GPS-721-MRTU
User Manual v2.0

High-quality Industrial Data Acquisition and Control Products

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 assumes no liability for damages resulting from 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

Trademarks

Names are used for identification purposes only and may be registered trademarks of their respective companies.

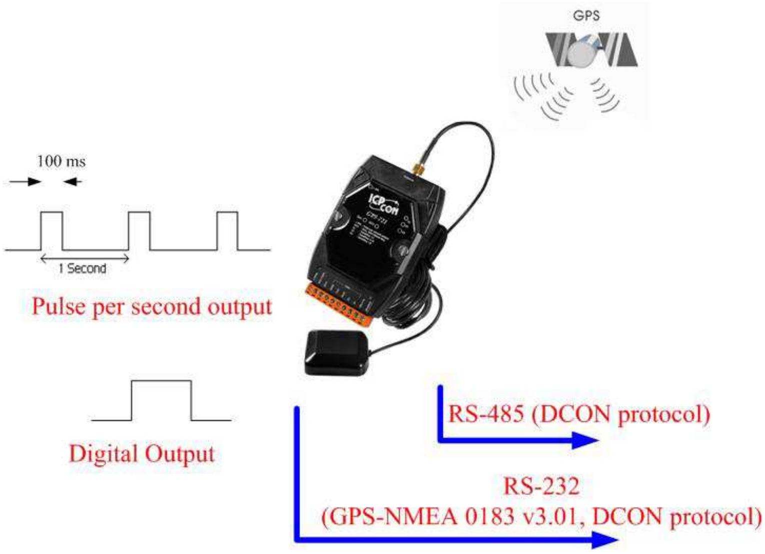
Table of Contents

1 Introduction

ICP DAS data acquisition and control modules provide analog-to-digital, digital-to-analog, digital input/output, timer/counter and other functions. These modules can be remotely controlled by using a set of commands, called the DCON protocol, in addition to standard Modbus RTU protocol commands. DCON protocol commands are written in ASCII format and Modbus RTU commands are written in hexadecimal format. Communication between the module and the Host is via a standard RS-485 bidirectional serial bus. Baud Rates are software programmable and transmission speeds of up to 115.2 Kbps can be selected.

The GPS-721-MRTU module ensures high sensitivity and low power consumption with an ultra-small form factor, and also supports the NMEA 0183 GPS protocol. The embedded GPS module is powered by a MediaTek solution and can provide superior performance, even in an urban environment, or an environment that features dense foliage. In addition to the ability to acquire GPS data, the GPS-721-MRTU includes a single Digital Output channel and a single PPS (pulse per second) channel that can be used for precise timekeeping and time measurement. PPS functionality can be combined with an additional external time source in order to synchronize signal operations, and also provides the full date and time for accurate and precise applications.

Consequently, the GPS-721-MRTU can act as a powerful remote GPS module as well as a general purpose GPS module with an RS-232 interface. It can also be applied to applications related to automotive navigation, personal positioning and navigation, marine navigation, and satellite time correction. etc.

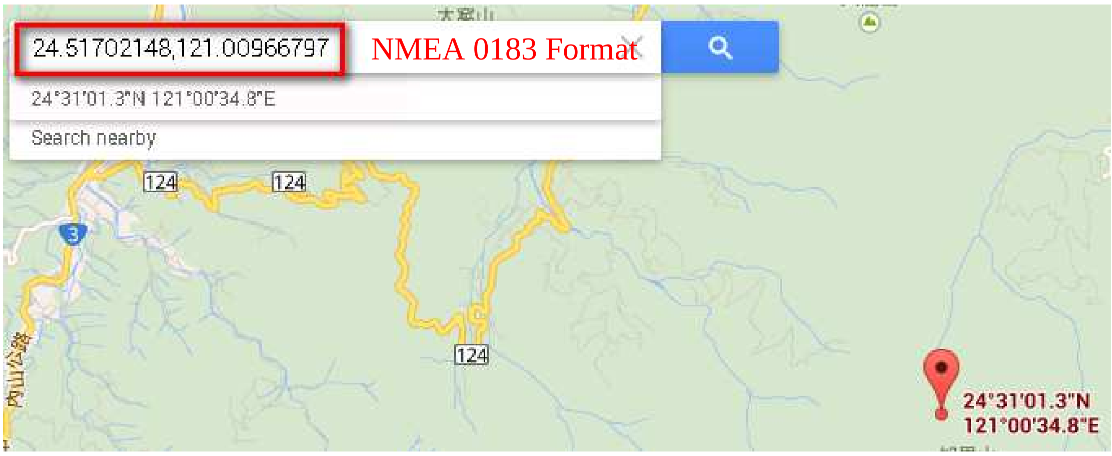


ICP DAS now also provides an option that allows NMEA 0183 format GPS data to be converted to Google Maps format. This feature is available for both the DCON and Modbus protocols from firmware version 2.10 or later.

Previously, incorrect GPS information would be displayed in Google Maps if NMEA 0183 format data was used without prior conversion. The following is a brief example.

Previously, if the GPS information displayed by the GPS-721-MRTU was entered into Google Maps without being converted beforehand, the location displayed would be incorrect because the data was entered in NMEA 0183 format. The following is a brief example.

In this example, GPS information for the ICP DAS Headquarters is used. If the values 24.51702148,121.00966797 are entered into Google Maps, an incorrect location will be displayed because the latitude and longitude values are in NMEA 0183 format, and will be shown as 24°31’01.3”N, 121°00’34.8”E, as illustrated below:



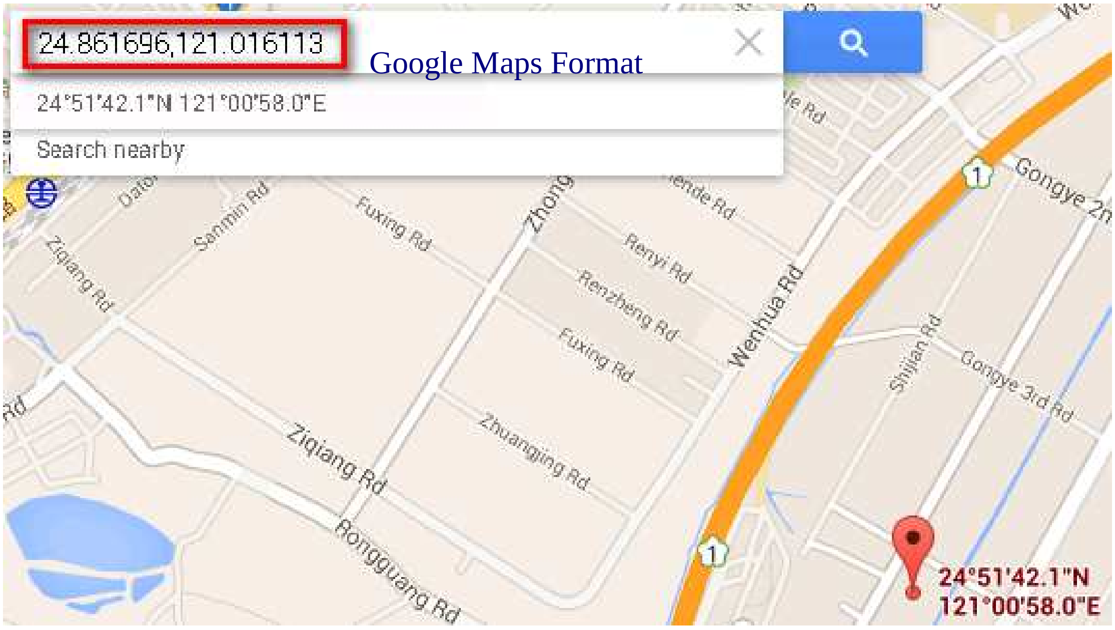
24.51702148,121.00966797

NMEA 0183 Format

24°31'01.3"N 121°00'34.8"E

Search nearby

However, after conversion, if the GPS information for the ICP DAS Headquarters is again entered in Google Maps format, 24.861696,121.016113 in this case, the correct location will be displayed and the latitude and longitude will be shown as 24°51’42.1”N, 121°00’58.0”E, as illustrated below:

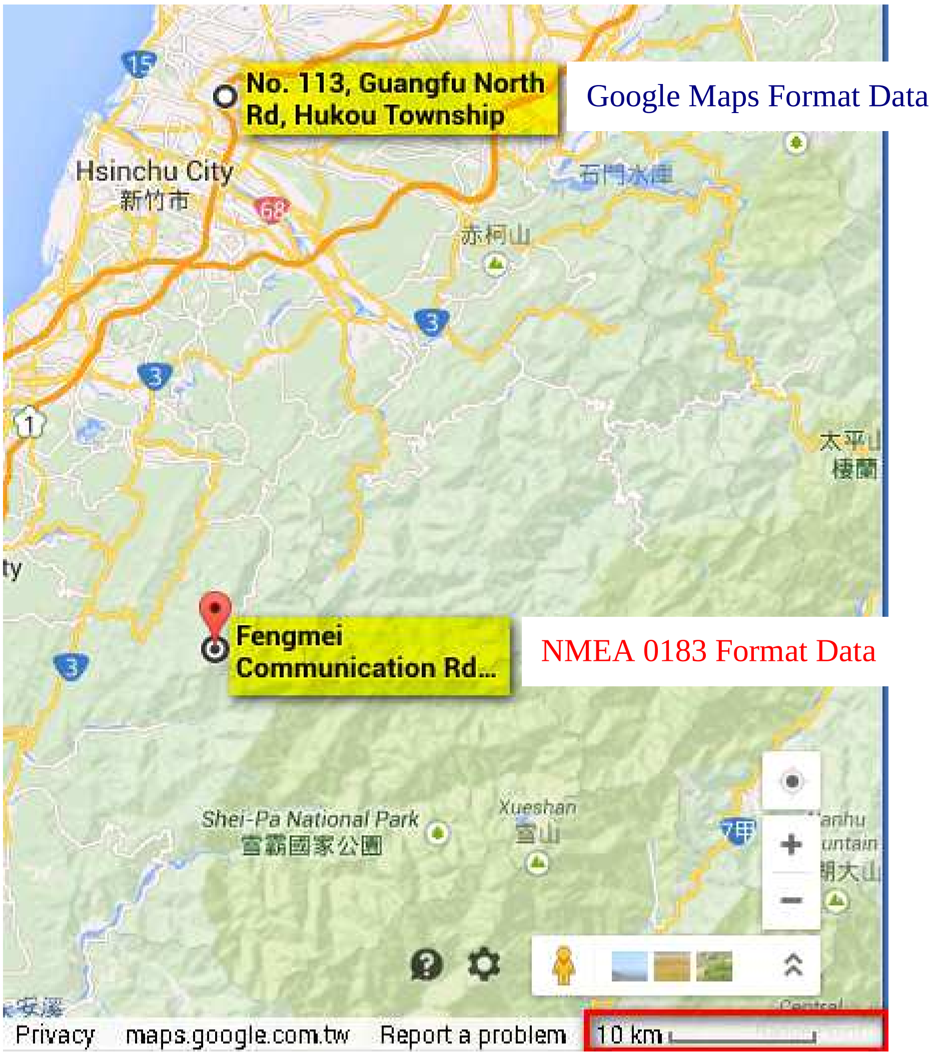


24.861696,121.016113

24°51'42.1"N 121°00'58.0"E

Search nearby

As can be seen from the image below, there is a significant difference in the position of these two:



No. 113, Guangfu North Rd, Hukou Township
Google Maps Format Data
Hsinchu City
新竹市
石門水庫
赤柯山
Fengmei Communication Rd...
NMEA 0183 Format Data
Shei-Pa National Park
雪霸國家公園
Xueshan
雪山
Privacy
maps.google.com.tw
Report a problem
10 km

1.1 Features

- MediaTek High-sensitivity GPS Solution
- Supports 66-channel GPS and the NMEA 0183 v3.01 standard
- RS-485 supports both the DCON and the Modbus RTU protocols
- RS-232 supports the NMEA 0183 v3.01 format, together with the DCON or the Modbus RTU protocol.
- Embedded 1-channel Digital Output, 1-channel PPS (pulse per second), 1 RS-485 port and 1 RS-232 port
- PPS: 100 ms pulse output/second for precise timekeeping and time measurement
- +10 to +30 V_{DC} Reverse Polarity Protection
- System LED indicators
- DIN-Rail design

1.2 Applications

- Automotive navigation
- Personal positioning and navigation
- Marine navigation
- Satellite time correction
- Precise timekeeping and time measurement

1.3 Packing List

GPS-721-MRTU Module plus
ANT-115-03 External Antenna

Software Utility CD

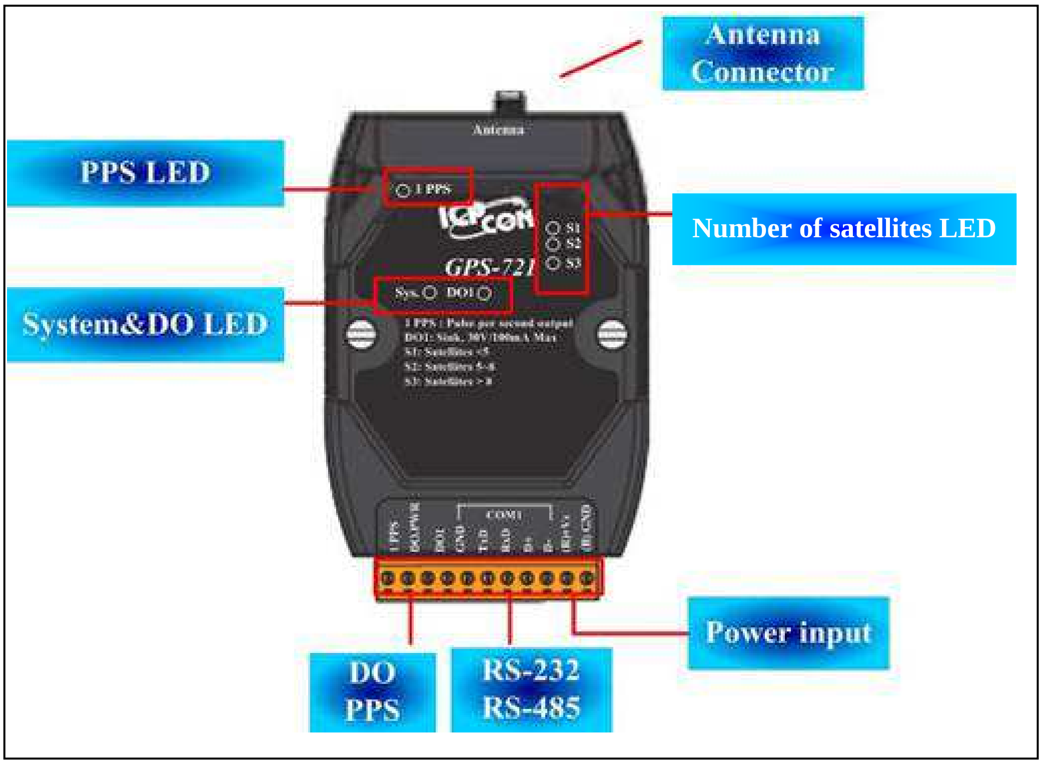
Quick Start Guide

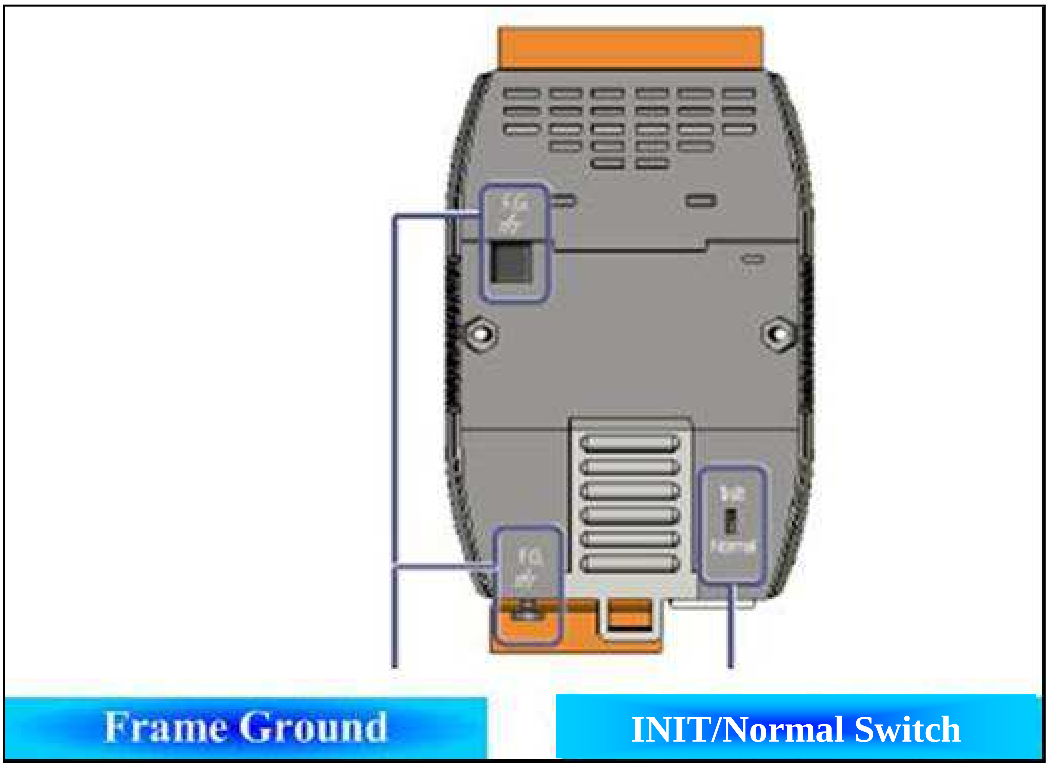
RS-232 Cable

Item	Quantity
GPS-721-MRTU Module plus External GPS antenna (ANT-115-03, Length: 5 m)	1
RS-232 Cable (CA-0910)	1
Software CD	1
Quick Start Guide	1

Ensure that all of these items are included in the shipping package for the GPS-721-MRTU module. If any of the items are missing or damaged, or if there are any other problems with the module, contact the supplier.

1.4 Hardware


Antenna Connector
Antenna
PPS LED
1 PPS
Number of satellites LED
S1
S2
S3
GPS-721
System&DO LED
Sys.
DO1
1 PPS : Pulse per second output
DO1: Sink, 30V/100mA Max
S1: Satellites <5
S2: Satellites 5-8
S3: Satellites > 8
1 PPS
DO.PWR
DO1
GND
TxD
RxD
D+
D-
(R)+Vs
(B)GND
COM1
Power input
DO
PPS
RS-232
RS-485


Frame Ground
INIT/Normal Switch

1.5 Specifications

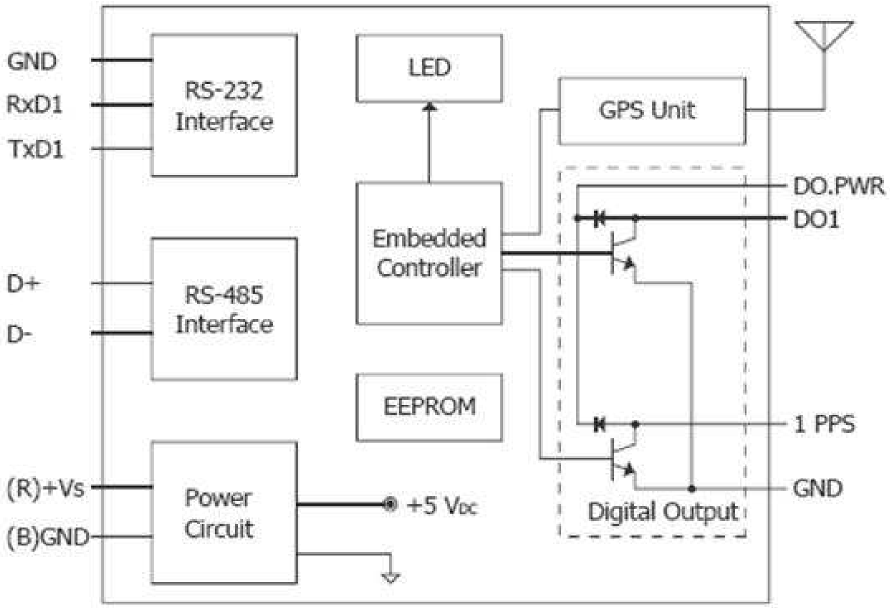
GPS Receiver	
Chip	MediaTek Solution
Frequency	L1 1575.42 MHz, C/A Code
Channels	32
Position Accuracy	Capable of SBAS (WAAS, EGNOS, MSAS)
Max. Altitude	< 18,000 m
Max. Velocity	< 515 m/s
Acquisition Time	Cold Start (Open Sky) = 42 s (Typical)
Sensitivity	Tracking = Up to -158 dBm Cold Start = Up to -142 dBm
Protocol	NMEA 0183 v3.01
GPS Antenna (Default)	
Length	5 m
Frequency Range	1575.42 MHz ±1.023 MHz.
Gain	30 dBi ±4.5 dBi (cable loss) Mounted on the 60 mm * 60 mm ground plane at 90°.
Output Impedance	50 Ω
VSWR	2.0 Max
GPS Output	
1 PPS	Pulse Per Second Output (Default 100 ms pulse/sec)
RS-232 Interface	For GPS Information Output
LED Indicators	
Power/Communication	1
GPS	3
Power	
Protection	Reverse Polarity
Frame Ground for ESD	Yes
Protection	
Required Supply Voltage	+10 V_{DC} to +30 V_{DC} (Non-regulated)
Power Consumption	2.5 W
Mechanical	

Dimensions (L x W x H)	117 mm x 72 mm x 35 mm
Weight	200 g
Housing	Plastic
Environment	
Operating Temperature	-25°C to +75°C
Storage Temperature	-40°C to +80°C
Humidity	5 to 95% RH, Non-condensing

I/O Specifications

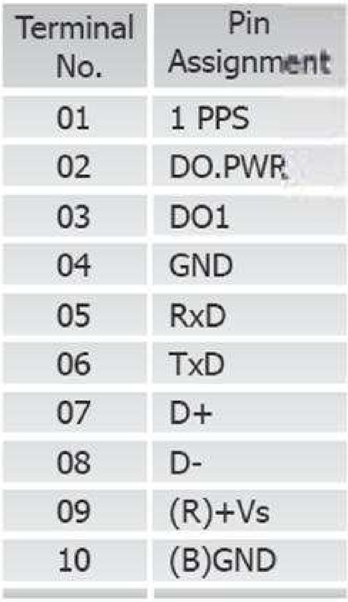
Digital Output	
Output Channels	1 (Sink)
Output Type	Non-isolated Open Collector
Output Current	100 mA
Load Voltage	+5 V_{DC} to +30 V_{DC}

1.6 Block Diagram for the GPS-721-MRTU

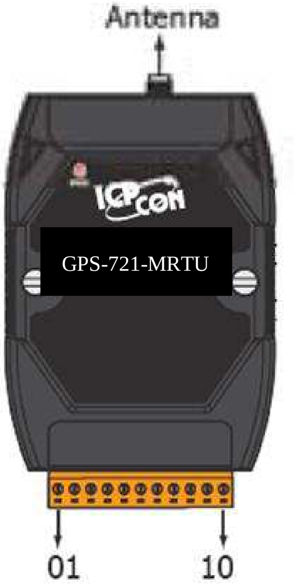


1.7 Pin Assignments and LED Indicators

Pin Assignments



Terminal No.	Pin Assignment
01	1 PPS
02	DO.PWR
03	DO1
04	GND
05	RxD
06	TxD
07	D+
08	D-
09	(R)+Vs
10	(B)GND

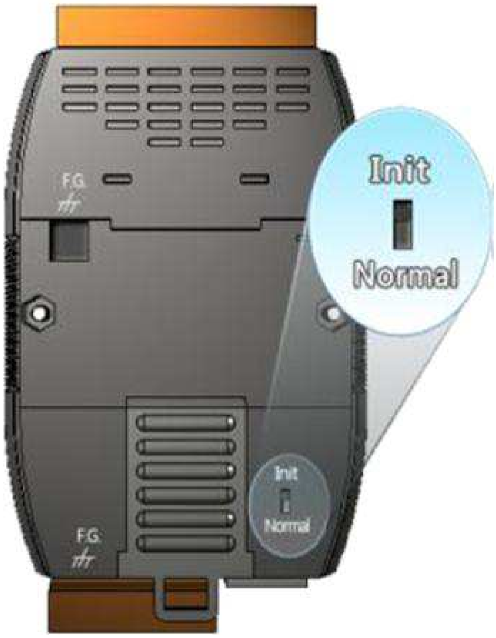


LED Indicators

LED	Status	Description
Sys.	ON	The module is operating/functioning correctly
	OFF	The module has encountered an error
DO1	ON	Digital Output is active
	OFF	Digital Output is inactive
S1	ON	Fewer than 5 signals received from GPS satellites.
	OFF	The module is unable to receive sufficient GPS signals
S2	ON	Between 5 and 8 signals received from GPS satellites
	OFF	Less than 5 signals received from GPS satellites
S3	ON	More than 8 signals received from GPS satellites
	OFF	Less than 8 signals received from GPS satellites
1 PPS	ON	The PPS signal is active
	OFF	The PPS signal is inactive

1.8 Operation Modes

The back panel of the GPS-721-MRTU module contains the frame ground and the INIT/Normal Switch. The module has two operation modes, INIT and Normal, which can be configured via the switch mechanism on the chassis. The following is a description of the two modes.



Mode	Description
INIT	After setting the module to "INIT" mode and restarting the module, a connection can be established using the default module address and communication settings, allowing new parameters for the module to be set. INIT Mode: ● Protocol: DCON ● Module Address: 00 ● Communication Baud Rate: 9600bps ● Checksum: Disabled Note: The DCON command to set the address, the Baud Rate and the checksum is %AANNTTCCFF. Refer to Section 2.1 for details.
Normal	In Normal mode, the module will operate based on the configured settings.

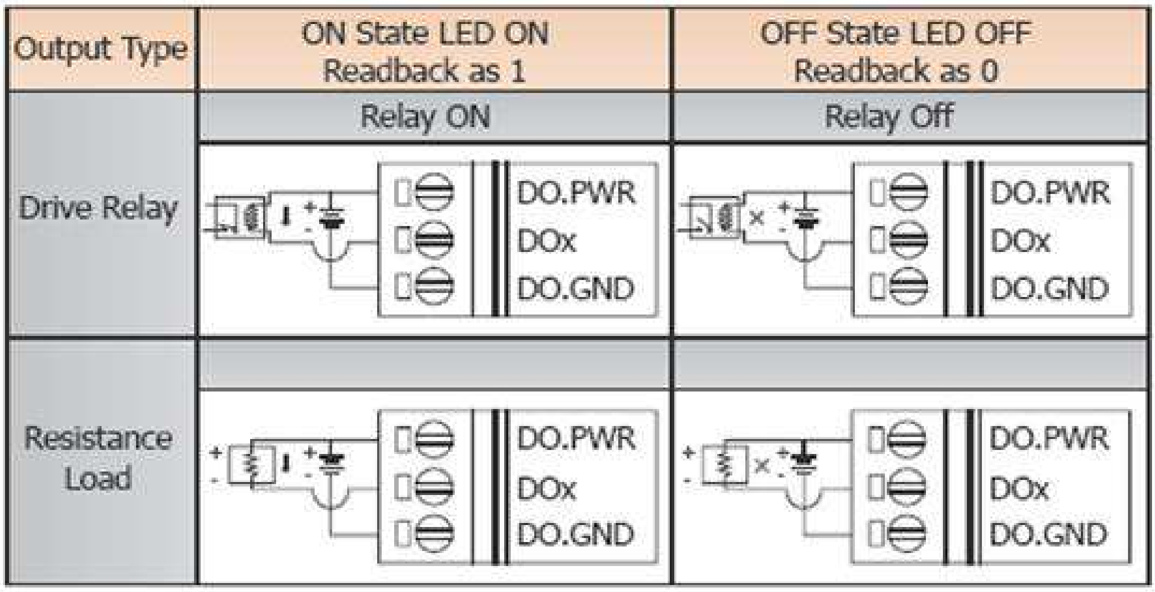
1.9 Wiring Recommendations

ICP DAS recommends that the following items are kept in mind when installing the GPS-721-MRTU module.

- Use a 26-12 AWG wire for signal connections
- Strip the wire to a length of 7 mm ±0.5 mm
- Use a crimp terminal for wiring
- Avoid high-voltage cables and power equipment as much as possible
- For RS-485 communication, use insulated and twisted pair 24 AWG wire, e.g., Belden 9841

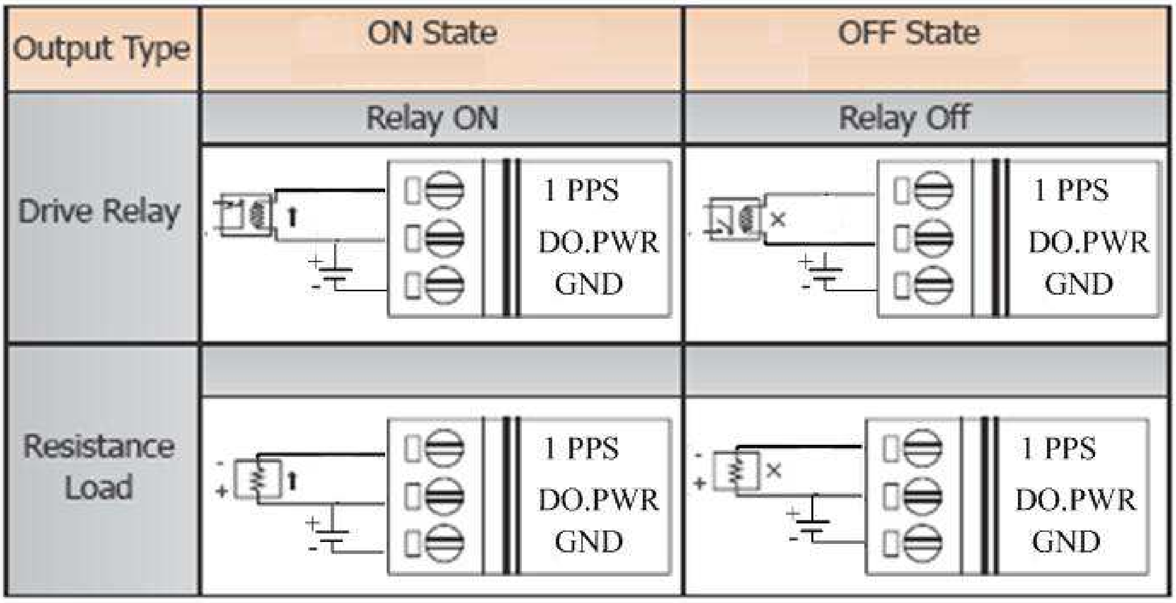
1.10 Wiring

- **Wiring for Digital Output**



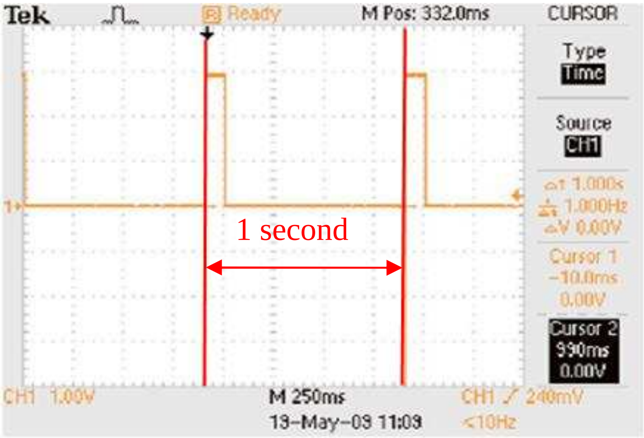
Output Type	ON State LED ON Readback as 1	OFF State LED OFF Readback as 0
Drive Relay	Relay ON	Relay Off
	DO.PWR DOx DO.GND	DO.PWR DOx DO.GND
Resistance Load		
	DO.PWR DOx DO.GND	DO.PWR DOx DO.GND

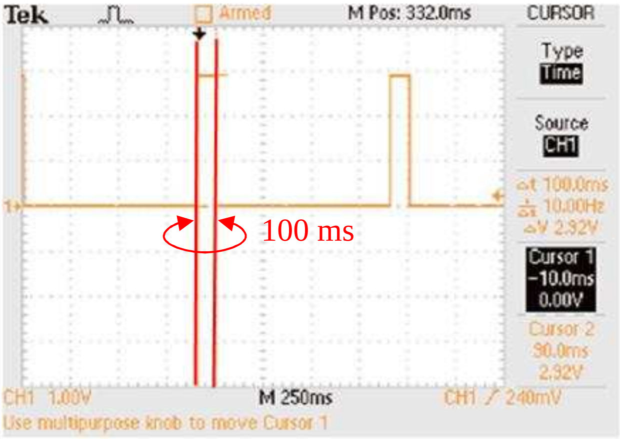
- **Wiring for PPS Output**



Output Type	ON State	OFF State
Drive Relay	Relay ON	Relay Off
	1 PPS DO.PWR GND	1 PPS DO.PWR GND
Resistance Load		
	1 PPS DO.PWR GND	1 PPS DO.PWR GND

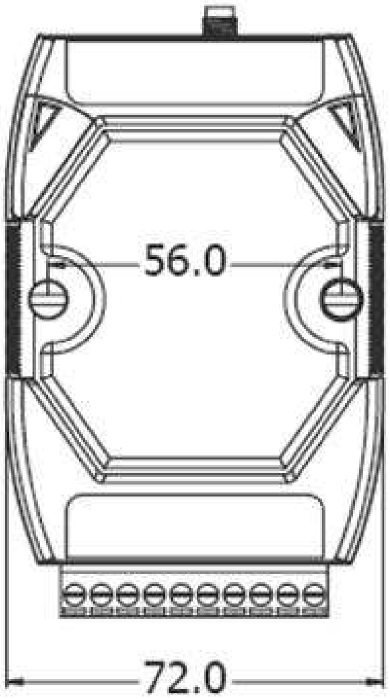
The PPS channel will output a 100 ms pulse each second when effective GPS signals are being received, as shown in the following figure.



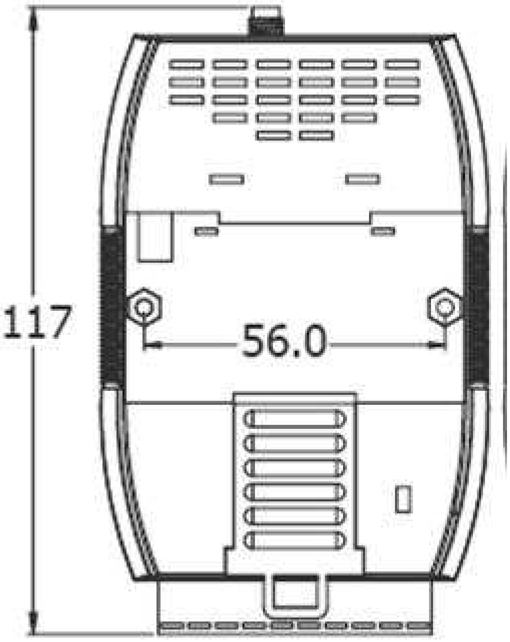


Note: A Pulse per Second (PPS) is an electrical signal that very precisely indicates the start of a second. PPS signals can be used for precise timekeeping and time measurement. PPS functionality can be combined with another time source that provides the full date and time in order to accurately and precisely ascertain the time.

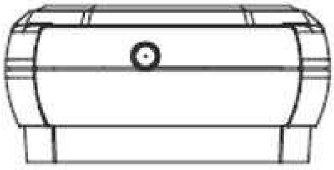
1.11 Dimensions (Units: mm)



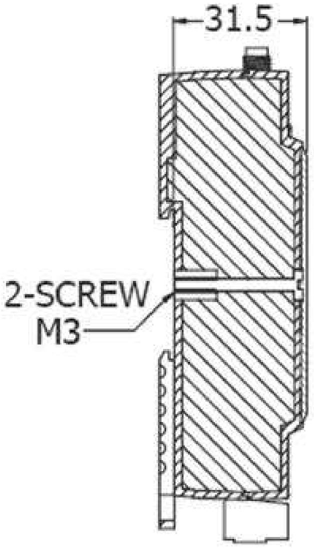
Front View



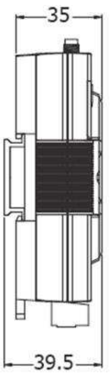
Rear View



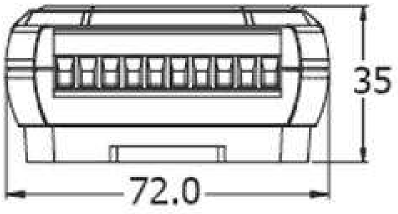
Top View



Side View



Din-Rail Mounting Bracket



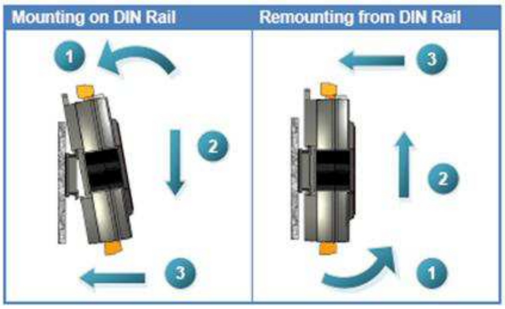
Bottom View

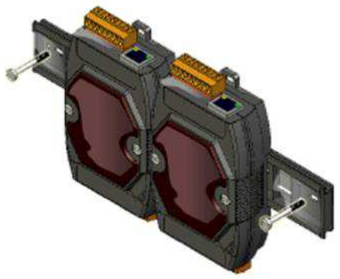
1.12 Mounting

The GPS-721-MRTU can be mounted on either a DIN-Rail or a wall, or it can be piggybacked to other modules using the bottom of the chassis.

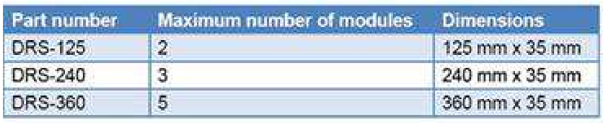
- **DIN-Rail Mounting**

  The GPS-721-MRTU includes simple rail clips that allow for reliable mounting on a standard 35 mm DIN-Rail.

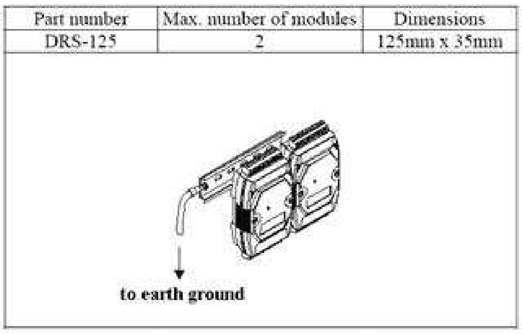




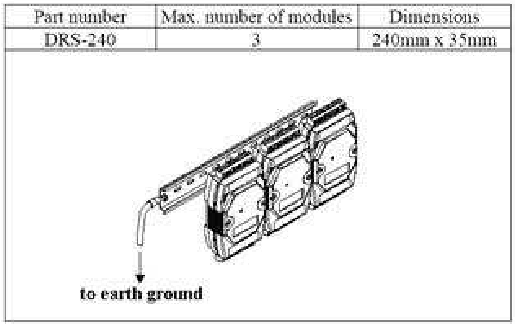
Three new DIN-Rail models are available, and each is made of stainless steel, which is stronger than those made of aluminum. There is a screw at one end, and a ring terminal is included so that it can be easily connected to the earth ground.



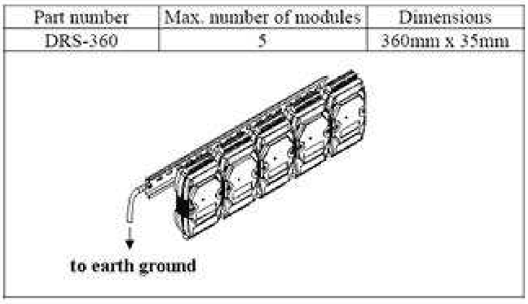
Part number	Maximum number of modules	Dimensions
DRS-125	2	125 mm x 35 mm
DRS-240	3	240 mm x 35 mm
DRS-360	5	360 mm x 35 mm



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



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

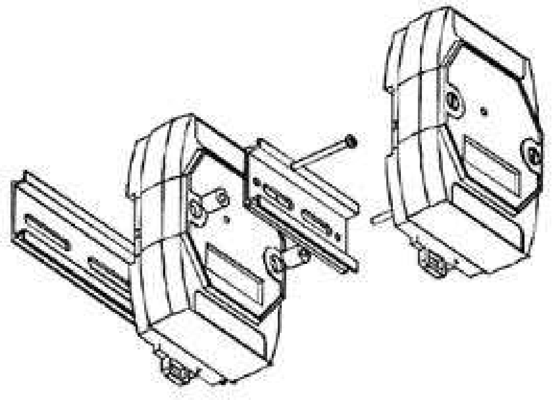


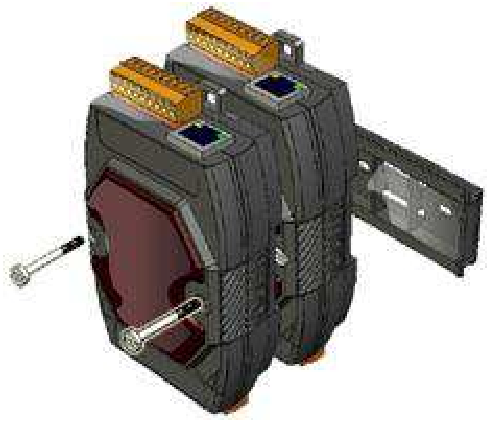
Part number	Max. number of modules	Dimensions
DRS-360	5	360mm x 35mm

Note: It is recommended that a 16 – 14 AWG wire is used to connect the DIN-Rail to the earth ground.

- **Piggyback Mounting**

The GPS-721-MRTU has two holes on either side of the module to enable piggyback mounting. Refer to the figure below for an illustration of how to piggyback the modules.

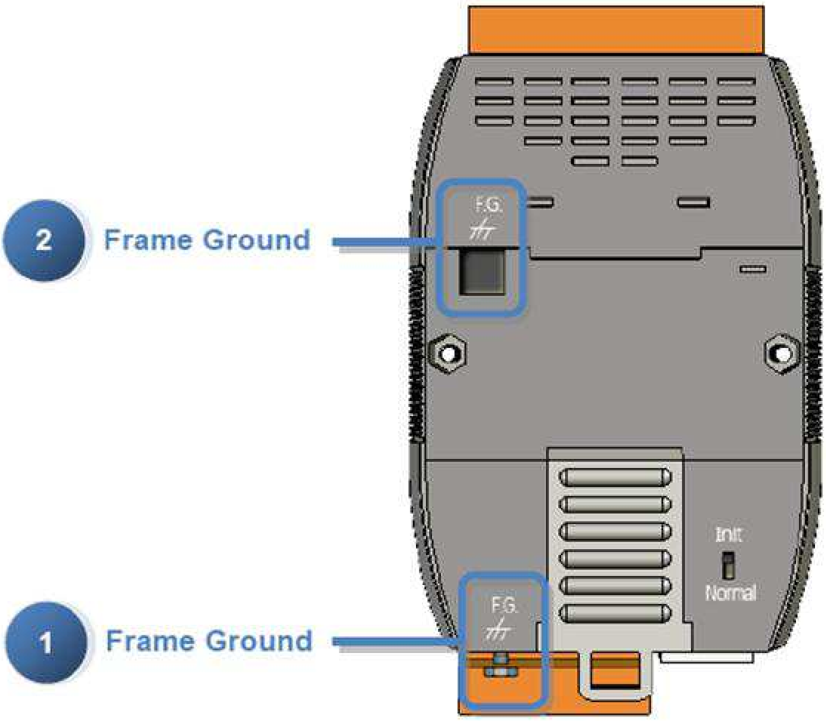




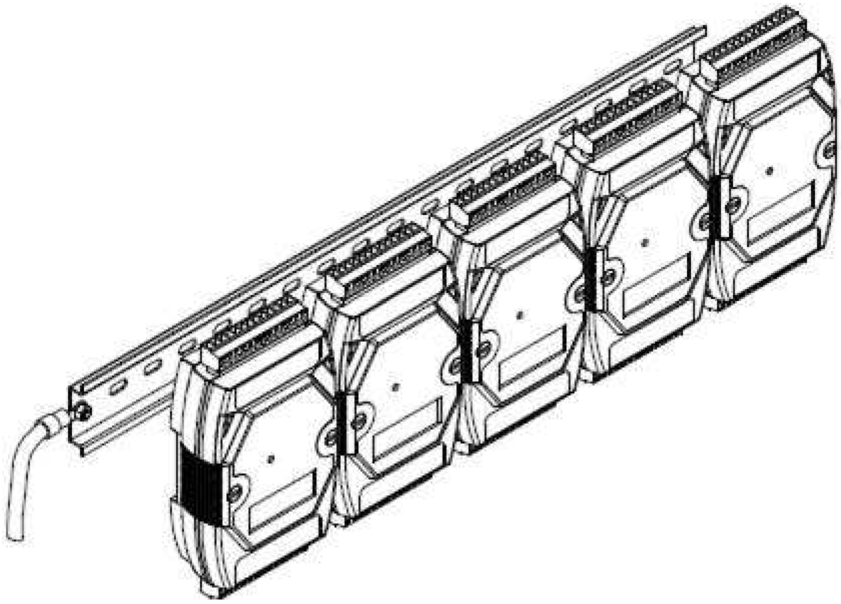
1.13 Frame Ground

Electronic circuits are constantly vulnerable to Electrostatic Discharge (ESD), which becomes worse in a continental climate area. Some I-7K and M-7K series modules feature a new design for the frame ground, which provides a path for bypassing the ESD, allowing enhanced static protection capabilities and ensuring that the module is more reliable. Either of the following options will provide better protection for the module:

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



New DIN-Rail models are available that can be easily connected to the earth ground. Each is made of stainless steel, which is stronger than those made of aluminum. There is a screw at one end, and a ring terminal is included, as shown in the figure below. Refer to Section 1.11 for more information about the new DIN-Rail models.



1.14 Configuration

- **Baud Rate Settings (CC)**

7	6	5	4	3	2	1	0
Parity		Baud Rate Code					

Parity (Bits 6 and 7)

Code	00	01	10	11
Parity	n,8,1	n,8,2	e,8,1	o,8,1

Baud Rate Code (Bits 0 to 5)

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

- **Type Code Settings (TT)**

The Type Code is set to 40 by default and cannot be changed.

- **Data Format Settings (FF)**

7	6	5	4	3	2	1	0
CU	CS	Reserved	GF	Reserved			

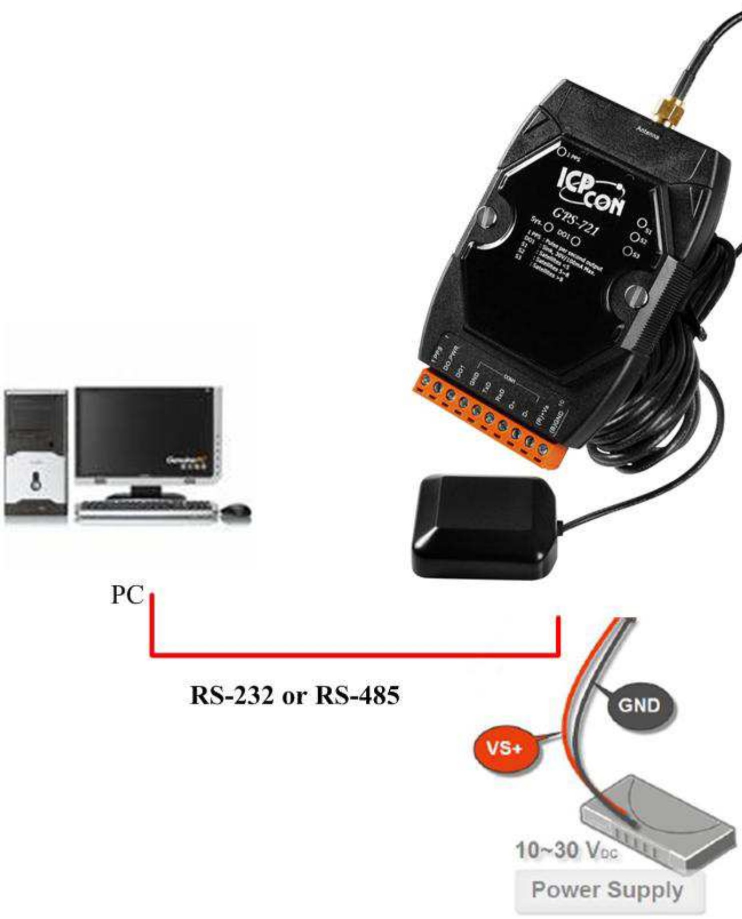
Key	Description
CS	The Checksum Settings 0: Disabled 1: Enabled
CU	The Counter Update Settings 0: The counter is updated when there is a falling edge in the input signal. 1: The counter is updated when there is a rising edge in the input signal.

GF	The format of the GPS information to be used for the #AA command 0: NMEA 0183 Format 1: Google Maps Format

Note: Reserved bits should be zero.

1.15 Quick Start

This section describes the installation process for the GPS-721-MRTU, and provides details of how to begin using the module.



1. Refer to Section 1.6 for details of the pin assignments.
2. Connect the module to the RS-485 network using the DATA+ and DATA- pins. See Section 1.6 for details of the pin assignments. The GPS-721-MRTU supports RS-232 connections, but if the Host is only equipped with an RS-232 interface, then an RS-232 to RS-485 converter will be required. Refer to the “I-7000 Bus Converter User Manual” for more information.

Default address and communication parameters:

●	**Protocol:**	**Modbus RTU**
●	**Module Address:**	**01**
●	**Communication Baud Rate:**	**9600 bps**

3. Configure the module by sending the %AANNTTCCFF command. See Section 2.1 for details.

   ICP DAS provides two free tools that allow the GPS-721-MRTU to be easily configured, the DCON Utility and the Send232 application, both of which can be downloaded from http://ftp.icpdas.com/pub/cd/8000cd/napdos/

4. All module functions can be implemented using the DCON commands described in Section 2.

5. If the Host is a PC with a Windows operating system installed, the DCON Utility can be used to allow easy configuration and reading of data. The DCON Utility can be downloaded from the ICP DAS website (http://www.icpdas.com), and documentation for the DCON Utility can be found in the ”Getting Started For I-7000 Series Modules” manual.

Note 1: The DCON Utility is provided to enable commands to be sent to the GPS-721-MRTU module. The graphic settings available on the GPS-721-MRTU module are not supported by the DCON Utility.

Note 2: If GPS-721-MRTU module is unable to receive a GPS signal, check the position of the antenna. If the problem persists, try repositioning the GPS antenna in an outdoor location.

1.16 Technical Support

Should you encounter any problems while using the GPS-721-MRTU module, and are unable to find help either in this manual or on our website, contact ICP DAS Product Support at.

Email: service@icpdas.com
Website: http://www.icpdas.com/sevices/support.htm

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

1. The Module name and serial number: The serial number can be found printed on the barcode label attached to the cover of the module.
2. The Firmware version: See Section 2.14 for information regarding the command used to identify the firmware version.
3. The Host configuration (type and operating system)
4. If the problem is reproducible, give full details, including a description of the procedure used to reproduce the problem.
5. Include details of any specific error messages displayed. If a dialog box with an error message is displayed, include the full text of the dialog box, including the text contained in the title bar.
6. If the problem involves other programs or hardware devices, describe the details of the problem in full.
7. Any comments and suggestions related to the problem are welcome.

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

2 DCON Protocol

All communication with the GPS-721-MRTU module consists of commands generated by the Host and responses transmitted by the module. Each module has a unique ID number that is used for addressing purposes and is stored in non-volatile memory. The module ID number is set to 00 by default and can be changed by sending the relevant user command. All commands to the module contain the ID number as the address, meaning that only the addressed module will respond. The only exception to this are commands #** (Section 2.2) and ~** (Section 2.22), which are sent to all modules, but, in both cases, the modules do not reply to the command.

Command Format:

Leading Character	Module Address	Command	[CHKSUM]	CR

Response Format:

Leading Character	Module Address	Data	[CHKSUM]	CR

CHKSUM A 2-character checksum that is present when the checksum option is enabled. See Section 1.13 (Data Format Settings) and Section 2 for details.

CR End of command character, carriage return (0x0D), used to end a frame.

Checksum Calculation:

1. Sum the ASCII codes of all the characters contained in the command/response string, except for the carriage return character (CR).
2. The checksum is equal to the sum value masked by 0FFh.

Example:

Command: $012(CR)

1. The sum of the string = “$”+”0”+”1”+”2” = 24h+30h+31h+32h = B7h

2. Therefore, the checksum is B7h, and so CHKSUM = “B7”

3. The DCON command string with the checksum = $012B7(CR)

Response: !01200600(CR)

1. The sum of the string = “!”+”0”+”1”+”2”+”0”+”0”+”6”+”0”+”0” = 21h+30h+31h+32h+30h+30h+36h+30h+30h = 1AAh

2. Therefore, the checksum is AAh, and so CHKSUM = “AA”

3. The DCON response string with the checksum = !01200600AA(CR)

Note: All characters should be expressed in upper case.

An Overview of the DCON Command Set

General Command Sets			
Command	**Response**	**Description**	**Section**
%AANNTTCCFF	!AA	Sets the Configuration of the Module	2.1
#**	No Response	Sends the Synchronized Sampling Command	2.2
#AA	!AA(Data)	Reads the UTC Time, Latitude, Longitude, and the number of GPS Satellite Signals available	2.3
#AAN	!AA(Data)	Individually reads the UTC Time, Latitude, Longitude, Date and the number of GPS Satellite Signals available	2.4

#AA00(Data)
#AA0A(Data)
	>	Sets the Value for the Digital Output Channel	2.5

#AAACDD	>	Sets the Digital Output for a Single Channel	2.6
#AA1CDD	>	Sets the Digital Output for a Single Channel	2.7
$AA2	!AANNTTCCFF	Reads the Configuration of the Module	2.8
$AA4	!S(Data)	Reads the Synchronized Sampling Data	2.9
$AA5	!AAS	Reads the Reset Status of the Module	2.10
$AA6	!AA(Data)	Reads the Status of the Digital Output Channel	2.11
$AAC	!AA	Clears the Status of the Latched Digital Output Channel	2.12
$AAD	!AA	Temporarily saves the Current Date	2.13
$AAF	!AA(Data)	Reads the Firmware Version of the Module	2.14
$AAI	!AA(Data)	Reads the Status of the INIT pin for the Module	2.15
$AALS	!AA(Data)	Reads the Status of the Latched Digital Output Channels	2.16
$AAM	!AA(Data)	Reads the Name of the Module	2.17
$AAP	!AASC	Reads the Communication Protocol currently being used by the Module	2.18
$AAPN	!AA	Sets the Communication Protocol to be used by the Module	2.19
@AA	>(Data)	Reads the Status of the Digital Output Channels	2.20

@AA(Data)	>	Sets the Value for all Digital Output Channels	2.21
~AAD	!AAF	Reads whether the Digital Output is Active or Inactive	2.23
~AADVV	!AA	Sets the Digital Output to Active	2.24
~AAI	!AA	Sets the Status of the Soft INIT function	2.25
~AAM	!AASC	Reads whether the Communication Mode for the RS-232 interface is NMEA 0183 or DCON	2.26
~AAMN	!AA	Sets the RS-232 Communication Mode for the RS-232 interface to either NMEA 0183 or DCON	2.27
~AAO(Name)	!AA	Sets the Name of the Module	2.28
~AARD	!AA(Data)	Reads the Response Delay Time	2.30
~AARDVV	!AA	Sets the Response Delay Time	2.31
~AATNN	!AA	Sets the Soft INIT Timeout Value	2.32

Host Watchdog Command Sets

Command	Response	Description	Section
~**	No Response	Host OK (The address in this command is zero and can be used to clear the Host Watchdog counter.)	2.22
~AAR	!AA	Enables the Date to be updated from a GPS message.	2.29
~AA0	!AASS	Reads the Status of the Host Watchdog	2.33
~AA1	!AA	Resets the Status of the Host Watchdog	2.34
~AA2	!AAVV	Reads the Timeout Settings for the Host Watchdog	2.35
~AA3EVV	!AA	Sets the Timeout Settings for the Host Watchdog	2.36
~AA4V	!AA(Data)	Reads the Power-on/Safe Value for the Host Watchdog	2.37
~AA5V	!AA	Sets the Power-on/Safe for the Host Watchdog Value	2.38

2.1 %AANNTTCCFF

Description:

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

Syntax:

%AANNTTCCFF[CHKSUM](CR)

% Delimiter character

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

NN The new address of the module in hexadecimal format (00 to FF)

TT The Type Code, which should be set to 40 for the GPS-721-MRTU module (Refer to Section 1.13 for details).

CC The new Baud Rate. Refer to Section 1.8 for details. Note that the Baud Rate will not be set unless the module is set to INIT mode.

FF The checksum and counter update direction settings of the module. See Section 1.13 for details.

Response:

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

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

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

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

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

Examples:

Command: %0109400600

Response: !09

Changes the address of module 01 to 09. The module returns a response indicating that the command was valid and includes the new address of the module.

Command: $092
Response: !09400600

Reads the configuration of module 09 and returns a response indicating that the command was valid, and showing that the Type Code is set to 40, the Baud Rate is 9600 bps, the Checksum is disabled and that the counter edge is falling.

Command: %0905400940
Response: ?09

Attempts to change the Baud Rate of module 09 to 57600bps, but the module returns a response indicating that the command was invalid command because it is not set to INIT* mode.

Command: %0905400940
Response: !09

Changes the Baud Rate of module 09 to 57600bps and the module is set to INIT* mode. The module returns a response indicating that the command was valid.

Related Commands:

Section 2.8 $AA2, Section 2.15 $AAI, Section 2.25 ~AAI, Section 2.32 ~AATNN

Related Topics:

Section 1.13 Configuration

Notes:

1 Changes to the address settings take effect immediately after a valid command is received. Changes to the Baud Rate and checksum settings take effect on the next power-on reset.

2 In order to change the Baud Rate, the checksum or other settings, the module must be set to INIT mode. Alternatively, send the following commands beforehand.

I. Send command ~AATNN (Refer to Section 2.32)
II. Send command ~AAI (Refer to Section 2.25)

III. Send command %AANNTTCCFF

If the commands are valid, the Baud Rate, checksum or others settings will take effect as soon as the module returns the response "!AA".

2.2 #**

Description:

This command is used to instruct all modules to sample their input values and store the data for later retrieval.

Syntax:

#[CHKSUM](CR)**

# Delimiter character

****** The synchronized sampling command

Response:

There is no response to this command. To access the data, another command, $AA4, must be sent. See Section 2.9 for details.

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

Examples:

Command: #**

Response: There is no response to this command.

Sends the synchronized sampling command to instruct all modules to sample their current input values. The data will be stored for later retrieval.

Command: $014

Response: !1010000

Sends a command to read the synchronized data and the module returns a response indicating that the command was valid. The status byte of the response is 1, which means that it is the first time the synchronized data has been read since the previous #** command was received.

Command: $014

Response: !0010000

Sends a command to read the synchronized data and the module returns a response indicating that the command was valid. The status byte of the response is 0, which means that it is **NOT** the first time the synchronized data has been read since the previous #** command was received.

Related Commands:

Section 2.9 $AA4

2.3 #AA

Description:

This command is used to read the UTC Time, the latitude and longitude, the quantity and other GPS satellite signal information from a specified module.

Syntax:

#AA[CHKSUM](CR)

# Delimiter character

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

Response:

Valid Command: **!AATTTTTT.TTT,LLLL.LLLL,C,NNNNN.NNNN, C,P,S**

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

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

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

TTTTTT.TTT The UTC Time

LLLL.LLLL The Latitude Value

C The Latitude Direction
Valid values = N or S (North or South)

NNNNN.NNNN The Longitude Value

C The Longitude Direction
Valid values = E or W (East or West)

P The Position Fix Indicator Value
0 = No fix possible, or unable to determine a valid position
1 = GPS SPS Mode, fix valid
2 = Differential GPS, SPS Mode, fix valid
6 = Estimated (Dead Reckoning) fix

S	The number of satellites in view Range = 0 - 12

Note that if there are insufficient GPS satellites in view, all data values will be set to 9.

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

Examples:

Command: #01

Response: !01999999.999,9999.9999,9,99999.9999,9,0,00

Attempts to read the UTC time, latitude, longitude, the Position Fix Indicator and the number of satellites in view for module 01, and the module returns a response indicating that the command was valid, but that the GPS data is invalid because there are insufficient GPS satellites in view.

Command: #01

Response: !01035035.00,2451.70629,N,12100.98908,E,1,9

Sends a command to read the GPS information and the module returns a response indicating that the command was valid, and that the UTC Time is 035035.000, the Latitude is 2451.7056 N, the Longitude is 12100.9903 E, the Position Fix Indicator is 1 (GPS SPS Mode, fix valid), and that there are 9 satellites in view.

Related Commands:

Section 2.4 #AAN

2.4 #AAN

Description:

This command is used to individually read the UTC time, the latitude and longitude, the date or the number of satellites in view for a specified module.

Syntax:

#AAN[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
N	The parameter to be read: 1: The UTC Time 2: The Latitude and Longitude in NMEA 0183 Format 3: The number of Satellites in view 4: The Date 5. The Latitude and Longitude in Google Maps Format

Response:

Valid Command:	**N=1**	**!AATTTTTT.TTT[CHKSUM](CR)**
	N=2	**!AALLLL.LLLL,C,NNNNN.NNNN,C [CHKSUM](CR)**
	N=3	**!AAS[CHKSUM](CR)**
	N=4	**!AADDMMYY[CHKSUM](CR)**
	N=5	**>LL.LLLLLL△C,NNN.NNNNNN△C [CHKSUM](CR)**
Invalid Command:	**?AA[CHKSUM](CR)**	

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
>	Delimiter character to indicate that the #AA5 command was valid
AA	The address of the responding module in hexadecimal format (00 to FF)

TTTTTT.TTT	The UTC Time
LLLL.LLLL	The Latitude value in degrees and minutes in NMEA 0183 format
C	The Latitude Direction Valid values = N or S (North or South)
NNNNN.NNNN	The Longitude value in degrees and minutes in NMEA 0183 format
C	The Longitude Direction Valid values = E or W (East or West)
S	The number of satellites in view. Range 0 - 12
DDMMYY	The Current Date: D (Day), M (Month), Y (Year)
LL.LLLLLL	The Latitude value in decimal degrees in Google Maps format
NNN.NNNNNN	The Longitude value in decimal degrees in Google Maps format
∆	Space character

Note that if there are insufficient GPS satellite signals, all data values will be set to 9.

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

Examples:

Command: #011

Response: !01999999.999

Attempts to read the UTC Time for module 01 and the module returns a response indicating that the command was valid, but that the GPS data is invalid because there were insufficient GPS satellites in view.

Command: #012

Response: !012451.7057,N,12100.9904,E

Reads the latitude and longitude information for module 01 NMEA 0183 format and the module returns a response indicating that the command was valid, showing that the latitude is 24° 51.7056’ N, and the longitude is 121°00.9903’ E.

Command: #013
Response: !019

Reads the number of satellites in view for module 01 and the module returns a response indicating that the command was valid, showing that there are 9.

Command: #014
Response: !01280314

Reads the date for module 01 and the module returns a response indicating that the command was valid, showing that the current date is March 28, 2008 (28/03/2014).

Command: #015
Response: >24.861750 N,121.016200 E

Reads the latitude and longitude information for module 01 in Google Maps format and the module returns a response indicating that the command was valid, showing that the latitude is 24.861750 N (degrees), and the longitude is 121.016200 E (degrees).

Related Commands:

Section 2.3 #AA, Section 2.13 $AAD, Section 2.29 ~AAR

Note:

The $AAD command (see Section 2.13) must be sent to synchronize the date information before using this command.

2.5 #AA00(Data) or #AA0A(Data)

Description:

This command is used to set the Digital Output value for a specified module.

Syntax:

#AA00(Data)[CHKSUM](CR)

#AA0A(Data)[CHKSUM](CR)

Delimiter character

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

00/0A The command to set the Digital Output value

(Data) A two-digit hexadecimal value representing the Digital Output value, where bit 0 corresponds to Digital Output channel 0. A bit value 0 of denotes that the Digital Output channel is set to OFF, and a bit value of 1 denotes that the Digital Output channel is set to ON.

Response:

Valid Command:	**>[CHKSUM](CR)**
Invalid Command:	**?AA[CHKSUM](CR)**
Ignored Command:	**![CHKSUM](CR)**

> Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

! Delimiter character to indicate that the command was ignored. The command will be ignored if a Host Watchdog timeout has occurred. If a Host Watchdog Timeout occurs, the module will be reset to the Safe value and all Digital Output commands will be ignored.

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

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

Examples:

Command: #010A01

Response: >

Sets Digital Output channel 0 of module 01 to ON, and the module returns a response indicating that the command was valid.

Command: #010001

Response: >

Sets Digital Output channel 0 of module 01 to ON, and the module returns a response indicating that the command was valid.

Command :#010A16

Response: ?01

Attempts to set Digital Output channels 0, 3, 6, 7 and 8 to OFF, and set Digital Output channels 1, 2 and 5 to ON, but a response indicating that the command was invalid is returned because the module only includes a single Digital Output channel.

Command: #010A01

Response: !

Attempts to set Digital Output channel 0 of module 01 to ON, but the module returns a response indicating that the command was ignored because a Host Watchdog timeout has occurred and so all Digital Output commands are ignored.

Related Commands:

Section 2.5 #AA00(Data), Section 2.6 #AAACDD, Section 2.7 #AA1CDD, Section 2.11 $AA6, Section 2.20 @AA, Section 2.21 @AA(Data)

Related Topics:

Section 1.13 Configuration

2.6 #AAACDD

Description:

This command is used to set a single Digital Output channel for a specified module.

Syntax:

#AAACDD[CHKSUM](CR)

#	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
A	The command to set a single Digital Output channel
C	Specifies the Digital Output channel to be set. As this module only has a single Digital Output channel, this value will always be 0.
DD	The command to set the Digital Output channel 00: Sets the Digital Output channel to OFF 01: Sets the Digital Output channel to ON

Response:

Valid Command:	**>[CHKSUM](CR)**
Invalid Command:	**?AA[CHKSUM](CR)**
Ignored Command:	**![CHKSUM](CR)**

>	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
!	Delimiter character to indicate that the command was ignored. The command will be ignored if a Host Watchdog timeout has occurred. If a Host Watchdog Timeout occurs, the module will be reset to the Safe value and all Digital Output commands will be ignored.
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: #01A001

Response: >

Sets Digital Output channel 0 of module 01 to ON, and the module returns a response indicating that the command was valid.

Command: #01A000

Response: >

Sets Digital Output channel 0 of module 01 to OFF, and the module returns a response indicating that the command was valid.

Command: #01A400

Response: ?01

Attempts to set Digital Output channel 4 to OFF, but a response indicating that the command was invalid is returned because the module only includes a single Digital Output channel.

Command: #01A001

Response: !

Attempts to set Digital Output channel 0 of module 01 to ON, but the module returns a response indicating that the command was ignored because a Host Watchdog Timeout has occurred and so all Digital Output commands are ignored.

Related Commands:

Section 2.5 #AA00(Data) / #AA0A(Data), Section 2.7 #AA1cDD, Section 2.11 $AA6, Section 2.20 @AA, Section 2.21 @AA(Data)

Related Topics:

Section 1.13 Configuration

2.7 #AA1CDD

Description:

This command is used to set a single Digital Output channel for a specified module.

Syntax:

#AA1CDD[CHKSUM](CR)

# Delimiter character

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

1 The command to set a single Digital Output channel

C Specifies the Digital Output channel to be set. As this module only has a single Digital Output channel, this value will always be 0.

DD The command to set the Digital Output channel
00: Sets the Digital Output channel to OFF
01: Sets the Digital Output channel to ON

Response:

Valid Command: **>[CHKSUM](CR)**
Invalid Command: **?AA[CHKSUM](CR)**
Ignored Command: **![CHKSUM](CR)**

> Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

! Delimiter character to indicate that the command was ignored. The command will be ignored if a Host Watchdog timeout has occurred. If a Host Watchdog Timeout occurs, the module will be reset to the Safe value and all Digital Output commands will be ignored.

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

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

Examples:

Command: #011001

Response: >

Sets Digital Output channel 0 of module 01 to ON, and the module returns a response indicating that the command was valid.

Command: #011400

Response: ?01

Attempts to set Digital Output channel 4 to OFF, but a response indicating that the command was invalid is returned because the module only includes a single Digital Output channel.

Command: #011201

Response: !

Attempts to set Digital Output channel 0 of module 01 to ON, but the module returns a response indicating that the command was ignored because a Host Watchdog Timeout has occurred and so all Digital Output commands are ignored.

Related Commands:

Section 2.5 #AA00(Data) / #AA0A(Data), Section 2.6 #AAACDD, Section 2.11 $AA6, Section 2.20 @AA, Section 2.21 @AA(Data)

Related Topics:

Section 1.13 Configuration

2.8 $AA2

Description:

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

Syntax:

$AA2[CHKSUM](CR)

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

Response:

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

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
TT	The Type Code for the module, which should be set to 40 for the GPS-721-MRTU module. See Section 1.13 for details).
CC	The Baud Rate code for the module. See Section 1.13 for details
FF	The checksum and counter update direction settings for the module. See Section 1.13 for details

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

Examples:

Command: $012

Response: !01400A80

Reads the configuration of module 01 and returns a response indicating that the command was valid, and showing that the Type code is set to 40, the Baud Rate is 115200 bps and that the Checksum is disabled and the counter edge is rising.

Command: %0505400940
Response: !05

Changes the Baud Rate of module 05 to 57600bps and the module is set to INIT* mode. The module returns a response indicating that the command was valid.

Command: $052
Response: !05400940

Reads the configuration of module 05 and returns a response indicating that the command was valid, and showing that the Type code is set to 40, the Baud Rate is 57600 bps and that the Checksum is enabled and the counter edge is falling.

Related Commands:

Section 2.1 %AANNTTCCFF

Related Topics:

Section 1.13 Configuration

2.9 $AA4

Description:

This command is used to read the synchronized sampling data that was stored when the last #** command (see Section 2.2) was sent.

Syntax:

$AA4[CHKSUM](CR)

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

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
S	The status of the synchronized data 0: This is **NOT** the first time that the data has been read 1: This is the first time that the data has been read
(Data)	The synchronized sampling data. See Section 1.8 for details of the data format

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

Examples:

Command: $014

Response: ?01

Attempts to read the synchronized sampling data for module 01, but returns a response indicating that the command was invalid because the Synchronized Sampling command, #** (see Section 2.2), was not sent in advance.

Command: #**

Response: There is no response to this command

Sends the synchronized sampling command to all modules..

Command: $014

Response: !1010000

Reads the synchronized sampling data for module 01. The module returns a response indicating that the command was valid containing the synchronized sampling data, and sets the status byte to 1 to signify that this is the first time the synchronized sampling data has been read.

Command: $014

Response: !0010000

Reads the synchronized sampling data for module 01. The module returns a response indicating that the command was valid containing the synchronized sampling data, and sets the status byte to 0 to signify that the synchronized sampling data has been read.

Related Commands:

Section 2.2 #**

Related Topics:

Section 1.13 Configuration

2.10 $AA5

Description:

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

Syntax:

$AA5[CHKSUM](CR)

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

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
S	The reset status of the module, where:

0: This is **NOT** the first time the command has been sent since the module was powered on, which denotes that there has been no module reset since the last $AA5 command was sent

1: This is the first time the $AA5 command has been sent since the module was powered on

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

Examples:

Command: $035
Response: !031

Reads the reset status for module 03 and returns a response indicating that the command was valid, and that it is the first time the $AA5 command has been sent since the module was powered-on.

Command: $035
Response: !030

Reads the reset status of module 03 and returns a response indicating that the command was valid, and that there has been no module reset since the last $AA5 command was sent.

Related Commands:

None

2.11 $AA6

Description:

This command is used to read the status of the Digital Output channel of a specified module.

Syntax:

$AA6[CHKSUM](CR)

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

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the invalid command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A single digit value representing the status of the Digital Output channel

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

Examples:

Command: $016

Response: !011

Reads the status of the Digital Output channel of module 01 and returns a response indicating that the command was valid with a value of 1, which denotes that the Digital Output is ON.

Related Commands:

Section 2.5 #AA00(Data) / #AA0A(Data), Section 2.6 #AAACDD, Section 2.7 #AA1CDD, Section 2.20 @AA, Section 2.21 @AA(Data)

Related Topics:

Section 1.13 Configuration

2.12 $AAC

Description:

This command is used to clear the status of the latched Digital Output channel of a specified module.

Syntax:

$AAC[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be cleared in hexadecimal format (00 to FF)
C	The command to clear the status of the latched Digital Output channels

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: @011

Response: >

Sets Digital Output channel 0 of module 01 to ON.

Command: $01C

Response: !01

Clears the status of the latched Digital Output channels of module 01 and returns a response indicating that the command was valid.

Command: $01L0

Response: !010

Reads the status of the low latched Digital Output channel of module 01 and returns a response indicating that the command was valid, with a value of 0 denoting that the low latched value has recently been set to ON.

Command: $01L1

Response: !011

Reads the status of the high latched Digital Output channel of module 01 and returns a response indicating that the command was valid, with a value of 0 denoting that the high latched value has recently been set to ON.

Related Commands:

Section 2.16 $AALS

Notes:

The status of both the high and the low latched Digital Output channels will be cleared when using this command.

2.13 $AAD

Description:

This command is used to synchronize the current date of the specified module.

Syntax:

$AAD[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be synchronized in hexadecimal format (00 to FF)
D	The command to synchronize the current date

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: $01D

Response: !01

Synchronizes the current date for module 01 and returns a response indicating that the command was valid. The date information is temporarily saved in the RAM.

Command: #014

Response: !0199,99,9999

Attempts to read the current date for module 01 and the module returns a response indicating that the command was valid, but that the date information is invalid because there are insufficient GPS satellites in view.

Command: #014

Response: !01280314

Reads the date for module 01 and the module returns a response indicating that the command was valid, showing that the current date is March 28, 2014 (28/03/2014).

Related Commands:

Section 2.4 #AAN

Notes:

1. To read the current date, use the #AAN command (see Section 2.4).
2. It takes approximately 2 seconds to read the date.

2.14 $AAF

Description:

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

Syntax:

$AAF[CHKSUM](CR)

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

Response:

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

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The Address of the responding module in hexadecimal format (00 to FF)
(Data)	A string indicating the firmware version of the module

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

Examples:

Command: $01F

Response: !0101.20

Reads the firmware version of module 01, and returns a response indicating that the command was valid, and that the firmware is version 1.2.

Related Commands:

None

2.15 $AAI

Description:

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

Syntax:

$AAI[CHKSUM](CR)

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

Response:

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

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The status of the INIT pin where: 0: The INIT pin is shorted to GND 1: The INIT pin is open

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

Examples:

Command: $01I

Response: !010

Reads the status of the INIT pin on module 01 and returns a response indicating that the command was valid with a value of 0, signifying that the INIT pin is shorted to GND.

Command: $01I

Response: !011

Reads the status of the INIT pin on module 01 and returns a response indicating that the command was valid with a value of 1, signifying that the INIT pin is open and that the configuration settings cannot be adjusted.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.8 $AA2

2.16 $AALS

Description:

This command is used to read the status of the latched Digital Output channels on a specified module.

Syntax:

$AALS[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
L	The command to read the status of the latched channels
S	The status to be read: 0: Reads the status of the low latched channels 1: Reads the status of the high latched channels

Response:

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

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	The status of the latched Digital Output channel represented by a single-digit hexadecimal value followed by 0.

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

Examples:

Command: @011

Response: >

Sets the Digital Output channel of module 01 to ON and returns a response indicating the command was valid.

Command: $01C

Response: !01

Clears the status of the latched Digital Output channel of module 01 and returns a response indicating that the command was valid.

Command: $01L0

Response: !010

Reads the status of the low latched Digital Output channel of module 01 and returns a response indicating that the command was valid, with a value of 0 denoting that the low latched value has recently been set to ON.

Command: $01L1

Response: !011

Reads the status of the high latched Digital Output channel of module 01 and returns a response indicating that the command was valid, with a value of 0 denoting that the high latched value has recently been set to ON.

Related Commands:

Section 2.12 $AAC

2.17 $AAM

Description:

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

Syntax:

$AAM[CHKSUM](CR)

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

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A string indicating the name of the module

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

Examples:

Command: $01M

Response: !01GPS721

Reads the name of module 01 and returns a response indicating that the command was valid, and that the name of the module is "GPS721".

Related Commands:

Section 2.28 ~AAO(Name)

2.18 $AAP

Description:

This command is used to read the communication protocol information for a specified module.

Syntax:

$AAP[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
P	The command to read the communication protocol information

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
S	The protocols supported by the module: 0: Only the DCON is supported 1: Both the DCON and the Modbus RTU protocols are supported 3: Both the DCON and the Modbus RTU/ASCII protocols are supported
C	The current protocol that is saved in the EEPROM and that will be used at the next power-on reset: 0: The protocol saved in the EEPROM is DCON 1: The protocol saved in the EEPROM is Modbus RTU 3: The protocol saved in the EEPROM is Modbus ASCII

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

Examples:

Command: $01P

Response: !0130

Reads the communication protocol information for module 01 and returns a response indicating that the command was valid, with a value of 30, which denotes that the module supports both the DCON and the Modbus RTU/ASCII protocols, and that the protocol that will be used at the next power-on reset is DCON.

Related Commands:

Section 2.19 $AAPN

2.19 $AAPN

Description:

This command is used to set the communication protocol for a specified module.

Syntax:

$AAPN[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
P	The command to set the communication protocol
N	The protocol to be used: 0: DCON Protocol 1: Modbus RTU Protocol 3: Modbus ASCII Protocol

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: $01P1

Response: ?01

Attempts to set the communication protocol to be used for module 01 to Modbus RTU, but returns a response indicating that the command was invalid because the module is not in INIT mode.

Command: $01P1

Response: !01

Sets the communication protocol to be used for module 01 to Modbus RTU and returns a response indicating that the command was valid.

Related Commands:

Section 2.18 $AAP

Note:

Before using this command, the rear slide switch must be in the INIT position. The new protocol information will be saved in the EEPROM and will become effective after the next power-on reset.

2.20 @AA

Description:

This command is used to read the status of the Digital Output channel of a specified module.

Syntax:

@AA[CHKSUM](CR)

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

Response:

Valid Command:	**>(Data)[CHKSUM](CR)**
Invalid Command:	**?AA[CHKSUM](CR)**

>	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A single-digit hexadecimal value representing the status of the Digital Output channel.

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

Examples:

Command: @01

Response: >0100

Reads the status of the Digital Output channel for module 01 and returns a response indicating that the command was valid, with a value of 0100, where 01 denotes that the Digital Output channel is ON.

Related Commands:

Section 2.5 #AA00(Data) / #AA0A(Data), Section 2.6 #AAACDD, Section 2.7 #AA1CDD, Section 2.11 $AA6, Section 2.21 @AA(Data)

Related Topics:

Section 1.13 Configuration

2.21 @AA(Data)

Description:

This command is used to set the value for the Digital Output channel of a specified module.

Syntax:

@AA(Data)[CHKSUM](CR)

@ Delimiter character

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

(Data) A single-digit hexadecimal value representing the data to be written to the Digital Output channel, where bit 0 of the value corresponds to the Digital Output channel. A bit value of 0 denotes that the Digital Output channel will be set to OFF, and a bit value of 1 denotes that the Digital Output channel will be set to ON.

Response:

Valid Command:	**>[CHKSUM](CR)**
Invalid Command:	**?AA[CHKSUM](CR)**
Ignored Command:	**![CHKSUM](CR)**

> Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

! Delimiter character to indicate that the command was ignored. The command will be ignored if a Host Watchdog Timeout has occurred. If a Host Watchdog Timeout occurs, the module will be reset to the Safe value and all Digital Output commands will be ignored.

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

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

Examples:

Command: @011
Response: >

> Sets the Digital Output channel of module to ON, and the module returns a response indicating that the command was valid.

Command: @019
Response: ?01

> Attempts to set Digital Output channel 9 of module to ON, but the module returns a response indicating that the command was invalid because Digital Output channel 9 doesn’t exist.

Command: @011
Response: !

> Attempts to set the Digital Output channel of module to ON, but the module returns a response indicating that the command was ignored because a Host Watchdog timeout has occurred.

Related Commands:

Section 2.5 #AA00(Data) / #AA0A(Data), Section 2.6 #AAACDD, Section 2.7 #AA1CDD, Section 2.11 $AA6, Section 2.20 @AA

2.22 ~**

Description:

If the Host Watchdog function has been enabled on the module, this command is used to inform all modules that the host is OK. Note that this command clears the counter of the Host Watchdog to prevent the module from being reset to safe mode. The module will be set to safe mode if a ~** command is received within the Host Watchdog timeout period to prevent unexpected events.

Syntax:

~[CHKSUM](CR)**

~	Delimiter character
**	The “Host OK” command

Response:

There is no response to this command.

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

Example:

Command: ~**

Response: No response

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

Command: ~010

Response: !0180

Reads the status of the Host Watchdog for module 01 and returns a response indicating that the command was valid, showing that the Host Watchdog is enabled and no Host Watchdog timeout has occurred.

Related Commands:

Section 2.33 ~AA0, Section 2.34 ~AA1, Section 2.35 ~AA2, Section 2.36 ~AA3EVV, Section 2.37 ~AA4V, Section 2.38 ~AA5V

Related Topics:

Appendix 1: Dual Watchdog Operation

2.23 ~AAD

Description:

This command is used to read whether the Digital Output signal for a specified module is active or inactive.

Syntax:

~AAD[CHKSUM](CR)

~ Delimiter character

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

D The command to read whether the Digital Output is active or inactive

Response:

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

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

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

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

VV A two-digit hexadecimal value representing the status of the Digital Output signal

7	6	5	4	3	2	1	0
Reserved						OAS	

Key	Description
OAS	Bit 1: Specifies the status of the Digital Output signal: 0. An output value of 0 indicates that the relay is inactive An output value of 1 indicates that the relay is active 1. An output value of 0 indicates that the relay is active An output value of 1 indicates that the relay is inactive

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

Examples:

Command: ~01D02

Response: !0102

Sets the Digital Output signals for module 01 to 02, which denotes that the Digital Output channel is in inactive mode, and returns a response indicating that the command was valid,

Command: ~01D

Response: !0102

Reads the status of the Digital Output signals for module 01 and returns a response indicating that the command was valid, with a value of 02, which denotes that the Digital Output channel is in inactive mode.

Related Commands:

Section 2.24 ~AADVV

2.24 ~AADVV

Description:

This command is used to set the Digital Output signal for a specified module to active or inactive.

Syntax:

~AADVV[CHKSUM](CR)

~ Delimiter character

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

D The command to set the Digital Output to active or inactive

VV A two-digit hexadecimal value representing the status of the Digital Output signal

7	6	5	4	3	2	1	0
Reserved						OAS	

Key	Description
OAS	Bit 1: Specifies the status of the Digital Output signal: 0. An output value of 0 sets the relay to inactive An output value of 1 sets the relay to active 1. An output value of 0 sets the relay to active An output value of 1 sets the relay to inactive

Response:

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

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

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

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

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

Examples:

Command: ~01D00

Response: !0100

Sets the Digital Output signals for module 01 to 00, which denotes that the Digital Output channels are in active mode, and returns a response indicating that the command was valid,

Command: ~01D

Response: !0100

Reads the status of the Digital Output signals for module 01 and returns a response indicating that the command was valid, with a value of 00, which denotes that the Digital Output channels are in active mode.

Related Commands:

Section 2.23 ~AAD

2.25 ~AAI

Description:

This command is used to enable the “Soft INIT” function, which temporarily switches the specified module to initialization mode. The “~AATNN” command (see Section 2.32) can be used to set the duration that the “Soft INIT” mode is active. The Baud Rate and checksum settings for the module can then be changed using the “%AANNTTCCFF” command (see Section 2.1).

Syntax:

~AAI[CHKSUM](CR)

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

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: ~01I

Response: !01

Enables the “Soft INIT” function on module 01 and returns a response indicating that the command was valid.

Command: %0107400780
Response: ?01

Attempts to change the Baud Rate and checksum settings for module 01, but the module returns a response indicating that the command was invalid because the duration for the "Soft INIT" function has not previously been set using the "~AATNN" command (see Section 2.32).

Command: ~01T10
Response: !01

Sets the duration for the "Soft INIT" function on module 01 to 10 (16 seconds) and returns a response indicating that the command was valid.

Command: ~01I
Response: !01

Enables the "Soft INIT" function on module 01 and returns a response indicating that the command was valid.

Command: %0107400740
Response: !01

Changes the Baud Rate and checksum settings for module 01 and returns a response indicating that the command was valid.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.32 ~AATNN

Related Topics:

Section 1.13 Configuration

2.26 ~AAM

Description:

This command is used to read the transmission protocol being used by a specified module, which can be either the DCON or the GPS protocol. Note that in GPS mode, no messages can be sent via the RS-485 port.

Syntax:

~AAM[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
M	The command to read the transmission mode being used for the RS-232 port

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
S	The transmission protocols supported by the module 0: DCON 3: DCON\GPS
C	The transmission protocol currently being used 0: DCON 1: GPS

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

Examples:

Command: ~03M

Response: !0330

Reads the transmission protocol being used for module 03 and returns a response indicating that the command was valid, and showing that the module supports both DCON and GPS protocols and that the DCON protocol is currently being used.

Command: ~03M1

Response: !03

Sets the transmission protocol to be used for module 03 to the GPS protocol and returns a response indicating that the command was valid. Note that the module must be set to INIT mode (see Section 1.7) before using this command. The new settings will become effective after the module is restarted.

Command: ~03M

Response: !0331

Reads the transmission protocol being used for module 03 and returns a response indicating that the command was valid, and showing that the module supports both DCON and GPS protocols, and that the module is currently using the GPS protocol. Note that the module needs to be restarted before the new transmission protocol becomes effective.

Related Commands:

Section 2.27 ~AAMN

Related Topics:

Section 1.7 Operating Modes

2.27 ~AAMN

Description:

This command is used to set the transmission protocol for a specified module. The GPS-721-MRTU supports either the DCON or the GPS transmission protocol. Note that in GPS mode, no messages can be sent via the RS-485 port.

Syntax:

~AAMN[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
M	The command to set the transmission protocol
N	The transmission protocol to be used 0: DCON 1: GPS

Response:

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

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was in valid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: ~03M

Response: !0330

Reads the transmission protocol being used for module 03 and returns a response indicating that the command was valid, and showing that the module supports both DCON and GPS protocols and that the DCON protocol is currently being used.

Command: ~03M1

Response: !03

Sets the transmission protocol to be used for module 03 to the GPS protocol and returns a response indicating that the command was valid. Note that the module must be set to INIT mode (see Section 1.7) before using this command. The new settings will become effective after the module is restarted.

Command: ~03M

Response: !0331

Reads the transmission protocol being used for module 03 and returns a response indicating that the command was valid, and showing that the module supports both DCON and GPS protocols, and that the module is currently using the GPS protocol. Note that the module needs to be restarted before the new transmission protocol becomes effective.

Command: ~03M1

Response: ?03

Attempts to set the transmission protocol to be used for module 03 to the GPS protocol and returns a response indicating that the command was invalid because the module needs to be set to INIT mode before changing the communication protocol.

Related Commands:

Section 2.26 ~AAM

Related Topics:

Section 1.7 Operating Modes

2.28 ~AAO(Name)

Description:

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

Syntax:

~AAO(Name)[CHKSUM](CR)

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

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: ~01O87000

Response: !01

Sets the name of module 01 to “GPS721” and returns a response indicating that the command was valid.

Command: $01M

Response: !01GPS721

Reads the name of module 01 and returns a response indicating that the command was successful, with the name “GPS721”.

Command: ~01O123456789ABCDEF

Response: ?01

Attempts to set the name of module 01 to "123456789ABCDEF", but returns a response indicating that the command was unsuccessful, because the name is longer than 6 characters.

Related Commands:

Section 2.17 $AAM

2.29 ~AAR

Description:

This command is used to enable the date for a specified module to be updated from a GPS message.

Syntax:

~AAR[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be updated in hexadecimal format (00 to FF)
R	The command to enable the date to be updated from a GPS message

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: #014

Response: !01999999

Attempts to read the date for module 01 from a GPS message and returns a response indicating that the command was valid, but that the date was not updated because there were insufficient GPS satellites in view.

Command: !01R

Response: >

Enables the date for module 01 to be updated from a GPS message, and returns a response indicating that the command was valid.

Command: #014

Response: >01230314

Reads the date for module 01 from a GPS message, and returns a response indicating that the command was valid, showing that the date is March 23, 2014 (23/03/14).

Related Commands:

Section 2.4 #AAN

2.30 ~AARD

Description:

This command is used to read the Response Delay Time for a specified module.

Syntax:

~AARD[CHKSUM](CR)

~ Delimiter character

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

RD The command to read the Response Delay Time for the module

Response:

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

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

! Delimiter character to indicate that the command was valid

? Delimiter character to indicate that the command was invalid

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

(Data) The Response Delay Time Value in ms (0 to 30)

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

Examples:

Command: ~03RD1E

Response: !03

Sets the Response Delay Time for module 03 to 1E (30 ms) and returns a response indicating that the command was valid.

Command: ~03RD

Response: !0300

Reads the Response Delay Time for module 03 and returns a response indicating that the command was valid, with a value of 00 (0 ms).

Related Commands:

Section 2.31 ~AARDVV

2.31 ~AARDVV

Description:

This command is used to set the Response Delay Time for a specified module.

Syntax:

~AARDVV[CHKSUM](CR)

$	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
RD	The command to set the Response Delay Time for the module.
VV	The Response Delay Time in ms (0 to 30)

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: ~03RD1E

Response: !03

Sets the Response Delay Time for module 03 to 1E (30 ms) and returns a response indicating that the command was valid.

Command: ~03RD

Response: !0300

Reads the Response Delay Time for module 03 and returns a response indicating that the command was valid, with a value of 00 (0 ms).

Command: ~03RD1F

Response: ?03

Attempts to set the Response Delay Time for module 03 to1F (31 ms), but returns a response indicating that the command was invalid because the value was not within the valid range (0 to 30),

Related Commands:

Section 2.30 ~AARD

2.32 ~AATNN

Description:

This command is used to set the duration that the “Soft INIT” function is active on a specified module. Note that before using this command, the “~AAI” command (see Section 2.25) must be sent to set the module to “Soft INIT” mode. The Baud Rate and checksum settings for the module can then be changed using the “%AANNTTCCFF” command (see Section 2.1).

Syntax:

~AATNN[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
T	The command to set the duration that the “Soft INIT” function is active
NN	A two-digit hexadecimal value representing the duration that the Soft INIT function is active. For example: a value of 10 = 16 seconds. Note that the maximum duration is 60 seconds

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: ~01I

Response: !01

Enables the "Soft INIT" function on module 01 and the module returns a response indicating that the command was valid.

Command: %0107400780

Response: ?01

Attempts to change the Baud Rate and checksum settings for module 01, but the module returns a response indicating that the command was invalid because the duration for the “Soft INIT” function has not previously been set using the "~AATNN" command.

Command: ~01T10

Response: !01

Sets the duration for the Soft INIT function on module 01 to 10 (16 seconds) and returns a response indicating that the command was valid.

Command: ~01I

Response: !01

Enables the “Soft INIT” function on module 01 and returns a response indicating that the command was valid.

Command: %0107400740

Response: !01

Changes the Baud Rate and checksum settings for module 01 and returns a response indicating that the command was valid.

Related Commands:

Section 2.1 %AANNTTCCFF, Section 2.25 ~AAI

Related Topics:

Section 1.13 Configuration

2.33 ~AA0

Description:

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

Syntax:

~AA0[CHKSUM](CR)

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

Response:

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

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
SS	A two-digit hexadecimal value that represents the status of the Host Watchdog, where:

Bit 2: 0 indicates that no Host Watchdog timeout has occurred, and 1 indicates that a Host Watchdog timeout has occurred.

Note: The status of the Host Watchdog is stored in the EEPROM and can only be reset by using the ~AA1 command. See Section 2.34 for more details.

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

Examples:

Command: ~010

Response: !0104

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

Command: ~010

Response: !0100

Reads the status of the Host Watchdog for module 01 and returns a response indicating that the command was valid, and showing that the Host Watchdog on the module is disabled and that a timeout has not occurred.

Related Commands:

Section 2.22 ~**, Section 2.34 ~AA1, Section 2.35 ~AA2, Section 2.36 ~AA3EVV, Section 2.37 ~AA4V, Section 2.38 ~AA5V

Related Topics:

Appendix 1: Dual Watchdog Operation

2.34 ~AA1

Description:

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

Syntax:

~AA1[CHKSUM](CR)

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

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was valid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: ~010
Response: !0104

Reads the status of the Host Watchdog for module 01 and returns a response indicating that the command was valid, and that a Host Watchdog timeout has occurred.

Command: ~011
Response: !01

Resets the status of the Host Watchdog timeout for module 01 and returns a response indicating that the command was valid.

Command: ~010

Response: !0100

Reads the status of the Host Watchdog for module 01 and returns a response indicating that the command was valid, and that no Host Watchdog timeout has occurred.

Related Commands:

Section 2.22 ~**, Section 2.33 ~AA0, Section 2.35 ~AA2, Section 2.36 ~AA3EVV, Section 2.37 ~AA4V, Section 2.38 ~AA5V

Related Topics:

Appendix 1: Dual Watchdog Operation

2.35 ~AA2

Description:

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

Syntax:

~AA2[CHKSUM](CR)

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

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
E	The status of the Host Watchdog 0: The Host Watchdog is disabled 1: The Host Watchdog is enabled
VV	A two-digit hexadecimal value that represents the timeout value in tenths of a second, for example, 01 denotes 0.1 seconds and FF denotes 25.5 seconds.

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

Examples:

Command: ~013164

Response: !01

Enables the Host Watchdog for module 01 and sets the Host Watchdog timeout value to 64 (10.0 seconds). The module returns a response indicating that the command was valid.

Command: ~012
Response: !011FF

Reads the Host Watchdog timeout value of module 01 and returns a response indicating that the command was valid with a value of 1FF, which denotes that the Host Watchdog is enabled and the Host Watchdog timeout value is 25.5 seconds (FF).

Related Commands:

Section 2.22 ~**, Section 2.33 ~AA0, Section 2.34 ~AA1, Section 2.36 ~AA3EVV, Section 2.37 ~AA4V, Section 2.38 ~AA5V

Related Topics:

Appendix 1: Dual Watchdog Operation

2.36 ~AA3EVV

Description:

This command is used to enable or disable the Host Watchdog for a specified module, and set the Host Watchdog timeout value. If a Host Watchdog timeout has occurred, the Host Watchdog will be disabled.

Syntax:

~AA3EVV[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be configured in hexadecimal format (00 to FF)
3	The command to enable or disable the Host Watchdog
E	The command to set the Host Watchdog: 0: Disables the Host Watchdog 1: Enables the Host Watchdog
VV	A two-digit hexadecimal value that represents the Host Watchdog timeout value in tenths of a second, for example, 01 denotes 0.1 seconds and FF denotes 25.5 seconds

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: ~013164

Response: !01

Enables the Host Watchdog for module 01 and sets the Host Watchdog timeout value to 64 (10.0 seconds). The module returns a response indicating that the command was valid.

Command: ~010

Response: !0104

Reads the status of the Host Watchdog for module 01 and returns a response indicating that the command was valid, and showing that a Host Watchdog timeout has occurred. The Host Watchdog will be disabled.

Command: ~010

Response: !0180

Reads the status of the Host Watchdog for module 01 and returns a response indicating that the command was valid, and showing that the Host Watchdog is enabled.

Command: ~012

Response: !01164

Reads the Host Watchdog timeout value for module 01. The module returns a response indicating that the command was valid, with a value of 164, which denotes that the Host Watchdog is enabled and that the Host Watchdog timeout value is 10.0 seconds.

Related Commands:

Section 2.22 ~**, Section 2.33 ~AA0, Section 2.34 ~AA1, Section 2.35 ~AA2, Section 2.37 ~AA4V, Section 2.38 ~AA5V

Related Topics:

Appendix 1: Dual Watchdog Operation

2.37 ~AA4V

Description:

This command is used to read either the Digital Output Power-on value or the Digital Output Safe value for a specified module.

Syntax:

~AA4V[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be read in hexadecimal format (00 to FF)
4	The command to read the Digital Output Power-on value or the Digital Output Safe value.
V	The Digital Output Power-on or Safe value P: Power-on Value S: Safe Value

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)
(Data)	A single-digit hexadecimal value representing the Digital Output Power-on value or the Digital Output Safe value.

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

Examples:

Command: ~014P

Response: !011

Reads the Digital Output Power-on value for module 01 and returns a response indicating that the command was valid, with a value denoting that the Digital Output will be set to 1 at the next power-on reset. See Section 2.23 for details.

Command: ~014S

Response: !010

Reads the Digital Output Safe value for module 01 and returns a response indicating that the command was valid, with a value denoting that the Digital Output will be set to 0 when a Host Watchdog timeout occurs. See Section 2.23 for details.

Related Commands:

Section 2.38 ~AA5V

2.38 ~AA5V

Description:

This command is used to set the current Digital Output value as either the Digital Output Power-on value or the Digital Output Safe value.

Syntax:

~AA5V[CHKSUM](CR)

~	Delimiter character
AA	The address of the module to be set in hexadecimal format (00 to FF)
5	The command to set the Digital Output Power-on value or the Digital Output Safe value
V	The Digital Output Power-on or Safe value P: Power-on Value S: Safe Value

Response:

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

!	Delimiter character to indicate that the command was valid
?	Delimiter character to indicate that the command was invalid
AA	The address of the responding module in hexadecimal format (00 to FF)

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

Examples:

Command: ~015P

Response: !01

Sets the Digital Output Power-on value for module 01 and returns a response indicating that the command was valid.

Command: @011

Response: >

Sets the Digital Output value for module 01 to 1, and returns a response indicating that the command was valid.

Command: ~015S

Response: !01

Sets the Digital Output Safe value for module 01 and returns a response indicating that the command was valid.

Command: @010

Response: >

Sets the Digital Output value for module 01 to 0, and returns a response indicating that the command was valid.

Command: ~014S

Response: !010

Reads the Digital Output Safe value for module 01 and returns a response indicating that the command was valid, with a value denoting that the Digital Output will be set to 0 when a Host Watchdog timeout occurs. See Section 2.23 for details.

Related Commands:

Section 2.37 ~AA4V

3 Modbus RTU Protocol

The Modbus protocol was originally developed for Modicon controllers by Modicon Inc.

Detailed information related to the Modbus RTU protocol can be found at:

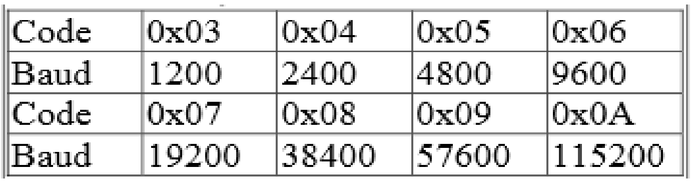
http://www2.schneider-electric.com/sites/corporate/en/products-services/automation-control/automation-control.page.

More valuable information can also be found at: http://www.modbus.org for more valuable information.

Modbus Address Mapping		
Address	**Description**	**Attribute**
00001, 10001	Reads the current status of the Digital Output channel or sets the Digital Output to either active or inactive	R/W
00129, 10129	Reads/sets the Digital Output Safe Value	R/W
00161, 10161	Reads/sets the Digital Output Power–on Value	R/W
00270, 10270	Reads the status of the Host Watchdog timeout. Write 1 to clear.	R/W
00260, 10260	Reads/sets the Modbus Host Watchdog Mode 0: Refer to Appendix 1 – Dual Watchdog Operation 1: Digital Output commands can be used to clear the status of the Host Watchdog timeout	R/W
00257, 10257	Reads/sets the Communication Protocol 0: DCON 1: Modbus RTU	R/W
00258, 10258	Reads/sets the Output Message Format: 0: Protocol Command Communication 1: NMEA 0183 Message	R/W
00261, 10261	Enables or disables the Host Watchdog 0: Disabled 1: Enabled	R/W

10273	Reads the Reset Status 0. This is **NOT** the first time the module has been read since being powered on 1. This is the first time the module has been read since being powered on	R
00274	Enables the Date to be updated from a GPS message. Write 1 to enable.	W
00275	Resets the GPS module. Write 1 to reset.	W
40001-40002	Reads the Latitude value (NMEA 0183 format) in degrees and minutes (Float Type) Value/100, +/-ddmm.mmmm, + = north, - = south *A value of 99999.99 is returned if fewer than 3 satellites are available.	R
40003-40004	Reads the Longitude value (NMEA 0183 format) in degrees and minutes (Float Type) Value/100, +/-dddmm.mmmm, + = east, - = west * A value of 99999.99 is returned if fewer than 3 satellites are available.	R
40005	Reads the Year value of the Date * A value of 99 is returned if fewer than 3 satellites are available.	R
40006	Reads the Month value of the Date * A value of 99 is returned if fewer than 3 satellites are available.	R
40007	Reads the Day value of the Date * A value of 99 is returned if fewer than 3 satellites are available.	R
40008	Reads the Hour value of the UTC Time * A value of 99 is returned if fewer than 3 satellites are available.	R
40009	Reads the Minutes value of the UTC Time * A value of 99 is returned if fewer than 3 satellites are available.	R
40010	Reads the Seconds value of the UTC Time * A value of 99 is returned if fewer than 3 satellites are available.	R

40011	Reads the GPS data quality. 0 = Invalid 1 = GPS fix 2 = DGPS fix	R
40012	Reads the number of satellites currently in view.	R
40013	Reads the Latitude value (NMEA 0183 format) in degrees (Integer Type) Value/100, +/-ddmm, + = north, - = south * A value of 9999 is returned if fewer than 3 satellites are available.	R
40014	Reads the Longitude value (NMEA 0183 format) in degrees (Integer Type) Value/100, +/-dddmm, + = east, - = west * A value of 9999 is returned if fewer than 3 satellites are available (For firmware version 2.00). * A value of 0x9999 is returned if fewer than 3 satellites are available (For firmware version 2.10 or later).	R
40015	Reads the Latitude float value (NMEA 0183 format) in minutes (Integer Type) Value/10000, +/-mmmm, + = north, - = south	n
40016	Reads the Longitude float value (NMEA 0183 format) in minutes (Integer Type) Value/10000, +/-mmmm, + = north, - = south	R
40017~40018	Reads the Latitude value (Google Maps format) in degrees (Float Type) Value/100, +/-dd.dddddd, + = north, - = south *A value of 99999.99 is returned if fewer than 3 satellites are available.	R
40019~40020	Reads the Longitude value (Google Maps format) in degrees and minutes (Float Type) Value/100, +/-ddd.dddddd, + = east, - = west * A value of 99999.99 is returned if fewer than 3 satellites are available.	R
40021	Reads the Latitude decimal value (Google Maps format) in degrees (Integer Type) Value , +/-dd, + = north, - = south	R

<table>
<tr><td>40022</td><td>Reads the Latitude float value (Google Maps format) in degrees (Integer Type)
Value/10000, +/-dddd, + = north, - = south</td><td>R</td></tr>
<tr><td>40023</td><td>Reads the Longitude decimal value (Google Maps format) in degrees (Integer Type)
Value, +/-ddd, + = east, - = west</td><td>R</td></tr>
<tr><td>40024</td><td>Reads the Longitude float value (Google Maps format) in degrees (Integer Type)
Value/10000, +/-dddd, + = east, - = west</td><td>R</td></tr>
<tr><td>40033</td><td>Reads the Latitude degree value for use in Google Maps (Integer type)</td><td>R</td></tr>
<tr><td>40034~40035</td><td>Reads the Latitude minutes value for use in Google Maps (Float type)</td><td>R</td></tr>
<tr><td>40036</td><td>Reads the direction of the Latitude for use in Google Maps (Integer type, 0 = north; -1 = south)</td><td>R</td></tr>
<tr><td>40037</td><td>Reads the Longitude degree value for use in Google Maps (Integer type)</td><td>R</td></tr>
<tr><td>40038~40039</td><td>Reads the Longitude minutes value for use in Google Maps (Float type)</td><td>R</td></tr>
<tr><td>40040</td><td>Reads the direction of the Longitude for use in Google Maps (Integer type, 0 = east; -1 = west)</td><td>R</td></tr>
<tr><td>40481-40482</td><td>Reads the Firmware Version information</td><td>R</td></tr>
<tr><td>40483-40484</td><td>Reads the Name of the Module</td><td>R</td></tr>
<tr><td>30485, 40485</td><td>Reads/sets the Module address. The valid range is 1 to 247</td><td>R/W</td></tr>
<tr><td>40486</td><td>Reads/sets the Baud Rate and the Data Format
Bits 5:0 (Baud Rate)
Baud Rate, valid range: 0x03 to 0x0A

<table>
<tr><td>Code</td><td>0x03</td><td>0x04</td><td>0x05</td><td>0x06</td></tr>
<tr><td>Baud</td><td>1200</td><td>2400</td><td>4800</td><td>9600</td></tr>
<tr><td>Code</td><td>0x07</td><td>0x08</td><td>0x09</td><td>0x0A</td></tr>
<tr><td>Baud</td><td>19200</td><td>38400</td><td>57600</td><td>115200</td></tr>
</table>
Bits 7:6
00: no parity, 1 stop bit
01: no parity, 2 stop bits</td><td>R/W</td></tr>
</table>

	10: even parity, 1 stop bit 11: odd parity, 1 stop bit	
30488, 40488	Reads/sets the Modbus Response Delay Time in milliseconds. The valid range is 0 to 30 ms.	R/W
30492, 40492	Reads/sets the Host Watchdog Timeout Counter. Write 0 to clear.	R/W
30489, 40489	Reads/sets the Host Watchdog Timeout Value, The valid range is 0 to 255, in intervals of 0.1 seconds	R/W
312345	Informs all modules that the host is OK	W

1. Modbus RTU Function Description:
   (0xxxx): 0x05, 0x0F Function Code
   (1xxxx): 0x01 Function Code
   (3xxxx): 0x06, 0x10 Function Code
   (4xxxx): 0x03 Function Code

Function Code	Description	Section
0x01	Read Coils	3.1
0x02	Read Discrete Inputs	3.2
0x03	Read Multiple Registers	3.3
0x04	Read Multiple Input Registers	3.4
0x05	Write a Single Coil	3.5
0x06	Write Multiple Registers	3.6
0x0F	Write Multiple Coils	3.7
0x46	Read/Write Module Settings	3.8

If the function specified in the message is not supported, then the module responds as below. Note that the address mapping for the Modbus protocol is Base 0.

Error Response

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

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

3.1 01 (0x01) Read the Coils

This function code is used to read the current digital output value from the GPS-721-MRTU module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x01
02-03	Starting Channel Number or Address Mapping	2 Bytes	Refer to the Modbus Address Mapping Table for more information.
04-05	Output Channel Number or Bit Count	2 Bytes	0x0001-0x0020 (Bit count)

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x01
02	Byte Count	1 Byte	The Byte Count of the Response (B=(bit count + 7)/8)
03	Bit Value	B Bytes	(Bit value)

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x81
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 05 01 00 00 00 01 [CRC]

Response: 05 01 01 01 [CRC]

Reads the Digital Output value

[Command]

Bytes 2 to 3: 00 00 (The starting channel number)

Byte 4 to 5: 00 01 (The output channel number)

Bytes 6 to 7: FD 8D (CRC)

[Response]
Byte 1: 01 (The Function Code)
Byte 2: 01 (The byte count of the response)
Byte 3: 01 (The current value for Digital Output channel 0)
Byte 4 to 5: 11 70 (CRC)

Command: 05 01 01 03 00 01 [CRC]
Response: 05 01 01 00 [CRC]
Reads the mode for the Modbus Host Watchdog.

Command: 05 01 01 10 00 01 [CRC]
Response: 05 01 01 01 [CRC]
Reads the reset status of the modules.

Command: 05 01 01 04 00 01 [CRC]
Response: 05 01 01 00 [CRC]
Reads whether or not the Host Watchdog is enabled.

Command: 05 01 01 0D 00 01 [CRC]
Response: 05 01 01 00 [CRC]
Resets the status of the Host Watchdog timeout.

Command: 05 01 01 00 00 09 [CRC]
Response: 05 01 02 01 00 [CRC]
Reads the protocol being used by the module.

3.2 03 (0x03) Read Multiple Registers

This function code is used to read the current configuration values and GPS data from the GPS-721-MRTU module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x03
02-03	Starting Channel Number or Address Mapping	2 Bytes	Refer to the Modbus Address Mapping Table for more information.
04-05	Input Channel Number	2 Bytes	0x0001-0x0020 (Word count)

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x03
02	Byte Count	1 Byte	The Byte Count of the Response (B=2 * Word Count)
03-	Register Values	(B*2) Bytes	Register values

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x83
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 05 03 01 E2 00 02 [CRC]

Response: 05 03 04 07 21 00 47 [CRC]

Reads the name of the module

[Command]

Bytes 2 to 3: 01 E2 (Address mapping)

Bytes 4 to 5: 00 02 (Word count)

[Response]

Byte 2: 04 (The Byte count of the response)

Bytes 3 to 6: 07 21 00 47

07 21 (The Low word of the module name)

00 47 (The High word of module name)

Command: 05 03 01 E4 00 01 [CRC]

Response: 05 03 02 00 05 [CRC]

Reads the address of the module.

Command: 05 03 01 E7 00 01 [CRC]

Response: 05 03 02 00 00 [CRC]

Reads the Modbus response delay time.

Command: 05 03 01 EB 00 01 [CRC]

Response: 05 03 02 00 00 [CRC]

Reads the Host Watchdog timeout value.

Command: 00 03 30 38 00 01 [CRC]

Response: No response

Informs all modules that the Host is OK.

3.4 04 (0x04) Read Multiple Registers

This function code is used to read the current configuration values and GPS data from the GPS-721-MRTU module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x04
02-03	Starting Channel Number or Address Mapping	2 Bytes	Refer to the Modbus Address Mapping Table for more information.
04-05	Input Channel Number	2 Bytes	0x0001-0x0020 (Word count)

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x04
02	Byte Count	1 Byte	The Byte Count of the Response (B=2 * Word Count)
03-	Register Values	(B*2) Bytes	Register values

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x84
02	Exception Code	1 Byte	Refer to the Modbus standard for more details

Examples

Command: 05 04 01 E2 00 02 [CRC]
Response: 05 04 04 07 21 00 47 [CRC]
Reads the name of the module.

Command: 01 04 01 E4 00 01 [CRC]
Response: 01 04 02 00 01 [CRC]
Reads the address of the module.

Command: 05 04 01 E7 00 01 [CRC]
Response: 05 04 02 00 00 [CRC]
Reads the Modbus Response Delay Time.

Command: 05 04 01 EB 00 01 [CRC]
Response: 05 04 02 00 00 [CRC]
Reads the Host Watchdog timeout value.

Command: 00 04 30 38 00 01 [CRC]
Response: No response
Informs all modules that the Host is OK.

3.5 05 (0x05) Write a Single Coil

This function code is used to write the Digital Output value for the GPS-721-MRTU module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x05
02-03	Output Channel Number	2 Bytes	Refer to the Modbus Address Mapping Table for more information.
04-05	Output Value	2 Bytes	A value of 0xFF00 sets the output to ON A value of 0xFF00 sets the output to OFF.

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x05
02-03	Output Channel Number	2 Bytes	The value is the same as bytes 02 and 03 of the Request.
04-05	Output Value	2 Bytes	The value is the same as bytes 04 and 05 of the Request.

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x85
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 05 05 00 80 FF 00 [CRC]

Response: 05 05 00 80 FF 00 [CRC]

Sets the power-on value for Digital Output channel 0 to ON.

Command: 05 05 00 00 FF 00 [CRC]

Response: 05 05 00 00 FF 00 [CRC]

Sets Digital Output channel 0 to ON.

Command: 05 05 01 03 FF 00 [CRC]

Response: 05 05 01 03 FF 00 [CRC]

Sets the Host Watchdog mode to Mode 1.

Command: 05 05 01 03 00 00 [CRC]

Response: 05 05 01 03 00 00 [CRC]

Sets the Host Watchdog mode to Mode 0.

Command: 05 05 01 04 FF 00 [CRC]

Response: 05 05 01 04 FF 00 [CRC]

Enables the Host Watchdog.

Command: 05 05 01 0D FF 00 [CRC]

Response: 05 05 01 0D FF 00 [CRC]

Clears the status of the Host Watchdog

Command: 05 05 01 00 FF 00 [CRC]

Response: 05 05 01 00 FF 00 [CRC]

Sets Bit 0 of the communication protocol.

[Command]

Bytes 2 to 3: 01 00 [Protocol (Bit 0)]

Bytes 4~5: FF 00

Protocol (Bit 1 and Bit 0)

01	00
Modbus RTU	DCON

Command: 05 05 01 01 FF 00 [CRC]

Response: 05 05 01 01 FF 00 [CRC]

Sets Bit 1 of the communication protocol.

[Command]

Bytes 2 to 3: 01 01 [Protocol (Bit 1)]

Bytes 4 to 5: FF 00

Protocol (Bit 1 to Bit 0)

01	00
Modbus RTU	DCON

3.6 06 (0x06) Write Multiple Registers

This function code is used to configure the settings for the GPS-721-MRTU module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x06
02-03	Address Mapping	2 Bytes	Refer to the Modbus Address Mapping Table for more information.
04-05	Register Values	2 Bytes	Register values

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x06
02-03	Address Mapping	2 Bytes	The value is the same as bytes 02 and 03 of the Request.
04-05	Register Values	2 Bytes	Register values

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x86
02 -	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command:05 06 01 E7 00 10 [CRC]

Response:05 06 01 E7 00 10 [CRC]

Sets the Modbus Response Delay Time (0 to 30, in 1 ms intervals).

[Command]

Bytes 2 to 3: 01 E7 (Address mapping)

Bytes 4 to 5: 00 10 (The Response Delay Time = 16 ms)

[Response]

Byte 1: 06 (Function Code)

Command:05 06 01 E8 00 C8 [CRC]
Response:05 06 01 E8 00 C8 [CRC]
Sets the Host Watchdog timeout value (0 to 255, in 0.1 second intervals).

Command:09 06 01 EB 00 00 [CRC]
Response:09 06 01 EB 00 00 [CRC]
Clears the Host Watchdog timeout counter.

3.7 15 (0x0F) Write Multiple Coils

This function code is used to write the Digital Output values for the GPS-721-MRTU module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x0F
02-03	Starting Channel Number	2 Bytes	Refer to the Modbus Address Mapping Table for more information.
04-05	Output Channel Number	2 Bytes	0x0001-0x0020 (Bit count)
06	Byte Count	1 Byte	B=(bit count + 7)/8
07	Output Value	2 Bytes	A bit corresponds to a channel. When the bit is 0, it denotes that the value of the channel that was set is OFF. If the bit is 1, it denotes that the value of the channel that was set is ON.

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x0F
02-03	Starting Channel Number	2 Bytes	The value is the same as bytes 02 and 03 of the Request.
04-05	Input Channel Number	2 Bytes	0x0001-0x0020

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x8F
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 05 0F 00 A1 00 01 01 01 [CRC]

Response: 05 0F 00 A1 00 01 [CRC]

Sets the Safe value for Digital Output channel 0.

Command: 05 0F 00 00 00 01 01 01 [CRC]

Response: 05 0F 00 00 00 01 [CRC]

Sets the Digital Output value for Digital Output channel 0.

3.8 46 (0x46) Read/Write the Module Settings

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

Sub-function Code	Description	Section
00 (0x00)	Reads the name of the module	3.8.1
04 (0x04)	Sets the address of the module	3.8.2
05 (0x05)	Reads the communication settings	3.8.3
06 (0x06)	Sets the communication settings	3.8.4
32 (0x20)	Reads the Firmware Version information	3.8.5
39 (0x27)	Sets the Digital Output Power-on value	3.8.8
40 (0x28)	Reads the Digital Output Power-on value	3.8.9
41 (0x29)	Sets the active status for the Digital Input and Digital Output	3.8.10
42 (0x2A)	Reads the active status for the Digital Input and Digital Output	3.8.11
53 (0x35)	Reads the Response Delay Time	3.8.12
54 (0x36)	Sets the Response Delay Time	3.8.13

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

3.8.1 00 (0x00) Read the Name of the Module

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

Request

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

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x00
03-06	Module Name	4 Bytes	0x07 0x21 0x00 0x47 for the GPS-721-MRTU.

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 05 46 00 [CRC]

Response: 05 46 00 07 21 00 47 [CRC]

Reads the name of the module.

3.8.2 04 (0x04) Set the Address of the Module

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

Request

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

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x04
03	Confirmation	1 Byte	0: OK Others: Error
04-06	Reserved	3 Bytes	0x00 0x00 0x00

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 01 46 04 02 00 00 00 [CRC]

Response: 02 46 04 00 00 00 00 [CRC]

Sets the address of the module.

3.8.3 05 (0x05) Read the Communication Parameters

This sub-function code is used to read the configuration of the communication parameters used for a module.

Request

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

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x05
03	Protocols Supported	1 Byte	0,1: DCON/Modbus RTU 3: DCON/Modbus RTU/ASCII
04	Baud Rate	1 Byte	0x00 to 0x0A (Refer to the Modbus Address Mapping table for details).
05	Reserved	1 Byte	0x00
06	Parity	1 Byte	0: N81 1: N82 2: E81 3: O81
07	Reserved	1 Byte	0x00
08	Protocol Mode	1 Byte	0: DCON 1: Modbus RTU
09-10	Reserved	2 Bytes	0x00 0x00

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for

			more details.

Examples

Command: 02 46 05 00 [CRC]

Response: 02 46 05 03 0A 00 00 00 01 00 00 [CRC]

Reads the communication protocol currently being used by the module (Modbus RTU).

3.8.4 06 (0x06) Set the Communication Parameters

This sub-function code is used to configure the communication parameters to be used by a module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x06
03	Reserved	1 Byte	0x00
04	Baud Rate	1 Byte	0x00 to 0x0A (Refer to the Modbus Address Mapping table for details).
05	Reserved	1 Byte	0x00
06	Parity	1 Byte	0: N81 1: N82 2: E81 3: O81
07	Reserved	1 Byte	0x00
08	Protocol Mode	1 Byte	0: DCON 1: Modbus RTU
09-10	Reserved	2 Bytes	0x00 0x00

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x06
03	Reserved	1 Byte	0x00
04	Baud Rate	1 Byte	0: OK Others: Error
05	Reserved	1 Byte	0x00
06	Parity	1 Byte	0: OK Others: Error
07	Reserved	1 Byte	0x00

08	Protocol Mode	1 Byte	0: OK Others: Error
09-10	Reserved	2 Bytes	0x00 0x00

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 02 46 06 00 09 00 02 00 03 00 00 [CRC]

Response: 02 46 06 00 00 00 00 00 00 00 00 [CRC]

Sets the configuration for module 02.

3.8.5 32 (0x20) Read the Firmware Version

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

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-Function Code	1 Byte	0x20

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x20
03	Major Version	1 Byte	0x00-0xFF
04	Minor Version	1 Byte	0x00-0xFF
05	Build Version	1 Byte	0x00-0xFF

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 01 46 20 [CRC]

Response: 05 46 20 02 00 00 [CRC]

Reads the firmware version information.

3.8.8 39 (0x27) Set the Power-on Value

This sub-function code is used to set the Power-on value for a module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x27
03	Power-on Value	1 Byte	0x00 (DO0)

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x27
03	Power-on Value	1 Byte	0 : OK Others : Error

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command:05 46 27 01 [CRC]
Response:05 46 27 00 [CRC]
Sets the Power-on value for Digital Output channel 0.

Command:05 46 28 [CRC]
Response:05 46 28 01 [CRC]
Reads the Power-on value for Digital Output channel 0.

3.8.9 40 (0x28) Read the Power-on Value

This sub-function code is used to read the power-on value for Digital Output channels 0 to 31of a module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x28

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x28
03	Power-on Value	1 Byte	0x00-0xFF (DOI0-DO7)
04	Power-on Value	1 Byte	0x00-0xFF (DO8-DO15)
05	Power-on Value	1 Byte	0x00-0xFF (DO16-DO23)
06	Power-on Value	1 Byte	0x00-0xFF (DO24-DO31)

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command:05 46 28 [CRC]

Response:05 46 28 00 [CRC]

Reads the Power-on value for Digital Output channel 0.

3.8.10 41 (0x29) Set the DI/O Active Status

This sub-function code is used to set the active state of the DI/O for a module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x29
03	DI/O Active Status	1 Byte	0x00-0x03

7	6	5	4	3	2	1	0
Reserved						OAS	IAS

Key	Description
IAS (Bit 0)	Specifies the active state of the Digital Input signal: 0. An input value of 0 sets the voltage to high An input value of 1 sets the Digital Input to no signal or low voltage 1. An input value of 0 sets the Digital Input to no signal or low voltage An input value of 1 sets the voltage to high
OAS (Bit 1)	Specifies the active state of the Digital Output signal: 0. An output value of 0 sets the relay to inactive An output value of 1 sets the relay to active 1. An output value of 0 sets the relay to active An output value of 1 sets the relay to inactive

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x29
03	DI/O Active Status	1 Byte	0 : OK Others : Error

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command:01 46 29 02 [CRC]
Response:01 46 29 00 [CRC]
Sets the active state for the DI/O.

Command:01 46 2A [CRC]
Response:01 46 2A 02 [CRC]
Reads the active state for the DI/O.

3.8.11 42 (0x2A) Read the DI/O Active Status

This sub-function code is used to read the active status of the DI/O for a module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x2A

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x2A
03	DI/O Active Status	1 Byte	0x00-0x03

7	6	5	4	3	2	1	0
Reserved						OAS	IAS

Key	Description
IAS (Bit 0)	Specifies the active state of the Digital Input signal 0. An input value of 0 indicates that the voltage is high An input value of 1 indicates that there is no signal or that the voltage is low 1. An input value of 0 indicates that there is no signal or that the voltage is low An input value of 1 indicates that the voltage is high
OAS (Bit 1)	Specifies the active state of the Digital Output signal 0. An output value of 0 indicates that the relay is inactive An output value of 1 indicates that the relay is active 1. An output value of 0 indicates that the relay is active An output value of 1 indicates that the relay is inactive

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details

Examples

Command:01 46 29 02 [CRC]

Response:01 46 29 00 [CRC]

Sets the active state for the DI/O.

Command:01 46 2A [CRC]

Response:01 46 2A 02 [CRC]

Reads the active state for the DI/O.

3.8.12 53 (0x35) Read the Response Delay Time

This sub-function code is used to read the Modbus Response Delay Time for a module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x35

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x35
03	Response Delay Time	1 Byte	(Data)

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 01 46 36 1E [CRC]
Response: 01 46 36 1E [CRC]
Sets the Modbus Response Delay Time to 30 ms

Command: 01 46 35 [CRC]
Response: 01 46 35 1E [CRC]
Reads the Modbus Response Delay Time (30 ms)

3.8.13 54 (0x36) Set the Response Delay Time

This sub-function code is used to set the Modbus Response Delay Time for a module.

Request

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x36
03	Response Delay Time	1 Byte	(Data)

Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0x46
02	Sub-function Code	1 Byte	0x36
03	Response Delay Time	1 Byte	(Data)

Error Response

00	Address	1 Byte	1 to 247
01	Function Code	1 Byte	0xC6
02	Exception Code	1 Byte	Refer to the Modbus standard for more details.

Examples

Command: 01 46 36 1E [CRC]

Response: 01 46 36 1E [CRC]

Sets the Modbus Response Delay Time to 30 ms.

Command: 01 46 35 [CRC]

Response: 01 46 35 1E [CRC]

Reads the Modbus Response Delay Time (30 ms).

4 Appendix 1 - Dual Watchdog Operation

Dual Watchdog = Module Watchdog + Host Watchdog

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

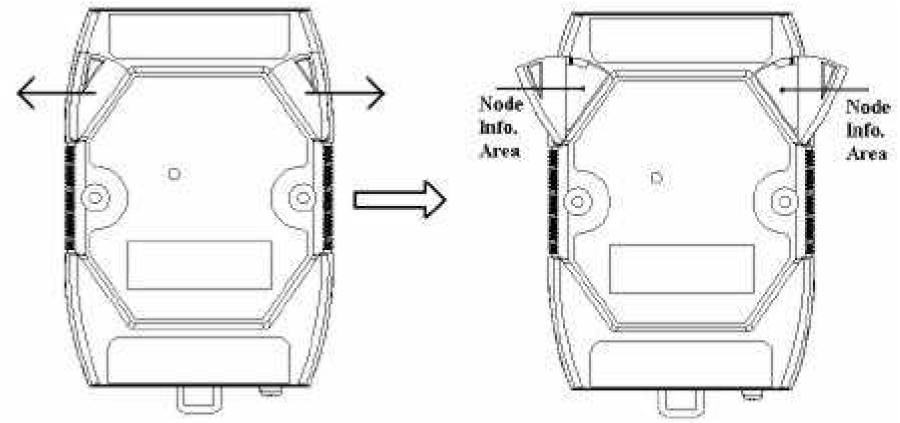
The Host Watchdog is a software function that monitors the operating status of the Host. Its purpose is to prevent problems due to network/communication errors or Host malfunctions. When a Host Watchdog timeout occurs, the module will reset all outputs to a safe state in order to prevent any erroneous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

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

For more information regarding the Dual Watchdog system, refer to Chapter 5 of the “Getting Started for I-7000 Series Modules” manual that can be downloaded from the ICP DAS website at http://www.icpdas.com.

5 Appendix 2 - Node Information Area

ICP DAS modules contain a built-in EEPROM that can be used to store configuration information, such as the module address, the Type Code, and the Baud Rate, etc. One minor drawback is that there may be no visual indications of the configuration of the module. New modules include “Node Information” areas that are protected by a cover, as shown below, and can be used to make a written record of the node information, such as the module address and the Baud Rate, etc. To access the Node Information areas, slide the covers outward, as shown in the figure below.


Node
Info.
Area
Node
Info.
Area