7024

Configuration Code Table : TT , CC

TT	Output Range
30	0 to 20 mA
31	4 to 20 mA
32	0 to 10 V
33	-10 to 10 V
34	0 to 5 V
35	-5 to 5V

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

Configuration Code Table : FF

7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
set to 0	checksum 0=disable 1=enable	**Slew Rate Control** **code voltage current** 0000 : immediate change 0001 : 0.0625 V/sec 0.125 mA/sec 0010 : 0.125 V/sec 0.250 mA/sec 0011 : 0.250 V/sec 0.500 mA/sec 0100 : 0.500 V/sec 1.000 mA/sec 0101 : 1.000 V/sec 2.000 mA/sec 0110 : 2.000 V/sec 4.000 mA/sec 0111 : 4.000 V/sec 8.000 mA/sec 1000 : 8.000 V/sec 16.000 mA/sec 1001 :16.00 V/sec 32.00 mA/sec 1010 :32.00 V/sec 64.00 mA/sec 1011 :64.00 V/sec 128.00 mA/sec 1100 :128.0 V/sec 256.00 mA/sec 1101 :256.0 V/sec 512.00 mA/sec 1110 :512.0 V/sec 1024.0 mA/sec 1111:1024.0V/sec 2048.0 mA/sec				00→engineering unit	

Data Format Table

TT	Output Range	Format	MAX	MIN
30	0 to 20 mA	Engineering Unit	+20.000	+00.000
31	4 to 20 mA	Engineering Unit	+20.000	+04.000
32	0 to 10 V	Engineering Unit	+10.000	+00.000
33	-10 to 10 V	Engineering Unit	+10.000	-10.000
34	0 to 5 V	Engineering Unit	+05.000	+00.000
35	-5 to 5 V	Engineering Unit	+05.000	-05.000

2. Command Set

Command Set Table for I-7021 / I-7024

Command	Response	Description	Reference
%AANNTTCCFF	!AA	Set the configuration of module	Sec. 2.1
#AA(data) (7021)	>	Analog output command	Sec. 2.2
#AAN(data) (7024)	>	Analog output of Channel N	Sec. 2.3
$AA0 (7021)	!AA	Perform 4-mA calibration	Sec. 2.4
$AA0N (7024)	!AA	Perform –10V/0mA calibration	Sec. 2.5
$AA1 (7021)	!AA	Perform 20-mA calibration	Sec. 2.6
$AA1N (7024)	!AA	Perform +10V/20mA calibration	Sec. 2.7
$AA2	!AATTCCFF	Read configuration of module	Sec. 2.8
$AA3VV (7021)	!AA	Trim calibration	Sec. 2.9
$AA3NVV (7024)	!AA	Trim calibration of Channel N	Sec. 2.10
$AA4 (7021)	!AA	Power-on value setting	Sec. 2.11
$AA4N (7024)	!AA	Channel N Power-on Output Configuration	Sec. 2.12
$AA5	!AAS	Reset status	Sec. 2.13
$AA6 (7021)	!AA	Last value read back	Sec. 2.14
$AA6N (7024)	!AA	Channel N Last value read back	Sec. 2.15
$AA7 (7021)	!AA	Perform +10V calibration	Sec. 2.16
$AA7N (7024)	!AA	Channel N Ponwe-on output readback	Sec. 2.17
$AA8 (7021)	!AA	Analog value read back	Sec. 2.18
$AA8N (7024)	!AA	Analog value read back for channel N	Sec. 2.19
$AAF	!AA(data)	Read the firmware version number	Sec. 2.20
$AAM	!AA(data)	Read the module name	Sec. 2.21

~**	No Response	HOST is OK	Sec. 2.22
~AA0	!AASS	Read Module Status	Sec. 2.23
~AA1	!AA	Reset Module Status	Sec. 2.24
~AA2	!AASTT	Read Host Watchdog Timer Value	Sec. 2.25
~AA3ETT	!AA	Enable Host Watchdog Timer	Sec. 2.26
~AA4 (7021)	!AA(data)	Read safe value	Sec. 2.27
~AA4N (7024)	!AA(data)	Read safe value of Channel N	Sec. 2.28
~AA5 (7021)	!AA	Set safe value	Sec. 2.29
~AA5N (7024)	!AA	Set safe value of Channel N	Sec. 2.30
~AAO(name)	!AA	Set module name	Sec. 2.31

2.1 %AANNTTCCFF

I-7021/I-7024

- **Description**: Set the configuration of module.

- **Syntax**: %AANNTTCCFF[chk](cr)
  % is a delimiter character
  AA=2-character module address, from 00 to FF(hex format)
  NN=new address, from 00 to FF(hex format)
  TT=Input range code, refer to Sec. 1.9
  CC=baud rate code, refer to Sec. 1.9
  FF=status code, refer to Sec. 1.9
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example for I-7024**:
  command: %0102300600(cr)
  response : !02(cr)
  address 01 is configured to a new address 02, 0-20 mA

  command: %0202320600(cr)
  response : !02(cr)
  change to 0-10V analog output

Refer to „I-7000 Bus Converter User Manual‟ chapter-5 for the following functions:

- **module status unknown**(Sec. 5.1), **change address**(Sec. 5.2)
- **change baud rate**(Sec. 5.3), **checksum enable/disable**(Sec. 5.4)

2.2 #AA(data)

I-7021

- **Description**: Send the analog output

- **Syntax**: #AA(data)[chk](cr)
  # is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  (data) = refer to Sec. 1.9.1 **data format table**
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → >[chk](cr)
  out of range → ?AA[chk](cr), D/A will go to the most close value
  command ignore → ![chk](cr)
  no response → syntax error or communication error or address error
  > is a delimiter character indicating a valid command
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

NOTE: If the module status=04 and the host watchdog fails, all D/A command will be ignored until module status is clear to 0 by ~AA1 command

- **Example**:

command: #0112.345(cr)
response : >

Current output=12.345 mA

command: #0210.000(cr)
response : >

Maybe 10.000 mA or 10.000 V depend on output type

command: #0330.000(cr)
response : ?AA

Out of range, the D/A will go to the most close value

command: #0401.234(cr)
response : !

The D/A command is ignored. The module status is 0x04.

2.3 #AAN(data)

I-7024

- **Description**: Send the analog output from Channel N

- **Syntax**: #AAN(data)[chk](cr)
  # is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  N=Channel No. (from 0 to 3)
  (data) = refer to Sec. 1.9.2 **data format table**
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → >[chk](cr)
  out of range → ?AA[chk](cr), D/A will go to the most close value
  command ignore → ![chk](cr)
  no response → syntax error or communication error or address error
  > is a delimiter character indicating a valid command
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

NOTE: If the module status=04 and the host watchdog fails, all D/A command will be ignored until module status is clear to 0 by ~AA1 command

- **Example**:

command: #010+12.345(cr)
response : >

Module address 1, Channel 0 Current output = 12.345 mA

command: #023-02.500(cr)
response : >

Module address 2, Channel 3 voltage output = -2.5 V

command: #020+30.000(cr)
response : ?AA

Out of range, the D/A will go to the most close value

command: #040+01.234(cr)
response : !

The D/A command is ignored. The module status is 0x04.

2.4 $AA0

I-7021

- **Description**: Perform 4-mA calibration. Refer to Sec. 1.8.1.1 for more information.

- **Syntax**: $AA0[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example:**
  command: $010(cr)
  response : !01(cr)

  address 01 perform 4 mA calibration

  command: $020(cr)
  response : !02(cr)

  address 02 perform 4 mA calibration

2.5 $AA0N

I-7024

- **Description**: Perform –10V/0mA calibration for channel N. Refer to Sec. 1.8.2.1 for more information.

- **Syntax**: $AA0N[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  N=Channel No. (from 0 to 3)
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example:**
  command: $0102(cr)
  response : !01(cr)

  Module address 01, Channel 2, perform –10V/0mA calibration

  command: $0201(cr)
  response : !02(cr)

  Module address 02, Channel 1, perform –10V/0mA calibration

2.6 $AA1

I-7021

- **Description**: Perform 20-mA calibration. Refer to Sec. 1.8.1.2 for more information.

- **Syntax**: $AA1[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:
  command: $011(cr)
  response : !01(cr)

  address 01 perform 20 mA calibration

  command: $021(cr)
  response : !02(cr)

  address 02 perform 20 mA calibration

2.7 $AA1N

I-7024

- **Description**: Perform +10V/20mA calibration for channel N. Refer to Sec. 1.8.2.2 for more information.

- **Syntax**: $AA1N[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  N=Channel No. (from 0 to 3)
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:
  command: $0112(cr)
  response : !01(cr)

  Module address 01, channel 2, perform +10V/20mA calibration

  command: $0210(cr)
  response : !02(cr)

  Module address 02, channel 0, perform +10V/20mA calibration

2.8 $AA2

I-7021 / I-7024

- **Description**: Read the configuration of module.
- **Syntax**: $AA2[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Response**: valid command → !AATTCCFF[chk](cr),
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  TT, CC, FF: refer to Sec. 1.9
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Example**:
  command: $012(cr)
  response : !01300600(cr)

  address 01, 0 to 20 mA, 9600 BPS, checksum disable, engineering unit

  command: $022(cr)
  response : !02320700(cr)

  address 02, 0 to 10V, 19200 BPS, checksum disable, engineering unit

NOTE: If the user use %AANNTTCCFF command to change module configuration, the new configuration code will be stored into EEPROM immediately. The configuration code includes module address, module type, baud rate code, checksum enable/disable code, calibration code, power-on value and safe value. **The EEPROM data of I-7000 can be read infinite times and can be written about 100,000 times max.** Therefore the user should not change configuration code often for testing.
The $AA2 command is used to read EEPROM data only, therefore the user can send this command to I-7000 module infinitely.

2.9 $AA3VV

I-7021

- **Description**: Trim the analog output for calibration. Refer to Sec. 1.8.1 for more information.

- **Syntax**: $AA3VV[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  VV=2-character HEX value, 1 count=4.88 uA or 2.44 mV
  00 to 5F : increase analog output 0 to 95 count
  A1 to FF : decrease analog output 95 to 1 count
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

command: $01302(cr)
response : !01(cr)

Increase analog output 2 count = 2*4.88 uA or 2*2.44 mV, depend on output type)

command: $023FE(cr)
response : !02(cr)

Decrease analog output 2 count = 2*4.88 uA or 2*2.44 mV, depend on output type)

2.10 $AA3NVV

I-7024

- **Description**: Trim the analog output for calibration for channel N. Refer to Sec. 1.8.2 for more information.

- **Syntax**: $AA3NVV[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  N=Channel No. (from 0 to 3)
  VV=2-character HEX value, 1 count=2.44 uA or 1.22 mV
  00 to 5F : increase analog output 0 to 95 count
  A1 to FF : decrease analog output 95 to 1 count
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

  command: $013202(cr)
  response : !01(cr)

  For channel 2, to increase analog output 2 count = 2*2.44 uA or 2*1.22 mV, depend on output type)

  command: $0231FE(cr)
  response : !02(cr)

  For channel 1, to decrease analog output 2 count = 2*2.44 uA or 2*1.22 mV, depend on output type)

2.11 $AA4

I-7021

- **Description**: Save the current output value as the power-on value when power on.

- **Syntax**: $AA4[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:
  command: #0212.345
  response : >(cr)

  Module 2 analog output as 12.345 mA

  command: $024(cr)
  response : !02(cr)

  To set the Start-Up analog output for module 2 as 12.345V

2.12 $AA4N

I-7024

- **Description**: Save the current output value as the power-on value for channel N

- **Syntax**: $AA4N[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  N=Channel No. (from 0 to 3)
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:
  command: #020-01.234
  response : >(cr)

  Channel 0 analog output –1.234V

  command: $0240(cr)
  response : !02(cr)

  To set the Start-Up analog output for channel 0 as –1.234V

2.13 $AA5

I-7021 / I-7024

- **Description**: Read back the reset status. This is the only command to detect the module watchdog failure. **If the module is down, the module watchdog circuit will reset this module. This reset will cause the analog output of module going to their power-on value. The power-on value may be different from those analog output before module reset. Therefore the user has to send output command again to module for keeping the same output state before and after module watchdog reset.** Refer to Sec. 3.6 for more information.

- **Syntax**: $AA5[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AAS[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  S = 0, it has not been reset since the last reset status read
  1, it has been reset since the last reset status read
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

Note: When first power-on, the user should read the module **once** and will find that the S=1. Then the user should read the module **continually** and find that the S=0. **If S is changed to 1, this module has been reset by module watchdog circuit at least once. And all output are going to their power-on value now.** Therefore the user has to send output command again to control all output values to the desire states. Refer to Sec. 3.5 for more information.

- **Example**:

command: $015(cr)
response : !011(cr)

It is first time power-on

command: $015(cr)
response : !010(cr)

It is normal

command: $015(cr)
response : !010(cr)

It is normal

command: $015(cr)
response : !011(cr)

This module is **reset** by module watchdog. Therefore all output are going to their **start-up values** now.

2.14 $AA6

I-7021

- **Description**: to read back the lastest output value **without A/D measurement.** Refer to Sec. 1.6.1 for more information.

- **Syntax**: $AA6[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response : syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  (data) = refer to Sec. 1.9.1 **data format table**
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:
  command: #0212.345(cr)
  response : >(cr)

  Module 2 analog output as 12.345 mA

  command: $026(cr)
  response : !0212.345(cr)

  Last output value for module 2 is 12.345V

2.15 $AA6N

I-7024

- **Description**: to read back the lastest output value for channel N **without A/D measurement.** Refer to Sec. 1.6.3 for more information.

- **Syntax**: $AA6N[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  N=Channel No. (from 0 to 3)
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response : syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  (data) = refer to Sec. 1.9.2 **data format table**
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

command: #010+12.345
response : >(cr)

The analog output for channel 0 is 12.345 mA

command: $0160(cr)
response : !010+12.345(cr)

Last value=12.345 mA

2.16 $AA7

I-7021

- **Description**: Perform +10 volt calibration. Refer to Sec 1.8.1.3 for more information.

- **Syntax**: $AA7[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:
  command: $017(cr)
  response : !01(cr)

  address 01 perform +10 volt calibration

  command: $027(cr)
  response : !02(cr)

  address 02 perform +10 volt calibration

2.17 $AA7N

I-7024

- **Description**: to read back the power-on output value of channel N.

- **Syntax**: $AA7N[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  N=Channel No. (from 0 to 3)
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

command: #020-01.234 response : >(cr)	Channel 0 analog output –1.234V
command: $0240(cr) response : !02(cr)	To set the Start-Up analog output for channel 0 as –1.234V
command: #020-03.456 response : >(cr)	Channel 0 analog output –3.456V
command: $0270 response : !02-01.234(cr)	The read back Start-Up value of channel 0 is –1.234V
command: $0260 response : !02-03.456(cr)	The last output value of channel 0 is –3.456V

2.18 $AA8

I-7021

- **Description**: Read back the analog output value through the current path. This command can read back the voltage or current output depended on the output type. If the user use 0-10V range, the pin_1 and pin_2 must be short together. (Refer to Sec. 1.6.1 & Sec. 1.6.2 for more information)

- **Syntax**: $AA8[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  (data) = refer to Sec. 1.9.1 **data format table**
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

command: $018(cr)
response : !0112.345(cr)

Current through path = 12.345 mA

command: $028(cr)
response : !0210.000(cr)

Current through path=10.000 mA

command: $032(cr)
response : !03320600(cr)
command: $038
response : !0301.234

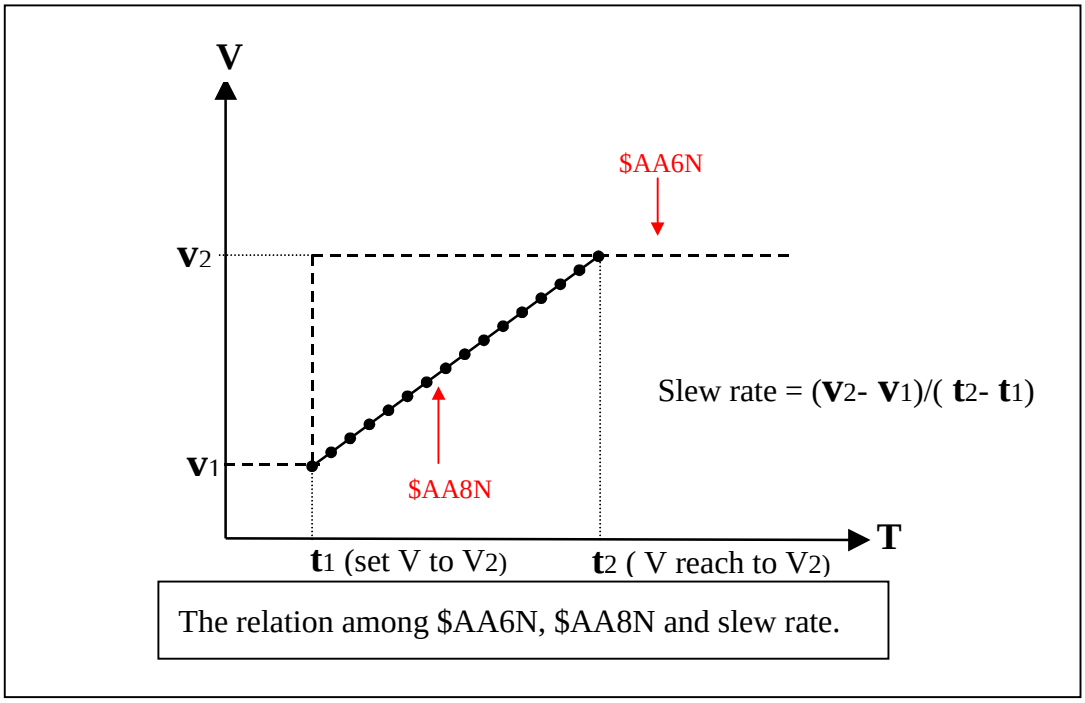
- 0-10V range
- Pin_1 & pin_2 short together
- voltage read back=1.234V

2.19 $AA8N

- **Description**: When sending a command to assign the analog output value for a specific channel of 7024. The analog output is updated gradually at the specific slew rate until the desired output value is reached. This command can read the analog value during updating process.

- **Syntax**: $AA8[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  (data) = refer to Sec. 1.9.2 **data format table**
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D



The relation among $AA6N, $AA8N and slew rate.

- **Example**:

command: $012(cr) response : !0132060C(cr)	The configuration for this 7024 as follows: Output range: 0 to 10V, slew rate: 0.25V/sec Checksum: Disable
command: #010+01.000(cr) response : >(cr)	Set channel:0 output value to 1.000 V
command: #010+09.800(cr) response : >(cr)	Set channel:0 output value to 9.800 V
:	
:	
command: $0180(cr) response : !01+01.372(cr)	The reading back value is 1.372V
:	
command: $0180(cr) response : !01+04.821(cr)	The reading back value is 4.821V
:	
command: $0180(cr) response : !01+06.772(cr)	The reading back value is 6.772V
:	
command: $0180(cr) response : !01+08.291(cr)	The reading back value is 8.291V
:	
command: $0180(cr) response : !01+09.800(cr)	The reading back value is 9.800V
:	
command: $0180(cr) response : !01+09.800(cr)	The reading back value is 9.800V

2.20 $AAF

I-7021 / I-7024

- **Description**: Read the version number of firmware.

- **Syntax**: $AAF[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response or address error → syntax error or communication error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  data=5-character for version number
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

  command: $01F(cr)
  response : !01A1.0(cr)

  module 01 version A1.0

  command: $02F(cr)
  response : !02A1.1(cr)

  module 02 version A1.1

2.21 $AAM

I-7021 / I-7024

- **Description**: Read the module name.

- **Syntax**: $AAM[chk](cr)
  $ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  data=4-character for module name
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

  command: $01M(cr)
  response : !017024(cr)

  Module name of 01 is 7024

  command: $02M(cr)
  response : !027024(cr)

  Module name of 02 is 7024

2.22 ~**

- **Description**: Host send this command to tell all modules that the Host is OK. Refer to Sec. 3.5 for more information.

- **Syntax**: ~**[chk](cr)
  ~ is a delimiter character
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: no response

- **Example**:
  command: ~**(cr)
  response : No Response

2.23 ~AA0

I-7021 / I-7024

- **Description**: Read the module status about the host watchdog. The module status will be latch until ~AA1 command is sent. **If the module status is equal to 0x04, it means the host watchdog timeout, then all output command will be ignored. And the output would be the safe-value that pre-defined by the command ~AA5(for 7021) or ~AA5N(for 7024).**

- **Syntax**: ~AA0[chk](cr)
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AASS[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  SS=2-character HEX status value as following:
  Bit_0, Bit_1 = reserved
  Bit_2 = 0 → OK,
  1 → host watchdog failure
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

command: ~010(cr)
response : !0100(cr)

Status of module 01 is OK

command: ~020(cr)
response : !0204(cr)

Status of module 02 is . host watchdog failure. → HOST is down now

2.24 ~AA1

I-7021 / I-7024

- **Description**: Reset the module status. The module status will be latched until ~AA1 command is sent. **If the module status=4, all output command will be ignored.** Therefore the user should read the module status first to make sure that the module status is 0. If the module status is not 0, only ~AA1 command can clear the module status. Refer to Sec., 3.2 and 3.3 for more information.
- **Syntax**: ~AA1[chk](cr)
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response or address error → syntax error or communication error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

command: ~010(cr)
response : !0104(cr)
Module's host watchdog status is 0x04 → host is down

command: #010+05.000(cr)
response : !(cr)
Output command is ignored

command: ~011(cr)
response : !01(cr)
clear module status

command: ~010(cr)
response : !0100(cr)
module status=0x00

command: #010+05.000(cr)
response : >(cr)
Output command is OK

2.25 ~AA2

I-7021 / I-7024

- **Description**: Read the status and timer value of host watchdog. The host watchdog timer is designed for host watchdog. When the host watchdog is enable, the host must send ~** command to all modules before the timer is up. When the ~** command is received, the host watchdog timer is reset and restart. Use ~AA3ETT to enable/disable/setting the host watchdog timer. Refer to Sec. 3.5 for more information.

- **Syntax**: ~AA2[chk](cr)
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AASTT[chk](cr)
  invalid command → ?AA[chk](cr)
  no response or address error → syntax error or communication error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  S=0: host watchdog is disable
  S=1: host watchdog is enable
  TT=2-character HEX value, from 00 to FF, unit=0.1 second
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

  command: ~012(cr)
  response : !01000(cr)

  Host watchdog timer of module 01 is disable

  command: ~022(cr)
  response : !0210A(cr)

  Host watchdog timer of module 02 is enable and =0.1*10 =1 second.

2.26 ~AA3ETT

I-7021 / I-7024

- **Description**: Enable/disable the timer value of host watchdog. The host watchdog timer is designed for software host watchdog. When the software host watchdog is enable, the host must send ~** command to all modules before the timer is expired. When the ~** command is received, the host watchdog timer is reset and restart. Use ~AA2 to read the host watchdog enable status & value. Refer to Sec. 3.5 for more information.

- **Syntax**: ~AA3ETT[chk](cr)
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  E=0 is disable and 1 is enable
  TT=2-character HEX value, from 00 to FF, unit=0.1 second
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response or address error → syntax error or communication error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:
  command: ~013000(cr)
  response : !01(cr)

  disable host watchdog timer of module 01

  command: ~02310A(cr)
  response : !02(cr)

  host watchdog timer of module 02 is enable and equal to 0.1*10 =1 second.

2.27 ~AA4

I-7021

- **Description**: Read the safe value.
  - When the module is **first power-on**, all output channels will go to their **power-on value**.
  - When the module is **down**, the module watchdog will reset the module and all output channels will **go to their power-on value**. **These power-on value may be different to old value before the module is reset.** Therefore the user must send out a new output command to control all output to the desire states.
  - When the host watchdog is enable and the **host is down**, all output will go to their **safe values** and module host status will change to 0x04. **If the module statue is 0x04, all output command will be ignored.** The module host status can be clear by ~AA1 command. Therefore the user must send ~AA1 command first, then send out a new output command to control all output to the desire states. Refer to Sec. 3.2 for more information.
- **Syntax**: ~AA4[chk](cr) → read safe value
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response or address error → syntax error or communication error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  (data): refer to Sec. 1.9
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Example**:
  command: ~014(cr)
  response : !0105.000(cr)

  Safe value = 5.0 volt

  command: ~024(cr)
  response : !0200.000(cr)

  Safe value = 0 volt

2.28 ~AA4N

I-7024

- **Description**: Read the safe value of channel N.
  - When the module is **power on firstly**, all output channels will go to their **power-on value**.
  - When the module is **down**, the module watchdog will reset the module and all output channels will **go to their power-on value**. **These power-on values may be different to previous analog output value before the module is reset.** Therefore the user must send out a new output command to control all output to the desire states.
  - When the host watchdog is enable and the **host is down**, all output will go to their **safe values** and module host status will change to 0x04. **If the module host statue is 0x04, all output command will be ignored.** The module host status can be clear by ~AA1 command. Therefore the user must send ~AA1 command firstly, then send out a new output command to control all output to the desire states. Refer to Sec. 3.2 for more information.
- **Syntax**: ~AA4N[chk](cr)
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  N=the channel no.(from 0 to 3)
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Response**: valid command → !AA(data)[chk](cr)
  invalid command → ?AA[chk](cr)
  no response or address error → syntax error or communication error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  (data): refer to Sec. 1.9
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Example**:

command: ~0140(cr)
response : !01+05.000(cr)

The safe vlaue of channel 0 is 5.000V

command: ~0141(cr)
response : !01+01.234(cr)

The safe value of channel 1 is 1.234V

2.29 ~AA5

I-7021

- **Description**: Set current value of D/A output as safe value.
  - When the module **power on firstly**, all output channels will go to their **power-on value**.
  - When the module is **down**, the module watchdog will reset the module and all output channels will **go to their power-on value**. **These power-on value may be different to old value before the module is reset.** Therefore the user must send out a new output command to control all output to the desire states.
  - When the host watchdog is enable and the **host is down**, all output will go to their **safe values** and module host status will change to 0x04. **If the module host statue is 0x04, all output command will be ignored.** The module host status can be clear by ~AA1 command. Therefore the user must send ~AA1 command first, then send out a new output command to control all output to the desire states. Refer to Sec. 3.2 for more information.

- **Syntax**: ~AA5[chk](cr) → set safe value
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Example**:
  command: #0100.000(cr)
  response : >(cr)
  command: ~015(cr)
  response : !01(cr)

Safe value is 0 volt

2.30 ~AA5N

I-7024

- **Description**: Set current value of D/A output as safe value for channel N.
  - When the module **power on firstly**, all output channels will go to their **power-on value**.
  - When the module is **down**, the module watchdog will reset the module and all output channels will **go to their power-on value**. **These power-on values may be different to previous output value before the module is reset.** Therefore the user must send out a new output command to control all output to the desire states.
  - When the host watchdog is enable and the **host is down**, all output will go to their **safe values** and module host status will change to 0x04. **If the module host statue is 0x04, all output command will be ignored.** The module host status can be clear by ~AA1 command. Therefore the user must send ~AA1 command first, then send out a new output command to control all output to the desire states. Refer to Sec. 3.2 for more information.

- **Syntax**: ~AA5N[chk](cr)
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D
- **Example**:

command: #010+12.345(cr)	To set channel:0 ‘s analog output to 12.345mA
response : >(cr)	
command: ~0150(cr)	To set safe value of channel:0 to 12.345mA
response : !01(cr)	

2.31 ~AAO(name)

I-7021 / I-7024

- **Description**: Set module name.

- **Syntax**: ~AAO(name)[chk](cr)
  ~ is a delimiter character
  AA=2-character HEX module address, from 00 to FF
  (name)=4-character/5-character module name
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Response**: valid command → !AA[chk](cr)
  invalid command → ?AA[chk](cr)
  no response → syntax error or communication error or address error
  ! is a delimiter character indicating a valid command
  ? is a delimiter character indicating a invalid command
  AA=2-character HEX module address
  [chk]=2-character checksum, if checksum disable → no [chk]
  (cr)=0x0D

- **Example**:

  command: $01M(cr)
  response : !017024(cr)
  command: ~01O8024(cr)
  response : !01(cr)

  Change module name from 7024 to 8024

  command: $01M(cr)
  response : !017080D(cr)
  command: ~01O8080D(cr)
  response : !01(cr)

  Change module name from 7080D to 8080D

Note: This command is designed for OEM/ODM user. The user can use it to change the module name for other purpose.

3. Operations Principle & Application Notes

3.1 INIT*_pin Operation Principle

All I-7000 modules contain an EEPROM to store configuration information. Therefore the user is difficult to find out the status of the I-7000 modules. The user can connect the INIT*_pin to GND_pin and power on the module. The I-7000 modules will **go to the factory default setting without change the EEPROM data.** The factory default setting is given as following:

address = 00
baud rate = 9600
checksum = DISABLE
data format = 1 start + 8 data bits + 1 stop bit

If the user disconnect the INIT*_pin and GND_pin, the I_7000 module will be auto configured according to the EEPROM data. The user is easy to find the EEPROM configuration data in the default setting. The steps are shown as following:

Step 1 : power off and connect INIT*_pin to GND_pin
Step 2 : power on
Step 3 : send command string **$002[0x0D]**
Step 4 : record the status of this I-7000 module
Step 5 : power off and disconnect INIT*_pin and GND_pin
Step 6 : power on

Refer to . I-7000 Bus Converter User Manual. Sec. 5.1 for more information.

3.2 D/A Operation Principle

(1) The D/A output of each channel will **go to their power-on value when power on firstly.**

(2) Every channel's D/A output will change to desire value if the #AA(data) (for 7021) or #AAN(data)(for 7024) command is received. Then the channel's D/A will keep in the same value until next #AA(data) (for 7021) or #AAN(data)(for 7024) command is send.

(3) If the I-7000 module reset by the hardware watchdog, **all D/A will go to their power-on value immediately.** The host computer can use $AA5 command to detect this condition. If the host computer send command #AA(data) (for 7021) or #AAN(data)(for 7024) to those modules now, those modules will change their D/A without any warning information. **Therefore it is recommended to use $AA5 command to detect the I-7000 status.**

(4) If the host watchdog failure, all the D/A will **go to their safe-value immediately and the module host status is set to 04.** If the host computer send out #AA(data) (for 7021) or #AAN(data)(for 7024) to those modules now, those modules will **ignore this command and return** . **!**. **as warning information.** The host **can use** . **~AA1**. **command to clear the module host status to 00,** then the I-7000 will accept the #AA(data) (for 7021) or #AAN(data)(for 7024) again.

3.3 I-7000 Module Status

(1) when module watchdog reset
- all D/A go to their power-on value
- **module status no change**
- accept host D/A command to change D/A state

(2) when host watchdog failure
- all D/A go to their safe-value
- **module host watch status=04 → host watchdog fails**
- **ignore** all host D/A command until module status is clear to 0 by ~AA1 command

3.4 Dual WatchDog Operation Principle

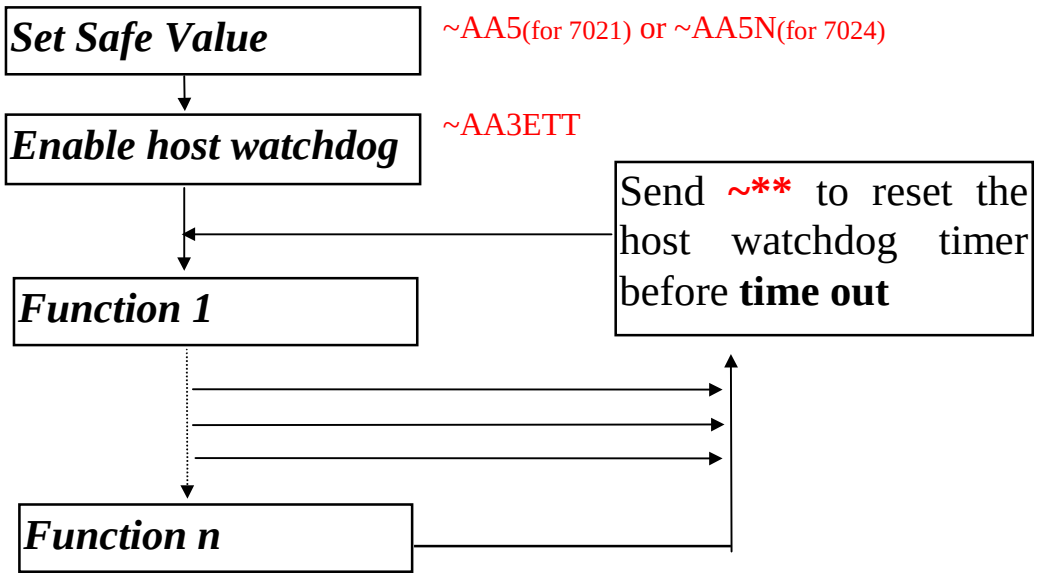
Dual watchdog = host watchdog + module watchdog
The host watchdog is a software watchdog.
The module watchdog is a hardware watchdog.

The I-7000 series are designed for industry applications, therefore they can work in the harsh envioronment. There are many couple noise or energy transient in such environment. The modules may be down if these noise is really too large. **The module watchdog can reset the module if it is down for too large signal.** Sometimes even the host may be down for hardware or software reasons. The host watchdog can monitor the status of host. **If the host is down, all the output of I-7000 modules will go to their pre-defined safe states for safety protection.**

If the RS-485 **network is open,** all the host command can not send to remote modules. This is very dangerous in real world application. The I-7000 output module will force their output going to their pre-defined safe state for safety consideration if the host watchdog is enable. **This dual watchdog feature will increase the system reliability very much.**

3.5 Host WatchDog Applications Notes

The host watchdog is designed to monitor the host computer. If the host computer fails, the output of the I-7000 modules will automatically go to their safe states to avoid unpredictable damaged. The flow chart for the host computer is given as following.



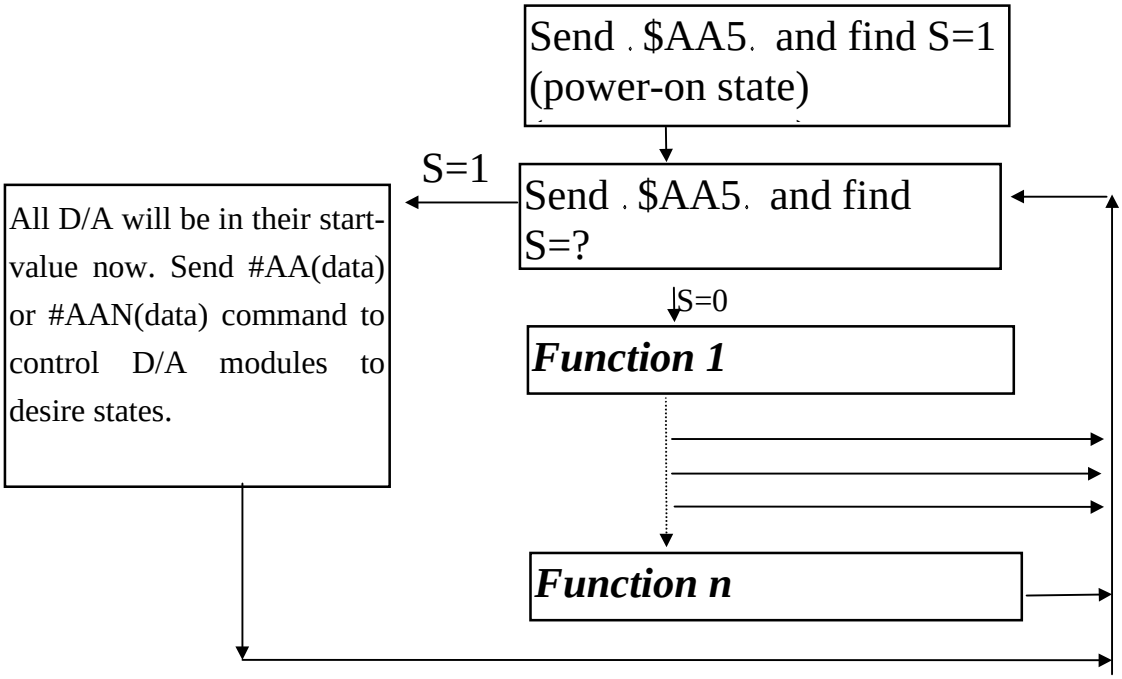
1. Refer to Sec. 1.5 for wire connection. Power on and run **test.exe**
2. press **2**
3. press **$012Enter]** → Receive=**!01320600**
4. press **2**
5. press **#010+00.000[Enter]**→ Receive = >
6. press **2**
7. press ~**0150[Enter]** → Receive=**!01**
8. press **2**
9. press **#010+05.000[Enter]**→ Receive=>
10. press **2**
11. press **~01311E[Enter]** → Receive=**!01**
12. wait 3 second, the led of I-7024 is flashed and the D/A output=0 volt.
13. press **2**
14. press **~011[Enter]**→ Receive=**!01**

- step 3: this is a I-7024 module, baudrate = 9600 output range 0 to 10V.
- step 5: set channel 0 D/A=0.0 volt
- step 7: set current D/A states as safe-value for channel 0
- step 9: set channel 0 D/A=5.0V
- step 11: enable the host watchdog and timer = 1EH*0.1s = 3 sec.
- step 12: this step simulate that the host fails and the D/A go to safe state
- step 14: clear the module status to 00

3.6 Module WatchDog Applications Notes

The $AA5 command is designed to detect the condition of module watchdog failure. **If the module is down, the module watchdog circuit will reset this module. After this resetting, the output state of module will go to their power-on value. The power-on value may be different from those output value before module reset. Therefore the user need to send output command to module for keeping the same output state before and after the reset of the module watchdog.**

The flow chart of module watchdog failure detection is given as following.



3.7 Using External Power for Current Output

The user can cascade multiple current devices to I-7021. If the JP1 is set to internal power, the load resistance is limited to 500 ohm. If the JP1 is set to external power, the load resistance can be 1050 ohm. Refer to Sec. 1.5.1 for application wiring.

If the user use external power, it is strongly recommended to use PWR-24 to increase the accuracy. The user should not use switching power supply for external power. The PWR-24 is a linear power adapter. The specifications of PWR-24 are given as following:

PWR-24/110V

- Primary excitation current: input 110V, 60Hz, 60mA
- Primary Load current: input 110V, AC 60 Hz, 52mA
- Secondary DC voltage(no load): 28.6V, +/- 5%
- Secondary DC voltage(load=100mA): 24.0V, +/-5%
- HI-POT TEST:
  PRI. TO SEC. & CORE: AC 1240/min
  SEC TO CORE: AC 500 V

PWR-24/220V

- Primary exciting current: input 220V, 50Hz, 55mA
- Primary Load current: input 220V, 50 Hz, 30mA
- Secondary DC voltage(no load): 29.2V, +/- 5%
- Secondary DC voltage(load=100mA): 24.0V, +/-5%
- HI-POT TEST:
  PRI. TO SEC. & CORE: AC 1240/min
  SEC TO CORE: AC 500 V

Dimension:

- 51mm x 41mm x 63mm

Order Information

- PWR-24 /220V: power adapter
- PWR-24/110V : power adapter