JDAM-9024
4 Analog Output Channel and
4 Digital Input Channel Module

USER'S MANUAL (V1.0)

健昇科技股份有限公司

JS AUTOMATION CORP.

新北市汐止區中興路 100 號 6 樓
6F., No.100, Zhongxing Rd.,
Xizhi Dist., New Taipei City, Taiwan
TEL：+886-2-2647-6936
FAX：+886-2-2647-6940
http://www.automation.com.tw
http://www.automation-js.com/
E-mail：control.cards@automation.com.tw

Contents

1. Introduction

1.1 Overview

The JDAM-9000 analog modules is a set of intelligent sensor to computer interface modules containing built-in microprocessor. They provide data comparison, and digital communication functions. Some modules provide analog I/O lines for controlling and monitoring analog signals.

1.2 Module Compatibility

The JDAM-9000 series are fully compatible to Advantech® ADAM-4000 series, ADlink® NµDAM-6000 series and ICP® I-7000 series.

1.3 Communication and Programming

JDAM modules can connect to and communicate with all computers and terminals. They use RS-485 transmission standards, and communicate with JDAM-ASCII format protocol(default) or Modbus-RTU protocol, which means that JDAM modules can be programmed in virtually any high-level language. Up to 256 JDAM modules may be connected to an RS-485 multi-drop network by using the JDAM RS-485 repeater, extending the maximum communication distance to 4,000 ft.

1.4 Software Configuration and Calibration

JDAM modules contain no ports or switches to set. By merely issuing a command from the host computer, you can change an analog input module to accept several ranges of voltage input. Remote configuration can be done by using the command set's configuration and calibration commands. By storing configuration and calibration parameters in a nonvolatile EEPROM, modules are able to retain these parameters in case of power failure.

1.5 JDAM ASCII and Modbus RTU protocol

Some JDAM-9000 modules support both JDAM-ASCII and Modbus-RTU protocols, and the factory default setting of these modules is JDAM-ASCII protocol. If you would like to configure the modules to Modbus-RTU protocol, please refer to address 00257 of Modbus-RTU function.

1.6 Watchdog Timer

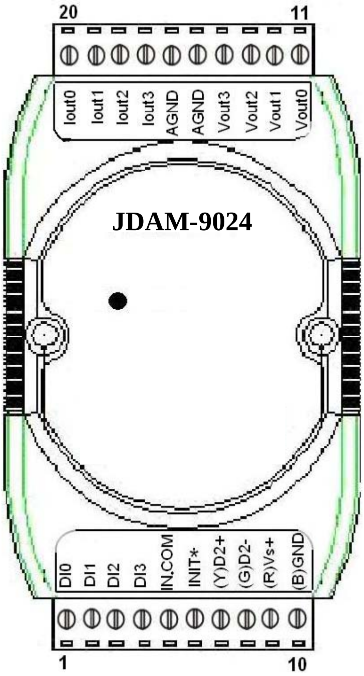
A watchdog timer supervisory function will automatically reset the JDAM modules in the event of system failure. Maintenance is thus simplified.

1.7 Power Requirements

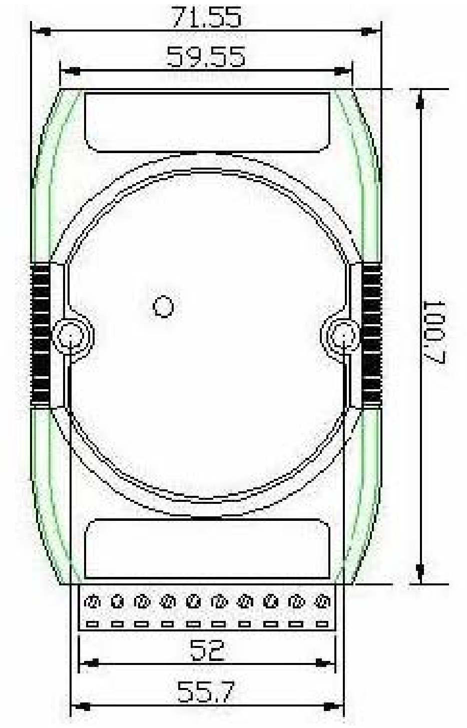
Although the modules are designed for standard industrial unregulated 24Vdc power supply , they accept any power unit that supplies power within the range of +10 to +30Vdc . The power supply ripple must be limited to 5V peak-to-peak, and the immediate ripple voltage should be maintained between +10 and +30Vdc.

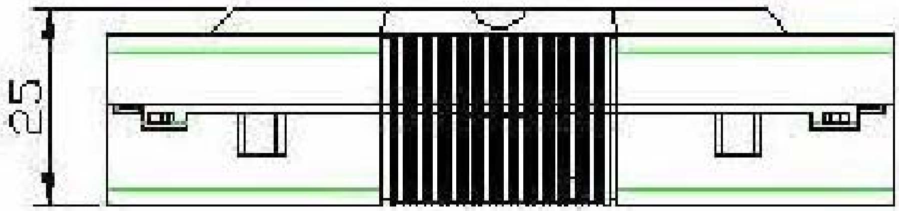
2. About the JDAM Analog Input Modules

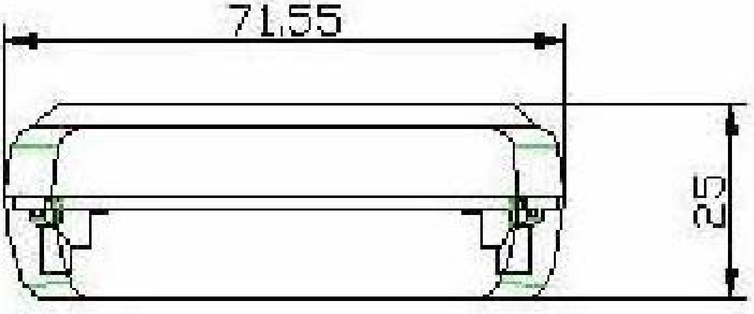
2.1 Outline of JDAM-9024 modules

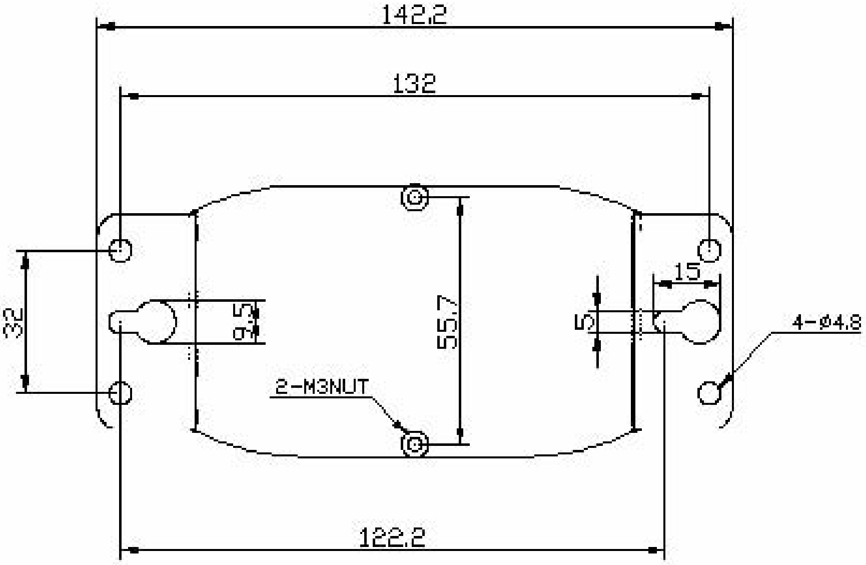


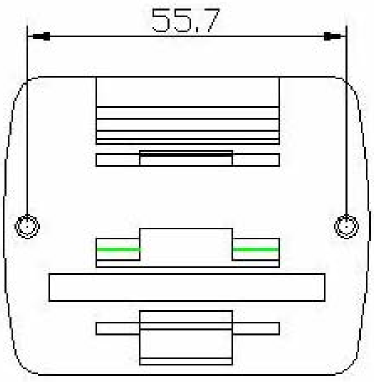
2.2 Module Dimension

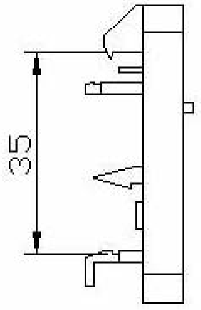



25


71.55
25


142.2
132
32
9.5
55.7
5
15
4-Ø4.8
2-M3NUT
122.2


55.7


35

2.2.1 Specifications of JDAM-9024

JDAM-9024 is a 4-channel analog output module with mixed type I/O. Under some circumstances, it is, however, a demand for multiple analog outputs to fulfill particular applications without many duplicate modules. JDAM-9024 is designed to achieve this purpose by integrating four A/O channels and four isolated D/I channels into only one module. The four digital input channels function as an interlock for emergency latch output. The LED indicators are used for status reading and both JDAM ASCII and Modbus-RTU protocols are supported.

JDAM-9024 provide multi-range A/O support, allows its four A/O channels working at the same time with different and more output ranges. For example, it can have 0~20mA and ±10 V at its output. To ensure the operation of machines and facilities, JDAM-9024 has the functionality of slew rate control. Output slope is programmable through ramping/clamping the slew rate. Unlike traditional mechanism, JDAM-9024 permits users to substitute its default value at the start up. Users can easily set up and configure the module to be more adaptive.

Specifications:

- Support Protocol: JDAM-ASCII(default) and MODBUS-RTU
- Interface: RS-485, 2 wires
- Speed (bps): 1200, 2400, 4800, 9600, 19.2K, 38.4K , 115.2K
- Analog output :
  - Output type: mA, V
  - Analog Channels Numbers: 4
  - Analog Resolution: 14 bits
  - Output Range: 0~20 mA, 4~20 mA, 0~+5V, ±5V, 0~+10V and ±10V
  - Programmable Output Slope : 0.125 to 2048 mA/Second
    0.0625 to 1024 V/Second
  - Current Load Resistor: External 24V/1050 ohms
  - Voltage Output : 5mA max.
  - Accuracy :±0.1% of FSR for current output
    ±0.2% of FSR for voltage output
  - Zero Drift: Voltage output : ±30μV/°C
    Current output : ±0.2μA/°C
  - Span Temperature Coefficient: ±25 ppm/°C
  - Isolation voltage : 3000Vdc
- Isolation Digital Input:
  - Channel: 4
  - Logical level 0: +1V max.
  - Logical level 1: +10 ~ +30Vdc
  - Isolation voltage : 3750Vrms

- Watchdog Function:
  - Module internal watchdog timer: 200 ms
  - Power failure threshold: 4.65V
  - Safety value: 2 digital output channels
  - Host programmable watchdog: 100 ms ~ 25.500 sec
- Overvoltage protection: ±35V
- Power input : +10V to +30Vdc
- Consumption: 2.4W

2.3 Block diagram of JDAM-9024 modules

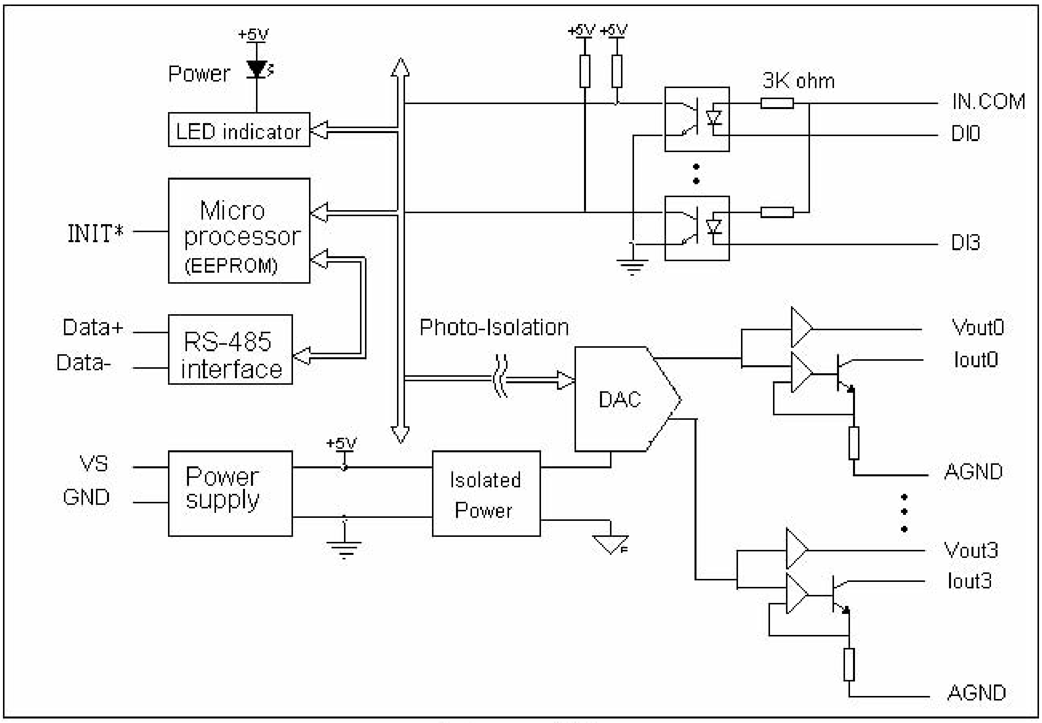


Figure 2.1

2.4 Wire connection

2.4.1 Analog output Wire connection

Voltage output Wire connection

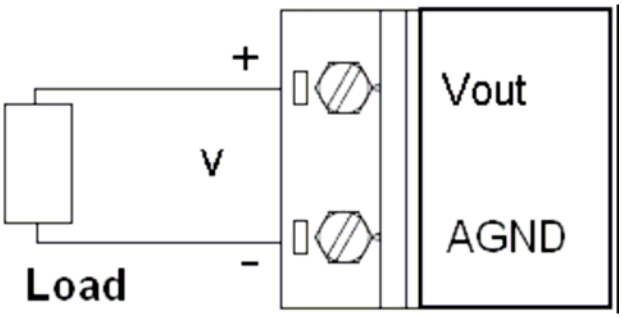


Figure 2.2

Current output Wire connection

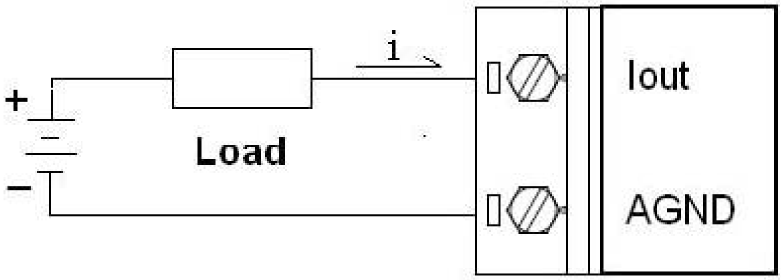


Figure 2.3

2.4.2 Isolation digital input wire connection

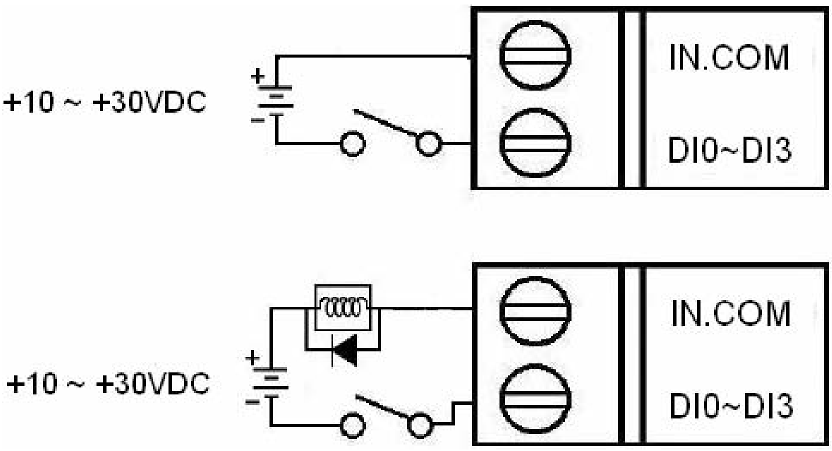


Figure 2.4

2.4.3 Power and Initial wire connection

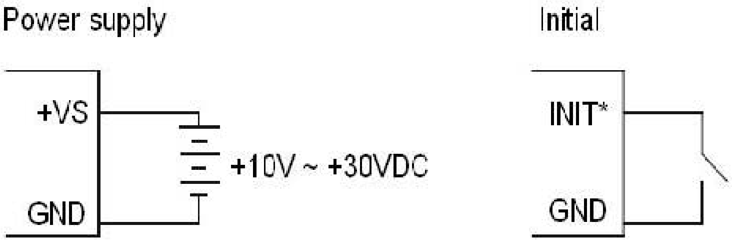


Figure 2.5

2.5 JDAM-9024 pin assignments

pin	name	description
1	DI0	Digital Input Channel 0
2	DI1	Digital Input Channel 1
3	DI2	Digital Input Channel 2
4	DI3	Digital Input Channel 3
5	IN.COM	Digital common source (+10V ~ +30Vdc)
6	INIT*	Initial state setting
7	DATA+	RS-485 series signal, positive
8	DATA	RS-485 series signal, negative
9	+Vs	Power supply, +10V~+30V
10	GND	Ground
11	Vout0	Voltage output channel 0
12	Vout1	Voltage output channel 1
13	Vout2	Voltage output channel 2
14	Vout3	Voltage output channel 3
15	AGND	Analog output ground
16	AGND	Analog output ground
17	Iout3	Current output channel 3
18	Iout2	Current output channel 2
19	Iout1	Current output channel 1
20	Iout0	Current output channel 0

The module accepts baud rate, checksum and communication protocol configuration setting under the **INIT*** mode. (**INIT*** connect to the ground)

3. Installation

This chapter provides guidelines to what is needed to set up and install an JDAM network. A quick hookup scheme is provided that lets you configure modules before they are installed in a network. To help you to connect JDAM modules with sensor inputs, several wiring examples are provided. Finally, you will find at the end of this chapter a programming example using the JDAM command set. Be sure to carefully plan the layout and configuration of your network before you start.

3.1 Set up an JDAM network

The following list gives an overview of what is needed to setup, install and configure an JDAM environment. A host computer that can output JDAM ASCII format or Modbus-RTU protocol with an RS-232C or RS-485 port. Power supply for the JDAM modules (+10 to +30Vdc) JDAM Series Utility software

3.2 Host computer

Any computer or terminal that can output in COMMAND over either RS-232 or RS-485 can be connected as the host computer. When only RS-232 is available, an JDAM-9520 module (RS-232/RS-485 converter) is required to transform the host signals to the correct RS-485 protocol. The converter also provides opto-isolation and transformer-based isolation to protect your equipment.

For the ease of use in industrial environments the JDAM modules are designed to accept industry standard +24Vdc unregulated power. Operation is guaranteed when using any power supply between +10 and +30Vdc. Power ripples must be limited to 5V peak to peak while the voltage in all cases must be maintained between +10 and +30Vdc . All power supply specifications are referenced at module connector. When modules are powered remotely, the effects of line voltage drops must be considered.

3.3 Power supply

All modules use on-board switching regulators to sustain good efficiency over the +10 ~ +30Vdc input range, therefore we can assume that the actual current draw is inversely proportional to the line voltage. The following example shows how to calculate the required current that a power supply should be able to provide.

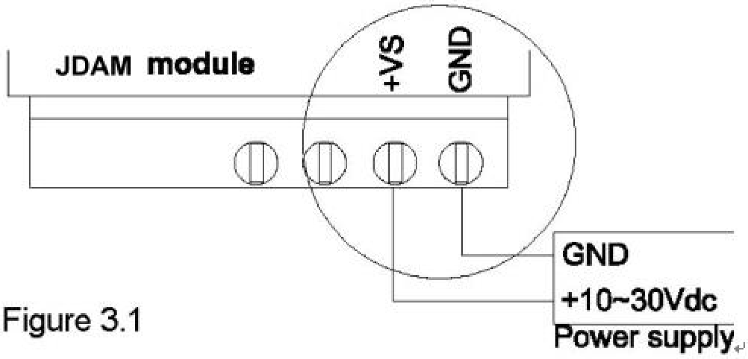


Figure 3.1

3.4 Communication Wiring

We recommend that shielded-twisted-pair cables that comply with the EIA RS-485 standard be used with the JDAM network to reduce interference.

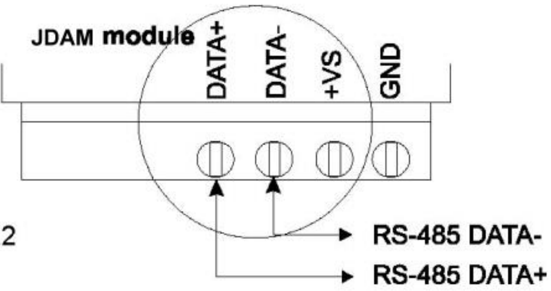


Figure 3.2

3.5 JDAM Utility Software

A menu-driven utility program called “DOSEDAM.EXE” for DOS or “WINEDAM.EXE and RTUEDAM8000.EXE for Windows is provided for JDAM module configuration, monitoring and calibration. It also includes a terminal emulation program that lets you easily communicate through the JDAM command set

3.6 JDAM Isolated RS-232/RS485 Converter

When the host computer or terminal has only a RS-232 port, an JDAM-9520 Isolated RS-232/RS-485/422 converter connected to the host’s RS-232 port is required. This module equips a “Auto baud rate detector” inside, therefore it can detect the baud rate and data format automatically and control the direction of RS-485 precisely.

3.7 Initializing the Module

All JDAM modules in a RS-485 network must have an unique address ID. Therefore, to configure the address ID before using is necessary :

- INIT* State settings:

  The JDAM I/O modules must be set at INIT* State when you want to change the default settings, such as the ID address, baud rate, check-sum status etc. All JDAM I/O modules have an special pin labeled as INIT*. The module will be in Default State if the INIT* pin is shorted to ground when power ON. Under this state, the default configuration is set as following :

  Address ID is 00
  Baud rate is 9600 bps
  Check-sum disable
  Protocol is JDAM-ASCII format protocol.

Therefore, the communication between host and the module will can be easily set as the same configuration, the initialization of a module will be possible no matter what configuration is set under operating state.

3.8 Initialization Procedure

1. Power off the host computer and the installed JDAM-9520 to COM port of host computer.
2. Connect a brand new JDAM module with the RS-485. Set the module in Default State by shorting the INIT* pin to GND. Refer to Figure 3.3 for detailed wiring.
3. Power on the power supply for JDAM modules.
4. Use the JDAM utility to configure the address ID, baud rate, check-sum status and command sets of the module.

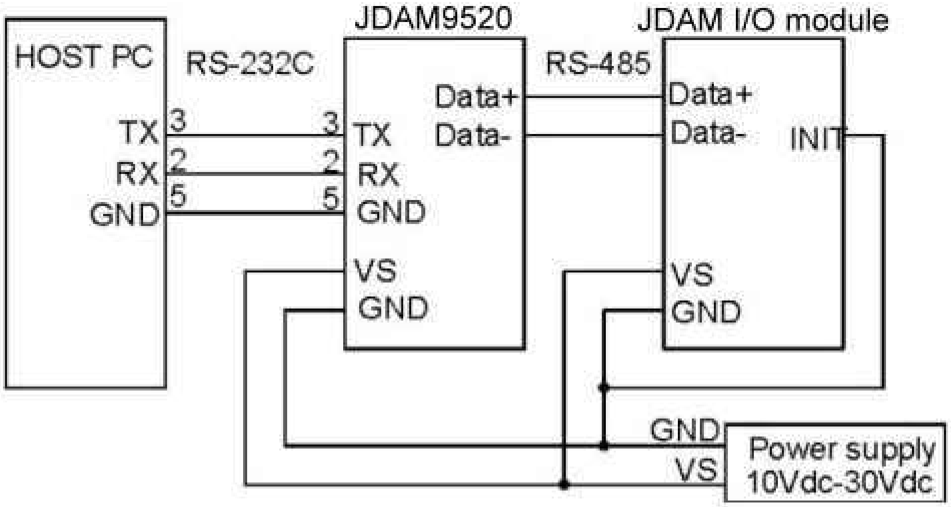


Figure 3.3

3.9 Install a New JDAM to a Existing Network

1. Equipments for Install a New Module
2. A existing JDAM network
3. New JDAM modules.
4. Power supply (+10 to +30Vdc)

Installing Procedures:

1. Configure the new JDAM module according to the initialization procedure.
2. The baud rate and check-sum status of the new module must be identity with the existing RS-485 network. The address ID must not be conflict with other JDAM modules on the network.
3. Power off the JDAM power supply of the existing RS-485 network.
4. Wire the power lines for the new JDAM with the existing network. Be careful about the signal polarity as wiring.
5. Wire the RS-485 data lines for the new JDAM with the existing network. Be careful about the signal polarity as wiring.
6. Wire to the input or output devices.
7. Power on the JDAM local power supply.
8. Use the JDAM utility to check entire network.

3.10 Protocol Switching

To switch to the JDAM-ASCII format protocol:

1. Uses address 00257 of Modbus function and set to a value of 0.
2. After a power-on reset, the communication protocol will be changed to JDAM-ASCII format protocol.

To switch to the Modbus RTU protocol:

1. Sends the $AAPN command and set N to a value of 1.

   Note: It is necessary to short the pin **INIT*** to ground.

   After a power-on reset, the communication protocol will be changed to the Modbus-RTU protocol.

3.11 Configuration Table of JDAM-9024

3.11.1 Baud Rate settings (CC)

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

3.11.2 Analog output Type Setting (TT)

Type code	30 hex	31 hex	32 hex	33 hex	34 hex	35 hex
Min. Input	0mA	4mA	0V	-10V	0V	-5V
Max. Input	20mA	20mA	+10V	+10V	+5V	+5V

3.11.3 Data format settings (FF)

The JDAM analog output modules can be configured to transmit data to the module in engineering units and hexadecimal binary data format

Engineering units: Only for JDAM-ASCII mode

- This data format including three components.
  1. sign (+ or -)
  2. digits
  3. decimal point
- Data is composited with a sign (+ or -) followed with 5-digits and a decimal point.
- It does not exceed 7-characters.
- Over Range(+9999.9) , Under Range(-9999.9).

Bit	7	6	5	4	3	2	1	0

Bit7: - reserved = 0

Bit6: - Checksum Bit

= 0 - Disable checksum/CRC (default)

= 1 - Enable checksum/CRC

Bit5~bit2: -reserved

= 0

Bit1~bit0 : -Data format

= 00 -Engineer unit format

Hexadecimal binary format: Only for Modbus-RTU mode

3.11.4 Analog output types and data format table

Type code	Range	Format	Max..	Min.	Output Resolution
30 hex	0mA ~ 20mA	Engineer unit	+20.000	+00.000	2.4414μA
		Hex Binary	3FFF	0	1.2207μA
31 hex	4mA ~ 20mA	Engineer unit	+20.000	+04.000	2.4414μA
		Hex Binary	3FFF	0	0.9766uA
32 hex	0V ~ +10V	Engineer unit	+10.000	+00.000	1.2207mV
		Hex Binary	3FFF	0	0.6104mV
33 hex	-10V ~ +10V	Engineer unit	+10.000	-100.000	1.2207mV
		Hex Binary	3FFF	C000	0.6104mV
34 hex	0V ~ +5V	Engineer unit	+05.000	+00.000	1.2207mV
		Hex Binary	3FFF	0	0.3052mV
35 hex	-5V ~ +5V	Engineer unit	+05.000	-05.000	1.2207mV
		Hex Binary	3FFF	C000	0.3052mV

Note: Hexadecimal binary format only for Modbus RTU mode

3.11.5 Slew Rate(SS)

The slew rate is defined as the discrepancy between the present number of milliamps (or Volts) per second and the required output currents (or voltages). An JDAM analog output module may be configured for a specific slew rate.

Slew rate	V/Sec.	mA/Sec.		Slew rate	V/Sec.	mA/Sec.
00	Immediate			08	8.0	16.0
01	0.0625	0.125		09	16.0	32.0
02	0.125	0.25		0A	32.0	64.0
03	0.25	0. 5		0B	64.0	128.0
04	0.5	1.0		0C	128.0	256.0
05	1.0	2.0		0D	256.0	512.0
06	2.0	4.0		0E	512.0	1024.0
07	4.0	8.0		0F	1024.0	2048.0

Note:

1. Type and slew rate are set by command "$AA9NTTSS" (see p46)
2. The analog output value at 33 conversions per second.
3. It’s needed to short the INIT* pin to ground while changing baud rate and/or enable/disable checksum.

3.12 Default Settings

The JDAM-9024 Factory default settings:

- Address ID : 01
- Baud rate : 9600 bps, (no parity, 8 data bits, 1 stop bit)
- Check-sum is disabled.
- Engineer unit format.
- Analog output type: type 32 hex (0V ~ +10V)
- Analog output slew rate is Immediate.
- Protocol : JDAM-ASCII format protocol.

4. JDAM-ASCII format protocol Command Sets

4.1 Introduction

In JDAM-ASCII format protocol mode The modules can communicate with ASCII format commands. All communications to and from the module are performed in ASCII. The JDAM command is composed by numbers of characteristics, including the leading code, address ID, the variables, the optional check-sum byte, and a carriage return to indicate the end of a command. The host computer can only command only one JDAM module except those synchronized commands with wildcard address command “#**”. The Module may or may not give response to the command. The host should check the response to handshake with the modules.

4.2 Format of ASCII protocol Commands

Syntax: (Leading code)(Addr)(Command)[Data] <Cksum><CR>

Every command begins with a delimiter character. There are five valid characters: a dollar sign $, a pound sign #, a percentage ,a wave sign ’~’ ,sign % and an at sign @. The delimiter character is followed by a two-character address (hex) 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 commands is terminated by a carriage return (cr).

Conventions

Leading Code	The first characteristic of the JDAM command, such as %,$,#,~, @, ...etc (1- character)
Addr	Module’s address ID, the value is in the range of 00 – FF (Hex) 2- character
Command	Command codes or value of variables
Data	Data needed by some output command
Checksum	Checksum in brackets indicate optional parameter, only checksum is enable then this field is required (2- character)
<CR>	carriage return(0x0D)

Note:

all commands should be issued in ASCII uppercase characters. There is no spacing between characters.

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 0ffh
3. [Checksum]={(Leading code)+(addr)+(command)+[data]}   MOD 0x100

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 :   !01400600(cr)

Sum of string='!'+'0'+'1'+'4'+'0'+'0'+'6'+'0'+'0'

=21h+30h+31h+34h+30h+30h+36h+30h+30h=1ACh

The checksum is ACh, and [CHK]="AC"

Response string with checksum=!01400600AC(cr)

4.3 Response of Commands

The response message depends on JDAM command. The response is also composed with several characteristics, including leading code, variables, and carriage return for ending. There are two kinds of leading code for response message, "!" or ">" means valid command and "?" means invalid. By checking the response message, user can monitor the command is valid or invalid. But under the following conditions, there will have no response message.

- The specified address ID is not exist.
- Syntax error.
- Communication error
- Some special commands does not have response.

4.4 Summary of Command Set

There are four categories of JDAM-9024 commands. The first is the General commands. The second is the Analog Commands, The third is the Digital input Commands and the last is Host Watchdog Command Sets. All the commands used in the JDAM analog output module are list in the following table.

4.4.1 General Command Sets

Command	Response	Description	Page
%AANNTTCCFF	!AA	Set Module Configuration	22
$AA2	!AATTCCFF	Read Configuration	25
$AA5	!AAS	Read Reset Status	26
$AAF	!AA(data)	Read Firmware Version	27
$AAM	!AA(Data)	Read Module Name	28
~AAO(Data)	!AA	Set Module Name	29
$AAPN	!AA	Sets the communication protocol.	30
$AAP	! AASC	Reads the communication protocol information.	31
~AAI	!AA	Soft INIT	32
~AATnn	!AA	Sets the Soft INIT Time-out	33
$AAS1	!AA	Reloads the module factory default	35
$AARS	!AA	Reset the module to initial power on state.	36

4.4.2 Analog Command Sets

Command	Response	Description	Page
#AAN(data)	>	Output Analog Value	37
$AA0N	!AA	0mA/-10V Calibration	38
$AA1N	!AA	20mA/10V Calibration	39
$AA3NVV	!AA	Trim Calibration	40
$AA4N	!AA	Set PowerOn Value for Channel N	41
$AA6N	!AA(Data)	Last Value of Channel N Readback	42
$AA7N	!AA	Read PowerOn Value	43
$AA8N	!AA(Data)	Current Value Readback	44
$AA9N	!AATTSS	Read DA Configuration of channel N	45
$AA9NTTSS	!AA	Set DA Configuration of channel N	46

4.4.3 Digital input Command Sets

Command	Response	Description	Page
@AA	>(Data)	Read Emergency DI Input port	47

4.4.4 Watchdog Command Sets

Command	Response	Description	Page
~**	No Response	Host OK	48
~AA0	!AASS	Reads module status.	49
~AA1	!AA	Reset Module Status	51
~AA2	!AAVV	Read host watchdog timeout interval	52
~AA3EVV	!AA	Set Host Watchdog Timeout interval	53
~AA4N	!AA(Data)	Read Safe Value	54
~AA5N	!AA	Set Safe Value	55
~AA8E	!AA	Enable/Disable channel(N) Emergency flag	56

4.5 Set Module configuration

Modules:	**For 9024 module**	
Description:	Configure the basic setting of JDAM, including the address ID, input range, baud rate, and data format.	
Command:	%AANNTTCCFF[CHK](cr)	
Syntax:	%	Command leading code
	AA	Module address ID (00 to FF)
	NN	New JDAM address ID (00 to FF)
	TT	= 00 -Reserved (see p46)
	CC	Set new baud rate of module (See p15)
	FF	Data format (See p16)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	New Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note: When you want to change the checksum or baud rate, the INIT* pin must be grounded at first. (see Appendix A)

Example 1:

Change ID address from 01 to 03 (Assume current baud rate is 9600 and checksum disabled)

Command: %0103000600(cr)

Response: !03(cr)

response new module ID address 03 (change ID address only)

Example 2:

Change baud rate from 9600 to 19200(Assume current ID is 03,
baud rate is 9600, and checksum disabled).

Because that the baud rate is changed from 9600 to 19200, the following procedures should be done before sending this command:

1. Power off the module
2. Short INIT* pin to Ground
3. Power on the module
4. send command string
5. Command: %0003000700(cr)
6. Response: !03(cr)
7. response module ID address 03
8. Power off module
9. Open INIT* pin and power on module again

Example 3:

Enable checksum(Assume current ID is 03, baud rate is 9600 and checksum disabled).

Because that the checksum is changed from disable to enable, the following procedures should be done before sending this command:

1. Power off the module
2. Short INIT* pin to Ground (see Appendix A)
3. Power on the module
4. send command string
5. Command: %0003000640(cr)
6. Response: !03(cr)
7. response module ID address 03
8. Power off module
9. Open INIT* pin and power on module again (checksum enabled)

Example 4:

Change baud rate from 9600 to 19200 and enable checksum (Assume current ID is 03, baud rate is 9600 and checksum disabled). Because that both the baud rate and checksum is changed , the following procedures should be done before sending this command:

1. Power off the module
2. Short INIT* pin to Ground (see Appendix A)
3. Power on the module
4. send command string
5. Command: %0003000740(cr)
6. Response: !03(cr)
7. response module ID address 03
8. Power off module
9. Open INIT* pin and power on module again (Baud rate changed to 19200 and checksum enabled)

It is recommended to use the setup utility to configure the module

Related topics: $AA2, $AA9NTTSS, $AA9N

4.6 Read Configuration

Modules:	**For 9024 module**	
Description:	Read module configuration	
Command:	$AA2[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	2	Command for reading configuration
	CHK	Check sum
	(cr)	Carriage return
Response:	!AATTCCFF[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	TT	= 00 (see p45)
	CC	Baud rate (see p15)
	FF	Data format of module (see p16)
	CHK	Check sum
	(cr)	Carriage return

Example 1:

Read configuration of module with ID address=01

Command: $012(cr)

Response : !01000600(cr)

Read address ID =01 module configuration
= 00 - TT (see p45)
= 06 - 9600 baud rate
= 00 - no checksum,

Related command: %AANNTTCCFF , $AA9NTTSS, $AA9N

4.7 Read reset status

Modules:	**For all modules**	
Description:	Read reset status	
Command:	$AA5[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	5	Command for read reset status
	CHK	Check sum
	(cr)	Carriage return
Response:	!AAS[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	S	= 0 - the module is not been reseted
		= 1 - the module is been reseted
	CHK	Check sum
	(cr)	Carriage return

Example :

Read address 01 Read reset status and return module is bee reseted

Command: $015(cr)

Response : !011(cr) -the module is been reseted

Command: $015(cr)

Response : !010(cr) -the module is not been reseted

Related command: $AARS

4.8 Read Firmware Version

Modules:	**For all modules**	
Description:	Read Firmware Version	
Command:	$AAF[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	F	Command for Read Firmware Version
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	(data)	firmware version of module(max. 6 chars.)
	CHK	Check sum
	(cr)	Carriage return

Example :

Read address 01 Read Firmware Version and return version A00.02

Command: $01F(cr)

Response: !01A00.02 (cr) - BIOS version A00.02

Related command:

4.9 Read Module Name

Modules:	**For all modules**	
Description:	Read Module Name	
Command:	$AAM[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	M	Command for Read Module Name
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	(data)	A string showing the name of the module (max. 6 chars.)
	CHK	Check sum
	(cr)	Carriage return

Example:

Read name of module 01 and return the module name “9024”

Command: $01M(cr)

Response : !018024(cr)

Related command: ~AAO

4.10 Sets the name of a module

Modules:	**For all modules**	
Description:	Sets the name of a module	
Command:	~AAO(data)[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	O	Command to Sets the name of a module
	(data)	New name of the module (max. 6 characters).
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example:

(1) Read name of module 01 and return the module name “9024”

Command: $01M(cr)

Response : !018024(cr)

(2) Sets the name of the module 01 to be “JDAM" and returns a valid response.

Command: ~01OEDAM (cr)

Response : !01 (cr)

(3) Read address 01 Read Firmware Version, return the module name “JDAM”

Command: $01M(cr)

Response : !01EDAM(cr)

Related command: $AAM

4.11 Set the communication protocol

Modules:	**For all modules**	
Description:	Set the communication protocol	
Command:	$AAPN[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	P	Command to Set the communication protocol
	N	The protocols supported by the module = 0 - JDAM-ASCII format protocol (default) = 1 -Modbus-RTU protocol
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note:

1. Before the command is issued, the INIT* pin should be connected to GND.
2. The new protocol is saved in the EEPROM and will be effective after the next power on reset.

Example :

Sets the communication protocol of module 01 to Modbus-RTU and returns an valid response

Command: ~01P1 (cr)

Response : !01(cr)

Related command: ~AAP

4.12 Reads the communication protocol information

Modules:	**For all modules**	
Description:	Reads the communication protocol information	
Command:	$AAP[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	P	Command for Read protocol information
	CHK	Check sum
	(cr)	Carriage return
Response:	!AASC[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	S	The protocols supported by the module = 0 - only ASCII protocol is supported = 1 - both the ASCII and Modbus RTU protocols are supported
	C	The protocols supported by the module = 0 -JDAM-ASCII format protocol = 1 - Modbus-RTU protocol
	CHK	Check sum
	(cr)	Carriage return

Example :

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

Command: $01P(cr)

Response : !0110(cr)

Related command: $AAPN

4.13 Soft INIT* command

Modules:	**For all modules**	
Description:	The Soft INIT* command is used to enable modification of the Baud Rate and checksum settings using software only.	
Command:	~AAI[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	I	Command to set the Soft INIT
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note: The ~AATnn command should be sent prior to sending this command.

Example : Sets the soft INIT* of module 01 and returns a valid response.
Command: ~01I(cr)
Response : !01(cr)

Related command: %AANNTTCCFF, ~AATnn

4.14 Sets the soft INIT* timeout value

Modules:	**For all modules**	
Description:	Sets the soft INIT* timeout value.	
Command:	~AATnn[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	T	Command to set the soft INIT time out value
	nn	Two hexadecimal digits representing the timeout value in seconds. The maximum timeout value is 60 seconds. When changing the Baud Rate or checksum settings without altering the INIT* pin, the ~AAI and %AANNTTCCFF commands should be sent consecutively and the time interval between the two commands should be less than the soft INIT timeout. If the soft INIT timeout is 0, then the Baud Rate and checksum settings cannot be changed using software only. The power-on reset value of the soft INIT timeout is 0.
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example :

(1) Sets the soft INIT* timeout value of module 01 and returns a valid response.

Command: ~01I (cr)		Response: !01(cr)

(2) Attempts to change the Baud Rate of module 01 to 19200 without first altering the INIT * pin. The module returns an invalid response because the soft INIT timeout value is 0.

Command: %0101000700 (cr)		Response: ?01(cr)

(3) Sets the soft INIT* timeout value of module 01 to 16 seconds and returns a valid response.

Command: ~01T10 (cr)		Response: !01(cr)

(4) Sets the soft INIT* timeout value of module 01 and returns a valid response.

Command: ~01I (cr)		Response: !01(cr)

(5) Changes the Baud Rate of module 01 to 19200 without first altering INIT * pin. The module returns

Command: %0101000700 (cr)		Response: !01(cr)

Related command: %AANNTTCCFF, ~AATI

4.15 Reloads factory default

Modules:	**For all modules**	
Description:	Reloads the module factory default setting.	
Command:	$AAS1[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	S1	Command to reload the factory default
	CHK	Check sum
	(cr)	Carriage return
Response:	!AAV[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note:

Before the command is issued, the INIT* pin should be connected to GND and after reponse command is issued, the module will be rebooted.

Example:

Reloads the module factory default setting and return valid.

Command: $05S1(cr)

Response : !05(cr)

Related command: %AANNTTCCFF, $AA2

4.16 Reset the module to initial power on state

Modules:	**For all modules**	
Description:	To stop current operation , reset the module to initial power on state.	
Command:	$AARS[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	RS	Command for reset the module
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example :

To stop current operation and reset the module to power-on state or safe output.

Command: $05RS(cr)

Response : !05(cr)

Related command:

4.17 Output Analog Value

Modules:	**For 9024 module**	
Description:	Output Analog Value for channel N	
Command:	#AAN(data) [CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	N	Analog output channel (0 to 3)
	(data)	analog output value, (see p15)
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!AA[CHK](cr)	Ignore Command
	>	Delimiter for valid command
	?	Delimiter for Invalid command
	!	Delimiter for the module's host watchdog status is set or Emergency input is active and the output command will be ignored.
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example:

Output address 01 value +02.456V for channel 2 and return success.

Command: #012+02.456

Response: > (cr)

Example:

Output address 01 value +02.456V for channel 2 and return emergency input is active(low) and the output command will be ignored. (see p56)

Command: #012+02.456

Response: ! (cr)

Related command: $AA9, $AA9NTTSS, $AA6N, $AA8N

4.18 Perform 0mA/-10V Calibration

Modules:	**For 9024 module**	
Description:	Tells the module to store parameters for channel(N) 0mA/-10V calibration. (see Appendix B)	
Command:	$AA0N [CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	0	Command for performing 0mA/-10V calibration
	N	Channel to calibrate (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example(1):

Perform 0mA/-10V Calibration for Channel 3

Command: $0203(cr)

Response: !02 (cr)

Related command: $AA3NVV, $AA1N, #AAN(data)

4.19 Perform 20mA/+10V Calibration

Modules:	**For 9024 module**	
Description:	Tells the module to store parameters for channel(N) 20mA/+10V calibration. (see Appendix B)	
Command:	$AA1N [CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	1	Command for performing 20mA/+10V calibration
	N	Channel to calibrate (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example:

Perform 20mA/+10V Calibration for Channel 3 and return valid.

Command: $0213(cr)

Response: !02 (cr)

Related command: $AA3NVV, $AA0N, #AAN(data)

4.20 Trim Calibration

Modules:	**For 9024 module**	
Description:	Trims specified module a number of units up/down (see Appendix B)	
Command:	$AA3NVV [CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	3	Command for trimming calibration
	N	Channel to trim (0 to 3)
	VV	2’s complement hexadecimal to trim the analog output value. 00 to 5F to increase 0 to 95 counts, and FF to A1 to decrease 1 to 95 counts. Each count indicates 2.44µA or 1.22mV.
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA [CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	delimiter for invalid command or the calibration is not enabled
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example:

Trim address 02 channel 1 output 52 counts and return success.

Command: $023134(cr)

Response: !02 (cr)

Related command: $AA0N, $AA1N, #AAN(data)

4.21 Set Power-On Value for Channel N

Modules:	**For 9024 module**	
Description:	Stores a default output value in a specified module. The output value will take effect upon startup. (see Appendix C)	
Command:	$AA4N[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	4	command for setting Power-On Value
	N	Channel to set (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example:

(1) Set address 02 channel 1 output +01.50, return success.

Command: #021+01.500 (cr)

Response: >(cr)

(2) Set address 02 channel 1 Power-On Value, return success. The Power-On Value of channel 1 is set to 01.50 now

Command: $0241(cr)

Response: >(cr)

Related command: #AAN(data)

4.22 Last Value of Channel N Readback

Modules:	**For 9024 module**	
Description:	Returns either last value sent to specified module by #AAN command. (see 3.11.5)	
Command:	$AA6N[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	6	command for reading last output value
	N	Channel to readback (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	(data)	the last output command value. (see p17)
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example :

(1) Set address 02 channel 1 output +02.567, return success.

Command: #021+02.567(cr)

Response: !02 (cr)

(2) Read address 02 channel 1 last output value, return +02.567

Command: $0261

Receive : !00+02.567(cr)

Related command: #AAN(data)

4.23 Read Power-On Value of Channel N

Modules:	**For 9024 module**	
Description:	Read Power-On Value of Channel N	
Command:	$AA7N[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	7	command for reading Power-On Value
	N	Channel to readback (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	(data)	Power-On Value. (see p17)
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example :

Read address 02 channel 1 PowerOn Value, return +05.000.

Command: $0271(cr)

Response: !02+05.000 (cr)

Related command: $AA4N , #AAN(data)

4.24 Current Value Readback

Modules:	**For 9024 module**	
Description:	Readback the current analog output value for channel N.	
Command:	$AA8N[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	8	command for read current output value
	N	Channel to readback (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	(data)	current analog output value. (see p17)
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example :

(1) Read address 01 configuration, return output type 0 to 10V, and slew rate is 1.0V/Second.

Command: $0190(cr) Response: !013205 (cr)

(2) Read address 01 channel 0 current value, return 0.0V.

Command: $0180 (cr) Response: !01+00.000 (cr)

(3) Set address 01 channel 0 output 10.0V, return success.

Command: #010+10.000 (cr) Response: > (cr)

(4) Read address 01 channel 0 last output command value, return +10.000V.

Command: $0160 (cr) Response: !01+10.000 (cr)

(5) Wait 1 seconds and Read address 01 channel 0 current value, return 9.0V.

Command: $0180 (cr) Response: !01+09.000 (cr)

(6) Wait 6 seconds, Read address 01 channel 0 current value, return 3.0V.
Command: $0180 (cr) Response: !01+03.000 (cr)

Related command: #AAN(data), $AA9, $AA9N

4.25 Read DA Configuration of Channel N

Modules:	**For 9024 module**	
Description:	Read type and slew rate of Channel N	
Command:	$AA9N[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	9	command for reading DA configuration
	N	channel to read DA configuration (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AATTSS[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	TT	Analog output type. (see p17)
	SS	Analog output slew rate. (see p17)
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example :

Read address 01 channel 2 DA configuration, return +/-10V output and slew rate is 5 (1.0V/Second).

Command: $0192(cr)

Response: !013305 (cr)

Related command: $AA9NTTSS

4.26 Set DA Configuration of Channel N

Modules:	**For 9024 module**	
Description:	Set type and slew rate for Channel N	
Command:	$AA9NTTSS[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	9	command for set DA configuration
	N	channel to set DA configuration (0 to 3)
	TT	Analog output type. (see p17)
	SS	Analog output slew rate. (see p17)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example:

Set address 01 channel 2 type to 30 (0 to 20mA) and slew rate is 5 (1.0V/Second).

Command: $01923005 (cr)

Response: !01(cr)

Related command: $AA9N

4.27 Reads Emergency digital input ports

Modules:	**For 9024 module**	
Description:	Reads Emergency digital input ports.	
Command:	@AA[CHK](cr)	
Syntax:	@	Command leading code
	AA	Module address ID (00 to FF)
	CHK	Check sum
	(cr)	Carriage return
Response:	>(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	(data)	Status of the digital input ports, a four-digit hexadecimal value. The first word represents the status of DI(15~8) and the second word represents the status of DI(7~0). Value 0 of bit means the channel is input low level, value 1 of bit means the channel is input high level.
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example :

Read address 01 digital input status and return Dl(0,2) high level and Dl(1,3) low level.

Command: @01(cr)

Response: >0005(cr)

Related command: ~AA0, ~AA8E

4.28 Host OK

Modules:	**For all modules**	
Description:	Host sends this command to all modules for broadcasting the information “Host OK” (see Appendix D).	
Command:	~** [CHK](cr)	
Syntax:	~	Command leading code
	**	command for all modules
	CHK	Check sum
	(cr)	Carriage return
Response:	No response.	

Example :

Send Host OK to all modules.

Command: ~** (cr)

Response: No response

Related command: ~AA0, ~AA1, ~AA2, ~AA3EVV, ~AA4N, ~AA5N

4.29 Reads the module status

Modules:	**For 9024 module**	
Description:	Reads the watchdog status & emergency input flag.	
Command:	~AA0[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	0	Command for reading module status
	CHK	Check sum
	(cr)	Carriage return
Response:	! AASS[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	SS	Module Status(hex ascii). bit(7) - Host watchdog enable flag, = 0 - Disable. = 1 -Enable. bit(6~3) - Emergency input E/Disable flag, one channel per bit of bit(3~6) for channel(0~3) and status is indicated as: = 0 - Disable. = 1 - Enable. bit(2) - Host watchdog timeout flag, = 0 - Clear. = 1 - Set. bit(1~0) - reserved.
	CHK	Check sum
	(cr)	Carriage return

Note:

(1) the watchdog timeout status will be stored in EEPROM of the module and can only be cleared by issuing ~AA1 command.

(2) the emergency input(DI) enable/disable flag will be stored in EEPROM of the module and can only be set by issuing ~AA8NE command.

Example(1):

Reads the host watchdog status of module 01 and returns 00, meaning that the host watchdog is disabled and no host watchdog time out has occurred.

Command: ~010<cr>

Response: !0100

Example(2):

Reads the host watchdog status of module 02 and returns 84, meaning that a host watchdog timeout has occurred.

Command: ~010<cr>

Response: !0184

Example(3):

Reads the host watchdog status of module 01 and returns 04, meaning that the host watchdog is disabled and host watchdog time out has occurred.

Command: ~010<cr>

Response: !0104

Example(4):

Reads the status of module 01 and returns 09, meaning that channel(0) emergency Input is enable.

Command: ~010<cr>

Response: !0109

Related command: ~**, AA1, ~AA2, ~AA3EVV, ~AA4N, ~AA5N, ~AA8E

4.30 Reset Module Status

Modules:	**For 9024 module**	
Description:	Resets the watchdog active status of a module (see p69).	
Command:	~AA1[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	1	Command for resetting watchdog active status
	CHK	Check sum
	(cr)	Carriage return
Response:	! AA [CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note: The module’s watch dog active status will be cleared after this command issued.

Example:

Resets the host watchdog time out status of module 01 and returns a valid response.

Command: ~011<cr>

Response: !01

Related command: ~**,AA0,~AA2,~AA3EVV,~AA4N, ~AA5N

4.31 Read Host Watchdog timer Timeout Value

Modules:	**For 9024 module**	
Description:	Reads the host watchdog time out value of a module (see p69).	
Command:	~AA2[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	2	Command for reading watchdog timeout value
	CHK	Check sum
	(cr)	Carriage return
Response:	! AAEVV[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	E	Host watchdog Enable/Disable status = 0 - Disable = 1 -Enable
	VV	Timeout value in Hex format from 01 to FF(=25.5 seconds), one unit is 0.1 sec
	CHK	Check sum
	(cr)	Carriage return

Example :

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

Command: ~012 (cr)

Response: !01FF (cr)

Related command: ~**, AA0, ~AA1, ~AA3EVV, ~AA4N, ~AA5N

4.32 Set host watchdog timeout value

Modules:	**For 9024 module**	
Description:	Enables/disables the host watchdog and set the host watchdog time out value of a module (see p69).	
Command:	~AA3EVV[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	3	Command for setting watchdog timeout value
	E	Enables/disables the host watchdog: = 0 - disable = 1 - enable
	VV	Timeout value in Hex format from 01 to FF(=25.5 seconds), one unit is 0.1 sec
	CHK	Check sum
	(cr)	Carriage return
Response:	! AA [CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note:

If host watchdog timer is enabled, the host should send Host OK command periodically within Timeout value to refresh the timer, otherwise the module will be forced to safety state. (see p69)

Example:

(1) Set module (ID=04) to have watchdog timeout value 10.0 seconds and enable
host watchdog.
Command: ~043164(cr)
Response: !04 (cr)

(2) Read watchdog timeout value form module (ID=04) and return
watchdog timeout value=10.0 seconds, and host watchdog is enabled.
Command: ~042(cr)
Response: !04164 (cr)

(3) Host OK Command: ~**(cr)

Stop sending any command string to modules for at least 10.0 seconds. The LED on the module will go to flash. The flash LED indicates the host watchdog is timeout and timeout status is set. ;wait…………. ;

(4) Read watchdog timeout status, and return Timeout status is set

Command: ~040(cr)

Response: !0484(cr)

(5) Reset the host watchdog time out counter register of module 04 and returns a valid response.

Command: ~041<cr>

Response: !01

Related command: ~**, ~AA0, ~AA1, ~AA2, ~AA4N, ~AA5N

4.33 Read Safe Value

Modules:	**For 9024 module**	
Description:	Read back safe value for watchdog timeout and Emergency input. (see p69)	
Command:	~AA4N[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	4	command for Read Safe Value
	N	channel number (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	(data)	Safe Value of module. (see p17 for format)
	CHK	Check sum
	(cr)	Carriage return

Example :

Read address 01 channel 2 Safe Value, return +02.500 Safe Value

Command: ~0142 (cr)

Response: !01+02.500(cr)

Related command: ~**, AA0, ~AA1, ~AA2, ~AA3EVV, ~AA5N

4.34 Set Safe Value

Modules:	For 9024 module	
Description:	Command for setting safe value for watchdog timeout and emergency Input, store the current output value as safe Value. (see p69)	
Command:	~AA5N[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	5	command for set Safe Value
	N	channel number (0 to 3)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA [CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	CHK	Check sum
	(cr)	Carriage return

Example :

(1) Output address 01 value +01.000V for channel 2, return success.

Command: #012+01.000

Response: > (cr)

(2) Set address 01 channel 2 Safe Value, return success.

Command: ~0152(cr)

Response: !01 (cr)

(3) Read address 01 channel 2 Safe Value, return +01.000 Safe Value

Command: ~0142 (cr)

Response: !01+01.000(cr)

Related command: ~**, AA0, ~AA1, ~AA2, ~AA3EVV, ~AA4N

4.35 Enable/Disable Emergency Input(DI)

Modules:	**For 9024 module**	
Description:	Enable/Disable Emergency Input(DI)	
Command:	~AA8NE [CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	8	command for Set Emergency DI Inputs
	N	channel number (0 to 3)
	E	Enables/disables emergency inputs(DI): = 0 - disable = 1 - enable
	CHK	Check sum
	(cr)	Carriage return
Response:	! AA [CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note:

(1) When a emergency input(DI) is active(low), the module will be forced to safety output state for channel N of A/O.

(2) If disable is selected then emergency input(DI) same as standard digital input. (see 4.27)

Example

(1) Output address 04 value +01.000V for channel 1, return success.
Command: #041+01.000
Response: > (cr)

(2) Set address 04 channel 1 Safe Value, return success.
Command: ~0451(cr)
Response: !01 (cr)

(3) Set module (ID=04) to enable channel(1) emergency inputs.
Command: ~04181 (cr)
Response: !04 (cr)

(4) Read module status from module (ID=04) and return the channel(1) emergency Input is enable.

Command: ~040(cr)

Response: !0410 (cr)

wait………

; The modules emergency Input(DI) channel(1) is active(low) and the AO channel N of the module is into safe output mode.

;

(5) Read Emergency Input port status from module (ID=04) and return the emergency input channel(1) is active(low).

Command: @040(cr)

Response: !040D(cr)

(6) Output address 04 value +01.000V for channel 1, return emergency input

is active and the output command will be ignored.

Command: #041+01.000

Response: ! (cr)

Related command: ~AA0, @AA, ~AA5N, #AAN,

5. Modbus RTU Protocol Command Sets

5.1 Introduction

The Modbus protocol is developed by Modicon Inc., originally developed for Modicon controllers. Detailed information can be found at http://www.automation.com.tw/ to find more valuable information. E-9000 series modules support the Modbus RTU protocol. The communication BaudRates rang from 1200bps to 115200bps. The parity, data bits and stop bits are fixed as no parity, 8 data bits and 1 stop bit. The following Modbus functions are supported.

5.2 MODBUS Data model

MODBUS bases its data model on a series of tables that have distinguishing characteristics.
The four primary tables are:

Primary	Object type	Type of	Comments
Discretes Input	Single bit	Read-Only	This type of data can be provided by an I/O system.
Coils	Single bit	Read-Write	This type of data can be alterable by an application
Input Reg.	16 bit word	Read-Only	This type of data can be provided by an I/O system
Holding Reg.	16 bit word	Read-Write	This type of data can be alterable by an application program.

5.3 MODBUS function code definition:

Function Code	Description
01 (0x01)	Read coils
02 (0x02)	Read Discrete Inputs
03 (0x03)	Read multiple Holding registers
04 (0x04)	Read multiple input registers
05 (0x05)	Write single coil
06 (0x06)	Write single register
15 (0x0F)	write Multiple coils
16 (0x10)	Write Multiple register
70 (0x46)	Read / write module settings

Error Response:

If the function specified in the message is not supported, then the module Response as follows:

Offset	Function	Length	Description
00	Address	1 Byte	1 to 247
01	Function code	1 Byte	Function code | 0x80
02	Exception code	1 Byte	01

If a CRC mismatch occurs, the module will not respond. (ref. "02208")

5.4 MODBUS Standard Register Designation

0xxxx	- Coils access,	(0x01, 0x05, 0x0F function code)
1xxxx	-Read discrete inputs,	(0x02 function code)
3xxxx	-Read input register,	(0x04 function code)
4xxxx	-Holding register access,	(0x03, 0x06, 0x10 function code)

xxxx - Element address of a data block, In the MODBUS data model each element within a data block is numbered from 1 to n.

Example:

00005	- Coils access and Starting Address = 0004 (0005-1)
10002	-Discrete inputs and Starting Address = 0001 (0002-1)
30257	-Input register and Starting Address = 0256 (0257-1)
40001	-Access holding register and Starting Address = 0000 (0001-1)

5.5 Modbus Address Mapping Table

There are two categories of jDAM-9024 commands. The first is the data access command sets, and The second is the Read/Write Module Command Sets. All the commands used in the JDAM analog output module are list in the following table.

5.5.1 Summary of command Sets

Address Mapping			
Address	**Item**	**Attribute**	**Sec.**
00257	Protocol, JDAM ASCII & Modbus select. For function (0x01) : = 1 - Modbus RTU For function (0x05) : 0xFF00 = Modbus RTU 0x0000 = JDAM ASCII For function (0x0F) : 1 = Modbus RTU.	R/W	5.5.2.1
00261	Host watchdog enable/disable For function (0x01) : = 1 - enable For function (0x05) : 0xFF00 = enable For function (0x0F) : 1 = enable.	R/W	5.5.2.1
00270	Host watchdog timeout status For function (0x01) : = 1 - set For function (0x05) : 0xFF00 = clear For function (0x0F) : 1 = clear	R/W	5.5.2.1
00272	Load factory calibration parameters For function (0x05) : 0xFF00 = enable For function (0x0F) : 1 = enable.	W	4.15
02208	CRC checking enable / disable(default) For function (0x01) : = 1 - enable For function (0x05) : 0xFF00 = enable For function (0x0F) : 1 = enable.	R/W	
02210	Reset the module to initial power-on status For function (0x05) : 0xFF00 = enable For function (0x0F) : 1 = enable.	W	4.16
00273	Read module reset status = 1 - first read after powered on = 0 - not the first read after powered on	R	4.7
02241	Enable/Disable DI0 Emergency input flag. For function (0x01) : 1 = enable For function (0x05) : 0xFF00 = enable For function (0x0F) : 1 = enable.	R/W	5.5.2.1
02242	Enable/Disable DI1 Emergency flag	R/W	5.5.2.1
02243	Enable/Disable DI2 Emergency flag	R/W	5.5.2.1
02244	Enable/Disable DI3 Emergency flag	R/W	5.5.2.1

Address Mapping			
Address	**Item**	**Attribute**	**Sec.**
10033	Read Emergency DI0 Input channel. 1 = input high level (ON) 0 = input low level (OFF)	R	5.5.2.1
10034	Read Emergency DI1 Input channel.	R	5.5.2.1
10035	Read Emergency DI2 Input channel.	R	5.5.2.1
10036	Read Emergency DI3 Input channel..	R	5.5.2.1

Address Mapping			
Address	**Item**	**Attribute**	**Sec.**
40001	Current analog output value of channel 0 Error Response (offset 02): = 0x00 - valid command = 0x02 - invalid start address = 0x03 - invalid data value = 0x04 - host WDT timeout = 0x05 - return if Emergency DI flag is active	R/W	5.5.2.1
40002	Current analog output value of channel 1	R/W	5.5.2.1
40003	Current analog output value of channel 2	R/W	
40004	Current analog output value of channel 3	R/W	
40193	Power on analog output value of channel 0	R/W	5.5.2.1
40194	Power on analog output value of channel 1	R/W	
40195	Power on analog output value of channel 2	R/W	
40196	Power on analog output value of channel 3	RW	
43001	Safe value of analog output channel 0	R/W	5.5.2.1
43002	Safe value of analog output channel 1	R/W	
43003	Safe value of analog output channel 2	R/W	
43004	Safe value of analog output channel 3	R/W	
40257	Type code channel(0)	R/W	3.11.2
40258	Type code channel(1)	R/W	
40259	Type code channel(2)	R/W	
40260	Type code channel(3)	R/W	
40289	slew rate control for channel 0	R/W	3.11.5
40290	slew rate control for channel 1	R/W	
40291	slew rate control for channel 2	R/W	
40292	slew rate control for channel 3	R/W	
40489	Host watchdog timeout value (0~255, in 100ms)	R/W	

40492	Host watchdog timeout count, write 0 to clear	R/W	
42202	Write data to number 2 of Display LED	W	
42203	Write data to number 3 of Display LED	W	
42204	Write data to number 4 of Display LED	W	
42205	Write data to number 5 of Display LED	W	
42206	Write data to number 6 of Display LED	W	
40481	Firmware version (low word)	R	
40482	Firmware version (high word)	R	
40483	Module name (low word)	R	
40484	Module name (high word)	R	
40485	Module address, valid range: 1 ~ 247	R/W	
40486	Baudrate setting(CC)	R/W	3.11.1
40488	Modbus response delay time in ms (0~30ms)	R/W	

Note: The "Function(0xnn)" indicate as modbus-RTU function code number.

5.5.2 Example of modbus RTU commands

5.5.2.1 Protocol, JDAM ASCII & Modbus select

◆ (00257) Read protocol and return modbus RTU is select.

Request : 01 01 01 00 00 01

Reponse: 01 01 01 01

◆ (00257) Set to JDAM ASCII protocol, return valid.

return valid.

Request : 01 05 01 00 00 00

Reponse: 01 05 01 00 00 00

◆ (00257) Read protocol and return JDAM ASCII is select

Request : 01 01 01 00 00 01

Reponse: 01 01 01 00

5.5.2.2 Host watchdog timeout operation

- (43001) Set +1.600V as safe output value of channel(0)
  return valid.   (see p16 data format)
  Request : 01 06 0B B8 0A 42
  Reponse: 01 06 0B B8 0A 42

- (40489) Write host watchdog timeout value(20 sec) return valid
  Request : 01 06 01 E8 00 C8
  Reponse: 01 06 01 E8 00 C8

- (00261) Set host watchdog timeout enable, return valid.
  Request : 01 05 01 04 FF 00
  Reponse: 01 05 01 04 FF 00

- (40001) Write +7.373V to analog output of channel(0),
  return valid.   (see p16 data format)
  Request : 01 06 00 00 2F 34
  Reponse: 01 06 00 00 2F 34
  ;
  wait 15 sec……
  ;

- (00270) Clear host watchdog timeout counter, return watchdog timeout is set.
  Request : 01 05 01 0D FF 00
  Reponse: 01 05 01 0D FF 00
  ;
  wait 25 sec……
  watchdog timeout and into safe output mode
  ;

- (00270) Read host watchdog timeout status, return host watchdog timeout flag is set.
  Request : 01 01 01 0D 00 01
  Reponse: 01 01 01 01

- (40001) Write +7.373V to analog output of channel(0), return host watchdog timeout occur. (see p16 data format)
  Request : 01 06 00 00 2F 34
  Reponse: 01 86 04

- (00270) Clear host watch dog timeout status, return host watchdog timeout flag is set.
  Request : 01 05 01 0D FF 00
  Reponse: 01 05 01 0D FF 00

- (40001) Write +7.373V to analog output of channel(0), return valid. (see p16 data format)
  Request : 01 06 00 00 2F 34
  Reponse: 01 06 00 00 2F 34

5.5.2.1 Emergency input active and into Safe output modet

- (43001) Set +1.600V as safe output value of channel(0) return valid. (see p16 data format)
  Request : 01 06 0B B8 0A 42
  Reponse: 01 06 0B B8 0A 42

- (02241) Enable emergency input of DI0, return valid.
  Request : 01 05 08 C0 FF 00
  Reponse: 01 05 08 C0 FF 00

- (40001) Write +7.373V to analog output of channel(0), return valid. (see p16 data format)
  Request : 01 06 00 00 2F 34
  Reponse: 01 06 00 00 2F 34

;

Emergency input of DI0 is active(low) and analog output of channel(0) into safe output mode

;

- (10033) Read emergency input of DI(0) status. return DI0 is active(low).
  Request : 01 02 00 20 00 04
  Reponse: 01 02 01 0E

- (40001) Write +7.373V to analog output of channel(0), return Emergency input of DI0 is active(low).
  Request : 01 06 00 00 2F 34
  Reponse: 01 86 05

- (10033) Read emergency input of DI(0) status. return DI0 is inactive(high).
  Request : 01 02 00 20 00 04
  Reponse: 01 02 01 0F

- (40001) Write +7.373V to analog output of channel(0), return valid. (see p16 data format)
  Request : 01 06 00 00 2F 34
  Reponse: 01 06 00 00 2F 34

6. Appendix A INIT* pin operation

The INIT* mode has two purposes, one for reading module current configuration, and another for configuring the module baud rate and checksum.

■ Reading module current configuration Each JDAM module has a built-in

EEPROM which is used to store the configuration information such as address ID, type, baud rate etc..

If the user unfortunately forget the configuration of the module. User may use a special mode called INIT* mode to resolve the problem. When the module is set to INIT* mode, the default settings are ID=00, baud rate=9600, and no checksum.

The following steps show you how to enable INIT* mode and read the current configuration:

Step1. Power off the module.
Step2. Connect the INIT* pin to GND pin.
Step3. Power on the module.
Step4. Send command $002<cr> in 9600 baud rate to read the current configuration stored in the EEPROM.
Step5. Power off the module again.
Step6. Open INIT* pin to force the module to normal mode.

■ Configuring the module baud rate and checksum

The module should be set to INIT* mode", While changing baud rate and/or checksum state by sending "Set module configuration" command.

The following steps show you how to enable INIT* mode and change baud rate and/or checksum state:

Step1. Power off the module
Step2. Connect the INIT* pin to GND pin
Step3. Power on the module
Step4. Send command %AANNTTCCFF in 9600 baud rate to set baud rate and/or checksum state (ID should be set to 00 in INIT* mode)
Step5. Power off the module again
Step6. Open INIT* pin to force the module to normal mode

7. Appendix B Calibration

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

- The current calibration procedure is as follows :

1. Connect meter and external power source to module's current output channel N.

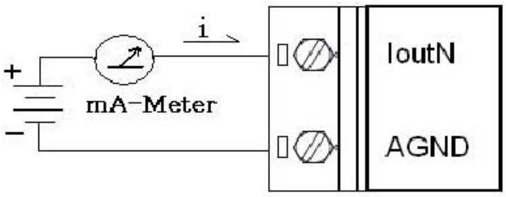


Figure 3.4

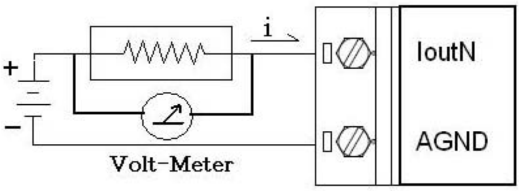


Figure 3.5

2. Warm up the module for 30 minutes.
3. Setting type to 30 (0 to 20mA) by command "$AA9NTTSS" (see p46)
4. Output 0mA by analog output command "#AAN(data)" (see p37)
5. Check the meter and trim the output until 0mA match by apply trim command "$AA3NVV" (see p40)
6. Repeat step(5) for trim calibration.
7. Perform 0mA calibration command for save min. calibration parameter. by command "$AA0N" (see p38)
8. Output 20mA by analog output command "#AAN(data)" (see p37)
9. Check the meter and trim the output until 20mA match by apply trim command "$AA3NVV" (see p40)
10. Repeat step(9) for trim calibration.
11. Perform 20mA calibration command for save min. calibration parameter. by command "$AA1N" (see p39)
12. Repeat steps 4 to 11 three times.

- The voltage calibration procedure is as follows :

1. Connect meter to module’s Voltage output channel N.

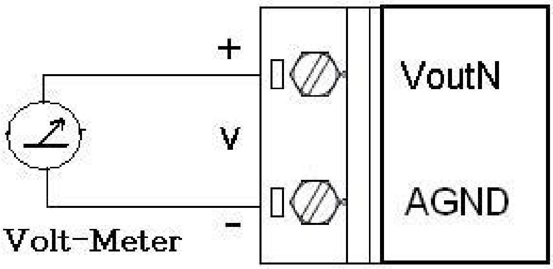


Figure 3.6

2. Warm up the module for 30 minutes.
3. Setting type to 33 (-10V to +10V) by command "$AA9NTTSS" (see p46)
4. Output -10V by analog output command "#AAN(data)" (see p37)
5. Check the meter and trim the output until -10V match by apply trim command "$AA3NVV" (see p40)
6. Repeat step(5) for trim calibration.
7. Perform -10V calibration command for save min. calibration parameter. by command "$AA0N" (see p38)
8. Output +10V by analog output command "#AAN(data)" (see p37)
9. Check the meter and trim the output until +10V match by apply trim command "$AA3NVV" (see p40)
10. Repeat step(9) for trim calibration.
11. Perform +10V calibration command for save min. calibration parameter by command "$AA1N" (see p39)
12. Repeat steps 4 to 11 three times.

Note: Reloads the factory default calibration parameters and clear user calibration by command "$AAS1" (see p35)

8. Appendix C Module Status

Power-On Reset or Module Watchdog Reset will let all output goto Power-On Value or Safe Value. And the module may accept the host’s command to change the output value. Host Watchdog Timeout will let all analog output goto Safe Value. The host watchdog timeout flag is set, and the output command will be ignored. The module’s LED will go to flash and user must reset the Module Status via command to restore normal operation.

9. Appendix D Dual Watchdog Operation

Dual Watchdog = Module Watchdog + Host Watchdog

The Module Watchdog is a hardware reset circuit to monitor the module’s operating status. While working in harsh or noisy environment, the module may be down by the external signal. The circuit may let the module to work continues and never halt. The Host Watchdog is a software function to monitor the host’s operating status. Its purpose is to prevent the network/communication from problem or host halt. While the timeout occurred, the module will turn the all output into safe state to prevent from unexpected problem of controlled target. The E-9000 module with Dual Watchdog may let the control system more reliable and stable.