

tM Series User Manual V1.1, 2023/07



Technical support: <a>service@icpdas.com Author: Wayne Chen, David Chen Editor: Janice Hong

Warranty

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

Warning

ICP DAS assumes no liability for any damage 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, not for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Copyright @ 2019 by ICP DAS Co., Ltd. All rights are reserved.

Trademark

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

Contact US

If you have any problem, please feel free to contact us. You can count on us for quick response. Email: <u>service@icpdas.com</u>

Revision History

This chapter provides revision history information to this document.

The table below shows the revision history.

Revision	Date	Description	
V1.1	Jul. 2023	Modify the download link in Section 6.4.1	
V1.1	Dec. 2020	Add Appendix G – Save or Load I/O Project	
V1.0	Jul. 2020	Initial issue	

Content of Table

Revision History	3
Content of Table	4
Chapter 1 Introduction	8
1.1 Features	9
1.2 Selection Guide	10
1.3 Specifications	11
1.3.1 Common Specification	11
1.3.2 Specification - AI	12
1.3.3 Specification - DI/DO	14
1.3.4 Specification - Multifunction I/O	17
1.4 Supported Software	19
1.4.1 DCON Utility Pro	19
1.4.2 EZ Data Logger	19
1.4.3 OPC Server	20
1.4.4 PacSDK	20
Chapter 2 Hardware Installation	21
2.1 Mounting	21
2.2 Dimensions	21
2.3 Wiring Connections	22
2.3.1 Wiring Recommendations	22
2.3.2 tM-AD2 Wiring	22
2.3.3 tM-AD5 Wiring	23
2.3.4 tM-AD5C Wiring	23
2.3.5 tM-AD8 Wiring	24
2.3.6 tM-AD8C Wiring	24
2.3.7 tM-TH8 Wiring	25
2.3.8 tM-R5 Wiring	25
2.3.9 tM-P3R3 Wiring	26
2.3.10 tM-PD3R3 Wiring	27
2.3.11 tM-P3POR3 Wiring	
2.3.12 tM-P4A4 Wiring	
2.3.13 tM-P4C4 Wiring	
2.3.14 tM-P8 Wiring	
2.3.15 tM-PDW8 Wiring	
2.3.13 (14) 1 2 400 44 (11) 5	

2.3.16 tM-C8 Wiring	33
2.3.17 tM-DA1P1R1 Wiring	34
2.3.18 tM-AD2P2C2 Wiring	35
2.4 Hardware Connection	36
2.5 Operating Switch and Parameters	37
Chapter 3 Software Tool - DCON Utility Pro	
3.1 Search I/O Modules	
3.2 "Configuration" Page	40
3.2.1 Common Settings for tM-P3R3, PD3R3, P3POR3, P4A4, P4C4, R5, C8, P8,	, and PDW841
3.2.2 AI items for tM-AD2 and AD4P2C2	42
3.2.3 AI Items for tM-AD5, AD8, AD5C, and AD8C	43
3.2.4 AI Items for tM-TH8 and DA1P1R1	44
3.3 "AI" Page	45
3.3.1 AI Page01 for tM-AD2	45
3.3.2 AI Page02 for tM-AD5, AD8, AD5C, and AD8C	46
3.3.3 AI Page03 for tM-TH8	47
3.4 Other I/O Pages	48
3.4.1 Configuration Page - User Defined Type	48
3.4.2 Configuration Page - AO	50
3.4.3 Configuration Page - AI/DO Alarm	52
3.4.4 Configuration Page - DO	54
3.4.5 Configuration Page - Host WDT	55
3.4.6 Configuration Page - DI	57
3.5 Command Line Tool	58
3.6 Calibration	59
3.6.1 Calibrate AO Module	59
Chapter 4 Software Tool – Modbus Master Tool	61
Chapter 5 Software Tool – EZ Data Logger	65
5.1 System Configuration	67
5.2 Workgroup Setting	
5.2.1 Install the Driver	69
5.2.2 Add a Device	70
5.2.3 Modify the Channel Settings or Add a Workgroup	71
5.2.4 Configure the Workgroup Settings	74
5.3 Start to Run (or Simulation)	75
5.3.1 Gauge	76

	5.3.2 Trend Line	77
	5.3.3 Layout	78
5.4	Open Database	80
Chapter 6	6 Software Development Tool	83
6.1	Using DCON Utility Pro	83
	Using PACSDK	
	- Modbus Development Tools and SCADA	
	6.3.1 Software Tool – OPC DA Server and UA Series Product	89
	6.3.2 Software Tool – nModbus	95
	6.3.3 Software Tool – LabVIEW	
	6.3.4 Software Tool – InduSoft	
6.4	Linux Modbus Development Tool	105
	6.4.1 Modbus tool for Linux	105
	6.4.2 Using LinPAC and the Python Application	
	6.4.3 Using LinPAC and the Perl Application	
6.5	Using Win-GRAF Development Software	114
Appendix	A Function Description	118
A.1	INIT Mode	
A.2	Dual Watchdog	119
	A.2.1 Power-on Value	120
	A.2.2 Safe Value	
A.3	Frame Ground	121
A.4	Slew Rate Control	122
A.5	AO Read-back Value	122
A.6	Advanced DI Features (Latch and Counter)	123
Appendi	KB Type Codes and Ranges for AI/AO Data	
B.1	Data Ranges for tM-AD2	125
B.2	Data Ranges for tM-AD5	126
В.З	Data Ranges for tM-AD5C	126
B.4	Data Ranges for tM-AD8	127
B.5	Data Ranges for tM-AD8C	127
B.6	Data Ranges for tM-AD4P2C2	
B.7	Data Ranges for tM-DA1P1R1	128
B.8	Data Ranges for tM-TH8	
	B.8.1 Steinhart–Hart Resistance (User-defined Type)	131
Appendi	C Modbus Register Mapping	
Appendi	C Modbus RTU Protocol Commands	
D.1	01 (0x01) Read Coil	137

D.2	02 (0x02) Read Discrete Inputs	.138
D.3	03 (0x03) Read Multiple Registers	
D.4		
D.5	05 (0x05) Force Single Coil	
D.6	06 (0x06) Preset Single Register	
D.7	15 (0x0F) Force Multiple Coils	
D.8	16 (0x10) Set Multiple Register	
Appendix	E DCON Protocol Commands	.145
E.1	General Commands	.147
E.2	I/O Commands	.147
E.3	Host Watchdog, Power-on Value, and Safe Value Commands	.148
E.4	DI Latched and Counter Commands	.149
E.5	Alarm Commands	.149
E.6	Module Calibration Commands	.150
E.7	User-defined Type Commands (for Thermistor)	.150
Appendix	F Quickly Save and Apply I/O Configurations	.238
F.1	Set, Write, and Save I/O Configurations	.239
F.2	Configure the I/O Module by a Configuration File	.242
Appendix	G Save or Load I/O Project	.245
G.1	Save Project	.245
G.2	Load Project	.248
Appendix	H Other Information	.252
H.1	How do I Search the Product Webpage?	.252
H.2	How do I Quickly Navigate a Specific Chapter?	.253

Chapter 1 Introduction

The tM series is a family of network data acquisition and control modules that are equipped with digital or analog input/output functions. The modules can be remotely controlled through an RS-485 serial bus by using DCON and Modbus RTU protocols. The selectable transmission speed of the RS-485 port is up to 115,200 bps.

Modbus has become a de facto standard communications protocol in industry and is now the most commonly available means of connecting industrial electronic devices. This makes the tM series perfect integration with the HMI, SCADA, PLC, and other software systems.

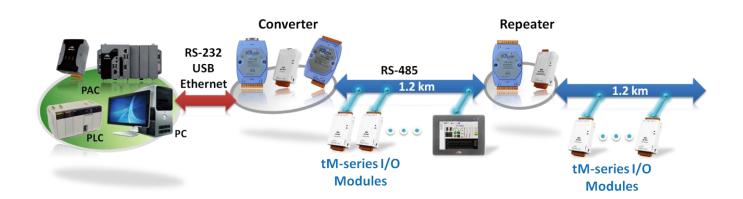


The tM series tiny RS-485 I/O modules support a variety of I/O types, like photo-isolated digital input, relay contact, photoMOS relay, and open-collector output. Compare to M-7000 series, the small number of channels designed for tM series is more cost-effective for distributed I/O applications.

The tM series provides the module watchdog and host watchdog to enhance the reliability and stability of the control system. For maximum space savings, the tM series is offered in an amazing tiny form-factor that makes it easily installed anywhere, even directly embedded into a machine. It is equipped with two removable terminal block connectors for easy wiring.

1.1 Features

- Tiny form-factor and Low Power Consumption
- Protocol supported:
 DCON, Modbus RTU/ASCII
- Supports RS-485 Multi-drop Network
 RS-485 bus supports baud rate up to 115200 bps



- Configurable I/O Types and Ranges
- Supports Dual-Watchdog
- Configurable Power-on Value and Safe Value
- Provides DI Latched Function
- Provides Low Speed Counter
- Isolated Digital Input and Output
- RoHS Compliant with no Halogen
- Easy DIN-Rail Mounting



1.2 Selection Guide

tM Series modules (Tabulating Date: 06, 20		ate: 06, 2019)		
Bus	RS-485			
Protocol	Modbus RTU, Modbus ASCII, DCON			
Models	AI AO DI DO			
tM-AD2	2-ch Single-ended, Voltage/Current			
tM-AD5	5-ch Differential, Voltage	- ch Differential, Voltage		
tM-AD5C	5-ch Differential, Current			
tM-AD8	ch Single-ended, Voltage			-
tM-AD8C	8-ch Single-ended, Current			
tM-TH8	8-ch Thermistor			

Models	AI	AO	DI	DO		
+14 DA1D1P1	tM-DA1P1R1 -		1-channel			
	-	Voltage/Current	Sink/Source	Form C Relay		
	4-channel		2-cha	annel		
tM-AD4P2C2	2-ch Single-ended, Voltage	-	C	NDN Cick		
	2-ch Single-ended, Current		Source	NPN, Sink		
			3-cha	annel		
tM-P3R3			Sink/Source	Form A Relay		
tM-PD3R3	-		Source + Dry			
tM-P3POR3			Sink/Source	PhotoMos Relay		
tM-P4A4			4-cha	annel		
tivi-P4A4	-		Sink	PNP, Source		
tM-P4C4			Source	NPN, Sink		
tM-R5				5-channel		
	-		-	Form A Relay		
tM-P8			8-channel			
(11)-FO			Sink/Source	-		
tM-PDW8	-		Sink/Source + Dry			
tM-C8				8-channel		
			-	NPN, Sink		

1.3 Specifications

The specifications of all tM series will be described in this section.

1.3.1 Common Specification

Communication			
Interface	RS-485		
Format	(N, 8, 1), (N, 8, 2), (O, 8, 1), (E, 8, 1)		
Baud Rate	1200 to 115200 bps		
Protocol	DCON, Modbus/RTU, Modbus/ASCII		
Dual Watchdog	Yes, Module (2.3 seconds), Communication (Programmable)		
LED Indicators			
Power	1 LED as Power Indicator		
EMS Protection			
	±4 kV Contact for Each Terminal		
	±8 kV Air for Random Point		
ESD (IEC 61000-4-2)	<u>tM-TH8:</u>		
	±2 kV Contact for Each Terminal		
	±3 kV Air for Random Point		
EFT (IEC 61000-4-4)	±4 kV for Power		
Isolation			
	Analog Input: 2500 VDC (Multi-function I/O: tM-AD4P2C2,		
Tatus modulo Isolation Field to Logic	tM-DA1P1R1)		
Intra-module Isolation, Field-to-Logic	Digital Input: 3750 VDC		
	PhotoMOS Relay Output: 2000 VDC		
Mechanical			
Dimensions (W x L x H)	52 mm x 98 mm x 27 mm		
Installation	DIN-Rail Mounting		
Temperature Range			
Operating	-25 °C to +75 °C		
Storage	-30 °C to +75 °C		
Humidity	10 to 95% RH, Non-condensing		

1.3.2 Specification - AI

Module Name		tM-AD2	tM-AD5	tM-AD5C	tM-AD8	tM-AD8C		
Analog Input					1	•		
		2	5	5	8	8		
Input Channels		single-ended	differential	differential	single-ended	single-ended		
Туре		0 to 500 mV 0 to 1 V 0 to 2.5 V 0 to 5 V 0 to 10 V 0 to 20 mA 4 to 20 mA	+/- 1 V +/- 2.5 V +/- 5 V +/- 10 V	0 to 20 mA 4 to 20 mA +/- 20 mA	0 to 500 mV 0 to 1 V 0 to 2.5 V 0 to 5 V 0 to 10 V	0 to 20 mA 4 to 20 mA		
Resolution	Normal Mode	14-bit	·					
Resolution	Fast Mode	12-bit						
Compling Data	Normal Mode	10 Hz total						
Sampling Rate	Fast Mode	200 Hz total						
	Normal Mode	± 0.1%	± 0.1%					
Accuracy	Fast Mode	± 0.5%						
Zero Drift		± 20 μV/°C						
Span Drift		± 25 ppm/°C						
Input Impedance	Voltage	10 MΩ	20 MΩ	-	10 MΩ	-		
Input Impedance	Current	125 Ω	-	125 Ω	-	125 Ω		
	Voltage	120 VDC		-	120 VDC	-		
Over Protection	Current	Yes, 50 mA at 110 VDC	-					
Open Wire Detection for 4 to 20 mA		Yes -						
Power Requireme	nts							
Reverse Polarity Protection		Yes						
Powered from Term	Powered from Terminal Block		0 to 30 VDC					
Consumption		0.7 W Max.	1 W Max. 0.7 W Max.					

Module Name	tM-TH8
Analog Input	
Input Channels	8
Туре	Thermistor
Thermistor Type	Precon ST-A3, Fenwell Type U, YSI L 100, YSI L 300, YSI L 1000, YSI B 2252, YSI B 3000, YSI B 5000, YSI B 6000, YSI B 10000, YSI H 10000, YSI H 30000, User-defined
Resolution	16-bit
Sampling Rate	8 Hz total
Accuracy	± 1%
Over-voltage Protection	120 VDC
Open Wire Detection	Yes
Individual Channel Configurable	Yes
Power Requirements	
Reverse Polarity Protection	Yes
Powered from Terminal Block	Yes, 10 to 30 VDC
Consumption	1 W Max.

1.3.3 Specification - DI/DO

Module Name		tM-P8	tM-PDW8	
Digital Input	/Counter	·		
Input Channels	5	8		
Туре		Wet Contact (Sink, Source)	Wet Contact (Sink, Source)	
.,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,			Dry Contact (Source)	
Wet Contact	ON Voltage Level	+3.5 VDC to 50 VDC		
	OFF Voltage Level	+1 VDC Max.		
Dry Contract	ON Voltage Level	-	Close to GND	
Dry Contact	OFF Voltage Level	-	Open	
	Max. Count	65535 (16-bit)		
Counters	Max. Input Frequency	100 Hz		
	Min. Pulse Width	5 ms		
Input Impedar	ice	10 KΩ, 0.5 W		
Over-voltage P	rotection	±70 VDC		
Power Requi	rements			
Reverse Polarity Protection		Yes		
Powered from Terminal Block		Yes, 10 to 30 VDC		
Consumption		0.2 W Max. 0.43 W Max.		

Module Name		tM-P3POR3	tM-P3R3	tM-PD3R3
Digital Input/	Counter			
Input Channels		3		
Туре		Wet Contact (Sink	, Source)	Dry Contact (Source)
Wat Contact	ON Voltage Level	+3.5 VDC to +50	VDC	-
	Wet Contact OFF Voltage Level			-
ON Voltage Level		-		Close to GND
	Dry Contact OFF Voltage Level			Open
	Max. Count	65535 (16-bit)		
Counters	Counters Max. Input Frequency		100 Hz	
Min. Pulse Width		5 ms		
Input Impedance		10 KΩ, 0.5W		-
Over-voltage Protection		70 VDC		-

Module Name		tM-P3POR3	tM-P3R3	tM-PD3R3	tM-R5
Relay Output					
Output Channels		3	5		5
Relay Type		PhotoMOS Relay	Power Relay, Form A (SPST N.O.)		
Operating Load Volt Range	tage	80 V (AC peak or DC)	250 VAC or 30 VDC		
Lood Current		1 A Max.	5 A Max.	_	
Load Current		3 A, (Peak, 1 ms, 1 shot)	S A Max.		
Output Off State Le Current	akage	1 uA	-		
Operating Time		5 ms (Max.)	6 ms		
Release Time		0.5 ms (Max.)	3 ms		
	VDE		5 A @250 VAC 30,000 ops (10 ops/minute) at 75°C		
Electrical Life	VDE		5 A @30 VDC 70,000 ops (10 ops/minute) at 75°C		
(Resistive load)	UL	-	5 A @250 VAC/30 VDC 6,000 ops		
	UL		3 A @250 VAC/30 VDC 100,000 ops		
Mechanical Life		-	20,000,000 ops at no load (300 ops/minute)		
Electrical Endurance	9	No Arcing, No Bounce, and No Switching Noise	-		
Power-on Value		Yes, Programmable			
Safe Value		Yes, Programmable			
Power Requireme	ents				
Reverse Polarity Pro	otection	Yes			
Powered from Terminal Block		Yes, 10 to 30 VDC			
Consumption		0.4 W Max.	0.8 W Max.	0.9 W Max.	1 W Max.

Module Na	me	tM-P4A4	tM-P4C4	tM-C8
Digital Inp	ut/Counter			
Input Chann	els	4		
Туре		Wet Contact (Sink)	Wet Contact (Source)	
On Voltage L	evel	+4.0 VDC to +50 VDC	+3.5 VDC to +50 VDC	
Off Voltage L	evel	+1 VDC Max.		
Input Imped	ance	10 KΩ, 0.66 W		No
	Channels	4		Digital Input
Countons	Max. Count	65535 (16-bit)		
Counters	Max. Frequency	100 Hz		
	Min. Pulse Width	5 ms		
Over-voltage	Protection	70 VDC		
Digital Out	put			
Output Chan	nels	4		8
Туре		Isolated Open Emitter (Source)	Isolated Open Collector (Sink)	
Max. Load C	urrent	650 mA/channel	700 mA/channel	
Load Voltage	2	+10 VDC to +40 VDC	+3.5 to +50 VDC	
Over-voltage	Protection	47 VDC	60 VDC	
Overload Pro	otection	Yes		1.4 A
Short Circuit	Protection	Yes		
Power-on Va	lue	Yes, Programmable		
Safe Value		Yes, Programmable		
Isolation				
Intra-module Isolation,		3750 VDC		
Field-to-Logi	с			
Power Req	uirements	Γ		
Reverse Pola	arity Protection	Yes		
Powered from	m Terminal Block	Yes, 10 to 30 VDC		
Consumptior	ı	0.5 W Max.		0.8 W Max.

1.3.4 Specification - Multifunction I/O

Module Name	AI	AO	DI	DO
tM-DA1P1R1	-	1	1	Relay: 1
tM-AD4P2C2	Voltage: 2 ; Current: 2	-	2	2

Module Name		tM-DA1P1R1	tM-AD4P2C2
Analog Input			
Input Channel	Voltage		2 cingle and d
Input Channel	Current		2 single-ended
Turno	Voltage		±1 V, ±2.5 V, ±5 V, ±10 V
Туре	Current		±20 mA, 0 to 20 mA, 4 to 20 mA
Resolution	Normal Mode		14-bit
Resolution	Fast Mode		12-bit
Sampling Data	Normal Mode		10 Hz total
Sampling Rate	Fast Mode	No Analog Input	200 Hz total
Acquiració	Normal Mode	- No Analog Input 	± 0.1%
Accuracy	Fast Mode		± 0.5%
Zero Drift			±20 μV/°C
Span Drift			±25 ppm/°C
Input Impedance	Voltage		10 MΩ
Input Impedance	Current		136 Ω
Over Protection	Voltage		120 VDC
Over Protection	Current		Yes, 50 mA at 110 VDC
Analog Output			
Output Channels		1	
Туре		0 to 10 V, 0 to 20 mA, 4 to 20 mA	
Resolution		12-bit	
Accuracy		+/-0.1% of FSR	No Analog Output
DA Output Response Time		10 ms	
Voltage Output Cap	ability	20 mA	
Current Load Resist	ance	500 Ω	

Module Name		tM-DA1P1R1	tM-AD4P2C2
Digital Input	/Counter		
Input Channel		1	2
T	Dry Contact	Source	-
Туре	Wet Contact	Sink/Source	Source
	ON Voltage Level	+3.5 VDC to +50 VDC	
Wet Contact	OFF Voltage Level	+1 VDC Max.	
Due Combo de	ON Voltage Level	Close to GND	
Dry Contact	OFF Voltage Level	Open	
Input Impedan	ice	10 KΩ, 0.5 W	
	Channels	1	2
Countration	Max. Count	65535 (16-bit)	
Counters	Max. Input Frequency	100 Hz	50 Hz
	Min. Pulse Width	5 ms	10 ms
Over-voltage P	rotection	70 VDC	
Digital/Relay	/ Output		
Channels		1	2
Туре		Power Relay, Form C	Isolated Open Collector (Sink)
Max Land Curr		NO: 10 A @ 250 VAC	
Max. Load Cur	rent	NC: 6 A @ 250 VAC	700 mA/Channel
Load Voltage		-	+3.5 VDC to +50 VDC
Over Dretection	voltage		60 VDC
Over Protection	Current] -	1.4 A (with short circuit protection)
Short Circuit P	rotection		Yes
Operating Load	d Voltage Range	250 VAC or 30 VDC	
Operating Time	e	15 ms Max.	
Release Time		5 ms Max.	-
Mechanical En	durance	1 X 10 ⁷ OPS	
Electrical Endu	rance	5 X 10 ⁴ OPS	
Power On Value		Yes, Programmable	
Safe Value		Yes, Programmable	
Power Requi	rements		
Reverse Polarit	y Protection	Yes	
Powered from	Terminal Block	Yes, 10 to 30 VDC	
Consumption		1.8 W Max.	1 W Max.
		1	I

1.4 Supported Software

ICP DAS provides some free software tool and development tools as follows, including:

1.4.1 DCON Utility Pro



DCON Utility Pro is used to search, configure, and test the I/O module through the RS-232/485 port. It supports DCON and Modbus protocols and can run on a Windows PC or a PAC. (Refer to <u>Chapter 3</u> for more information.)

1.4.2 EZ Data Logger



EZ Data Logger is the software that ICP DAS provides for users to easily build a SCADA system on Windows 2000/XP/Vista. It comes with two versions, "Lite" & "Professional". The Lite version is not only full-functioned but free to all ICP DAS users! (Refer to <u>Chapter 4</u> for more information.)

EZ Data Logger is a small data logger software. It can be applied to small scale remote I/O system. With its user-friendly interface, users can quickly and easily build a data logger software without any programming skill.

Functions:

- Support DCON, Modbus RTU, Modbus ASCII, and Modbus TCP protocols
- Support multiple COM Ports and TCP/IP connections
- Support Virtual Channel definition
- Support Control Logic (VB Script)
- Support Alarm Notifier (by sending SMS or E-Mail)
- Flexible workgroup configuration
- Real-time data trend (with zoom in and zoom out)
- Provide a Layout view
- Provide IP Camera Viewer
- Access database supported (can be exported to Excel file or CVS file)
- Provide Reporter to print trend line or data
- Provide High/Low alarm with audio warning
- Can search for DCON and Modbus modules
- Provide value scaling
- Support three-level authority management
- Do not need highly programming skills



NAPOPC_ST DA Server is a free OPC DA Server ("OPC" stands for "OLE for Process Control" and "DA" stands for "Data Access") for ICP DAS products.

Based on Microsoft's OLE COM (component object model) and DCOM (distributed component object model) technologies, NAPOPC_ST DA Server defines a standard set of objects, interfaces, and methods for use in process control and manufacturing automation applications to facilitate the interoperability.

Using NAPOPC_ST DA Server, the system integrates data with SCADA/HMI/Database software on the same computer and others. SCADA/ HMI/Database sends a request and NAPOPC DA Server fulfills the request by gathering the data of ICP DAS modules (License Free) and third-party devices (License Charge) to SCADA/HMI/Database.

For different OS of PAC products, ICP DAS provides several professional DA Servers:

Version	🕅 NAPOPC_ST	RAPOPC_XPE	X NAPOPC_CE5	X NAPOPC_CE6
Platform	Desktop Windows	Windows XP Embedded	Windows CE5	Windows CE6
Price	Free/\$	Free	Free	Free

For more information please visit <u>https://opc.icpdas.com</u>

1.4.4 PacSDK



The PacSDK is a software development kit that contains header files, libraries, documents, and tools required to develop an application for XPAC, WinPAC, and ViewPAC series. Refer to <u>section 6.2</u> for more information.

ICP DAS has released a new SDK, named PacSDK, which merged with and replaced the XPACSDK and the WinPACSDK. The applications that users developed can be easily applied to different PACs.

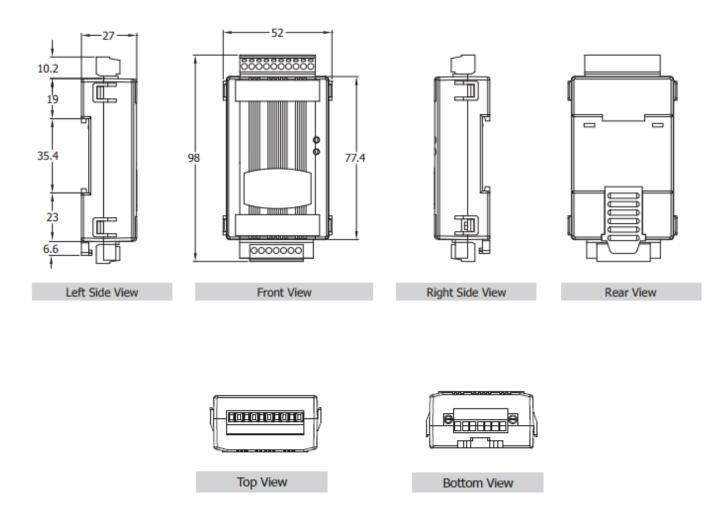
Chapter 2 Hardware Installation

2.1 Mounting



2.2 Dimensions

The following diagrams provide the dimensions of the tM series module and can be used as a reference when defining the specifications for any custom enclosures. All dimensions are in millimeters.

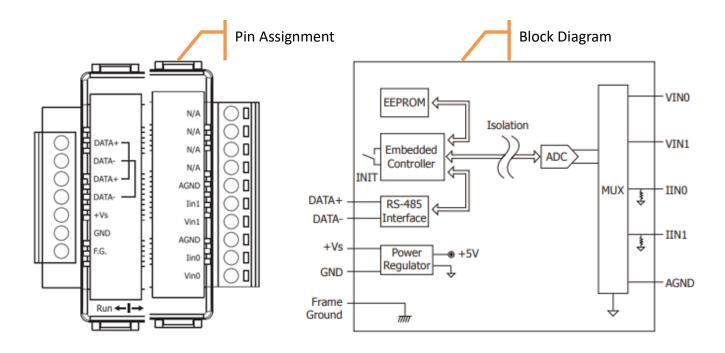


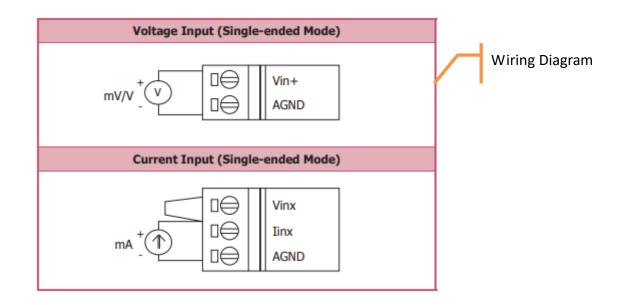
2.3 Wiring Connections

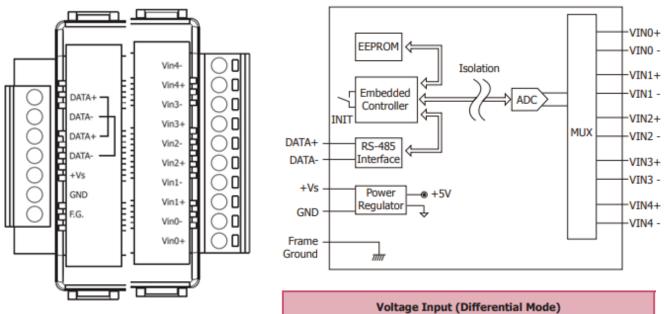
2.3.1 Wiring Recommendations

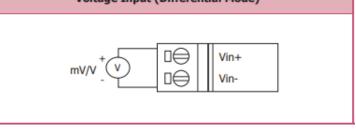
- Use 26-12 AWG wire for signal connections
- Strip the wire to a length of 7±0.5mm
- Use a crimp terminal for wiring
- Avoid high-voltage cables and power equipment as much as possible
- For RS-485 communication, use insulated and twisted pair 24 AWG wire, e.g., Belden 9841

2.3.2 tM-AD2 Wiring

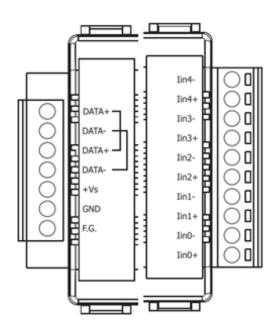


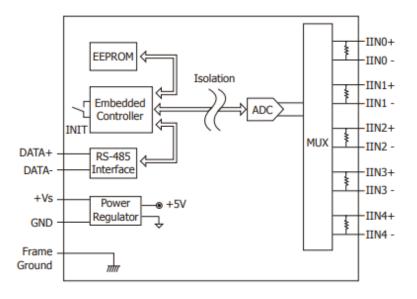


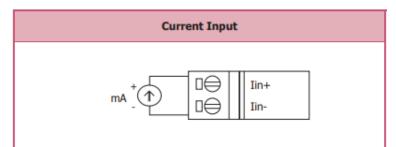


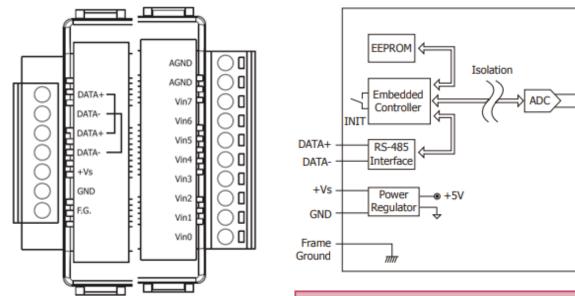


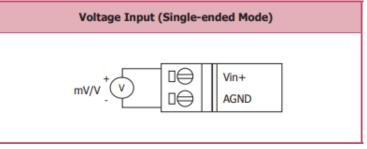
2.3.4 tM-AD5C Wiring



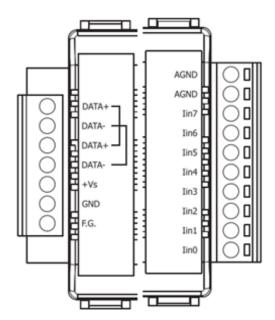


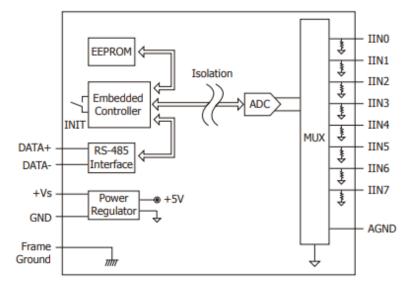


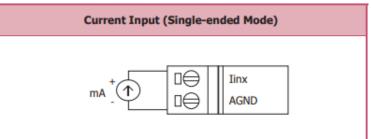




2.3.6 tM-AD8C Wiring







VIN0

VIN1

VIN2

VIN3

VIN4

VIN5

VIN6

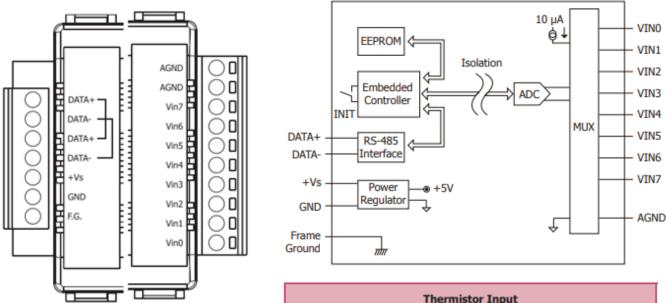
- VIN7

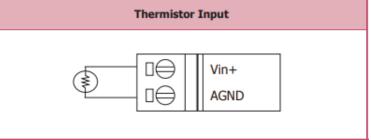
AGND

MUX

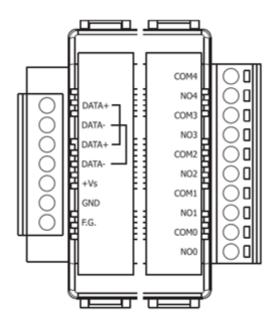
Ŷ

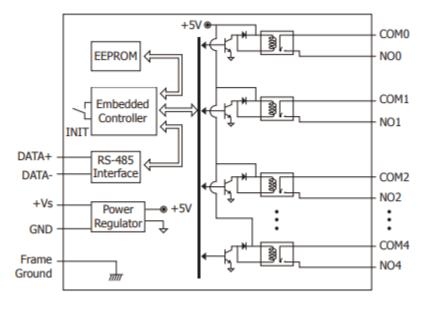
2.3.7 tM-TH8 Wiring



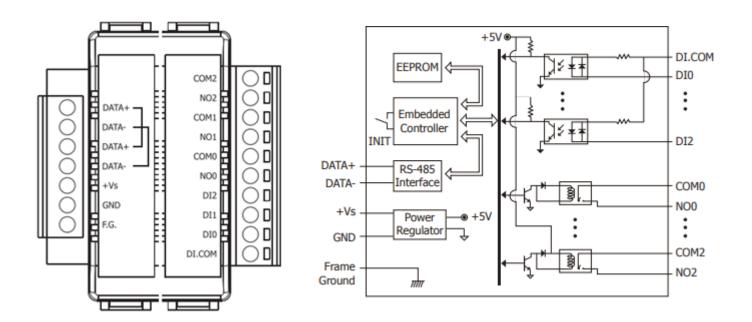


2.3.8 tM-R5 Wiring





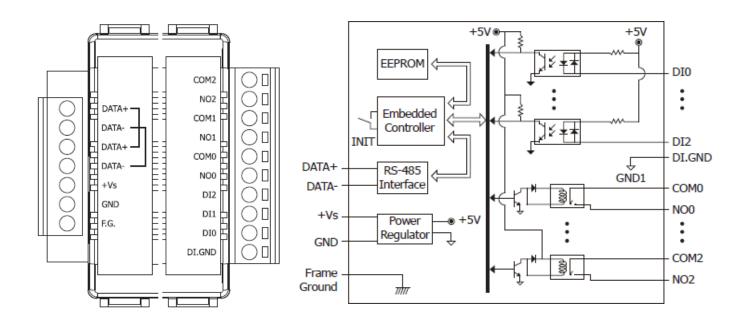
Power Relay	ON State Readback as 1	OFF State Readback as 0
Relay Output	COMx Relay Close AC/DC LOAD NOx To other channels	COMx Relay Open AC/DC X LOAD C NOx To other t channels



Input Type	ON State Readback as 1	OFF State Readback as 0
	+3.5 ~ +50 VDC	OPEN or <1 VDC
Wet Contact (Sink)	DIx 10K → → → → → → → → → → → → → → → → → → →	DIx 10K
	+3.5 ~ +50 VDC	OPEN or <1 VDC
Wet Contact (Source)	DIx 10K -+	DIx 10K

Power Relay	ON State Readback as 1	OFF State Readback as 0
Relay Output	AC/DC LOAD NOx COMx Relay Close To other channels	COMx Relay Open AC/DC X LOAD To other NOx channels

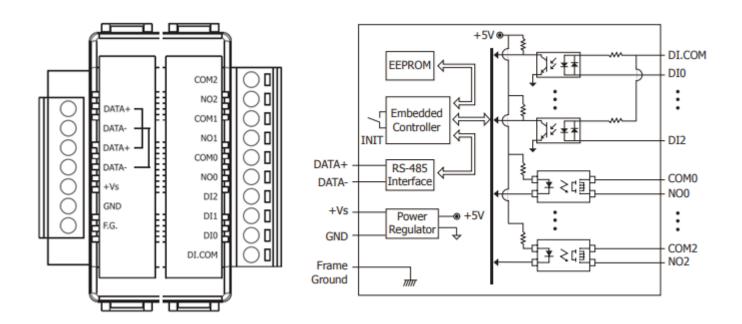
2.3.10 tM-PD3R3 Wiring

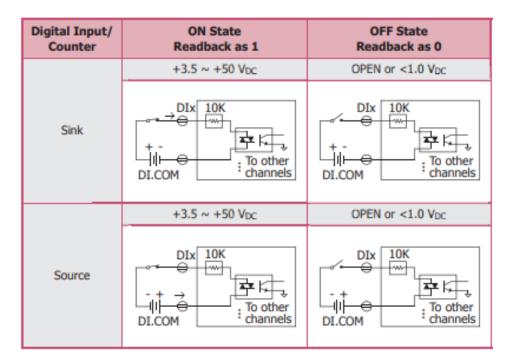


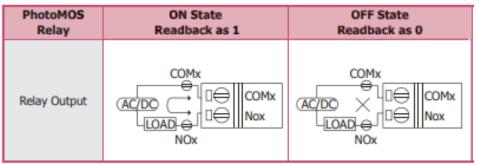
Input Type	ON State Readback as 1	OFF State Readback as 0
	Close to GND	Open
Dry Contact	→ DI.GND → DI.GND → → → → → → → → → → → → → → → → → → →	X DI.GND W +S5 V To other DIx C thannels

Power Relay	ON State Readback as 1	OFF State Readback as 0
Relay Output	COMx Relay Close AC/DC LOAD NOx Comx Relay Close To other channels	COMx Relay Open AC/DC × LOAD + NOx To other channels

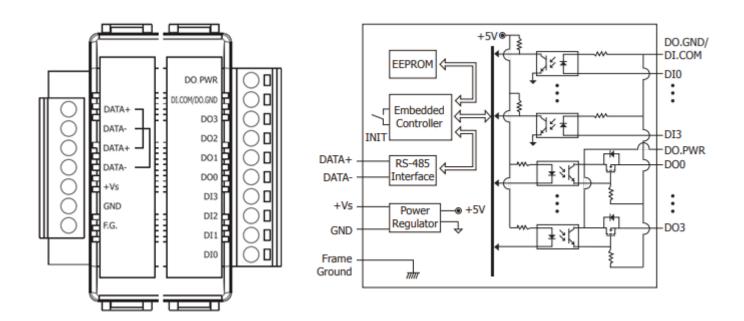
2.3.11 tM-P3POR3 Wiring

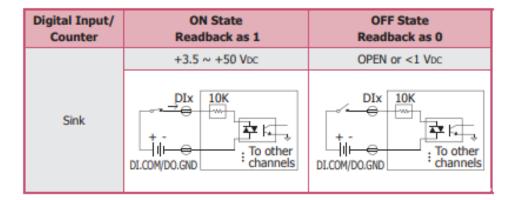


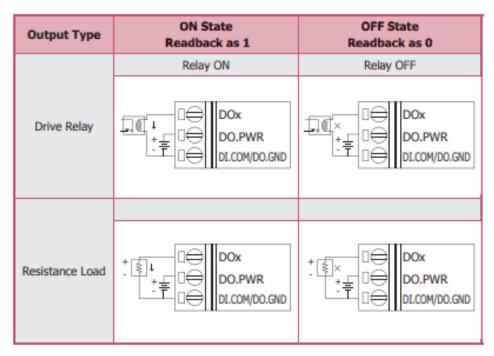




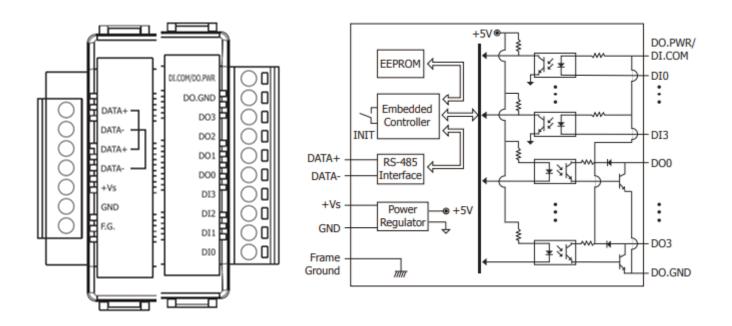
2.3.12 tM-P4A4 Wiring

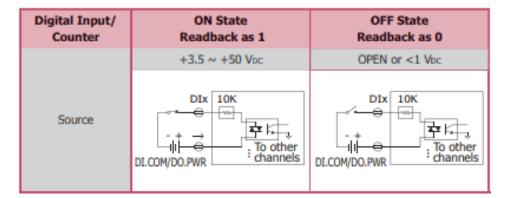


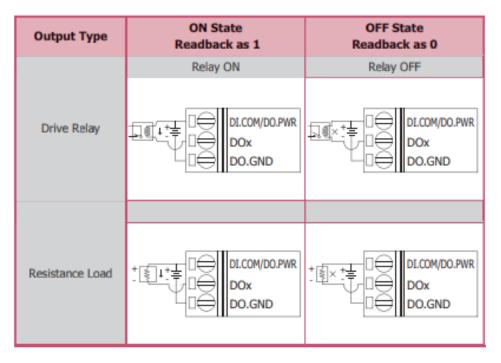




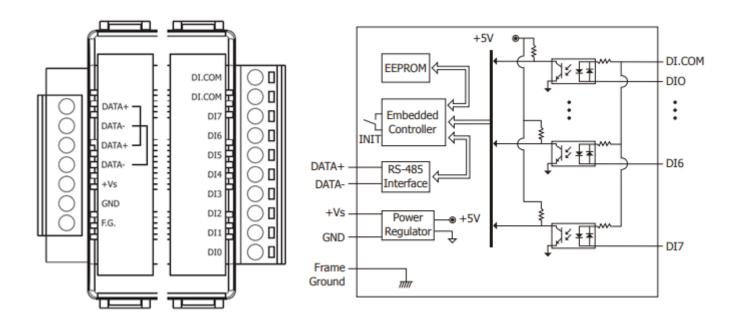
2.3.13 tM-P4C4 Wiring





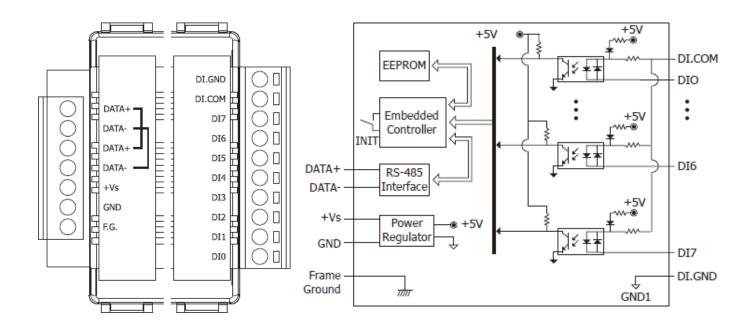


2.3.14 tM-P8 Wiring



Input Type	ON State Readback as 1	OFF State Readback as 0
	+3.5 ~ +50 V _{DC}	OPEN or <1 V _{DC}
Wet Contact (Sink)	DIx 10K →→→→→→→→→→→→→→→→→→→→→→→→→→→→→→→→→→→→	DIx 10K
	+3.5 ~ +50 V _{DC}	OPEN or <1 VDC
Wet Contact (Source)	DIx 10K	DIX 10K

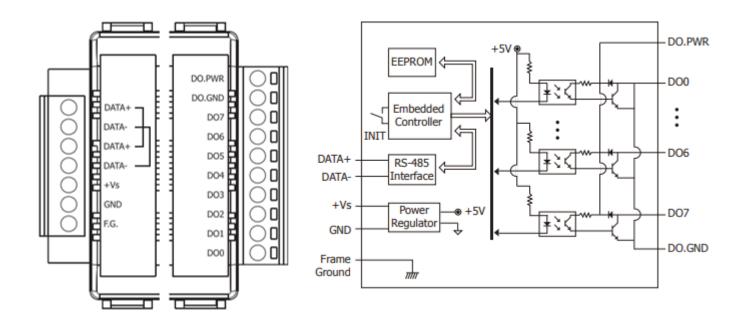
2.3.15 tM-PDW8 Wiring

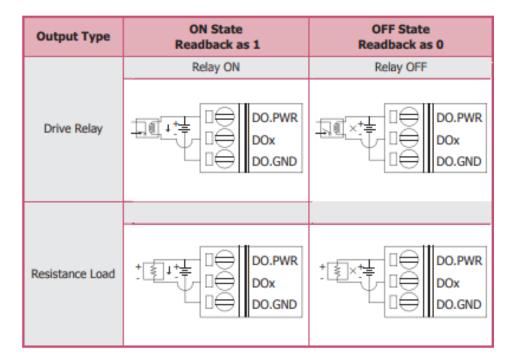


Input Type	ON State Readback as 1	OFF State Readback as 0
	+3.5 ~ +50 V _{DC}	OPEN or <1 V _{DC}
Wet Contact (Sink)	→→→ DIx 10K →→→ → → → → → → → → → → → → → → → → →	DIx 10K
	+3.5 ~ +50 V _{DC}	OPEN or <1 V _{DC}
Wet Contact (Source)	DIx 10K	DIx 10K

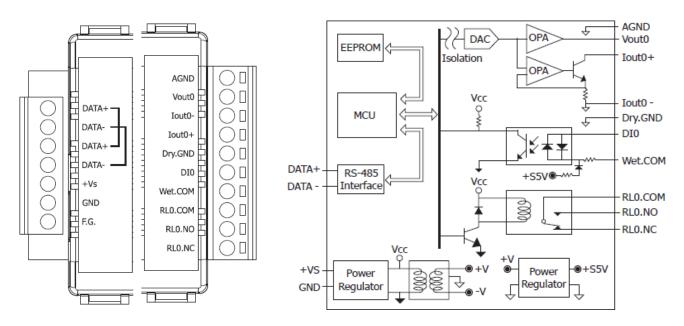
Input Type	ON State Readback as 1	OFF State Readback as 0
	Close to GND	Open
Dry Contact	→ DI.GND → DI.GND → → + S5 V + + S5 V + + + + + - + + + + + - + + + + + + + + + + + + + + + + + + +	X DI.GND

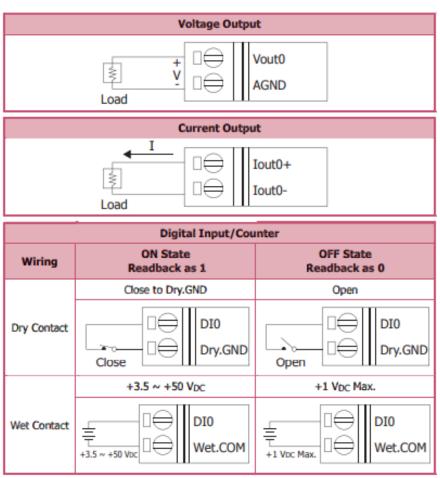
2.3.16 tM-C8 Wiring





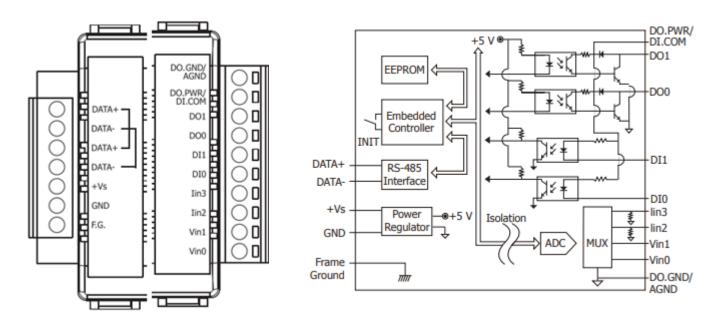
2.3.17 tM-DA1P1R1 Wiring

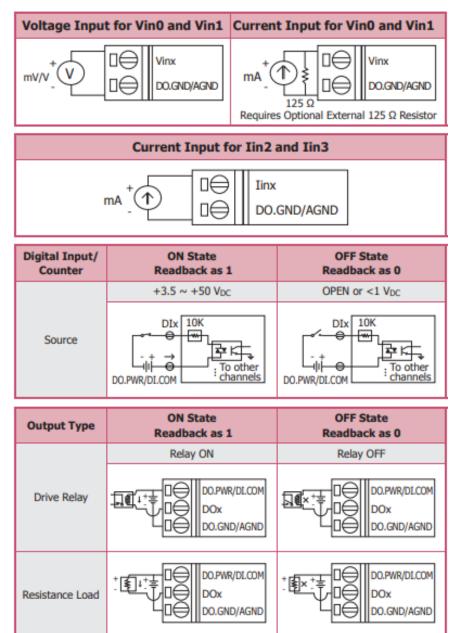




Relay Output			
ON State		OFF State	
Load1 → □⊖ RI	L0.NC L0.NO	↓ Load1 □⊖ RL0.NC Load2 × □⊖ RL0.NO	
	LO.COM		

2.3.18 tM-AD2P2C2 Wiring

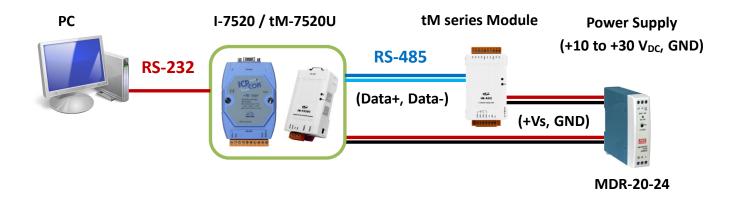




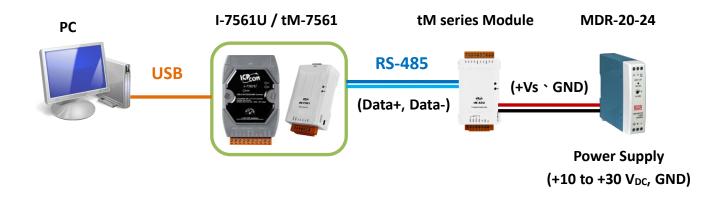
2.4 Hardware Connection

Before using tM series module, it must be configured by using DCON Utility Pro. tM series equips an RS-485 port that can be used to connect to a PC by using an RS-232 (or USB) to RS-485 converter.

1) Using RS-232 to RS-485 converter



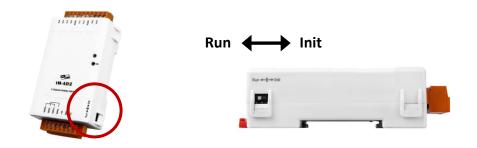
2) Using USB to RS-485 converter



2.5 Operating Switch and Parameters

tM series provides two operating modes, i.e., Run and Init, which can be set by sliding the switch to the Init or Run position on the right side of modules.

Operating Switch



Run Mode:

Normally, the operating switch of tM series module is set to "Run".

Init Mode:

If the communication parameters of the module are unknown, adjusting the switch to the "**Init**" position and then reboot the module. The setting values are always as follows:

Protocol:	<u>DCON</u> ,	Address:	<u>О</u> , Ва	aud Rate:	<u>960</u>	<u>0</u> (N,8,1),	Checksum:	<u>Disable</u> .	
DCON Utility Pro PC V	4.0.0.1 Beta Ve	rsion							
				5	1		FAQ		
COM1 COM5:*	ID tDA1P1R1	Address 0[00h]	Baud Rate 9600	Checksum Disabled	Format N,8,1	Status Remote I/O	Description [DCON]1*DI + 1*	*DO + 1*AO	Comments Supported

Default Communication parameters:

Parameters	Factory default settings (Run Mode)	Fixed Init settings (INIT Mode)	
Protocol	Modbus RTU	DCON	
Address	1	0	
Baud Rate	9600 bps		
Parity	n,8,1-no parity		
Checksum	Disable		

Chapter 3 Software Tool - DCON Utility Pro

First, using DCON Utility Pro to configure the communication parameters and I/O settings for each tM series module according to the application. DCON Utility Pro is a software tool that supports DCON and Modbus RTU/ASCII protocols and can be used to search, configure, and test I/O module.



Download the software and user manual on the web site. DCON Utility Pro can run on PC or PAC. https://www.icpdas.com/en/product/guide+Software+Utility_Driver+DCON__Utility_Pro

3.1 Search I/O Modules

In this case, PC (COM Port) and I/O module (RS-485) are connected through a tM-7561 (USB to RS-485 converter). Make sure that both PC and I/O module are connected properly, and the power supply is turned on. (The version is v3.0)

<u>Step 1</u>: Choose the COM port number that is used on PC and set conditions such as Baud Rate, Protocol, Checksum, and Format for searching.

DCON Utility Pro I	PC V 4.0.0.1
COM1:*	ID Address Baud Rate Checksum Format Status Description
	COM Port Search Options X
	COM1 ✓ Start 0 End 255 COM1 ecksum Format
	☑ 115200 □ 57600 □ 38400 □ 19200
	☑ 9600 □ 4800 □ 2400 □ 1200
	 □ Search RU-87PN Addr. Mode Timeout 200 ms ☑ Search And Get I/O Configurations
	Start Search Exit

Baud Rate: The default Baud Rate setting of the module is 9600, also you can check multiple options.

Baud Rate	Protocol Ch	ecksum Fo	rmat	
☑ 11520	57600	38400	19200	
☑ 9600	4800	2400	1200	

<u>Protocol</u>: When the module startup in "Init" mode, the default Protocol setting is DCON. When the module startup in "Run" mode, the default Protocol setting is Modbus RTU.

Baud Rate	Protocol	Checksum	Format		
☑ DCC	DN 🔽	Modbus RTU	J	Modbus ASCII	

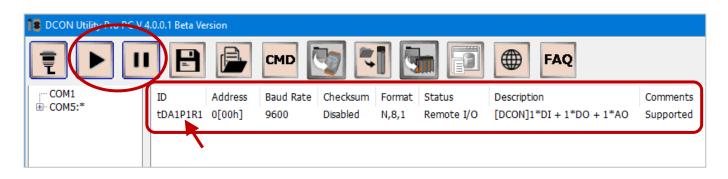
<u>Checksum</u>: The default Checksum setting is disabled.

Baud Rate Protoco	Checksum	Format
Checksum Disal	bled	Checksum Enabled

Format: The default Format setting is "N,8,1".

Baud Rate	Protocol	Checksum	Format	
▼ N,8,1		N,8,2 🗆	E,8,1	0,8,1

Step 2: After specifying search conditions, click the Start Search () button to search the module. Click Stop Search () to stop searching, and click the model (ID) to display the configuration window.



3.2 "Configuration" Page

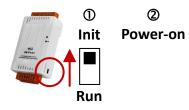
After searching and found the module, users can configure or test the I/O module. **Note:** DCON Utility Pro V3 (or later) support to open multiple setting window in Run mode.

In DCON Utility Pro, the "Configuration" page for each module can be used to set the communication parameters such as Protocol, Address, Baud Rate, etc. The rest of the setting pages for each module may be different so the following sections will describe each according to the IO type of modules.

Model	Description	Section	
tM-P3R3	3-channel Digital Input and 3-channel Relay Output Module		
tM-PD3R3			
tM-P3POR3	3-channel Digital Input and 3-channel PhotoMOS Relay Output Module		
tM-P4A4	4-channel Digital Input and 4-channel Digital Output Module		
tM-P4C4	4-channel Digital Input and 4-channel Digital Output Module	<u>3.2.1</u>	
tM-R5	5-channel Relay Output Module		
tM-P8	0 shamed Disited Jacout Madula		
tM-PDW8	 8-channel Digital Input Module 		
tM-C8	8-channel Digital Output Module		
tM-AD2	2-channel Voltage/Current Input Module		
tM-AD4P2C2	4-channel Analog Input, 2-channel Digital Input and 2-channel Relay Output Module	<u>3.2.2</u>	
tM-AD5	5-channel Differential Voltage Input Module		
tM-AD8	8-channel Single-ended Voltage Input Module		
tM-AD5C	5-channel Differential Current Input Module	<u>3.2.3</u>	
tM-AD8C	8-channel Single-ended Current Input Module		
tM-TH8	8-channel Thermistor Input Module		
tM-DA1P1R1	1-channel Analog Output, 1-channel Digital Input and 1-channel Relay Output Module	<u>3.2.4</u>	

3.2.1 Common Settings for tM-P3R3, PD3R3, P3POR3, P4A4, P4C4, R5, C8, P8, and PDW8

In Init mode, the module is searched with the default communication parameters. On the "Configuration" page, click the "Set Module Configurations" button, and then rebooting in Run mode to apply the settings.



Note: Only when the Protocol is set to **DCON**, the items marked with (INIT*) must be set under the INIT mode. When using the Modbus protocol, these items can be set after adjusting the switch to Init, even if the module starts in Run mode. After completing the setting, adjusting the switch to Run and then rebooting.

tC8 Firmware[A103]		\times
Configuration DO	Host WDT Commands Log Summary About	
Protocol (INIT*) Address Baud Rate (INIT*) Parity (INIT*) Checksum (INIT*)	DCON 0 0 000 N,8,1 Disabled The default communication parameters in Init mode.	
Response Delay	Image: Construction of the setting in the setting	

Protocol	Specify the Protocol setting It can be DCON, Modbus RTU, and Modbus ASCII
Address	Specify the unique NET-ID for the module (Range: 0-255)
Baud Rate	Specify the Baud Rate setting It can be 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
Parity	Specify the Parity, It can be "N,8,1-None Parity", "N,8,2-None Parity", "E,8,1-Even Parity", and "O,8,1-Odd Parity"
Checksum	Specify the Checksum, it can be Disabled or Enabled Note that the setting is "Disabled" when the Protocol is set to Modbus RTU/ASCII.
Respond Delay	Specify the delay time to respond. (Range: 0-30 ms)

3.2.2 AI items for tM-AD2 and AD4P2C2

The "Configuration" page provides settings for communication parameters (Section 3.2.1) and analog data.

tAD4P2C2 Firmware[A1	05]	×
Configuration AI/DO	Alarm DI Host WDT Commands Log Summary About	
Protocol (INIT*)	DCON ~	
Address	0 (00H)	
Baud Rate (INIT*)	9600 ~	
Parity (INIT*)	N,8,1 ~	
Checksum (INIT*)	Disabled ~	
Analog Format	Engineering Forma 🗸	
Sample Mode	Normal Mode V	
Response Delay	1 ms	
	Set Module Configurations	
Exit		

Analog Format	Specify the format for analog data <u>DCON Protocol</u> : It can be Engineering Format, Percent Format, or 2's Complement Format <u>Modbus RTU/ASCII Protocol</u> : It can be Engineering Format or 2's Complement Format
Sample Mode	Specify the mode for sampling data It can be Normal Mode or Fast Mode

3.2.3 AI Items for tM-AD5, AD8, AD5C, and AD8C

The "Configuration" page provides settings for communication parameters (Section 3.2.1) and analog data.

tAD5 Firmware[A104]	×
Configuration AI	Commands Log Summary About
Protocol (INIT*)	DCON ~
Address	0 (00H)
Baud Rate (INIT*)	9600 ~
Parity (INIT*)	N,8,1 ~
Checksum (INIT*)	Disabled ~
Analog Format	Engineering Forma 🗸
Sample Mode	Normal Mode v
Type Code	[08] +/- 10 V ~
Response Delay	0 ms
	Set Module Configurations
Exit	

Analog Format	Specify the format for analog data <u>DCON Protocol</u> : It can be Engineering Format, Percent Format, or 2's Complement Format <u>Modbus RTU/ASCII Protocol</u> : It can be Engineering Format or 2's Complement Format			
Sample Mode	Specify the mode for sampling data It can be Normal Mode or Fast Mode.			
Type Code Specify a type code with data range AD5: +/- 1 V, +/- 2.5 V, +/- 5 V, +/- 10 V AD8: 0 to 500 mV, 0 to 1 V, 0 to 2.5 V, 0 to 5 V, 0 to 10 V AD5C: +/- 20 mA, 0 to 20 mA, 4 to 20 mA AD8C: 0 to 20 mA, 4 to 20 mA (Reference: B.2 to B.5 Data Ranges)				

3.2.4 AI Items for tM-TH8 and DA1P1R1

The "Configuration" page provides settings for communication parameters (Section 3.2.1) and analog data.

H8 Firmware[A102]		
Configuration AI	User Defined Type Commands Log Summary About	
Protocol (INIT*)	DCON ~	
Address	0 🗧 [00H]	
aud Rate (INIT*)	9600 ~	
arity (INIT*)	N,8,1 ~	
necksum (INIT*)	Disabled ~	
nalog Format	Engineering Forma 🗸	
Response Delay	0 ms Set Module Configurations	
Exit	nge==>Please connect to input source or check the input connector	

	Configure the format of analog data.
Analog Format	DCON Protocol: tM-TH8 : It can be Engineering Format, Percent Format, 2's Complement Format, or Ohms tM-DA1P1R1 : It can be Engineering Format, Percent Format, or 2's Complement Format
	Modbus RTU/ASCII Protocol: tM-TH8, tM-DA1P1R1: It can be Engineering Format or 2's Complement Format

3.3 "AI" Page

Model	Description	Section
tM-AD2	2-channel Voltage/Current Input Module 3.3.	
tM-AD5	5-channel Voltage Input Module	
tM-AD8	8-channel Voltage Input Module	2.2.2
tM-AD5C	5-channel Current Input Module <u>3.3.2</u>	
tM-AD8C	8-channel Current Input Module	
tM-TH8	8-channel Thermistor Input Module	<u>3.3.3</u>

The "AI" page provides settings for detecting or configuring analog input channels of the module.

3.3.1 AI Page01 for tM-AD2

The "AI" page can be used to detect data and specify the type code for each analog input channel.

tAD2 Firmware[A106]		×
Configuration AI AI Value ✓ CH:00 -000.001 ✓ CH:01 -000.001	1 [08] 0 ~ 10 V [7FFFh] ~ [08] 0 ~ 10 V [7FFFh] ~	
Exit		

CH: xx	Check the box to detect the current value of the channel		
Type Code	Specify a type code with data range tM-AD2: 0 to 500 mV, 0 to 1 V, 0 to 2.5 V, 0 to 5 V, 0 to 10 V, 0 to 20 mA, 4 to 20 mA (Reference: B.1 Data Ranges)		
Set all channels as CH0	Configure all channels to the same type code as CH: 00		

3.3.2 AI Page02 for tM-AD5, AD8, AD5C, and AD8C

The "AI" page can be used to detect data for each analog input channel.

Note: tM-AD5 and tM-AD5C equip five analog input channels, i.e., CH: 00 to CH: 04.

tM-AD8 and tM-AD8C equip eight analog input channels, i.e., CH: 00 to CH: 07.

tAD8 Firmwar	e[A100]				×
Configuratio		Commands Log	Summary	About	
	AI Value				
✓ CH:00					
CH:01	-000.004				
CH:02	-000.003				
CH:03	-000.004				
✓ CH:04	-000.003				
CH:05	-000.004				
CH:06	-000.004				
CH:07	-000.003				
	1				
Exit					

	Check the box to dete DCON, Modbus RTU/	ect the current value of t	the channel
CH: xx	Protocol	Analog Format	Under Range
	DCON	Fuchania	-9999.9
	Modbus	Modbus Engineering	-32768
	DCON, Modbus	2's Complement	8000
	DCON	Percent	-999.99
	Note: Only applicable for th	ne measurement ranges	s of 0-20 mA and 4-20

3.3.3 AI Page03 for tM-TH8

The "AI" page can be used to detect data and configure the type code and temperature settings for each analog input channel.

tTH8 Firmware[A102]		×				
-	e Commands Log Summary About					
AI Value Typ	pe Code	Temperature Offse				
CH:00 -9999.90 [UnderRange] prev	Con Type III 10K @ 25°C -30 ~ 240°F 🗸	00.00 + -				
☑ CH:01 -9999.90 [UnderRange] [60)] PreCon Type III 10K @ 25°C -30 \sim : \smallsetminus	00.00 + -				
☑ CH:02 -9999.90 [UnderRange] [60)] PreCon Type III 10K @ 25°C -30 \sim : \smallsetminus	00.00 + -				
☑ CH:03 -9999.90 [UnderRange] [60)] PreCon Type III 10K @ 25°C -30 \sim : \smallsetminus	00.00 + -				
☑ CH:04 -9999.90 [UnderRange] [60)] PreCon Type III 10K @ 25°C -30 \sim : \smallsetminus	00.00 + -				
☑ CH:05 -9999.90 [UnderRange] [60)] PreCon Type III 10K @ 25°C -30 \sim : \smallsetminus	00.00 + -				
☑ CH:06 -9999.90 [UnderRange] [60)] PreCon Type III 10K @ 25°C -30 \sim : \smallsetminus	00.00 + -				
✓ CH:07 -9999.90 [UnderRange] [60	0] PreCon Type III 10K @ 25°C -30 \sim : \smallsetminus	00.00 + -				
Temperature Format	Set all channels as CH0					
Exit						
Analog Input Under Range==>Please conr	Analog Input Under Range==>Please connect to input source or check the input connector					

	Check the box to detect the current value of the channel DCON, Modbus RTU/ASCII				
	Protocol	Analog Format	Under Range	Over Range	
CH: xx	DCON	- Engineering	-9999.9	+9999.9	
	Modbus		-32768	32767	
	DCON, Modbus	2's Complement	8000	7FFF	
	DCON	Percent	-999.99	+999.99	
Type Code	Specify a type code with temperature range for the thermistor (Reference: B.8 Data Ranges)				
Set all channels as CH0	Configure all channels to the same type code as CH: 00				
Temperature Offset	Specify an offset to adjust the temperature reading				
Temperature Format	Specify the temperature scale				
•	It can be Celsius (°C	C) or Fahrenheit (°F)			

3.4 Other I/O Pages

3.4.1 Configuration Page - User Defined Type

"User Defined Type" is a customized configuration page that is used for tM-TH8, a thermistor input module. The following example shows how to set the type code for a custom thermistor.

1. To get Steinhart-Hart coefficients from the thermistor vendor. On the "Set By Steinhart Coefficients" page, enter A, B, and C coefficients in **Float Format**, click the "Calculate" button and then click the "Setting" button.

For example, there is a thermistor with the part number 'YSI H Mix 10000', and three Steinhart-Hart coefficients are listed in the table below.

Coefficient	Float Format	Hex Format
A	1.02949264911967E-03	3A86F00B
В	2.39078592663805E-04	397AB12C
С	1.56816365983255E-07	3428615B

tTH8 Firmware[A102]	×
Configuration AI User Defined Type Commands Log Summary About	
Set By Steinhart Coefficients Set By Resistor and Temperature Help	
1 Float Format Hex Format Select Type Code	
A 1.02949264911967E-03 3A86F00B 0x70 ~	
B 2.39078592663805E-04 397AB12C 2 Calcualte	
C 1.56816365983255E-07 3428615B Clear	
Please input A , B , C Coefficients in float format	
After clicking the "Calculate" button, the values in hex format will be displayed.]
RunMonitor Exception at step 0 Access to the port is denied.	

 Otherwise, you can also get a Resistance-Temperature table from the thermistor vendor. On the "Set By Resistor and Temperature" page, enter three sets of Resistance-Temperature values (Ohms, °C) and click the "Calculate" button to calculate three Steinhart-Hart coefficients. For accurate results, it is recommended to configure (Rx, Tx) by using the following rules:

(1) -40°C \leq **T1, T2, T3** \leq 150°C (2) |**T2 - T1**| \leq 50°C (3) |**T3 - T2**| \leq 50°C

For example,

Temperature (°C)
T1:0
T2: 50
T3: 100

<u>Note</u>: If the resistance is greater than 204700 ohms, it will be treated as under range.

tTH8 Firmware[A102]						×
Configuration AI U	Iser Defined Type	Commands Log	Summary	About		
Set By Steinhart Coeffi	icient: Set By Res	istor and Temper	ature Help			
Resist	tor Value	Ter	mperature	°C	Select Type Code	
R1(ohms) 29490		T1 0			0x70 ~	
R2(ohms) 3893		T2 50			Calcualte	
R3(ohms) 816.8		T3 100			Clear	
Plea	ase input resistor ar	nd temperature v	alues		Setting	
Exit						
Analog Input Under Range	e==>Please conne	ct to input source	e or check t	the input o	connector	

3. After clicking the "Calculate" button, it will automatically switch to the "Set By Steinhart Coefficients" page and show the results. Finally, click the "Setting" button to apply the settings.

Float Format Hex Format Select Type Code A 0.00102949264911967 3A86F00B 0x70 ~ B 0.000239078592663805 397AB12C Calcualte C 1.56816365983256E-07 3428615B Clear	Set By S	Steinhart Coefficients Set By Re	sistor and Temperature Help	
A 0.00102949204911907 SA80P00B Low C B 0.000239078592663805 397AB12C Calcualte		Select Type Code		
B 0.000239078592663805 397AB12C	Α	0.00102949264911967	3A86F00B	0x70 ~
C 1.56816365983256E-07 3428615B Clear	В	Calcualte		
	с	1.56816365983256E-07	3428615B	Clear
Please input A , B , C Coefficients in float format		Setting		

3.4.2 Configuration Page - AO

tM-<u>DA1</u>P1R1 equips one analog output channel.

tDA1P1R1 Firmware[A103]	×
Configuration AO DO DI Host WDT Commands Log Summary About	
Type Code Slew Rate CH:0 [02] 0 ~ +10 V immediate Set Set [Power On Value] Set [Safe Value]	
Exit	

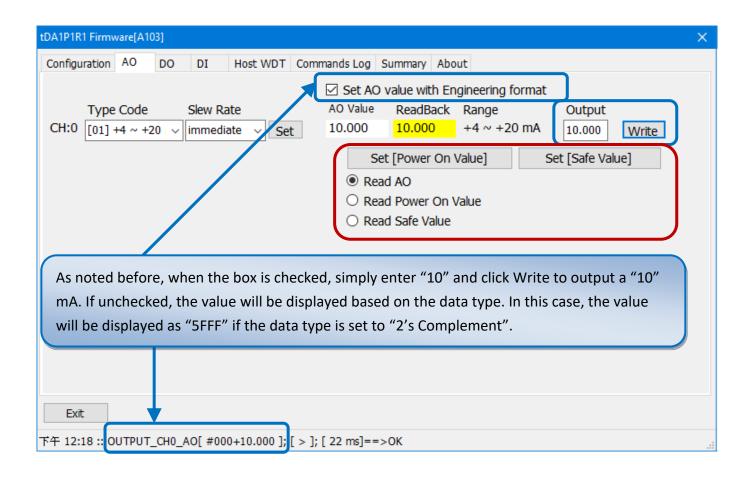
Type Code	Specify a type code with data range (tM-DA1P1R1: 0 to 10 V, 0 to 20 mA, 4 to 20 mA, reference: B.7 Data Ranges)
Slew Rate	Specify the change of voltage or current per second (V/sec, mA/sec) (Reference: A.4 Slew Rate Control)
Set	Click this button to apply the Type Code or Slew Rate settings

☑ Set AO value with the Engineering format:

- 1) Check the box to display or set the AO Value and the ReadBack value with the Engineering format.
- 2) <u>Unchecked</u>, to display value with the data format setting (reference: Section 3.2.4 Configuration page Analog Format)

For example, the Type Code is set to "+4 to +20mA" and Analog Format is set to "**2's Complement**". To output 10 mA, enter "10" in the Output field, and click the Write button. If the box is checked, the value will display as "10", otherwise, the value will display as "**5FFF**".

AO Value	Display the title and value according to the selected item such as	
ReadBack	Read AO/Power-on value/Safe value	
Note: The Readback value varies according to the Slew Rate setting so that may be different fro		
the configured AO value. Reference: A.5 AO Read-back Value		
RangeDisplay the data range according to The type code setting		
Output / Write	Specify an output value, and then click "Write" to apply the settings	



Set to Power-on Value	Enter a value in the output field, and click this button to apply the settings
Set to Safe Value	Reference: A.2 Dual Watchdog
Read AO	Choose the radio button to read the AO value
Read Power-on Value	Choose the radio button to read the current Power-on value or Safe value,
Read Safe Value	at which point the values cannot be set.

3.4.3 Configuration Page - AI/DO Alarm

tM-AD4P2C2 equips four analog input channels and two relay output channels. Both the Alarm Mode and High/Low Alarm Limit setting are used to configure a warning light or an alarm device that be connected to a relay output channel.

tAD4P2C2 Firmware[A105]						×
Configuration AI/DO Alarm DI	Host WDT Commands Log	Summary	About			
AI Value ✓ CH:00 +00.0000 ✓ CH:01 -00.0011 ✓ CH:02 +000.000 ✓ CH:03 -000.001	Type Code [05] +/- 2.5 V [05] +/- 2.5 V [0D] +/- 20 mA [0D] +/- 20 mA		Alarm Mode Disable v Disable v	High Alarm Limit 2.5 2.5 2	Low Alarm Limit -2.5 -2.5 Set Alarm	
1 Exit	Set all channels as CH0		DO Bit Status DO DO DO1 Set [Power Read DO Read Power Read Safe V	On Value	Low Alarm Status CH:00 Clear CH:01 Clear Set [Safe Value]	

CH: xx	Check the box to detect the current value of the analog input channel		
	pecify a type code with data range		
Type Code	tM-AD4P2C2: +/- 1 V, +/- 2.5 V, +/- 5 V, +/- 10 V,		
	+/- 20 mA, 0 to 20 mA, 4 to 20 mA		
	(Reference: B.6 Data Ranges)		
Set all channels as CH0	Configure all channels to the same type code as CH: 00		

Only when Alarm Mode = Disable, you can configure the following items:

DOx	Check the box to set the DO value, Power-on value, or Safe value
Set to Power-on Value	After selecting the DOx box, click this button to apply the settings
Set to Safe Value	(Reference: A.2 Dual Watchdog)
Read DO	Choose the radio button to read the DO status
Read Power-on Value	Choose the radio button to read the current Power-on value or Safe value,
Read Safe Value	at which point the values cannot be set.

tAD4P2C2 Fi	rmware[A105]						×
Configuratio	on AI/DO Alarm DI	Host WDT Commands Log	Summary	About			
⊡ сн:00	AI Value +000.002	Type Code [08] +/- 10 V	~	Alarm Mode Latch ~ Momentar ~	High Alarm Limit	Low Alarm Limit	
✓ CH:01	+000.001 -000.001	[08] +/- 10 V [0D] +/- 20 mA	~			Set Alarm	J
⊘ сн:оз	-000.001	[0D] +/- 20 mA Set all channels as CH0		DO Bit Status DO DO0 DO1 Set [Power Read DO Read Power Read Safe V	r On Value	Low Alarm Status CH:00 Clear CH:01 Clear Set [Safe Value]	
Exit							
下午 01:47:	:GET_CH1_ALARM_MODE	[@00RAC1]; [!021]; [15 r	ms]==>OK				

Alarm Mode	Specify the alarm mode, it can beDisable:DisabledMomentary:An alarm occurred when the value is out of the limitLatch:An alarm occurred when the value is out of the limit. The status of the alarm can be cleared.				
High Alarm Limit	After choosing a type code, set the upper limit of the alarm				
Low Alarm Limit	After choosing a type code, set the lower limit of alarm				
Set Alarm	Click the "Set Alarm" button to apply the settings for Alarm Mode, High Alarm Limit, and Low Alarm Limit				
(High/Low) Alarm Status	 Used to display or clear the status of the alarm <u>CH: xx</u>, if the value is out of the limits, the text-box will be displayed in red <u>Clear</u>, when Alarm Mode is set to Latch, the button can be used to clear the status of the alarm by clicking it 				

3.4.4 Configuration Page - DO

The "DO" page can be used to set or display the DO value, Power-on value, or Safe value.

Model	Description		
tM-DA1P1 <u>R1</u>	One Digital Output Channel		
tM-P3 <u>R3</u> , tM-PD3 <u>R3</u> , tM-P3 <u>POR3</u>	Three Relay Output Channels		
tM-P4 <u>A4</u> , tM-P4 <u>C4</u>	Four Digital Output Channels		
tM-R5	Five Relay Output Channels		
tM-C8	Eight Digital Output Channels		

×

Bit Status (CH: xx)	Check the CH: xx box to specify the value
DO Value	After selecting the CH: xx boxes, a hexadecimal value will be displayed
Set to Power-on Value	After selecting the CH: xx boxes, click the button to apply the settings
Set to Safe Value	(Reference: A.2 Dual Watchdog)
Read DO	Choose the radio button to read the DO status
Read Power-on Value	Choose the radio button to read the current Power-on value or Safe value
Read Safe Value	(While the radio button is selected, these two values cannot be set.)

3.4.5 Configuration Page - Host WDT

The "Host WDT" page can be used to enable the host watchdog and reset the status of the host watchdog timeout. (Reference: Appendix A.2 Host Watchdog)

Model	Description			
tM- <u>DA1</u> P1 <u>R1</u>	One Analog Output and One Digital Output Channels			
tM-AD4P2 <u>C2</u>	Two Digital Output Channels			
tM-P3 <u>R3</u> , tM-PD3 <u>R3</u> , tM-P3 <u>POR3</u>	Three Relay Output Channels			
tM-P4 <u>A4</u> , tM-P4 <u>C4</u>	Four Digital Output Channels			
tM-R5	Five Relay Output Channels			
tM-C8	Eight Digital Output Channels			

1) When using the **DCON** protocol, the Host WDT page will be displayed as follows.

(Reference: Section 3.2.1 Common Settings)

tAD4P2C2 Firmware[A105]				×
Configuration AI/DO Alarm	DI Host WDT	Commands Log	Summary	About
Enable WDT Au	ito Send Host OK)		
WDT Timeout	25.50 (0.1 ~ 25.5 sec)	Set Timer		
Reset Watchdog State	JS			
Exit				

Enable WDT	Check the box to enable the watchdog		
Auto Send Host OK	Check the box to automatically notify that the host works normally Note that the function is for testing and the timer is disabled		
WDT Timeout / Set Timer	Specify a watchdog timeout value and click the "Set Timer" button to apply the setting		
Reset Watchdog Status	After the timeout, it needs to clear the Timeout status to write an AO/DO value.		

2) When using the **Modbus RTU (or Modbus ASCII)** protocol, the Host WDT page will be displayed as follows. (Reference: Section 3.2.1 Common Settings)

AD4P2C2 Firmware[A105]	<
Configuration AI/DO Alarm DI Host WDT Commands Log Summary About	
Enable WDT Enable Output When WDT Timeout	
WDT Timeout 25.50 Set Timer	
(0.1 ~ 25.5 sec)	
Reset Watchdog Status	
Exit	

Enable WDT	Check the box to enable the watchdog			
Enable Output When WDT Timeout	Check the box to allow modifying an AO/DO value, even if the Watchdog timeout has occurred			
WDT Timeout / Set Timer	Specify the duration of the Watchdog timeout, and click the "Set Timer" button to apply the setting			
Reset Watchdog Status	Click the button to clear the Timeout status			

For example:

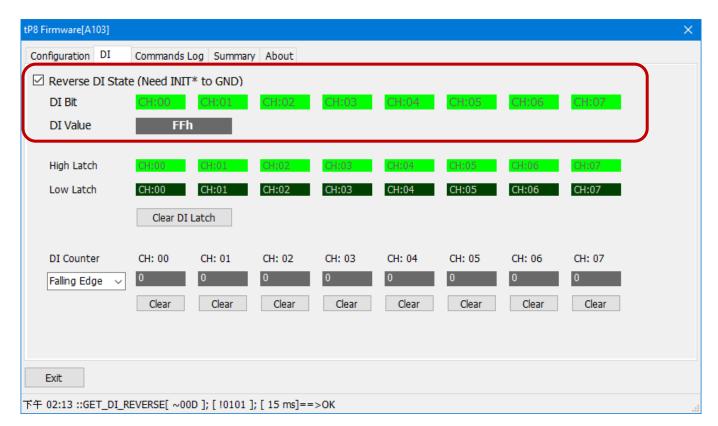
- Check the Enable WDT checkbox, set WDT Timeout as 5 seconds, and click the Set Timer button. After 5 seconds, a warning message "Watchdog Timeout" will be displayed on the status bar. The output value will be set as the predefined Safe Value, which cannot be changed.
 - <u>Note:</u> When using the Modbus protocol, you can check the "Enable Output When WDT Timeout" box to allow the change of an AO/DO value, even if the timeout occurred.
- 2. Click the "Reset Watchdog Status" button to clear the Timeout status, and the predefined Power-on value will be loaded after a reboot. Otherwise, the predefined Save value will be loaded.
 - <u>Note:</u> When using the DCON protocol, you can check the "Auto Send Host OK" box to automatically send a signal which indicates the Host works normally. Also, uncheck the box and click the "Set Timer" box again to start the watchdog timer.

3.4.6 Configuration Page - DI

The DI page is used to display or clear the status of DI, DI Latch, or DI Counter.

Model	Description		
tM-DA1 <u>P1</u> R1	One digital input channel		
tM-AD4 <u>P2</u> C2	Two digital input channels		
tM- <u>P3</u> R3, tM- <u>PD3</u> R3, tM- <u>P3</u> POR3	Three digital input channels		
tM- <u>P4</u> A4, tM- <u>P4</u> C4	Four digital input channels		
tM-P8, tM-PDW8	Eight digital input channels		

<u>DI Page</u>



Reverse DI Status		Check the box to reverse the status of all DI channels				
DI Bit Status		Display the status for each DI channel (ON or OFF)				
DI Value		Using a hexadecimal value to indicates the status of DI channels				
	High Latch	Display the status of active-high for each DI channel				
DI Latch	Low Latch	Display the status of active-low for each DI channel				
	Clear DI Latch	Click the button to clear the latched status for all DI channels (Reference: Appendix A.6 Advanced DI Features)				
DI Counter	Falling/Rising Edge	The counter value will be increased by 1 when received a high-to-low or a low- to-high signal				
	Clear	Click the button to clear the counting value of the DI channel				

3.5 Command Line Tool

"Command Line" tool can be used to test and debug tM series modules by using the Modbus RTU protocol and DCON protocol commands.

- 1. Search the tM module and then click the **Command Line** button to open the configuration page.
- In the "Terminal Command Line Tool" window, the module settings will be filled in automatically. Either enter a command in the **Command** field (1) or use the preset commands (2) and click the Send button.

🚺 DCON Utility Pro	PC V 4.0.0.	1 Beta Versio	on					
	11	-			7	1		FAQ
	ID tP4C4	Address 2[02h]	Baud Rate 115200	e Checksum Disabled	Format N,8,1	Status Remote I/O	Description [DCON]4*DI + 4*DO	Comments Supported
Terminal Comman	d Line Tool							×
COM Port	COM5	~	Protocol	DCON		[
Baud Rate	115200	~	Format	N,8,1-No	ne Parity	~		
Checksum	Disabled	~	Address	2	~	~	Send	
Timeout	100	→ ms	Select ID	tP4C4		/		
Command	\$02M							
Response	!02tP4C4			1.				
GET_MODULE_NAME GET_MODULE_FIRMWARE GET_MODULE_CONFIG GET_MODULE_CONFIG GET_MODULE_PROTOCOL SET_MODULE_PROTOCOL READ_DI READ_DI READ_DI_LOW_LATCH CLEAR_DI_LATCH SET_DI_REVERSE GET_DI_REVERSE READ_CH3_DI_COUNTER OUTPUT_DO RFAD_DO 2. When using DCON protocol, refer to Appendix E DCON Protocol Commands Appendix E DCON Protocol Commands								
	Export Com	mands		(lear		Save to path\log_repor	t\

3. Click the "Save to \logger_report\" button to save results to a file in the "log_report" folder under the installation folder of DCON Utility Pro. The file name is "Command_Line_Result_Log_mm_dd_xx.txt".

Command_Line_Result_Log_12_2_15.ts	xt - Note	epad	—		×
File Edit Format View Help					
下午 03:10:54-> 下午 03:07 :: [\$	502M]	; [!02tP4C4]; [12 ms	==>0	K ^
下午 03:10:57-> 下午 03:07 :: [\$					
Ln 1, Col 1	100%	Windows (CRLF)	ANS	l	

3.6 Calibration

Normally, the module has been calibrated before shipping. Also, users can use the following commands to calibrate the module by using Command Line in the DCON Utility Pro.

Warning:

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



3.6.1 Calibrate AO Module

tM-DA1P1R1 provides a single analog output channel. Follow these steps to calibrate the module by using DCON Utility Pro.

- 1. Warm up the module for 30 minutes.
- 2. Set (<u>\$AA9NTS</u>) / read (<u>\$AA2</u>) the type code.
 - E.g., \$019020, means to set the type code of analog output channel 0 of module 01 as 2 (i.e., 0 to +10V), and the slew rate is set to 0 (Immediate).

<u>Note:</u> The module must be set to the DCON protocol mode before calibrating.

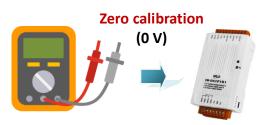
Terminal Commar	nd Line Tool			×
COM Port	COM5 ~	Protocol	DCON	
Baud Rate	115200 ~	Format	N,8,1-None Parity	✓
Checksum	Disabled \checkmark	Address	1 ~	✓ Send
Timeout	100 ~ ms	Select ID	tDA1P1R1	
Command	\$0110	2		
Response	!01			
GET_MODULE_ GET_MODULE_ GET_MODULE_ GET_MODULE_ GET_MODULE_ SET_MODULE_ OUTPUT_CH0_ READ_CH0_AO OUTPUT_DO READ_DO SET_CH0_AO_I GET_CH0_AO_I GET_CH0_AO_I GET_CH0_AO_S	FIRMWARE CONFIG PROTOCOL PROTOCOL AO RANGE_SLEW RANGE_SLEW POWERON	· · · · · ·	午 05:34 : [#010+00.000]; 午 05:34 : [\$013000]; [!01 午 05:35 : [\$0100]; [!01]; 午 05:35 : [#010+10.000]; 午 05:35 : [\$013000]; [!01 午 05:35 : [\$0110]; [!01];	>]; [13 ms]==>OK ; [13 ms]==>OK
	Export Commands		Clear	Save to path\log_report\

Calibration voltages or current used by the tM-DA1P1R1:

Type Code (T)	0	1	2	4
Zero Output	0mA	4mA	0V	0V
Span Output	+20mA	+20mA	+10V	+5V

- Send command to output Zero calibration voltage/current. (E.g., 0 V)
 Format: #AAN(Data) = # [AA] Address [N] Channel No. [Data] +00.000 (V, mA)
 Command: #010+00.000
- 4. Adjust the value

Repeat to send the trim command and observe the meter until the meter's reading is close to zero calibration voltage/current.



Format:\$AA3NVV = \$ [AA] Address[3] Set analog output[N] Channel No.[VV] 00 to 5F means to increase 0 to 95 counts ;FF to A1 means to decrease 1 to 95 countsCommand:\$01301F(1F stands for add 31 * 2.44 mV or add 31 * 4.88 uA)

5. After adjusting the value, send the Zero calibration command. Note that the value will be stored in an EEPROM.

Format: AAON = Address [0] Zero calibration [N] Channel No. Command: 0100

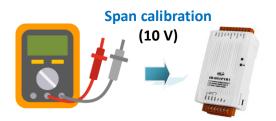
6. Send command to output **Span calibration** voltage/current. (E.g., 1**0 V**)

 Format:
 #AAN(Data) = # [AA] Address
 [N] Channel No.
 [Data] +10.000 (V, mA)

 Command:
 #010+10.000

7. Adjust the value

Repeat to send the trim command and observe the meter until the meter's reading is close to span calibration current/voltage.



Format:\$AA3NVV = \$ [AA] Address[3] Set analog output[N] Channel No.[VV] 00 to 5F means to increase 0 to 95 counts ;FF to A1 means to decrease 1 to 95 countsCommand:\$01301F(1F stands for add 31 * 2.44 mV or add 31 * 4.88 uA)

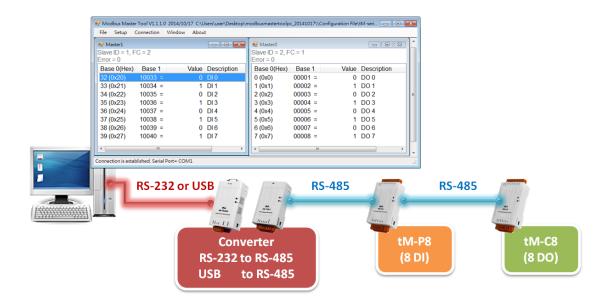
8. After adjusting the value, send the **Span calibration** command. Note that the value will be stored in an EEPROM.

Format: AA1N = [AA] Address [1] Span calibration [N] Channel No. Command: 0110

Chapter 4 Software Tool – Modbus Master Tool

Modbus Master Tool supports Modbus RTU/ASCII protocol that can be used to simulate and test I/O modules on a PC (or PAC). First, download the software and the user manual on FTP.

Web: https://www.icpdas.com/en/product/guide+Software+Development__Tools+Modbus__Tool#674 Tool: https://www.icpdas.com/en/download/show.php?num=1026 (PC or WinCE)



Hardware Wiring for testing:

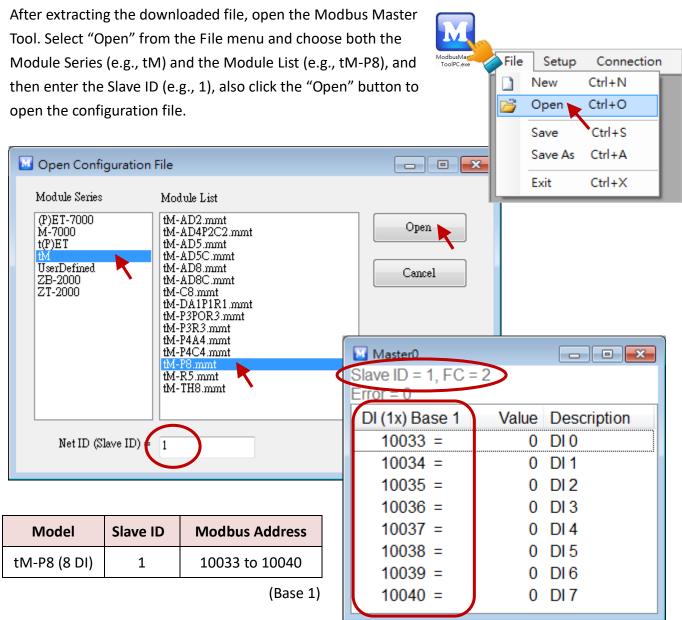
	Converter	t	tM-P8 (8 DI)	tM-C8 (8 DO)
	Data+	C	Data+	 Data+
	Data-	(Data-	 Data-
Power Supply	+Vs		+Vs	+Vs
(+10 ~ +30 V _{DC} , GND)	GND	—— (GND	GND
			DI.COM	 DO.PWR
	RS-232 or USB			DO.GND
PC (COM1 or USB)		[DIO to DI7	 DO0 to DO7

In this example, using the tM series DI and DO module. Follow the steps below to conduct the setting.

Model	Description	Slave ID	Baud Rate	Protocol	Modbus Address
tM-P8	8 DI	1	0600		10033 to 10040
tM-C8	8 DO	2	9600	Modbus RTU	00001 to 00008

<u>Note:</u> Refer to Appendix C to look up the Modbus registers for tM series modules.

Step 1: Open the configuration file



Step 2: Add a window and configure the DO

Select "New Window" from the Setup menu to add a window, and select "Definition" to configure the DO settings, and then click the OK button.

	U Definition	\frown			×
Setup Connection Window	Slave ID:	2			ОК
Definition	Туре:	Read Coils S	Status (0xxxx) for	DO	
New Window Set Value	Addresses:	Base 0	Base 1		Cancel
Set Description	Address:	1			
	Length:	8	00001	to	80000
	Format:	Hex	•		
	Descriptions	🔲 Clear All E	Descriptions		

Model	Slave ID	Modbus Ac	ldress	Dout	ale-click o	n the "Description"
tM-C8 (8 DO)	2	00001 to 0	8000		to add a	•
File Setup Co	nnection Wir	ndow About	(Base 1)			
Master0			Y Mas	ter1		
Slave ID = 1, FC Error = 0	= 2		Slave I	D = 2, FC =		
DI (1x) Base 1	Value D	escription	DO (Dx) Base 1	Value	Description
10033 =	0 D	10	00	0001 =	0	DO0
10034 =	0 D	11	00)002 =	0	D01
10035 =	0 D	12	00)003 =	0	DO2
10036 =	0 D	13	00)004 =	0	DO3
10037 =	0 D	14	00)005 =	0	DO4
10038 =	0 D	15	00)006 =	0	DO5
10039 =	0 D	16	00)007 =	0	DO6
10040 =	0 D	17	00	= 8000	0	DO7
•						4
						:

Step 3: Create a Serial Port connection

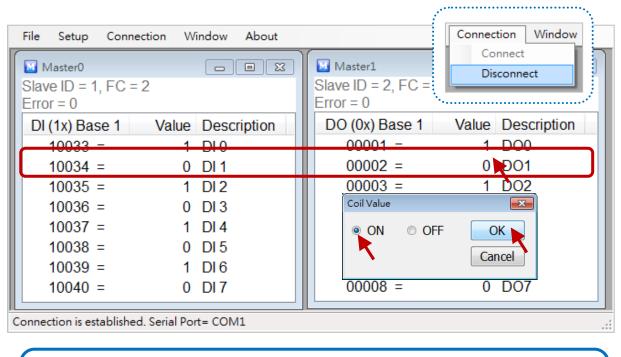
In this case, the tM module communicates with the COM1 of a PC through the Modbus RTU protocol. Select "Connect" from the "Connection" menu, and configure all parameters as illustrated in the figure below, and then click the "OK" button to establish the connection.

Note: The communication parameters of the tM module must be predefined by using the DCON Utility Pro, refer to Chapter 3 for more information.

		Connecti	on	Window		
		Con	nect			
		Disc	onneo	ct		
		_				
🔟 Connect						—
I	nterface:	COM1	•	Sca	an Interval(ms):	440
В	audrate:	9600	•		Timeout(ms):	200
	Data Bit:	8	•	Delay Be	tween Poll(ms):	20
	Parity:	0-None Parity	•			
:	Stop Bit:	1	•			
	Mode:	® rtu © asc	II		Cancel	ОК

Step 4: Test the DI and DO modules

After connecting with modules, both the current DI and DO status of modules will be displayed. Double-click on any DO channel to modify the DO value and then to view the change of DI value. **Note:** To disconnect with modules, simply click "Disconnect" from the "Connection" menu.



DI (1x) Base 1	Value Description	DO (0x) Base 1	Value	Description
10033 =	1 DL0	00001 =	1	DO0
10034 =	1 DI1	00002 =	1	DO1

Step 5: Save the configuration file

After completing the test, click "Save As" from the "File" menu to save the configuration file.



Chapter 5 Software Tool – EZ Data Logger

<u>Note:</u> EZ Data Logger no longer provides a new function since July 2019. It is recommended to use eLogger HMI.

Support Items	EZ Data Logger	eLogger
Hardware platform	PC	PC, XPAC, WinPAC, ViewPAC, iPPC Win-GRAF series PAC
OS	Windows XP, 7, 10	Windows XP, 7, 10, WinCE 6, 7
НМІ	Yes	Yes
Web HMI	No	Yes
Data Logger	Access, MS SQL, MySQL	CSV, MS SQL, MySQL
Protocol	Modbus RTU, Modbus TCP, and DCON	Modbus RTU, Modbus TCP, and I-8K/I-87K on slot
3rd Party Driver Support	No	Driver develops tool to add plug-in 3rd party driver

eLogger is a free and easy HMI development platform designed by ICPDAS. eLogger can not only be used to design local and Web Server HMI but can also achieve remote PAC control via a Web browser on PC or smartphone. With Win-GRAF, it is easy to create a professional monitoring application without any advanced programming background.

Website: https://www.icpdas.com/en/product/guide+Software+eLogger+eLogger

eLogger Features:

- Support Modbus RTU/ASCII and Modbus TCP Protocols
- Support Real-time Data Trend

The maximum of five lines in one plot.

• Support Local HMI:

Support a variety of HMI elements and a maximum of 32 pages.

• Support Web Server HMI:

Support administrator login.

• Support Account Management (for Local HMI):

3-level operating management: Admin, Power User, User.

• Support Remote Maintenance:

eLogger is not only supported to upload a program or web pages and execute (or stop) the program through the Internet.

• Support Data Logging

Local database: Support CSV files. Remote database: Support Microsoft SQL Server 2005 or later.





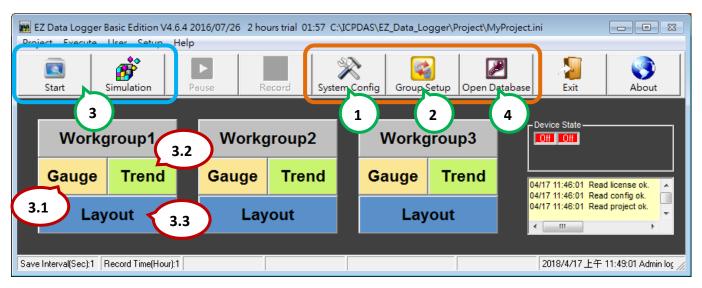


EZ Data Logger is a small data logger software. It supports DCON and Modbus RTU/ASCII protocols, and users can quickly and easily build a data logger software without any programming skill.

To download the software and the user manual (.chm) on the website. https://www.icpdas.com/en/product/guide+Software+EZ__Data__Logger+EZ__Data__Logger

Install and run EZ Data Logger. The software provides four major steps and will be described in the following content.





- Step 1: System Configuration
- Step 2: Workgroup setting
- Step 3: Start to Run (or Simulation)
 - 3.1 Gauge
 - 3.2 Trend Line
 - 3.3 Layout
- Step 4: Open Database

First, choose a preferred display language from the Project menu for the software.

Project Execute	User Setup	Help		
New Project				
Open Project		Pause	Record	
Save Project As		Fause	Record	
Open Database	e			
Open Error Me	ssage			_
language	×	English	K	
Exit		Simple Cl	hinese	
Gauge	irena	Tradition	al Chinese	

5.1 System Configuration

The "System Config" window provides the settings for the trend line and the database.

Start Sime	Pause	Record	System Config	Group Setup	Øpen Database	Exit	About
	System Config Ver 4.6.6					X	
1	Display Mo Sampling Time(so Graph XAxis(Minu Display Buffer S Data Forr	ode ⊂ Nume ec) 1 ite) 1 ize 28800 mat 123.456 □ Auto s		-	Save		
2	Local Database Cor		SQL Server Confi		MySQL Config		
	✓ Log Local Databa Save Interval(sec)			Disab Passw		rd	
	Record Time(Hour) Database File Path		00~00:59:59, _Data_Logger\log\	.,23:00:00~2	3:59:59)	• 	

Oscillograph Config			
Display Mode	Numeric: Log data by list. The minimum sampling time is 0.01 sec.		
	Date/Time: Log data by time. The minimum sampling time is 1 sec.		
Sampling Time (sec.)	The time interval for scanning IO channels, updating database, and display values.		
Graph X Axis (Minute)	The span of the trend display.		
Display Buffer Size	Maximum points of each trend line.		
Data Format	The format of the display values.		
Auto start when executing	Check the box to automatically log data when the program is launched.		
Local Database Config			
Save Interval (sec.)	Time interval of saving data in a database (a multiple of sampling time).		
Record Time (Hour)	EZ Data Logger will create a new database when the record time is up.		
	Auto renamed by Datayyyymmdd_hh.mdb when record time is up		
Database File Path	(yyyy: year mm: month dd: day hh: hour).		
	The database will be saved in the path.		

5.2 Workgroup Setting

The "Workgroup Setting" window can be divided into four major steps.

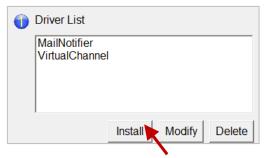
- Step 1: Install the Driver
- Step 2: Add a Device
- Step 3: Modify the Channel Settings or Add a Workgroup
- Step 4: <u>Configure the Workgroup Settings</u>

Start Simulat	ion Pause	Record	System Config	Group Setup	Dpen Database	Exit	About
Substitution Setting Workgroup Setting Driver List MailNotifier VirtualChannel	1	Ø Device	e List IChannel)		rk Group orkgroup 1 orkgroup 2 orkgroup 3 4	Add Delete Property
Ins Channel List Al List Counter List Contact List Nickname Tag	tall Modify Delet AO List Freq List Web Camera Location Gain O			Modify C DO List Control Logic	Delete	Al Channels AO Channels DI Channels OO Channels Counter Channels Freq Channels Virtual Channe Webcam Brow	; !ls
<				(i) Ada	dify d>> 3	Ø	Home

<u>Note:</u> For more information about the Virtual Channel, Control Logic, Web Camera, Mail Notifier, and User Level, refer to <u>EZ Data Logger User Manual</u> (.chm).

5.2.1 Install the Driver

1. Click the "Install" button on the Driver List.



2. Configure the Driver Mode, Timeout (ms), COM Port, Baud Rate, and Checksum to meet the needs of equipment, and then click the "Install" button.

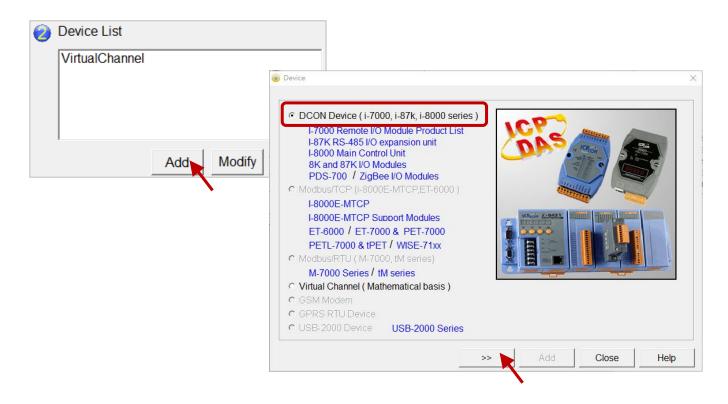
Driver Driver Mode DCON	Device List	×
Timeout (ms) 200 COM Port 9 • BaudRate 9600 • Checksum Disable •	DCON DCON Modbus Serial Modbus TCP Visual Channel GSM Modem Mail Notifier M2M RTU USB 2000	
Modify Install	Search +Device Close Help	

Also, simply click the "+Device" button to add a device (reference: Section 5.2.2 – Install the Device). Alternatively, click the "Search" button to search the connected device, and then click the "Close" button to close the window.

📮 Driver	×
Driver Mode DCON	Device List
Timeout (ms) 200	tM_P4C4 4*D0 + 4*DI
COM Port 9	
BaudRate 9600 -	
Checksum Disable	
Modify	Search +Device Close Help

5.2.2 Add a Device

1. Click the "Add" button on the Device List, and click the " >> " button in the Device window.



2. Choose a communication setting (e.g., DCON COM9), select the name of the tM series module also add an alias, and click the Add button.

Oevice	×
DCON COM9 C 7K C 87K on 87K4/87K8/RU-87Pn C 87K on 87K4/87K8/RU-87Pn C 87K on 87K4/87K8/RU-87Pn C 87K on 87K4/87K8/RU-87Pn C 87K on 8431/8831 C 87K on 8431/8831 C 9DS-700	
CZB-2000 Module Module Alias M-P4C4 Address Slot Al Num AO Num DI Num DO Num Counter Num Frequency Num	
Description	
A*DO + 4*DI	
< Add Close Help	

5.2.3 Modify the Channel Settings or Add a Workgroup

After clicking the module name (e.g., tM_P4C4) in the Device List, all I/O tags in the Channel List will be selected. To add them to the Work Group, specify the number of the workgroup (e.g., Workgroup1) and click the "Add >>" button.

Modify the Channel Settings

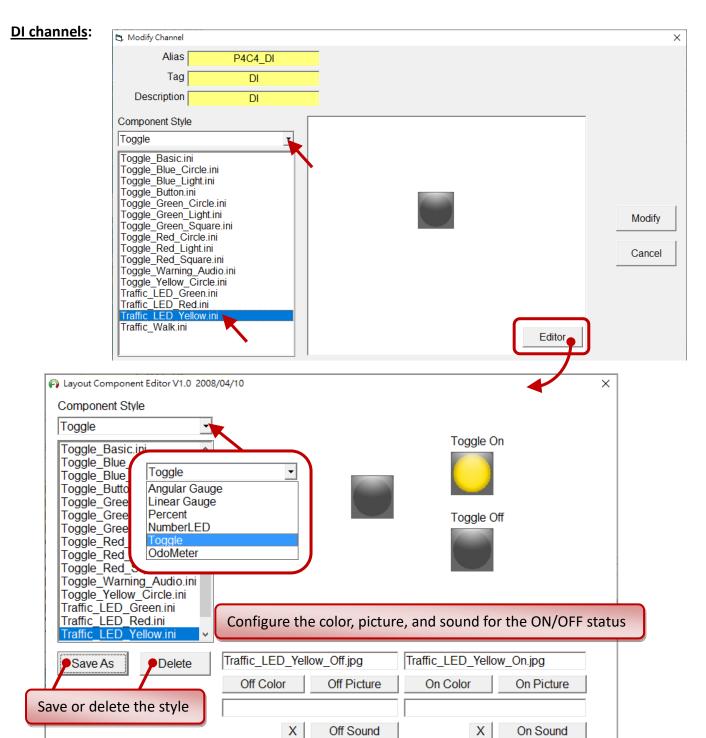
1. Choose a single (or multiple) I/O tag for the specified tM module and click the Modify button.

Workgroup Setting				– 🗆 🗙
 Driver List Modbus Seria DCON COM MailNotifier VirtualChanne 	9	Device List VirtualChanne MAD2 MP4C4	Add Modify Del	Work Group Workgroup1 Add Workgroup2 Delete Workgroup3 Delete Property Reset
Channel List Ar List Counter List Contact List	AO List AO List Freq List Web Camera	DI List	DO List Control Logic	Al Channels AO Channels DI Channels Channels
Nickname tM_P4C4_D0_0 tM_P4C4_D0_1 tM_P4C4_D0_2 tM_P4C4_D0_3	Tag Location tM_P4C4_DO_0 tM_P4C4_O tM_P4C4_DO_1 tM_P4C4_O tM_P4C4_DO_2 tM_P4C4_O tM_P4C4_DO_3 tM_P4C4_O	2h0 2h1 2h2 2h3	Description tM_P4C4_DO_0 tM_P4C4_DO_1 tM_P4C4_DO_2 tM_P4C4_DO_3	-tM_P4C4_D0_1 -tM_P4C4_D0_2 -tM_P4C4_D0_3 -counter channels -Freq Channels -Virtual Channels -Webcam Browsers
	ingle (or multiple)	I/O tag and m		Home

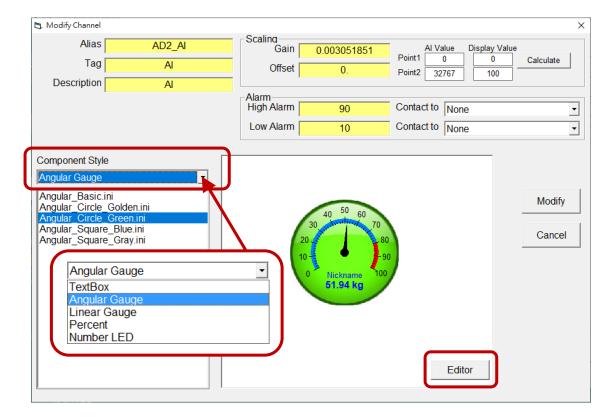
2. Change to the needed settings in the Modify Channel window, and then click the Modify button.

DO channels:	Modify Channel Alias P4C4-D0 Tag D0 Description D0 C Output confirm window	×
	Component Style Toggle Basic.ini Toggle Blue_Circle.ini Toggle_Blue_Light.ini Toggle_Green_Circle.ini Toggle_Green_Square.ini Toggle_Red_Light.ini Toggle_Red_Light.ini Toggle_Red_Square.ini Toggle_Varming_Audio.ini Toggle_Valow_Circle.ini Traffic_LED_Green.ini Traffic_LED_Yellow.ini Traffic_Walk.ini Connection Co	Modify Cancel

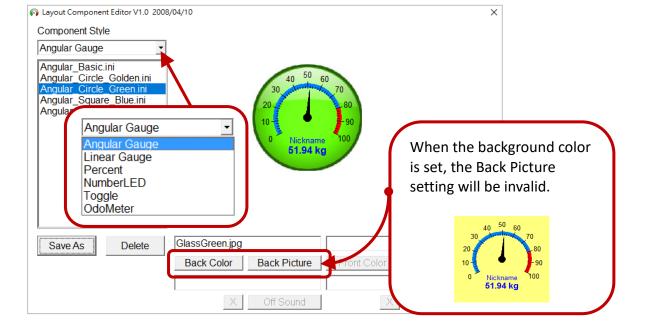
DI, DO channel			
Alias	The name that easy to identify I/O tag		
	Tag name that can be used for the Virtual Channel or Control Logic		
Тад	function. Note that the name must begin with the alphabet from a to z and		
	the first character cannot be a number.		
Description	Used to display the hint for the object		
Output confirm window	Check the box to display a confirmation window before outputting a value		
Component Style	Select the style of an object to be displayed on the Layout screen		
Editor	Click the button for modifying the style of an object		



AI channels:



Alias	The name that easy to identify I/O tag
Тад	Tag name that can be used for the Virtual Channel or Control Logic function. Note that the name must begin with the alphabet from a to z and the first character cannot be a number.
Description	Used to display the hint for the object
Scaling	Enter AI Values and Display Values, and click the Calculate button to calculate both the Gain value and Offset value.
Alarm	When the value is out of the High/Low Alarm value, a warning light comes on and plays sounds until it is closed. (Reference: 5.3.1 the Gauge screen)
Editor	Click the button for modifying the style of an object



5.2.4 Configure the Workgroup Settings

As described in Section 5.2.3, users can choose multiple I/O tags and add them to the specified workgroup. Each of the workgroups can add a maximum of 32 I/O tags.

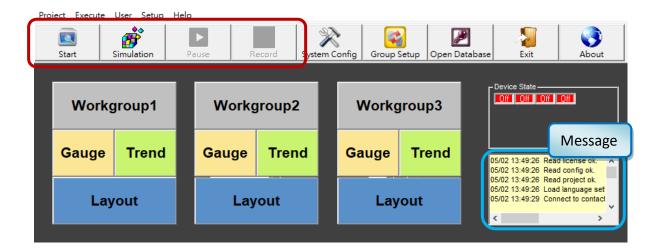
Workgroup Setting		Restore	- 🗆 X				
Driver List Modbus Serial COM1 DCON COM9 MailNotifier VirtualChannel	Device List VirtualChannel tM-AD2 tM_P4C4	Move Size Minimize Maximize x Close Alt+F4	Work Group Workgroup1 Add Workgroup2 3 Delete Property Reset				
Channel List Al List Counter List 1 Freq List	elete	Add Modify Delete DO List Control Logic	Al Channels AD2_Al0 AD2_Al1				
Contact List Web Camer		v Alarm Description High	AO Channels DI Channels P4C4_DI0 P4C4 DI1				
AD2_AI0 AI0 tM-AD2 Ch0 0.0030 AD2_AI1 AI1 tM-AD2 Ch1 0.0030		10 Al0 10 Al1 2	P4C4_Dl2 P4C4_Dl3 EDO Channels P4C4-DO0 P4C4-DO1 P4C4-DO2				
•		Modify Add>>	P4C4-D03 Counter Channels Freq Channels Home				

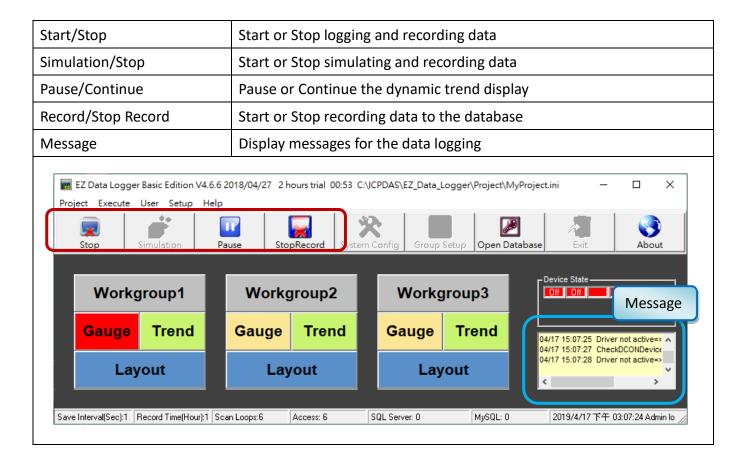
<u>Note:</u> If you cannot reduce the size of the window, simply right-click on the title bar and click Restore.

Work Group	
Add	Create a workgroup
Delete	Delete the specified workgroup
	Group name: Set a name that is easy to identify Group name Work Group Set a name that is easy to identify
Property	Open when executing: If I/O tags are added, check the Gauge, Trend, or Layout box to display the specified screen when the program is launched.
Reset	Remove all I/O tags
Channels	To delete a single I/O tag, right-click on the tag name and click Delete. P4C4_DI3
Home	Go back to the main window

5.3 Start to Run (or Simulation)

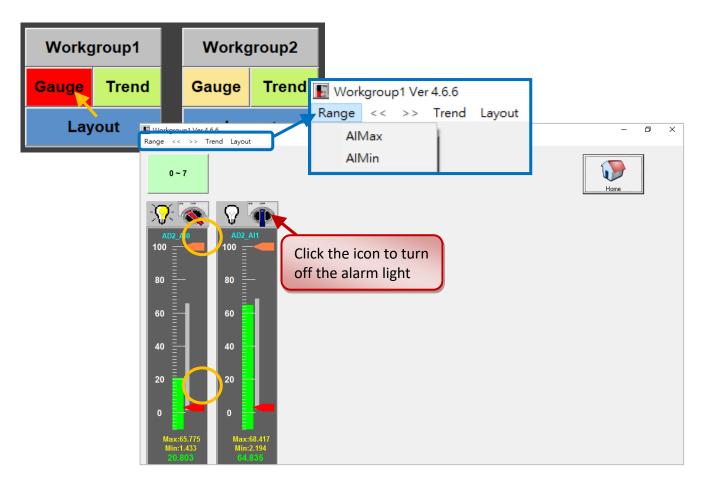
After configuring I/O channels and Workgroups, simply start to log data or execute the simulation.





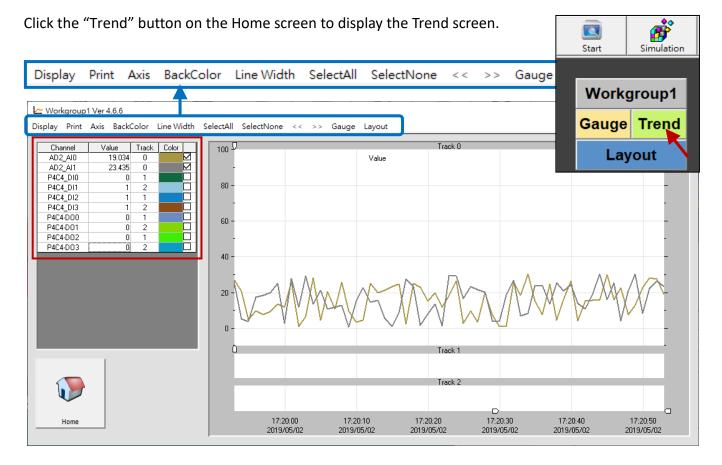
5.3.1 Gauge

Click the "Gauge" button on the Home screen to display the Gauge screen. If the logged or simulated AI value is exceeding the high/low alarm value, an alarm light will be turned on with an audio warning until the light is turned off.



Range	Used to set the maximum/minimum of analog input value if the switch is set to OFF. The orange arrow stands for the high alarm value (e.g., 90) The red arrow stands for the low alarm value (e.g., 10)
<<	Click to switch to the previous workgroup if exists
>>	Click to switch to the next workgroup if exists
Trend	Click to switch to the Trend screen
Layout	Click to switch to the Layout screen
0~7 8~15	If the number of channels is more than 8, click the button to switch the screen

5.3.2 Trend Line



Display	Pause or continuous the dynamic trend										
Print	Print the trend includes the group name and file time										
Axis	Specify the maximum and minimum values on the Y-axis and the range of time in minutes on the X-axis										
BackColor	Specify the background color of the trend										
Line Width	Specify the width of the trend line										
Select All/ Select None	Display or hide all trend lines										
<< / >>	Switch to the previous or the next workgroup										
Gauge	Switch to the "Gauge" screen	Set Output Value X									
Layout	Switch to the "Layout" screen	Nickname:P4C4-DO1 Tag:DO1									
Channel	The name of the channel	Description:DO1									
Value	When executing data logging (or the simulation), click any AO/DO channel on the list to set the output value	On Off Inverse © Manual © Auto									
Track	When executing data logging (or the simulation change the display track (1 or 2).	on), click any DI/DO channel to									
Color	Specify the color of the trend line. Note that t when executing data logging (or the simulatio	_									

5.3.3 Layout

Click the "Layout" button on the Home screen to display the Layout screen.

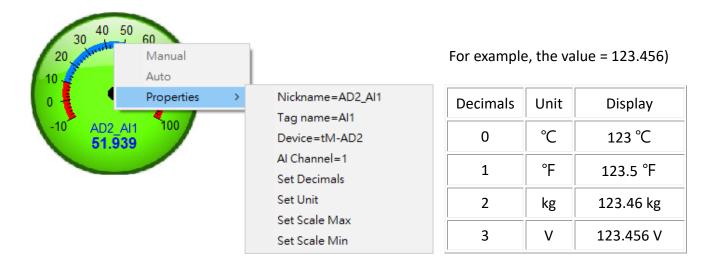
ck the "Layo	out" button or	n the Home scre	een to display	y the Layout sc	reen.	<u>s</u>
Picture	Reset Position	Font Color <<	>> Gauge	Trend Hide		Start Simulation
	Position Font Color	<< >> Gauge Trend P4C4_DI0	Hide P4C4_DI1	P4C4_DI2	P4C4_DI3	Workgroup1 Gauge Trend Layout
AD2_AI0	077 %	Off P4C4-D00	On P4C4-D01	On P4C4-D02	Off P4C4-D03	
20 30 40 20 30 40 10 40 10 40 10 40 10 40 14	0 50 60 70 80 90 2 Al1 100				Home	

Picture	Set or cancel the background picture, the jpg or bmp format is supported.
Reset Position	Reset the position for all objects on the screen
Font Color	Set the color of the tag name
<< / >>	Switch to the previous or the next workgroup
Gauge	Switch to the "Gauge" screen
Trend	Switch to the "Layout" screen
Hide	To hide the current screen

Right-click menu:

On the Layout screen, the right-click menu of the object provides some options that can be used to manually or automatically output value or set properties.

Note: The "Manual" and "Auto" options are used for logical control, refer to EZ Data Logger User Manual (.chm) for more information. When the control logic is active (reference: Section 5.2 -Workgroup Setting - Control Logic), EZ Data Logger will automatically output AO or DO value according to the logic. Also, select "Manual" to manually change the value.



Properties	
Nickname	The specified alias in the Channel list (Reference: Section 5.2.3)
Tag name	The specified tag name in the Channel list (Reference: Section 5.2.3)
Device	The specified name of a module in the Device list (Reference: Section 5.2.2)
I/O Channel = n	Display the number of the I/O channel
Set Decimals	Specify the display decimal point, it can be 0 to 3
Set Unit	Specify the display unit, e.g., °C
Set Scale Max	Specify the display maximum scale value, e.g., 100
Set Scale Min	Specify the display minimum scale value, e.g., -10

5.4 Open Database

In the Open Database window, it allows opening a log file (.mdb) and export data into an Excel or CSV file.

Start	Simulation	Pause	Record	System Config		Open Database	Exit	About
Start	Simulation	Pause	Record	system Conlig	Group setup		EXIL	About
	. D		ll SelectNor	N Auto	V Auto Dev	locales rue		
		Print SelectA	al selectivor	ie Y-Axis	A-AXIS Bac	CKCOIOF EXIT		
	Ver 4.3.2 2015// 4/2	2 X-Axis BackColor	Evit					- 🗆 X
Database			Workgroup		3	' ' Time'	Scale ' '	-
MyProject201905		^	Workgroup1		2019/5/3 下午 01	:54:53~2019/5/3下午 01:	59:59	
MyProject201905 MyProject201905		¥	Workgroup2 Workgroup3		Load Table	Load Trend	Print	Export
C:\ICPDAS\EZ_D	ata_Logger\log\MyP	roject20190!]			Load Hend	1-11112	
Name AD2_AI0	Value Track 0				Track O			1
AD2_AI1 tM-DA1P1R1_4000	0						Time	
P4C4_DI0 P4C4_DI1	1							
P4C4_DI2 P4C4_DI3	1 2	60-						
P4C4-D00 P4C4-D01	1 2							
P4C4-D02 P4C4-D03	1	40-						
		20 -						
		i i i i i i i i i i i i i i i i i i i						
		0-						
					Track 1			
					Track 2			_
<		>						
I IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	.og	▶ ► 00:00 1899/12/	29	18:00 1899/12/30	12:0 1899/1		06:00 899/12/30	00:00 1899/12/:

The menu bar provides functions to print or display data for each I/O channel and configure the style of a trend. Follow the steps listed on the next page to load the data table or trends.

Print	Print the data table or trends
Select All / Select None	Check or uncheck all box to display or hide the logged data of all channels
Y-Axis	Specify the maximum (or minimum) display value on Y-axis
X-Axis	Specify the display time scale on X-axis in minutes
BackColor	Specify the background color of the trend
Exit	Close the window

Note: When logging (or simulating) data, the settings of Y-Axis, X-Axis, and BackColor cannot be changed.

Step 1: Select a database:

Choose a log file in the list or click the browse button to open a file in a specific folder.

Database	
MyProject20190503_13.mdb	^
MyProject20190503_11_old.mdb	
MyProject20190503_11.mdb	Y
C:\ICPDAS\EZ_Data_Logger\log\MyProject20190503_11_old.mdb	

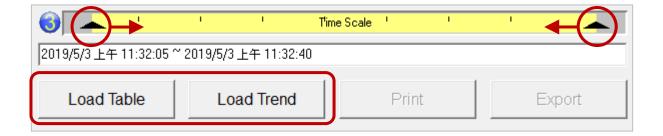
Step 2: Select a workgroup:

Choose a workgroup that contains I/O tags. (Reference: Section 5.2.3)



Step 3: Specify a time scale and load the table/trend:

Drag the arrow to the left or right to specify the time scale, and then click the **Load Table** or **Load Trend** button to load a data table or trends.



1) Load Table: Click the "Load Table" button to load a data table.

Note that check (or uncheck) the box for all channels to display or hide data.

Name	Value	Track	Cold			List	AD2 AI00	AD2 AI11	PACA DI02	PACA DI13	PACA DI24	PACA DISE		P4C4-D017	P4C4-DD28	P4C4-DD39	SamplingTime	AlarmLog 🔺
AD2 AI0	5,963	0	COR		►		5.963	27.384	0	0	0	0	0	0	0	0	2019/5/3 上午 11:32:05	AD2_Al0_Low Alarm.
AD2_AI1	27.384	n n		M	Ľ.	1	20.646	16.88	1	1	1	0	0	0	0	0	2019/5/3 上午 11:32:06	AD2_AI0 Low Alarm
P4C4 DI0	0	1		<u>N</u> NN		2	20.191	0.552	0	1	0	1	0	0	0	0	2019/5/3 上午 11:32:07	AD2 Al1 Low Alarm.
P4C4_DI1	0	2				3	19.501	22.15	1	0	1	0	0	0	0	0	2019/5/3 上午 11:32:08	AD2_Al1 Low Alarm
P4C4_DI2	0	1				4	20.161	14,701	0	0	1	1	0	0	0	0	2019/5/3 上午 11:32:09	
P4C4_DI3	0	2			-	5	14.408	17.457	1	1	0	0	0	0	0	0	2019/5/3 上午 11:32:10	
P4C4-D00	0	1		\bowtie		6	16.898	4.334	1	1	0	1	0	0	0	0	2019/5/3 上午 11:32:11	AD2 Al1 Low Alarm.
P4C4-D01	0	2		NNN	-	7	20.276	1.923	0	0	1	1	0	0	0	0	2019/5/3 上午 11:32:12	
P4C4-D02	0	1			-	8	22.413	23.069	0	1	1	0	0	0	0	0	2019/5/3 上午 11:32:13	AD2 Al1 Low Alarm
P4C4-D03	0	2			-	0	6.043	21.079	0	0	0	0	0	0	0	0	2019/5/3 上午 11:32:14	AD2_AI0 Low Alarm.
					-	9			0	0	0	0	0	0	0	0		ADZ_AIU LOW AIAIIII.
						10	3.442	16.193	1	1	0	1	0	0	0	0	2019/5/3 上午 11:32:15	
						11	20.054	4.251	1	1	0	0	0	0	0	0	2019/5/3 上午 11:32:16	AD2_AI0 Low Alarm
AD2 AI0 Low Alarm, Value	=5.963<10			~		12	25.941	28.086	0	1	1	0	0	0	0	0	2019/5/3 上午 11:32:17	AD2_Al1 Low Alarm
	-0.000 10					13	17.948	6.326	0	1	0	0	0	0	0	0	2019/5/3 上午 11:32:18	AD2_Al1 Low Alarm.
						14	10.446	20.551	0	0	0	1	0	0	1	0	2019/5/3 上午 11:32:19	AD2_Al1 Low Alarm
						15	3.314	29.893	1	1	0	0	0	0	1	0	2019/5/3 上午 11:32:20	AD2_AI0 Low Alarm.
						16	13.919	3.934	0	1	1	0	0	0	1	0	2019/5/3 上午 11:32:21	AD2_AI0 Low Alarm
· · · · · · · · · · · · · · · · · · ·				~		17	23.405	7.584	1	0	1	1	0	0	1	0	2019/5/3 上午 11:32:22	
₩ 4 1/29				· M		18	0.348	23 304	0	1	4	1	0	0	1	0	2010/5/3 上午 11:32:23	

- 2) Load Trend: Click the "Load Trend" button to load data trends.
 - * Check (or uncheck) the box for all channels to display or hide trend lines.
 - * Click on the trend to display values at the specific time.
 - * Click the scroll button to display an alarm message.



- 3) <u>Print:</u> Click the "Print" button to print the trend.
- 4) Export: Click the "Export" button to save data in an Excel or CSV file.

Export to an Excel file:

P2	P2 • : × ✓ fx														
	А	В	С	D	E	F	G	н	I	J	к	L	М	N	(🛋
1	List	AD2_AI00	AD2_AI11	P4C4_DI02	P4C4_DI13	P4C4_DI24	P4C4_DI35	P4C4-DO06	P4C4-D017	P4C4-DO28	P4C4-DO39	SamplingTime	AlarmLog		
2	0	5.963	27.38	0.00	0	0	0	0	0	0	0	11:32:05	w Alarm. Valu	ue=5.963<10	
3	1	20.646	16.88	1.00	1	1	0	0	0	0	0	11:32:06	Alarm Off. Va	lue=20.646>10	0
4	2	20.191	0.55	0.00	1	0	1	0	0	0	0	11:32:07	w Alarm. Valu	ue=0.552<10	
5	3	19.501	22.15	1.00	0	1	0	0	0	0	0	11:32:08	Alarm Off. Va	lue=22.150>10	0
6	4	20.161	14.70	0.00	0	1	1	0	0	0	0	11:32:09			-
	Þ	MyProje	ct20190503	_11_old	+				:	•					Þ

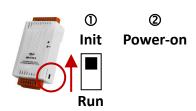
Export to a Text file:

檔案(F) 編輯(E) 格式(O) 檢視									^
1日 光(1) 湯 第二(1) 1日 20(0) 10(10)	(V) 說明(H)								
0, 5.963, 2 1, 20.646, 2, 20.191, 3, 19.501, 1 4, 20.161, 1	_AII1, P4C4_DI02, 7.384, 0, 16.88, 1, 0.552, 0, 22.15, 1, 4.701, 0, 7.457, 1,	P4C4_DI13, P4C4 0, 1, 1, 0, 0, 1,	L_DI24, P4C4_DI35 0, 0 1, 0 0, 1 1, 0 1, 1 0, 0	, P4C4-D006, F , 0, , 0, , 0, , 0, , 0,	24C4-D017, P4C4- 0, 0, 0, 0, 0, 0, 0,	DO28, P4C4-DO 0, 0, 0, 0, 0, 0, 0, 0,	39, Sam, 0, 2019/05/03 0, 2019/05/03 0, 2019/05/03 0, 2019/05/03 0, 2019/05/03 0, 2019/05/03	11:32: 11:32: 11:32: 11:32:	05,1 06,1 07,1 08,1 09,



Chapter 6 Software Development Tool

tM series modules support the DCON and Modbus protocols. When working with the DCON protocol, configure or get the module data by using DCON Utility Pro or PACSDK. When working with the Modbus protocol, develop your applications via co-work with software tools and SCADA described in Section 6.3.



6.1 Using DCON Utility Pro

- Step 1: The Protocol setting can be configured only the INIT switch is set.
- Step 2: Search for the tM module and set the Protocol to DCON, and then rebooting in Run mode.
- Step 3: Click the **Command Line** button and enter a DCON protocol command in the **Command** field. (Reference: Appendix E DCON Protocol Commands)

DCON Utility Pro PC V	4.0.0.1 Beta Version			
			1 🔙 🗊	FAQ
E COM5:* ID tP4	Address Baud 1 C4 2[02h] 11520		Status Descripti Remote I/O [DCON]4	ion Comments 4*DI + 4*DO Supported
Terminal Command Line	Tool			×
COM Port COM	5 v Proto	col DCON		
Baud Rate 1152	200 v Forma	t N,8,1-None Parity	~	
Checksum Disab	oled 🧹 Addre	ss 2 ~	~	Send
Timeout 100	√ ms Selec	t ID tP4C4		
Command \$02M	М	2		4
Response 102t	P4C4	3		
GET_MODULE_NAME GET_MODULE_FIRMW, GET_MODULE_CONFIG SET_MODULE_CONFIG GET_MODULE_PROTO SET_MODULE_PROTO READ_DI_HIGH_LATCH READ_DI_HIGH_LATCH CLEAR_DI_LATCH SET_DI_REVERSE GET_DI_REVERSE READ_CH3_DI_COUNT OUTPUT_DO RFAD_DO	i icol col H H	The commonly	used DCON protocol each module will be	
Export	t Commands	Clear	Save to p	path\log_report\

6.2 Using PACSDK

PACSDK are software development toolkits that can be used on a PC or PAC (e.g., XPAC, WinPAC, and ViewPAC series). PACSDK includes header files, libraries, documentation, and tools that can be used to develop VC/C#/VB.net applications to access I/O data of tM series module.

<u>Note:</u> PACSDK only supports the DCON protocol.



PACSDK

Items		PACSDK	Description
Header file		PACSDK.h	Using libraries when developing VC applications on
Library files		PACSDK.lib	Windows PC.
	For C program	PACSDK.dll	Loading libraries when executing VC applications on PC or PAC.
DLL files	For .NET program (i.e., C#, VB)	PACNET.dll	 Using management libraries when developing .NET applications Load libraries when executing .NET applications on PC or PAC.

Download Files on FTP:

1) PACSDK:

Users can download SDK according to the development platform (i.e., PC or PAC).

When using the following development platform - **PC or XPAC (WES)**, copy PACSDK.dll and PACNET.dll to the folder where the applications (.exe file) is located after downloading.



https://www.icpdas.com/en/product/guide+Software+Development__Tools+PAC__SDK

XPAC, iPPC (WES7)

For XP-8000-WES7, XP-9000-WES7, iPPC (WES7)

https://www.icpdas.com/en/download/show.php?num=2540

When using the following development platform - **XPAC (CE6)**, **WinPAC (CE5)**, or **WinPAC/ViewPAC (CE7)**, visit the following webpages to download and install files.

Development Tools: Visual Studio 2008 Professional Edition or earlier

XPAC (CE6)
For XP-8000-CE6 https://www.icpdas.com/en/download/show.php?num=2473
WinPAC (CE5)
WinCE5.0 PACs/ViewPACs without I/O Slot(s) https://www.icpdas.com/en/download/show.php?num=2594 WinCE5.0 PACs/ViewPACs with I/O Slot(s) https://www.icpdas.com/en/download/show.php?num=2593
WinPAC/ViewPAC (CE7)
WinCE7.0 PACs/ViewPACs without I/O Slot(s) https://www.icpdas.com/en/download/show.php?num=2409 WinCE7.0 PACs/ViewPACs with I/O Slot(s) https://www.icpdas.com/en/download/show.php?num=2348

2) PAC API Manual:



The user manual describes the way to install the PACSDK, set up the development environment, and use the **PAC_IO** API functions. (pac_standard_api_manual_x.x.x.pdf)

Download the user manual according to the development platform at the following path.

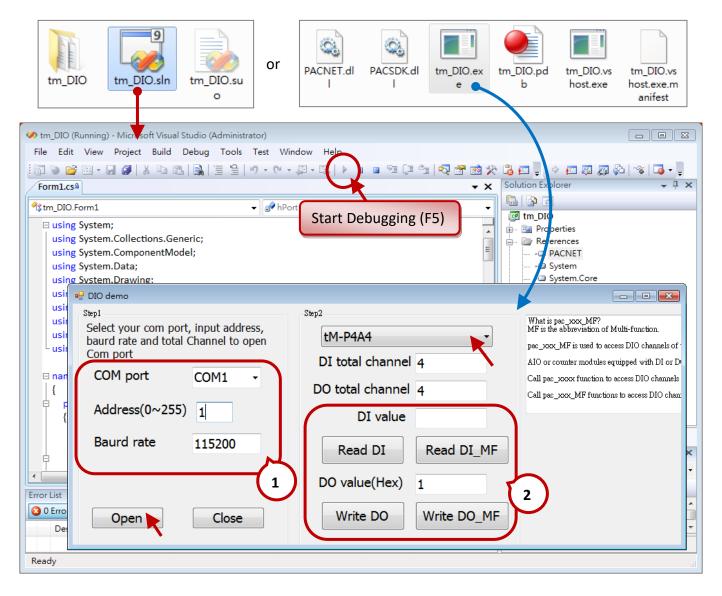
PC
https://www.icpdas.com/en/download/show.php?num=1049
PAC (WES)
https://www.icpdas.com/en/download/show.php?num=2527
PAC (WinCE)
https://www.icpdas.com/en/download/show.php?num=2407

PAC_Standard_API_Manual	
書鏡 ※ ● ● ● ● ● 新増 ● 編輯 ● 係 ■ 2.7. PAC_IO API 2.7.1 coc GotPit 2.7.2. pac_WriteDO/pac_WriteDO_MF	int iBitValue = 1; bool ret = PACNET.IO.WriteDOBit(hPort, iSlot, iDO_TotalCh, iChannel, iBitValue); PACNET.UART.Close(hPort); This chapter lists all commands for I/O modules.
2.7.3. pac_WriteDOBit 2.7.4. pac_ReadDO/pac_ReadDO_MF 2.7.5. pac_ReadDI/pac_ReadDI_MF 2.7.6. pac_ReadDIO/pac_ReadDIO_MF	[C#]
 2.7.7. pac_ReadDILatch 2.7.8. pac_ClearDILatch 2.7.9. pac_ReadDIOLatch 2.7.10. pac_ClearDIOLatch 2.7.11. pac_ReadDICNT/pac_ReadDICNT_MF 2.7.12. pac_ClearDICNT/pac_ClearDICNT_MF 2.7.13. pac_WriteAO/pac_WriteAO_MF 2.7.14. pac_ReadAI 2.7.15. pac_ReadAI 2.7.16. pac_ReadAIHex 2.7.17. pac_ReadAIAIIExt 2.7.18. pac_ReadAIAII 2.7.19. pac_ReadAIAII 2.7.19. pac_ReadAIAII 2.7.19. pac_ReadAIAIIHexExt 2.7.20. pac_ReadAIAIIHex 	<pre>// If using the remote I/O such as the I-7K, M-7K, tM series module IntPtr hPort; hPort = PACNET.UART.Open(""); byte iAddr = 1; int iChannel = 2; int iDO_TotalCh = 8; int iBitValue = 1; bool ret = PACNET.IO.WriteDOBit(hPort, PAC_REMOTE_IO(iAddr), iDO_TotalCh, iChannel, iBitValue); PACNET.UART.Close(hPort);</pre>
 2.7.21. pac_ReadCNT 2.7.22. pac_ClearCNT 2.7.23. pac_ReadCNTOverflow 2.7.24. pac_WriteModuleSafeValueDO/pac_WriteModuleSafeValueDO_MF 	Remarks The function can support for Local or Remote. When the module is local, the second Parameter's range is from 0 to 7. If remote, the second Parameter need use the
2.7.25. pac_ReadModuleSafeValueDO/pac_ReadModu +	macro, PAC_REMOTE_IO(0255), which range is from 0 to 255.

3) C# demo program (for DCON protocol):

www.icpdas.com/en/download/show.php?num=2876

In the case of a DIO demo, refer to Section 2.4 to complete the hardware connection, and double-click the tm_DIO.sln to open the project file, and then click the "Start Debugging" button. Alternatively, double-click the tm_DIO.exe at the "...\tm_DIO\bin\Debug" path.



- Step 1: Choose the COM Port, enter the address (Slave id) and baud rate for the tM module, and click the Open button to open the COM port. In this example, COM Port = COM9, Address = 1 and Baud rate = 9600. If you do not know the settings of the module, refer to Chapter3 to configure them by using DCON Utility Pro. Note: The Protocol setting of the module must be set as DCON.
- Step 2: Choose the module name (e.g., tM-P4A4) and then the number of I/O channels will automatically be displayed in the "DI (or DO) total channel" field. Also, click the "Read DI" button to read the DI value, or enter a hexadecimal value (e.g., "C" = 1101) in the "DO value (Hex)" field and click the "Write DO" button to write a DO value. Note: When using a multi-function I/O (e.g., tM-AD4P2C2), replaced to click the "Read DI_MF" or "Write DO_MF" button to access data.

After completing the previous steps, the screen will be displayed as below: **DI value = 15** (10) = 1111, indicates the status of DI3, DI2, DI1, and DI0 are ON. **DO value = c** (16) = 1101, indicates the status of DI3, DI2, and DI0 is ON, and the status of DI1 is OFF.

🖳 DIO demo		
Step1 Select your com port, input address, baurd rate and total Channel to open Com port COM port COM9 Address(0~255) 1 Baurd rate 9600 Open Close	Step2 tM-P4A4 DI total channel 4 DO total channel 4 DI value 15 Read DI Read DI_MF DO value(Hex) c Write DO Write DO_MF	What is pac_xxx_MF? MF is the abbreviation of Multi-function. pac_xxx_MF is used to access DIO channels o AIO or counter modules equipped with DI or : Call pac_xxxx function to access DIO channel Call pac_xxx_MF functions to access DIO cha

<u>Step 3</u>: Click the "Close" button to release the COM port.

Users can see the change of LED indicators on the module, or observe the DI or DO status by using DCON Utility Pro. Note that the COM port in use for the demo must be released.

- Step 1: Search for the I/O module. (Reference: Section 3.1)
- <u>Step 2</u>: Double-click the name of the module to open the configuration page, and then click the DI or DO tab to see the results.

[The DI Page]:	tP4A4 Firmware[A104]
	Configuration DO Host WDT DI bout
	DI Bit Status CHOU CHOI CHO2 CHO3 DI Value OFh
[The DO Page]:	tP4A4 Firmware[A104] Configuration DO Hest WDT DI About Bit Status
	CH:00 CH:01 V CH:02 V CH:03 DO Value OCh Set to [Power On Value] Set to [Safe Value] © Read DO © Read Power ON Value © Read Safe Value

6.3 Modbus Development Tools and SCADA

ICP DAS provides a variety of software tools and SCADA that supports Modbus RTU/ASCII protocol, and helpful for users to develop their applications.

6.3.1 Software Tool – OPC DA Server and UA Series Product

NAPOPC DA Server is a free OPC DA Server for ICP DAS products that can be used to integrate data from ICP DAS remote I/O module, PAC I/O, and other equipment. Also, NAPOPC DA Server allows the SCADA/HMI/Database software that supports OPC DA Client, to access I/O data to achieve the process control and manufacturing automation applications.



Also, users can choose an optional license of the Modbus protocol for the NAPOPC_ST DA Server (PC) to work with the third-party Modbus equipment.

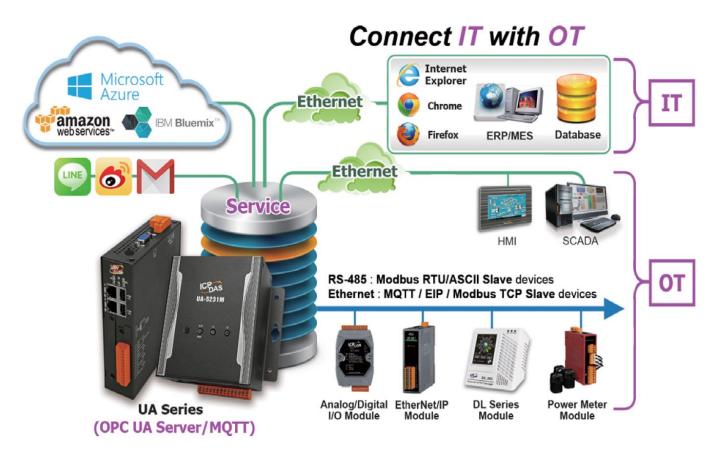
Website / Data Sheet/ DM:

https://opc.icpdas.com/opcda.htm

https://ftp.icpdas.com.tw/pub/cd/8000cd/napdos/napopcsvr/data_sheet/napopc_st_data_sheet_e.pdf https://www.icpdas.com/root/support/catalog/pdf/Brochure/NAPOPC/NAPOPC-DM-e.pdf

New Open Save Device Tag Multi. Gener. Search Expand Shrink Monitor Cut Copy Paste Delete New Open Save Device Tag Multi Gener Search Expand Shrink Monitor Cut Copy Paste Delete New Open Save Device Tag Multi Gener Search Expand Shrink Monitor Cut Copy Paste Delete New Open Save Cop Copy Paste Delete Cut Copy Paste Delete New Copy Copy Copy Copy Paste Delete Select Device Select Device Select Device Select Select Open Select Select	X ICPDAS - NAPOPC_ST DA Server (Ver 3.56) File Add Edit View Options Help		
DIs Ch00 Coi Ch01 Coi Coi DICounter Ch02 Coi LatchLowDIs Ch03 Coi LatchHighDIs Ch04 Coi Device Name Device20 Controller Setting Controller Setting	New Open Save Device Group Ta	ag Multi Gener Search Expand Shrink Monitor Cut Copy	Paste Delete
B-10 DS-720_7 B-10 ACS-11-MF_8 B-10 ACS-11-MF_9 B-10 WP5X0X_10 B-10 WP5231_12 B-10 PET-7026_11 B-10 PET-7026_11 B-10 PET-7026_11 B-10 PET-7026_14 B-10 S1-002_15 B-10 S1-002_16 B-10 D1-302_17 B-10 D1-302_19 Ready Request Tag Number Coll 122 Register 122 Ø Simulate I/O Pending Time 1000 Mode Mode OK Cancel	→ DIs → Ch00 Coi → Ch01 Coi → Ch02 Coi → LatchLowDIs → Ch03 Coi → LatchHighDIs ↔ Ch04 Coi → cb05-720_5 ↔ Ch05 Coi → cb05-720_7 ↔ Ch06 Coi → ACS-11-MF_8 ↔ Ch07 Coi → ACS-11-MF_9 ↔ Ch07 Coi → MVP5XXX_10 ↔ Ch07 Coi → MVP5231_12 ↔ SI-002_15 ↔ SI-002_15 → SI-002_15 ↔ SI-002_16 ↔ DL-302_17 → DL-302_19 DL-302_19 OL-302_19 OL-302_19	C DCON C FRnet Modbus Device Name □evice20 Controller Setting ○ ○ Modbus RTU ○ General Modbus Device ○ Modbus ASCII ○ ○ Modbus ASCII ○ ○ Modbus TCP ○ ○ Modbus TCP ○ ○ Tot 502 ○ ○ Modbus TCP ○ ○ Tot 502 ○ ○ Word Swap COM Port Setting ○ COM 1 ▼ Baud Rate 115200 ○ □ Request Tag Number Coi ○ □ ○ □ ○ □ ○ □ ○ □ ○ □ ○ □ □ □ ○ □ ○ □ ○ □ ○ □ ○ □ ○ □ ○ □	

UA Series IIoT Communication Server



UA series is a series of IIoT (Industrial IoT) Communication Server for integrating the system and devices of IT and OT. UA series feature the IIoT Gateway function that allows users to access remote I/O modules and controllers via Modbus TCP/ RTU/ASCII, MQTT, and EtherNet/IP communication protocols. IIoT gateway function can also convert these I/O data to conform to OPC UA or MQTT protocols for implementing in the applications of MES, ERP, SCADA, or Cloud service. Besides, UA series feature the Data Logger function that allows users to write I/O data directly into the remote database and save it to the local file (.CSV) as the historical records.

UA series supports uploading I/O data to the IoT Cloud platform such as Amazon AWS and Windows Azure. It can also connect to the Cloud logic service platform "IFTTT" which supports many web APPs for users to receive first-hand notification messages through the most commonly used mobile APPs when an event is triggered. UA series enhance the networking and interoperability between IT and OT. Through UA series, users can easily deploy Industrial IoT.

Visit the ICP DAS UA Series website for more information. https://www.icpdas.com/en/product/guide+IIoT+Controller_Server+Communication__Server In the example, using a UA series to connect to a tM-P4A4 module through an RS-485 port (i.e., Tty05) to read/write Modbus RTU I/O data. Also, the Modbus RTU to OPC UA data conversion function is added.

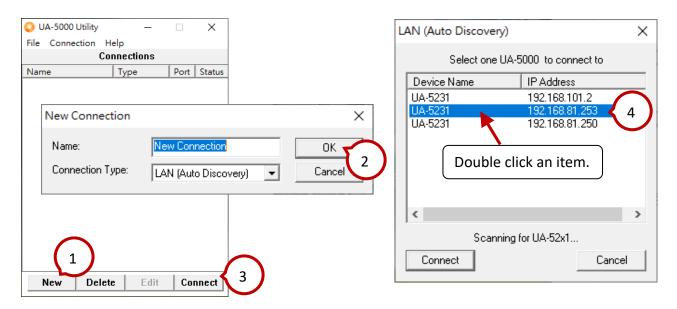


Follow the steps:

1. Log in to the configuration web page of a UA-5000 series.

If the IP address of UA-5000 series is unknown, downloading the UA-Series Utility and searching the UA device on the web.

https://www.icpdas.com/en/download/index.php?nation=US&kind1=&model=&kw=ua-



Double-click a UA-5000 list item, and the web browser will automatically be displayed. Enter the Username and Password (root) and click the Login button to log into the configuration page.

S UA-5231	× +	- 🗆 ×
$\ \ \leftarrow \ \ \rightarrow \ \ G$	▲ 不安全 192.168.81.253	* 🕒 🗹 😩 :
6	IIoT Communication Server	Username : root Password : •••• Language : English •

2. Choose "(Master) Modbus RTU/OPC UA" to add the Modbus RTU to the OPC UA conversion function.

← → C ③ 不安全 192.168.81.253/	Main.html		7	ۍ ۲	۵	\checkmark	
IIoT Communication Se							
ICP DAS Co., Ltd	rver	👉 Func	ction Wizard (Click here)				•
		Func	ction Wizard (Click here)				
System Setting Module Setting	IoT Platform Setting	Convort So	dule Communication Conversion				
System Setting Module Setting	IOT Flationn Setting	(IV	/laster)Modbus RTU / OPC UA 📐				
			/laster) Modbus TCP / OPC UA				- 1
I/O Status File Setting		(M	/laster) Modbus ASCII / OPC UA				- 1
		M	QTT / OPC UA				

3. Set the COM Port (e.g., Tty05) and communication parameters (9600, 8, n, 1) for the UA-5231, and then click Save.

IIOT Comm	nunication Server	(Master) Modbus RTU / OPC UA	•					
Controller COM Port S	Setting > Module Setting >	OPC UA Connection 🔖 Enable Converting Module ≽ Save Project ≽						
Run the project	COM Port Interface Setting Page							
L	Serial Port	ttyO5 •						
	Baud Rate	9600 •						
	Data Bits	8 bits •						
	Parity	None						
	Stop Bits	1 bit						
	Polling Rate(ms)	500						
		Save						

4. Click "Module Setting" to set the COM Port and enter a module name, and then click the "+" button to add the module setting.

	roller COM Port Setting > Module S	Setting OPC UA Connection	 Enable Converting Module > Save Project 	>
Run t	Modbus RTU Module L	ist		
	Serial Port	ttyO5	• 1	
	Load ICPDAS Module	Select The Module	Update ICPDAS Module List	
	Select All No.	*Module Name / Nickname	Edit	
		tM-P4A4		
	Copy Remo	vve 2	< 1 / 0 >	
	Remove all	Save		

Next, click the Edit button to display the configuration page.

Ŧ	2	▼ tM-P4A4	
	1	tM-P4A4	Edit

5. The tM-P4A4 module (4 DI and 4DO) is used in this example, refer to Appendix C to look up the Modbus address. After configuring the Data Model, Start Address, and Data Number settings, click the Add button to add the Modbus Mapping Table and click OK.

Module Content Setting			
No.	1		
Module Name	tM-P4A4		
Slave ID	1		
Timeout(ms)	500		
Modbus Mapping Table S	Setting		
Data Model	02 Input Status(1x)	•	
Start Address	32		
Data Number	4		
Create Tables	Add Success.		
Modbus Mapping Table		Address Setting	Nickname Setting
Coil Status(0x)	Input Status(1x)	Holding Registers(4x)	Input Registers(3x)
Number 4 N	ddress 32 Jumber 4 Type Bool Edit		
	ОК	Cancel	

Then, it will go back to the Modbus RTU Master List page. Click Save to save the settings.

	1	tM-P4A4	Edit
	Copy Remove		< 1 / 1 >
Remove all		Saver Success.	

6. Click **OPC UA Connection** to display the Local Server page. By default, the Anonymous Login function is enabled. Just click both two Save buttons.

> OPC UA Connection >	Server	
	Server Nam	e ICPDAS_OPC_UA_Server
	Pol	rt 48010
		Save Success.
	User Identity Tokens	
	Anonymous Log	in 🗹 Enabled
	User Password Log	in 🗹 Enabled
	Certificate Log	in Enabled
		Save Success.

7. Click **Enable Converting Module** and check the Module item to enable the data conversion function, and then click Save.

Enable	Converting Module						
Modbus R	TU Module List						
No.	*Module Name / Nickname	Edit	All Enabled				
1	tM-P4A4	Edit					
< 1 / 1 >							
Save Success.							

8. Click **Save Project** to save the project and click **Run the Project** to update and execute the new project. Now, UA-5231 can conduct the data conversion for the tM module.



Visit the website for more information about UA applications. https://www.icpdas.com/en/product/guide+IIoT+Controller_Server+Communication__Server

6.3.2 Software Tool – nModbus



nModbus is a C# 3.0 implementation of the Modbus protocol. It is developed and maintained voluntarily and provided free of charge. ICP DAS verified and improved the DLL based on the official released <u>NModbus net-2.0 1.11.0.0-source.zip</u>.

Programmers can use the DLL released by ICP DAS to develop a Modbus application for regular Windows-based PCs or WinCE based devices.

Download the DLL, nModbus API manual, and demos on the nModbus website: https://www.icpdas.com/en/product/guide+Software+Development__Tools+Modbus__Tool

nModbus API Manual:

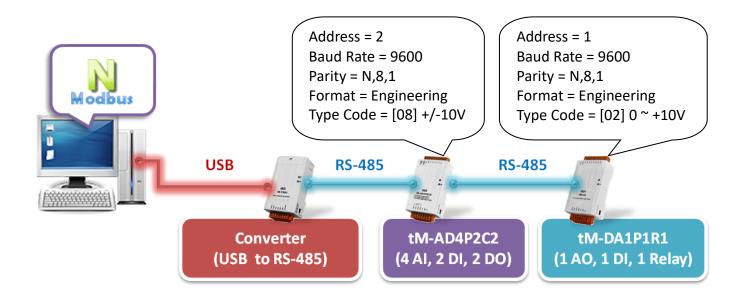
https://www.icpdas.com/en/download/show.php?num=1024

Demo: https://www.icpdas.com/en/download/show.php?num=1025

- ≻ <u>PC版</u>:
 - **DLL**: log4net.dll, nmodbuspc.dll
- ≻ <u>WinCE 版</u>:
 - **DLL**: cabc.dll, fc19.dll, nmodbusce.dll

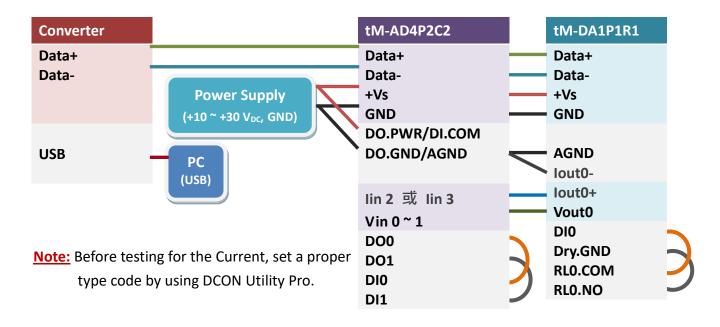
Download the Modbus RTU Master test demo that supports tM series module. www.icpdas.com/en/download/show.php?num=2876 (C# Demo for Modbus Protocol)

Before using the module, configure them by using DCON Utility Pro. (Reference: Chapter3)



Hardware Wiring for this example:

In this case, using tM-AD4P2C2 and tM-DA1P1R1 modules to test the DI/DO/AI/AO status. First, connect all devices properly as follows, refer to Section 2.3 to know the way of the hardware wiring.



Double click "modbusrtu_master_tm.exe" in the path "...\modbusrtu_master_tm\bin\Debug" on PC. Next, enter the communication parameters of a module, and click the "Open Com" button to connect.

Enter an AO value for the tM-DA1P1R1, and see the change of AIO and AI1 values for the tM-AD4P2C2. Also, click the DO button and see the change of DI status.

🖳 nModbus-TM Series		
Com COM9 -	Data Bit 8 - Stop Bit 1 - Slave ID(HEX) 1	Open Com
Baud. 9600 -	Parity None Module Module -	Close Com
AI AO DI DO	$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	

The current AIO and AI1 values of the tM-AD4P2C2 are 4.957V. Next, click the DO button to change the DI status.



6.3.3 Software Tool – LabVIEW

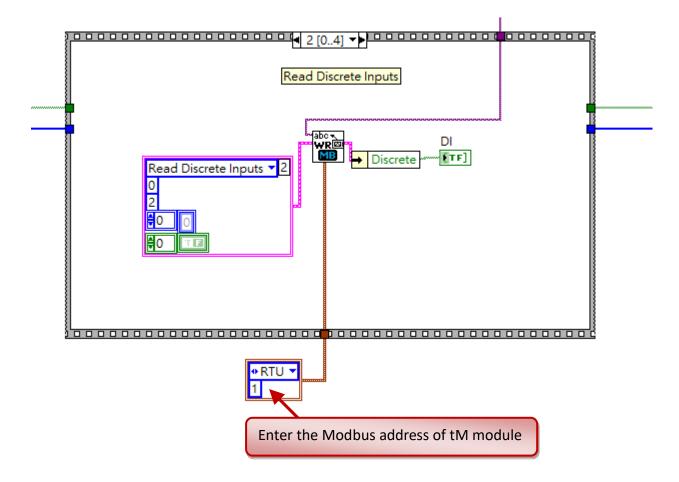
Laboratory Virtual Instrument Engineering Workbench (LabVIEW) is a system-design platform and development environment for a visual programming language from National Instruments. LabVIEW provides an easy-to-use graphical interface and supports a variety of hardware drivers and software analysis tools that help users to speed up the amount of time to develop applications. LabVIEW has been widely used for the test, measurement, and automated control in various laboratories or industries.

Download the Modbus RTU demo programs at the link, and then connect the tM module. https://www.icpdas.com/en/product/guide+Software+Development__Tools+LabVIEW__Tools

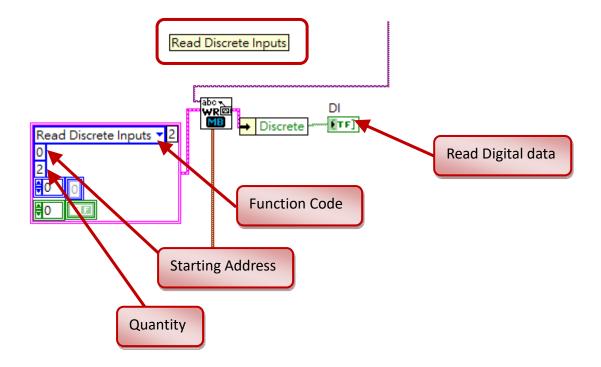
- **Note:** Before communicating with the tM module, refer to Chapter3: Software tool DCON Utility Pro to configure the module, and then confirm the COM port number and both the Modbus address and baud rate settings of the module.
- Step 1: Download the ModbusRTU.zip and unzip the file.
- Step 2: Choose a demo program for the I/O module that is connected with PC (e.g., MB_Serial_Example_ai_ao_di_do.vi), and set the proper COM port number and baud rate.

	_Example_ai_a									_		×
<u>F</u> ile	iew <u>P</u> roject					6-					9	mastr MB
	COM port		This c Mod Mod 1. Sel 2. Ru	demo based bus.IIb to d bus/RTU de bus/RTU m ect the COI n the demo the DO/AC	d on MB emonstra evice and odule. M port nu	Serial Exan ate how to write DO, . umber and	read DI, A AO to the input the	l from a same				^
	Data to write			Slave DIO		only Registe						
	A00	AO1		AI0	Al1	A12	AI3	A14	AI5			
<								STO				×

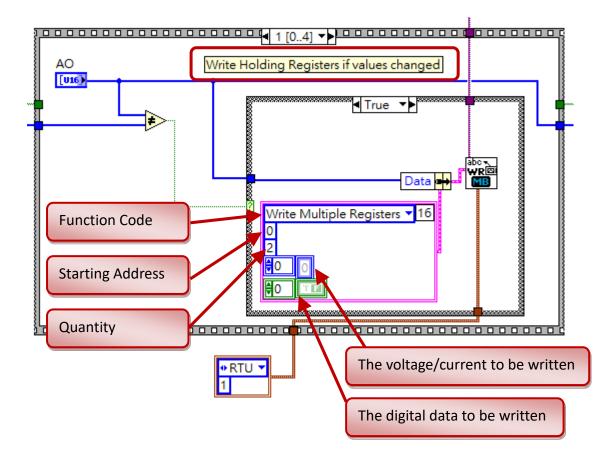
Step 3: Enter the Modbus address of the module.



Step 4: To read data from the module, set the function code, starting address, and the length of data.



Step 5: To write data to the module, set the function code, starting address, the length of data, and the data to be written.



Description of Modbus Function Codes

Function Code	Description		I/O Address
01	Read Coils	(Read D O data)	Охххх
02	Read Discrete Inputs	(Read DI data)	1xxxx
03	Read Holding Registers	(Read AO data)	4xxxx
04	Read Input Registers	(Read AI data)	Зхххх
05	Write Single Coil	(Write single DO data)	Охххх
06	Write Single Register	(Write single AO data)	4xxxx
15	Write Multiple Coils	(Write multiple DO data)	Охххх
16	Write Multiple Registers	(Write multiple AO data)	4xxxx

6.3.4 Software Tool – InduSoft

InduSoft Web Studio is a powerful, integrated collection of automation tools that includes all the building blocks needed to develop modern Human Machine Interfaces (HMI), Supervisory Control and Data Acquisition (SCADA) systems, and embedded instrumentation and control applications.

InduSoft Web Studio supports all Windows runtime platforms, ranging from 10 / 8 / 7 (32-bit/64-bit), Windows XP/ Vista, and Windows Server Editions, along with built-in support for local or remote (web) based visualization. InduSoft also conforms to industry standards such as Microsoft .NET, OPC, DDE, ODBC, XML, and ActiveX.

Notice:

ICP DAS Co., Ltd. has released InduSoft v8.1 license for Hardkey. Visit the website for the new features.

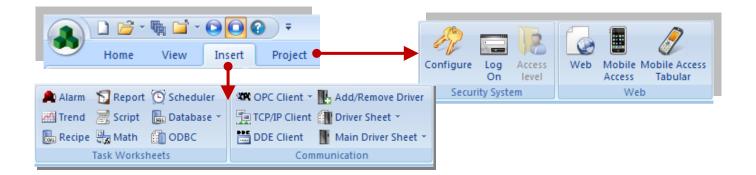


InduSoft website:

https://www.icpdas.com/en/product/guide+Software+InduSoft+InduSoft

Software Features:

- ♦ Graphics and Animation design tools
- ♦ Connect to any SQL database (MS SQL, MySQL, Sybase, Oracle), MS Access, or Excel, and ERP/MES systems
- ♦ Alarms, Events, Trends, Recipes, and Reports management tools
- ♦ Provides over 240 native communication drivers
- \diamond Supports web server, database, and overall system redundancy
- ♦ Two powerful scripting languages are supported; built-in InduSoft functions and standard VBScript.
- \diamond Remotely view screens as web pages using IE browser or InduSoft Secure Viewer
- ♦ Compare any configuration file or merge changes from multiple developers.
- Monitoring and interacting with process values on any browser (e.g., iOS Safari, Google Chrome, etc.) that supports HTML5.



Description of the demo program:



The demo program is used to read/write data from/to tM-P8 (8 DI) and tM-C8 (8 DO) modules through the Modbus RTU protocol. Refer to the InduSoft user manual for more details about the setting. First, click Run (F5) from the Home menu to execute the project.

For_tM_modules -Demo for tM series modules ShutDown ON ON OFF OFF DO2 DO3 DOO D01 OFF OFF OFF OFF D04 DO5 D06 D07

Follow the steps below:

1. Add variables

On Global tab, right-click on Project Tags and click Insert Tag to add the variable.

Home Vie	ew Insert Proj	ject Help						
Paste Ta	sks Debug *	Onnect	Tasks Dat		erify .	rt Resolution Global	DO 🔊	ce
roject Explorer	Д			MODBU001.DRV	Main01.scr			
▲ A Project: 項目1./			Name	Array	Туре	Description	Scope	
Project Tage			🔍 Filter text	🔍 Filter te	t 🔍 (All)	🔻 🔍 Filter text	🔍 (All) 🔻	-
Datasic		1	[Ј] ој	8	Boolean	·	Server -	-
Classes	~	2	[J] DO	8	Boolean	-	Server -	-
🛛 📜 Share				0	Boolean		Local -	_
🛛 👢 Syster	<u>O</u> pen			0	Integer		Local	-
b 🦺 Securi	Insert Tag			0	Real String		Server T	-
⊳	K			0	Integer	+	Server T	,
M Transl	<u>R</u> efresh			0	Integer	v	Server T	-
	Remove Unused	d Tags		0	Integer	-	Server T	-
🔊 Global 📃 Gr		P			integer		001101	-
atabase Spy			ф х	Output			1	ņ
Tag/Expression	Value	Quality Co	ntinuous 🔺	MODBU Read grou	p: 2,1X:0 > Ok (G:	roup number: 1)		
li[0]	0	GOOD		MODBU Read grou	p: 2,1X:0 > OK (G: p: 2,1X:0 > Ok (G:	roup number: 1) roup number: 1)		
lo[0]	0	GOOD		MODBU Read grou	p: 2,1X:0 > Ok (G: p: 2,1X:0 > OV (G:	roup number: 1)		
DI[1]	0	GOOD		MODBU Read grou	p: 2,1X:0 > Ok (G	roup number: 1)		
00[1]	0	GOOD GOOD	 ✓ ✓ 	MODBU Read grou MODBU Read grou MODBU Read grou MODBU Read grou MODBU Read grou MODBU Read grou MODBU Read grou	p: 2,1X:0 > Ok (G: p: 2,1X:0 > Ok (G:	roup number: 1) roup number: 1)		
DI[2] DO[2]	0	GOOD	V -		. ,	,		

2. Configure the variable and the method of the communication

On the **Comm** tab, right-click MODBU and click Insert to configure the variable. In this example, the Net-ID of tM-P8 (DI) is "2", and the Header is set to "1x: 0"; the Net-ID of tM-C8 (DO) is 3, and the Header is set to "0x: 0".

Note: Using DCON Utility Pro to set communication parameters of the tM module before using them.
 In this example, Protocol = Modbus RTU, Address = "2" (for tM-P8) or "3" (for tM- C8),
 Baud Rate = 9600, Parity = N,8,1, and Checksum = Disable.

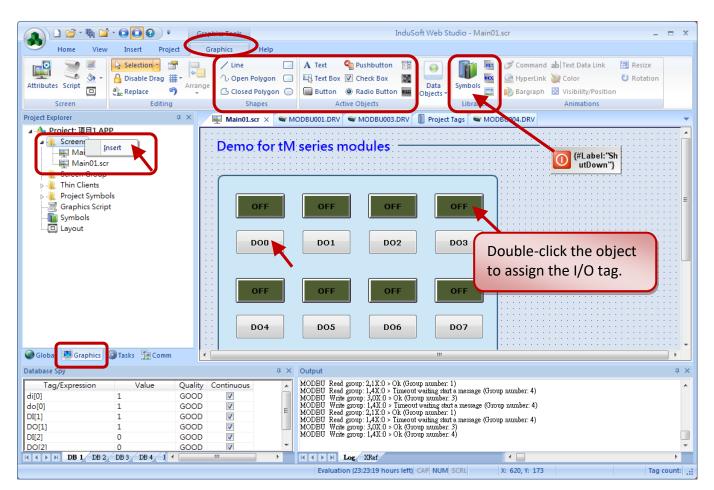
D =	InduSoft Web Studio - N	IODBU001.DRV _ = ×					
Home View Insert Project	Help						
↓ Cut ↓ Run ↓ ↓ ↓ ↓ ↓ ↓ ↓ Paste ↔ Find ↓ ↓ ↓ ↓ Clipboard Local Management ↓ ↓ ↓ ↓ ↓	Ownload O	Import Wizard Import Wizard Convert Resolution Import Wizard Register Controls Import Wizard Tools Tags					
Project Explorer Project Explorer		in01.scr					
▲ ·· A Proiect: 項目1.APP	Description:						
	DI	Increase priority					
MAIN D Insert	Read Trigger: Enable Read when Idle:	Read Completed: Read Status:					
2: AI Settings	1						
3: DO	Write Trigger: Enable Write on Tag Cha	nge: Write Completed: Write Status:					
UPC DA 2.05							
Click to set the way of	Station: Header:						
communications.	2 1X:0						
-	Tag Name	Address					
		The range of Modbus					
	2 D[1]	address for tM DI tags is					
🕥 Global 🖳 Graphics 🖓 Tasks 📜 Comm	•	from 33 to 40.					

Also, click Settings to specify the communication method for InduSoft to communicate with the module. In this case, using COM1.

Solution MODBU:			None		×
Serial Encapsulation:	None	•	Modem TCP/IP		
Serial Port			UDP/IP		
COM:	COM1	•	Stop Bits:	1	•
Baud Rate:	9600	•	Parity:	None	•
Data Bits:	8	•			
Signed/Unsigned:		Pro	tocol(ASCII or	RTU):	
Unsigned 👻		RTU 🗸			
Swap - Write Type:		Block Size / (ERO-xxx):MaxGap:			
No Swap / Write Item 👻		64			
Advanced				OK	Cancel

3. Design an HMI page and assign variables

On the "Graphics" tab, right-click Screen and click Insert to add a screen. Using objects listed on the Graphics menu to design an HMI page, and double-click the object to assign the I/O tag.



Pushbutton – DI

Object Properties	8
Replace Hint:	Pushbutton
Type: Maintained -	State: Normally Closed -
Tag/Expression:	Indicator: D[3]
Disable:	Reset:
E-Sign Config	Security: Key Shift Alt

<u>Button</u> – DO

bject Properties				
Replace.	Hint:		Command	
Type: Built-i	n Language 🛛 👻	Config	Bac	k to button.
*On Down On V	√hile On Up On Right Do	wn On Right Up	On Double Click	On Tou 🔨
*On Down On \ Tao	While On Up On Right Do	wn On Right Up	On Double Click	On Tou 🕚
		wn On Right Up	On Double Click	🕻 On Tou 🔨
Тао	Expression	wn On Right Up	On Double Click	On Tou 🕙

6.4 Linux Modbus Development Tool

ICP DAS LinPAC family is embedded with a flexible and open-source Linux system which can be used to control tM modules via DCON or Modbus protocols. This section illustrates three kinds of Modbus software tools for users to develop various applications.

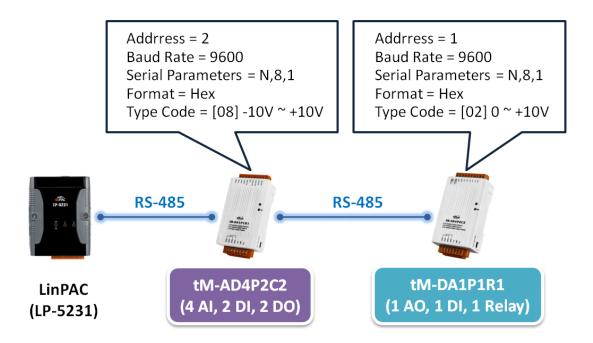
6.4.1 Modbus tool for Linux

The LinPAC series is the Linux-based PAC and support DCON and Modbus protocols. The user can visit the website for more information and download the LinPAC SDK. https://www.icpdas.com/en/product/guide+Software+Development Tools+Modbus Tool#2844

The following table lists the file path of the Modbus tool for AM335x SDK:

OS platform	File path of SDK
Windows	C: \cygwin\LinPAC_AM335x_SDK\examples\ xvboard or C: \cygwin\LinPAC_AM335x_SDK\examples\ modbus
Linux	root@LinuxPC-ICPDAS: /icpdas/linpac_am335x_sdk/i8k/examples/ xvboard or root@LinuxPC-ICPDAS: /icpdas/linpac_am335x_sdk/i8k/examples/ modbus

In this case, using the PAC (LP-5231) and two modules (tM-DA1P1R1 and tM-AD4P2C2) for testing. The parameters for the two modules will be set as follows:



First, set the Net-ID of the slave device, follow the steps below:

```
root@LP-5231:~# getmodbus 2 9600 1 4 484 1 100
1
root@LP-5231:~# setmodbus 2 9600 1 16 484 1 2 100
root@LP-5231:~# getmodbus 2 9600 2 4 484 1 100
2
```

Step 1: Use the command to **query** the Net-ID of the tM-AD4P2C2 module.

Command:

getModbus <comport> <baudrate> <NetID> <command> <address> <count> <timeout(ms)>

getModbus 2 9600 1 4 484 1 100

Step 2: Use the command to **modify** the Net-ID of the tM-AD4P2C2 module.

Command:

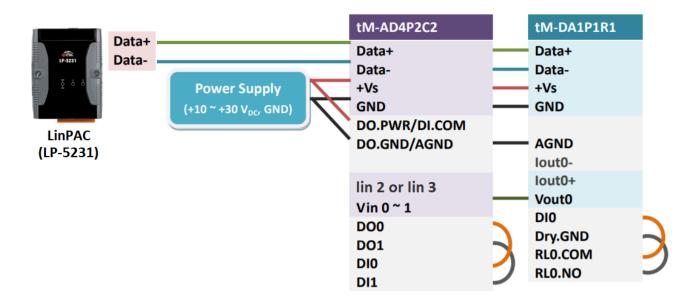
setModbus <comport> <baudrate> <NetID> <command> <address> <count> <value> <timeout(ms)>

```
# setModbus 2 9600 1 16 484 1 2 100 // Set the NetID as 2
```

Note:

- 1. Refer to Appendix C Modbus Register Mapping (Base 1) to set the address of the device.
- 2. The base address for tM series module is **0** (i.e., Base 0). For example, the Modbus registers 40485 is used to read/write module address (i.e., Net-ID). In this case, you should use the address 484 to get or set the Net-ID.

Hardware Wiring for this example:



□ Test the AI/AO channel

Wiring the AO channel of tM-DA1P1R1 to the AI channel of tM-AD4P2C2, and the setModbus.c and getModbus.c programs can be used to test AI/AO. Follow the steps below:

```
root@LP-5231:~# setmodbus 2 9600 1 16 32 1 65535 100
root@LP-5231:~# getmodbus 2 9600 1 3 32 1 100
65535
root@LP-5231:~# getmodbus 2 9600 2 4 0 1 100
32767
```

Step 1: Use the command to **set** the AO value of the tM-DA1P1R1 module.

setModbus 2 9600 1 16 32 1 65535 100 //Output 10 V

Step 2: Use the command to **read** the AO value of the tM-DA1P1R1 module.

```
# getModbus 2 9600 1 3 32 1 100
```

Step 3: Use the command to **read** the AI value of the tM-AD4P2C2 module.

```
# getModbus 2 9600 2 4 0 1 100
```

□ Test the DI/DO channel

Wiring the DO channel of tM-DA1P1R1 to its DI channel, and the setModbus.c and getModbus.c programs can be used to test DI/DO. Follow the steps below:

```
root@LP-5231:~# setmodbus 2 9600 1 15 0 1 1 100
wCount=1 iCount=8 iIndex=0
root@LP-5231:~# getmodbus 2 9600 1 1 0 1 100
1
root@LP-5231:~# getmodbus 2 9600 1 2 32 1 100
1
```

Step 1: Use the command to **set** the DO status of the tM-DA1P1R1 module.

setModbus 2 9600 1 15 0 1 1 100 //Set the status of DO channel to "ON"

Step 2: Use the command to read the DO status of the tM-DA1P1R1 module.

getModbus 2 9600 1 1 0 1 100

Step 3: Use the command to **read** the DI status of the tM-DA1P1R1 module.

```
# getModbus 2 9600 1 2 32 1 100
```

Wiring the DO channel of tM-AD4P2C2 to its DI channel, and the setModbus.c and getModbus.c demo programs can be used to test DI/DO. Follow the steps below:

```
root@LP-5231:~# setmodbus 2 9600 2 15 0 1 1 100
wCount=1 iCount=8 iIndex=0
root@LP-5231:~# getmodbus 2 9600 2 1 0 1 100
1
root@LP-5231:~# getmodbus 2 9600 2 2 32 1 100
1
```

Step 1: Use the command to **set** the DO status of the tM-AD4P2C2 module.

```
# setModbus 2 9600 2 15 0 1 1 100 //Set the status of DO channel to "ON"
```

Step 2: Use the command to **read** the DO status of the tM-AD4P2C2 module.

```
# getModbus 2 9600 2 1 0 1 100
```

Step 3: Use the command to **read** the DI status of the tM-AD4P2C2 module.

getModbus 2 9600 2 2 32 1 100

6.4.2 Using LinPAC and the Python Application

The LinPAC series supports the Python programming language. The user can find the Modbus tool for testing tM series module from the official website of Python. In this example, the LP-5231 module is connected to the tM-DA1P1R1 module and the modbus-tk tool is used for accessing Modbus registers.

For more information about the modbus-tk tool, visit <u>https://github.com/ljean/modbus-tk</u>. Follow these steps to install the software and test the module:

Step 1: Use the command to check if the version of Python is 2.5 or later.

# pythonversion							
root@LP-5231:~#	pythonversion						
Python 2.7.3							
root@LP-5231:~#							

Step 2: Use the command to install pyserial module.

```
# pip install pyserial
```

Step 3: Use the command to download the modbus-tk package.

wget https: //github.com/ljean/modbus-tk/archive/master.zip

Step 4: Use the command to unzip the modbus-tk package.

```
# unzip master.zip
```

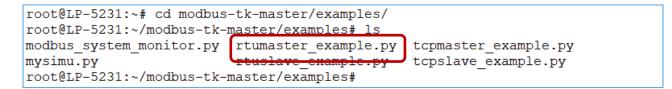
Step 5: Use these commands to install the modbus-tk tool.

- # cd modbus-tk-master
- # python setup.py build
- # python setup.py install

Step 6: Use the command to check if "pyserial" and "modbus-tk" have been installed successfully.

```
# pip list
root@LP-5231:~/modbus-tk-master# pip list
Package Version
-------
distribute 0.6.24dev-r0
modbus-tk 0.5.8
pip 18.1
pyserial 3.4
setuptools 0.6rc11
```

Step 7: Find the "rtumaster_example.py" demo program provided by modbus-tk.



Step 8: Modify the parameters of the rtumaster_example.py demo program.

innert we have the
import modbus_tk
import modbus_tk.defines as cst
from modbus tk import modbus rtu
#PORT = 1
PORT = '/dev/tty02' The communication port
del main():
"""main"""
logger = modbus_tk.utils.create_logger("console")
try:
#Connect to the slave The communication parameters
master = modbus rtu.RtuMaster(
<pre>serial.Serial(port=PORT, baudrate=9600, bytesize=8, parity='N', stopbits=1, xonxoff=0</pre>
master.set_timeout(5.0)
master.set_timeout(5.0) master.set_verbose(True)
master.set_verbose(True)
<pre>master.set_verbose(True) logger.info("connected")</pre>
master.set_verbose(True)
<pre>master.set_verbose(True) logger.info("connected")</pre>
<pre>master.set_verbose(True) logger.info("connected") logger.info(master.execute(1, cst.READ_HOLDING_REGISTERS, 32, 1)) # #</pre>
<pre>master.set_verbose(True) logger.info("connected") logger.info(master.execute(1, cst.READ_HOLDING_REGISTERS, 32, 1)) # # # The Modbus command # # # # # # # # # # # # # # # # # # #</pre>
<pre>master.set_verbose(True) logger.info("connected") logger.info(master.execute(1, cst.READ_HOLDING_REGISTERS, 32, 1)) # #</pre>

<u>Note</u>: The base address for tM series module is '0' (Base 0).

Step 9: Execute the demo program to read the AO value of the tM-DA1P1R1 module. The results will be displayed as illustrated in the figure below.

root@LP-5231:~/modbus-tk-maste	er/examples# python rtumaster example.py
2018-12-10 17:47:25,575 INFO	modbus_rtuinit MainThread RtuMaster /dev/tty02 is opened
2018-12-10 17:47:25,578 INFO	rtumaster_example.main MainThread connected
2018-12-10 17:47:25,580 DEBUG	modbus.execute MainThread -> 1-3-0-32-0-1-133-192
2018-12-10 17:47:25,606 DEBUG	modbus.execute MainThread <- 1-3-2-255-255-185-244
2018-12-10 17:47:25,607 INFO	rtumaster_example.main MainThread (65535,)

6.4.3 Using LinPAC and the Perl Application

The LinPAC series supports the Perl programming language. For testing tM series module, users can find the Modbus tool from the official website of Perl. In this example, the LP-5231 module is connected to the tM-DA1P1R1 module and the Device-Modbus-RTU tool is used for accessing Modbus registers. Follow the steps to install the software and test the module:

Step 1: Download and unzip the Device-Modbus-RTU package (Device-Modbus-RTU-0.022.tar.gz) from the website https://metacpan.org/release/Device-Modbus-RTU

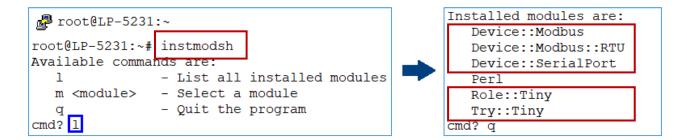
Step 2: Use the command to install the dependent module for Device-Modbus-RTU.

sudo cpan Role: : Tiny Try: : Tiny Device: : SerialPort Device: : Modbus

Step 3: Use the command to install the Device-Modbus-RTU tool.

- # cd Device-Modbus-RTU-0.022
- # perl Makefile.PL
- # make
- # make test
- # make install

Step 4: Use the 'instModsh' command to check if the Perl module has been installed successfully.



Step 5: Find the "write_new_addr.pl" and "simple_client_rtu.pl" demo programs provided by Device-Modbus-RTU.

```
root@LP-5231:~# cd Device-Modbus-RTU-master/examples/
root@LP-5231:~/Device-Modbus-RTU-master/examples# ls
arduino_client.ino server_rtu.pl simple client rtu.pl write new addr.pl
```

Step 6: Modify the parameters of demo programs.

□ Modify the COM port setting in the "write_new_addr.pl" and "simple_client_rtu.pl" scripts.

```
my $client = Device: : Modbus: : RTU: : Client->new(
    port => '/dev/ttyO2', // The number of COM port
    baudrate => 9600, // Bits per second
    parity => 'none', // Parity check
);
```

□ Modify the Modbus command setting in the "write_new_addr.pl" script to set the AO value.

```
my $req = $client->write_single_register( // Modbus function code
unit => 1, // The NetID of Slave device
address => 32, // The channel address
value => 65535 // Set the AO vale
);
```

□ Modify the Modbus command setting in the "simple_client_rtu.pl" script to read the AO value.

The "write_new_addr.pl" demo program:

```
#! /usr/bin/env perl
use Device::Modbus;
use Device::Modbus::RTU::Client;
use Data::Dumper;
use strict;
use warnings;
use v5.10;
my $client = Device::Modbus::RTU::Client->new(
   port => '/dev/tty02',
   baudrate => 9600,
                                                  The COM port setting
   parity => 'none',
my $req = $client->write single register(
   unit \Rightarrow 1,
    address => 32,
                                                  The Modbus command setting
    value => 65535
);
say "->" . Dumper $req;
$client->send request($req);
my $resp = $client->receive response;
say "<-" . Dumper $resp;
```

Note: The base address for tM series module is '0' (Base 0).

Step 7: Execute the demo program to control tM series module.

(1) The results of executing the "write_new_addr.pl" program.

```
root@LP-5231:~/Device-Modbus-RTU-master/examples# perl write new addr.pl
->$VAR1 = bless( {
                 'unit' => 1,
                 'function' => 'Write Single Register',
                 'value' => 65535,
                 'address' => 32,
                 'code' => 6
               }, 'Device::Modbus::Request' );
<-$VAR1 = bless( {
                 'unit' => 1,
                 'crc' => 45193,
                 'message' => bless( {
                                        'function' => 'Write Single Register',
                                        'value' => 65535,
                                        'address' => 32,
                                        'code' => 6
                                      }, 'Device::Modbus::Response' )
               }, 'Device::Modbus::RTU::ADU' );
```

(2) The results of executing the "simple_client_rtu.pl" program.

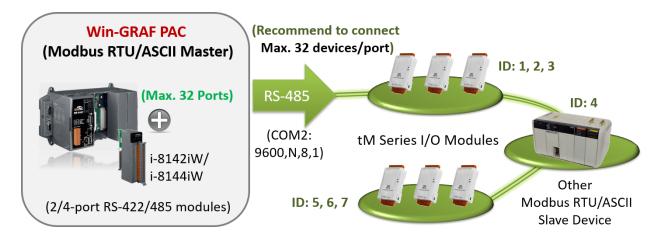
```
root@LP-5231:~/Device-Modbus-RTU-master/examples# perl simple client rtu.pl
->$VAR1 = bless( {
                 'unit' => 1,
                 'function' => 'Read Holding Registers',
                 'quantity' => 1,
                 'address' => 32,
                 'code' => 3
               }, 'Device::Modbus::Request' );
<-$VAR1 = bless( {
                 'unit' => 1,
                 'crc' => 62649,
                 'message' => bless( {
                                        'bytes' => 2,
                                        'function' => 'Read Holding Registers',
                                        'values' => [
                                                       65535
                                                     ],
                                        'code' => 3
                                      }, 'Device::Modbus::Response' )
               }, 'Device::Modbus::RTU::ADU' );
```

6.5 Using Win-GRAF Development Software

Win-GRAF is a powerful SoftLogic development software and PLC-like SoftLogic package that supports IEC 61131-3 Standard Open PLC Languages running on Windows 7/8 (or later). The Win-GRAF Runtime application can run on any Win-GRAF PAC, such as the WinPAC series WP-5238-CE7, WP-8xx8, and WP-8xx8-CE7, or the touch panel ViewPAC series VP-x208-CE7 and VP-x238-CE7, or the advanced CPU XPAC-CE6 series XP-8x38-CE6.

Using the Win-GRAF Workbench with ICP DAS Win-GRAF PACs, the control/monitor systems can easily implement an industrial level of data acquisition and logic control in various industry fields. Website: https://www.icpdas.com/en/product/guide+Software+Development__Tools+Win-GRAF

Win-GRAF PAC supports Modbus RTU/ASCII/TCP/UDP Master/Slave and DCON protocols. The following example will describe how to enable the Win-GRAF PAC as Modbus RTU/ASCII Master to connect to a tM series module.



Steps for using Win-GRAF:

1. Mouse click the "Open Fieldbus Configuration" tool button to open "IO Drivers" window.

🚾 Win-GRAF - Test_01									
<u>File E</u> dit <u>V</u> iew Insert Project	<u>T</u> ools	<u>W</u> indow	<u>H</u> elp						
🖆 🔒 🖹 🎒 🕹 🖡 🛍	X	<u>x</u> 2	1 (° 🛗	333 品。9	16 😨 🚮	🖻 §g 目	🖁 🕍 🚮		
Workspace	10 Dri	ivers							ΖX
⊡ 🗊 Test_01	臣				Name V	/alue	🍸 Name 👘	Тур	e
🗄 🚞 Exception programs	뮮						🔂 🖸	Flobal variable	es 🔼
🖕 🔤 Programs							📃 🛃 R	ETAIN variab	les
🖬 Main	*						📄 N	1ain	
🖕 📴 Watch (for debuggi	0						💾 🔁 p	OnBadIndex	
Soft Scope							📇 p	OnDivZero	~
📰 Initial values							<		>
📲 🚮 Binding Configuration							Name	Value	
🔤 § g Global defines	¢,								
🔤 🚰 Variables							<		
Types	Build								X
	< >	Build	Cross references	Runtime	Call stack 🖉	Breakpoints	Digital sa	ampling trace	Prompt

2. Click the "Insert Configuration" button on the left of the "IO Drivers" window, then click the "MODBUS Master" and "OK" to enable the Modbus Master setting.

10 Drivers						×
展		Name Value	7	Name	Туре	
<mark>倡、</mark> 品				🚮 G	ilobal variables	^
				🔡 R	ETAIN variables	
* =	Add Configuration					×
	Choose a configuration				ОКъ	ſ
	(All)					ζ.
	📥 MODBUS				Cancel	J
é.	MODBUS Max	ster 🖕				
	MODBUS Slav	/e				

 Click the "Insert Master/Port" button on the left side to open the setting window. Next, select the "Serial MODBUS-RTU", set COM Port (e.g., "COM2:9600,N,8,1") and Delay time (recommended value: 10 ms, it can be 0 to 10000), and then click "OK".

IO D	ivers *					×		
E	Ma MODBUS Master	Name Value		Name	Туре			
윫				<u></u>	Global variables	^		
40-	MODBUS Master Port				RETAIN variables			
₽					Main	=		
÷	O MODBUS on Ethernet	ОК		28	pOnBadIndex	_		
	Address:	Canc		28	pOnDivZero	~		
	Port: 502			<	>			
	Mo			ame	Value			
ġį,	MO Protocol: TCP - Open MODBUS UDP - MODBUS RTU							
	UDP - Open MODBUS				Ш	>	-	
		lf w	ish to	set it	as a Modbus AS	CII		
		🗾 🏓 Ma	ster, c	hange	e the setting to			
		"AS	CII:CO	DM2:9	600,N,8,1"			
	<u>Com. port:</u> COM2:9600,N,8,1							
	Delay between requests	Afte	er rec	eiving	the respond, wa	itin	ig for	10
	Delay (ms): 10	ms 🖌	to ser	nd the	next command			
	Try to reconnect after communication er	ror						
	Manage diagnostic info for slaves	(1) (1) (1) (1) (1) (1) (1) (1) (1) (1)	ect "D	isable	d" if you do not	wai	nt	
	Disabled (do not open and manage this				A Port setting.		-	

4. Click the "Insert Slave/Data Block" button on the left side to create a data block.

IO Drivers		
E Ma MODBUS Master	Name	Value
	Mode	RTU
	Address	COM2:9600,N,8,1
	Port	502
	Reconnect after error	
F	Slaves diagnostics	
	Delay between requests (ms)	10
	Disabled	

5. Each data block represents one Modbus Master request, add it according to the application. Configure the following settings in the "MODBUS Master Request" windows and click "OK".

Request				74.014
Description:				ок 💌
	1	1		Cancel
<u>S</u> lave/Unit:	1	_ \ { ª	·)	
MODBUS Reques	t			
<1 > Read Coil			м (b	
<2> Read Inp				
<3> Read Hold		s	*	
Data block				
Base <u>a</u> ddress:	1			
	-	-		
<u>N</u> b items:	8			
Activation				
• Periodic:	2000	ms	15000	
O On call	2000	_		
Ourcai			(on error)	
O a l			This option can no	ot apply
On change				uest.
On change			to the "Read" req	
Misc.	250	m	to the "Read" req	
	250] ms-	to the "Read" req	

a. <u>Slave/Unit</u>:

Enter the Net-ID of the Slave device. (In this case, the Net-ID is "1").

- b. <u>MODBUS Request</u>: In this case, it used to read DI data. Select "<2> Read Input Bits" option.
- <u>Base address</u>:
 Start from "1" by default.

Nb items:

The number of DI signals to read. (In this case, the number is "8").

d. <u>Periodic</u>:

Sending the request periodically. (In this case, to send once every two seconds.) "on error" means the next sending time when an exception occurred (e.g., 15 seconds).

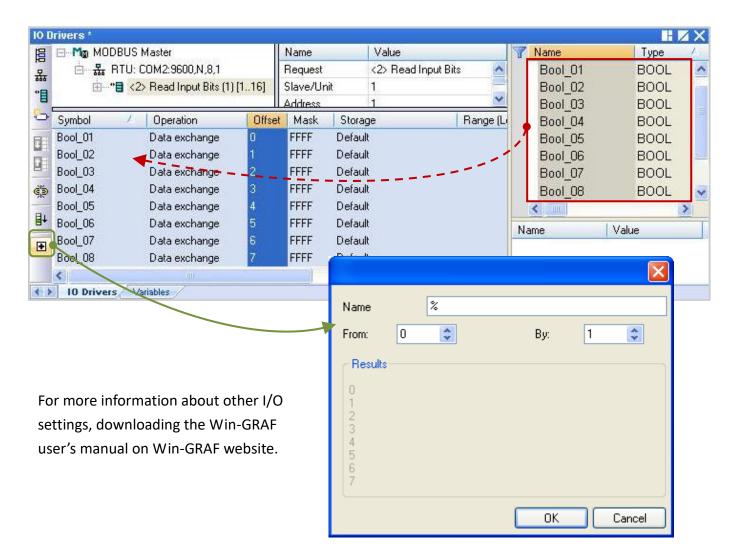
- e. <u>Timeout</u>: Set a timeout value. When time-out occurred, it will show the defined error code. The recommended value for the Modbus RTU/ASCII device is 200 to 1000 ms. E.g., 250 ms)
- 6. In the "I/O Drivers" window, a "Read Input Bits" data block is added. Next, open the "Variables" window and then add variables for accessing I/O data.

Workspace	IO Drivers		
E-01	He Mo MODBUS Master	Name	Value
Exception programs	品 BTU: COM2:9600,N,8,1	Request	<2> Read Input Bits
🛱 🛄 Programs	** 2> Read Input Bits (1) [116]	Slave/Unit	1
📑 🖬 Main		Address	1
🖨 🔤 Watch (for debuggi		Nb Item	16
Soft Scope	Tips:	Activation	Periodic
📰 Initial values	Press "F1" key to view the details	Period (ms)	2000
		Period on error	15000
	about MODBUS Master settings.	Timeout (ms)	250
🚮 Variables 🖕		Number of trials	1
🖳 🖪 Types	<u>]</u> +	Description	
Double click it to open t	he window.	Offset	Mask
	<		>

There are 8 variables (Name: "Boo_01 to "Boo_08"; Type: BOOL) that can be used to read 8 DI data.

Va	riables									íX
T	Name	Туре	Dim.	Attrib.	Syb.	Init value	User	Tag	Description	
	🗉 🚮 Global var	riables								^
	Bool_01	BOOL								
	Bool_02	BOOL								
	Bool_03	BOOL								
	Bool_04	BOOL								
	Bool_05	BOOL								
	Bool_06	BOOL								
	Bool_07	BOOL								
	Bool_08	BOOL								~
	<				-	Ì			>	
4	> 10 Drivers V	ariables								

- 7. Drag variables (i.e., "Boo_01" to "Boo_08") into the "Symbol" area of the data block, and choose all "Offset" fields, and then click the "Iterate Property" button on the left side to set the "Offset" value.
 - Note: The "Offset" value starts at "0" and the Modbus address of the variable is equal to this value plus 1 (Base address). If using a 32-bits (or more than 32-bits) data type (e.g., "DINT"), it requires two Modbus addresses.

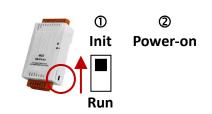


Appendix A Function Description

A.1 INIT Mode

tM series module built-in an EEPROM that can be used to save data such as the module address, the

type code, the baud rate, etc. Sometimes, the module cannot be used because the user forgot the communication parameters of the module and there is no information to look up the settings. tM series features the "INIT" mode to solve this problem.



When the module is powered on in "INIT" mode, the communication

parameters of the module are loaded as follows so that users can quickly found the module.

- 1. Protocol: DCON
- 2. Address: 00
- 3. Baud Rate: 9600 bps
- 4. Checksum: Disable

Under the "INIT" mode, the Command Line function of the DCON Utility Pro allows the user to enter "\$002" command to read the original module settings that are stored in an EEPROM. Refer to the \$AA2 command in Appendix D to understand the response value.

<u>Note</u>: The user only needs to enter "\$002", and the <CR> character will automatically be added and hidden by DCON Utility Pro.

	nd Line Tool		Protocol	DCON					>
OM Port		~			- Devite				
aud Rate	9600	~	Format	N,8,1-No	ne Parity	~		Send	
hecksum	Disabled	~	Address	0	~	~			
Timeout	100 🚿	/ ms	Select ID	tAD5		~			
Command	\$002								
esponse	101080700				_				
	101080700		_						
ET_MODULE			Т	「午 04:00 ::	[\$002]; [!(01080700];	[28 ms]==	=>OK	^
ET_MODULE	CONFIG								
ET_MODULE_ ET_MODULE_	PROTOCOL	!:	命令]	E確					
ET_MODULE_ ET_CHANNEL	PROTOCOL ENABLE_STATUS	01	: Modu	ule Addre	SS				
	_ENABLE_STATUS		: Type	Code: +	-/-10V				
EAD_CH1_AI		07			9200				
EAD_CH2_AI EAD CH3 AI		00	Data	Format	(Checksi	um=Disal	ole,		
					•			Engineering.)	
EAD_CH4_AI	E_DELAY_TIME		Sdillu	ie woue-	-ivuiillai.	י אוומוטצ ו	Unnar-		_

A.2 Dual Watchdog

For different application fields in smart buildings, important medical automation, traffic flow monitoring, and factory production process, systems may be shut down due to natural disasters (e.g., lightning strike) or the harsh environment (e.g., noise or external signals) or any unexpected situations.

If the system is installed at a place where hard to manually reboot, or if the crashed system will lead to a dangerous accident or a huge loss of capacity and cost, the dual watchdog can be used to automatically reboot the equipment and load a Safe Value to ensure the system is working properly and safely.

tM series I/O module features a dual watchdog protection mechanism.

Dual Watchdog = Module Watchdog + Host Watchdog

Module Watchdog:

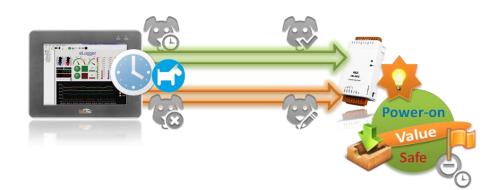


Module Watchdog is a hardware reset circuit that monitors the operating status of the module. Hardware watchdog using a timer that will regularly be reset to 0 and prevent it from timing out when the module is operating normally. If a timeout occurs, the module will automatically reboot and load a predefined Power-on Value.

Host Watchdog:



Host Watchdog is a software function to monitor the communication between the host and the I/O module. Software watchdog using a CPU timer of a host. When the system is working normally, it will regularly send a host-ok signal to the module. If a timeout occurs, the module will immediately load a Safe Value. If the status of time-out is cleared, a predefined Power-on Value will be loaded after a reboot.



Using DCON Utility Pro to test the host watchdog function, refer to <u>Section 3.4.5</u> - Configuration Page - Host WDT. On the "Host WDT" page, the user can set the duration of WDT (Watchdog Timeout), also clear the status of timeout.

A.2.1 Power-on Value

When the module is powered on or warm boot or reboot due to the module Watchdog timeout, a predefined Power-on Value will be loaded into the output channel.

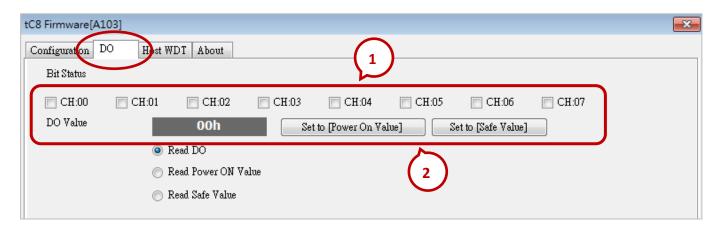
A.2.2 Safe Value

If host watchdog is enabled and the timeout occurs, a Safe Value for the output channel will be loaded. Note that the PWR LED on the tM module will be blinking if the timeout occurs. The time-out status will be stored in the EEPROM and will not be changed even if the module is rebooting. If the status of the timeout is cleared, a predefined Power-on Value will be loaded after a reboot.

Command	Description				
~AA1	Reset the host watchdog status of a module				
~AA5N	Set the current AO value as Safe Value				
~AA5V	Set the current DO value as Power-on Value or Safe Value				
\$AA4N	Set the current AO value as Power-on Value				

Refer to Appendix E.3 Host Watchdog for more details about commands.

Using DCON Utility Pro to set the Power-on Value or the Safe Value, also enable the host watchdog function (reference: Section 3.4.2 to Section 3.4.5). On the "AO/DO" page, the user can set the output value for each channel of the tM module.



Note:

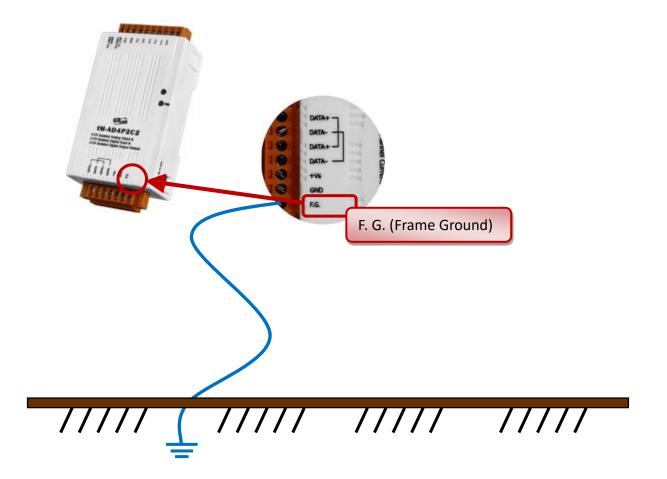
When using the DCON protocol, in case of the host watchdog timeout occurs, the output value will automatically be set to a Safe Value, and cannot be changed. When using the Modbus protocol, the DCON Utility Pro provides the function to change the AO/DO value even if the WDT timeout occurs.

Configuration AI/DO Alarm DI Host WDT About					
Enable WDT	Enable WDT Enable Output When WDT Timeout				
WDT Timeout	25.00	Set Timer			
(0.1 ~ 25.5 sec)					
Reset Watchdo	og Status				

A.3 Frame Ground

Electronic circuits are constantly vulnerable to Electro-Static Discharge (ESD), which becomes worse in a continental climate area. The tM modules feature a new design for the frame ground, which provides a path for bypassing ESD, allowing enhanced static protection (ESD) capability and ensures that the module is more reliable.

Connect the frame ground terminal to a wire/DIN rail and connect the wire/DIN rail to the earth ground will provide better protection for the module.

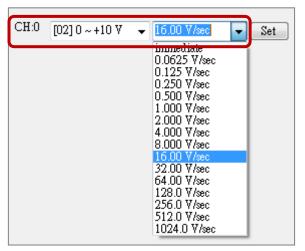


A.4 Slew Rate Control

The feature is used for an analog output module, e.g., tM-DA1P1R1. In general, the output of an analog output module changes instantaneously. That is, when the module receives an output command, its output changes to the specified value immediately.

However, it may require that the output change to the specified value gradually in some applications. The slew rate control is to adjust the output change rate.

<u>Note:</u> The tM-DA1P1R1 modules update the analog output every 10 ms.

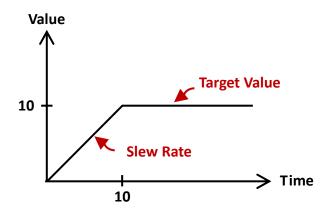


The AO page of DCON Utility Pro

For example, the current output value of the module is 0 V. The user can change the output value to 10 V immediately or specify a slew rate to change the value to 10 V gradually.

S	V/sec	mA/sec
0	Immediate	Immediate
1	0.0625	0.125
2	0.125	0.25
3	0.25	0.5
4	0.5	1.0
5	1.0	2.0
6	2.0	4.0
7	4.0	8.0
8	8.0 16.0	
9	16.0	32.0
Α	32.0	64.0
В	64.0	128.0
С	128.0	256.0
D	256.0	512.0
E	512.0	1024.0





A.5 AO Read-back Value

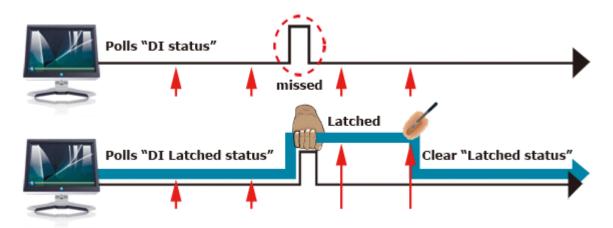
Some modules have built-in electrical circuits to measure the physical output value, also called the read-back value. The tM-DA1P1R1 module doesn't have these circuits so the read-back value equals the AO value. Besides, if users specify a slew rate, the AO value will be changed gradually so both the read-back value and AO value are different.

A.6 Advanced DI Features (Latch and Counter)

The DI channel of a module is not only used to read the digital input status, but also provides several kinds of advanced functions.

DI Latch Function

All DI channels of tM series modules provide the Latch function to keep the high/low events in the internal registers of the module. In general, the host controller polls modules one by one to get all DI status. Because RS-485 is a low-speed field bus, the polling will take time and probably miss a short duration signal. With the DI latch function, the short duration (>=5ms) signal will not be lost anymore.



tM series modules support multiple commands to read the high/low latched status, refer to Appendix E.4 – DI Latched Commands for more information.

Low-Speed Counter

The DI module not only detects the high/low level but also automatically counts the DI signal in the background. The signal under 100Hz can be detected and counted.



Appendix B Type Codes and Ranges for AI/AO Data

All type codes used for the tM series analog input/output module are listed as follows. Each type-code supports two kinds of data formats:

Engineering:

Normally, the AO value gets from the software is 1000 or 10000 times the real value. For example, <u>Type Code 05:</u> The value gets from the software is 15000 which means the real value is 1.5 V; The value gets from the software is -23000 which means the real value is -2.3 V.

<u>Type Code 08:</u> The value gets from the software is 2500 which means the real value is 2.5 V; The value gets from the software is -6500 which means the real value is -6.5 V.

Type Code	Range	Data Format	Min.	Max.
05		Engineering	-25000	+25000
05	-2.5 V to +2.5 V	2's Complement	8000h	7FFFh
06	-20 mA to +20 mA	Engineering	-20000	+20000
06		2's Complement	8000h	7FFFh
07		Engineering	+4000	+20000
07	+4 mA to +20 mA	2's Complement	0000h	FFFFh
08 -	101/1-1101/	Engineering	-10000	+10000
	-10 V to +10 V	2's Complement	8000h	7FFFh

2's Complement:

The value is usually displayed in a hexadecimal format on the document or the software. For example, "1230h" or "ABCDh" (the "h" refers to hexadecimal). The value gets from the software can be converted to the real value by using the following formula, for example,

<u>Type Code 08</u>: The value gets from the software is 19660 (Dec.) or 4CCCh (Hex.) which means the real value is 6.0 V.



Type Code 08: To output 6 V, the value 19660 (Dec.) or 4CCCh (Hex.) must be set in the software.



B.1 Data Ranges for tM-AD2

The tM-AD2 module only supports positive voltage or current. The range of positive voltage, current, and values for each type-code are listed in the table. For example,

<u>Type Code **05**</u>: The data ranges from 0 to +2.5 V. The value ranges from 0 (0000h) to +32767 (<u>7</u>FFFh).

<u>Type Code **1A**</u>: The data ranges from 0 to 20 mA. The value ranges from 0 (0000h) to +65535 (<u>F</u>FFFh).

Type Code	Range	Data Format	Min.	Max.
05		Engineering	0	+25000
05	0 V to +2.5 V	2's Complement	0000h	7FFFh
07	4 mA to 20 mA	Engineering	4000	20000
07	4 mA to 20 mA	2's Complement	0000h	FFFFh
0.8	0)/to :10)/	Engineering	0	+10000
08	0 V to +10 V	2's Complement	0000h	7FFFh
09		Engineering	0	+5000
09	0 V to +5 V	2's Complement	0000h	7FFFh
0A		Engineering	0	+10000
UA	0 V to +1 V	2's Complement	0000h	7FFFh
ОВ	0 V to +500 mV	Engineering	0	+5000
UB	0 V 10 +500 mV	2's Complement	0000h	7FFFh
0.0		Engineering	0	+20000
	0D 0 mA to +20 mA	2's Complement	0000h	7FFFh
1A	0 mA to 20 mA	Engineering	0	20000
IA	0 MA to 20 MA	2's Complement	0000h	FFFFh

Note: If the Analog Format is set to "% FSR", the data of any type code ranges from +000.00 to +100.00.

B.2 Data Ranges for tM-AD5

Type Code	Range	Data Format	Min.	Max.
05	-2.5 V to +2.5 V	Engineering	-25000	+25000
05	-2.5 V t0 +2.5 V	2's Complement	8000h	7FFFh
08	-10 V to +10 V	Engineering	-10000	+10000
08		2's Complement	8000h	7FFFh
00		Engineering	-5000	+5000
09	-5 V to +5 V	2's Complement	8000h	7FFFh
0.0	1.1/+0.11/	Engineering	-10000	+10000
UA	0A -1 V to +1 V	2's Complement	8000h	7FFFh

The tM-AD5 module supports positive and negative voltage and the value ranges from -32768 to 32767.

Note: If the Analog Format is set to "% FSR", the data of any type code ranges from -100.00 to +100.00.

B.3 Data Ranges for tM-AD5C

The tM-AD5C module supports positive and negative current and the value ranges from -32768 to 32767 or from 0 to 65535.

Type Code	Range	Data Format	Min.	Max.
07	4 m 4 to 20 m 4	Engineering	4000	20000
07	4 mA to 20 mA	2's Complement	0000h	FFFFh
0.0	0D -20 mA to +20 mA	Engineering	-20000	+20000
UU		2's Complement	8000h	7FFFh
1.0	1A 0 mA to 20 mA	Engineering	0	20000
1A		2's Complement	0000h	FFFFh

Note: If the Analog Format is set to "% FSR", the data of any type code ranges from -100.00 to +100.00.

B.4 Data Ranges for tM-AD8

The tM-AD5 module only supports positive voltage and the value ranges from 0 to 32767. Take the type code **0B** as an example, the data ranges from **0 to +500 mV** and the value ranges from 0 (0000h) to +32767 (<u>7</u>FFFh).

Type Code	Range	Data Format	-F.S	+F.S		
05	0 V to +2.5 V	Engineering	0	+25000		
05	0 V 10 +2.5 V	2's Complement	0000h	7FFFh		
08		Engineering	0	+10000		
08	0 V to +10 V	0 0 10 +10 0		2's Complement	0000h	7FFFh
00		Engineering	0	+5000		
09	0 V to +5 V	2's Complement	0000h	7FFFh		
0.0	0 V to +1 V	Engineering	0	+10000		
0A	0 0 10 +1 0	2's Complement	0000h	7FFFh		
OB	0 mV to +500 mV	Engineering	0	+5000		
UB	0 1110 to +500 1110	2's Complement	0000h	7FFFh		

Note: If the Analog Format is set to "% FSR", the data of any type code ranges from +000.00 to +100.00.

B.5 Data Ranges for tM-AD8C

The tM-AD5C module only supports positive current (0-20 mA and 4-20 mA). Take the type code **0D** as an example, the data ranges from **0 to +20 mA** and the value ranges from **0** (0000h) to +32767 (<u>7</u>FFFh).

Type Code	Range	Data Format	Min.	Max.
07	4 m 4 to 20 m 4	Engineering	4000	20000
07	4 mA to 20 mA	2's Complement	0000h	FFFFh
0.0		Engineering	0	20000
0D	0 mA to +20 mA	2's Complement	0000h	7FFFh
1.0	0 0 0	Engineering	0	20000
AL	1A 0 mA to 20 mA	2's Complement	0000h	FFFFh

Note: If the Analog Format is set to "% FSR", the data of any type code ranges from +000.00 to +100.00.

B.6 Data Ranges for tM-AD4P2C2

The tM-AD4P2C2 module supports both positive and negative voltage/current, and the value ranges from -32768 to 32767 or from 0 to 65535.

Type Code	Range	Data Format	Min.	Max.
05	-2.5 V to +2.5 V	Engineering	-25000	+25000
05	-2.5 V t0 +2.5 V	2's Complement	8000h	7FFFh
06	20 m 4 to + 20 m 4	Engineering	-20000	+20000
06	-20 mA to +20 mA	2's Complement	8000h	7FFFh
07	4 m 4 to 20 m 4	Engineering	4000	20000
07	4 mA to 20 mA	2's Complement	0000h	FFFFh
0.9	10)//to 110)/	Engineering	-10000	+10000
08	-10 V to +10 V	2's Complement	8000h	7FFFh
00	-5 V to +5 V	Engineering	-5000	+5000
09	-5 V 10 +5 V	2's Complement	8000h	7FFFh
0A	-1 V to +1 V	Engineering	-10000	+10000
UA	-1 V 10 +1 V	2's Complement	8000h	7FFFh
0.0		Engineering	-20000	+20000
0D	-20 mA to +20 mA	2's Complement	8000h	7FFFh
1.0	0 m 4 to 20 m 4	Engineering	0	20000
1A	0 mA to 20 mA	2's Complement	0000h	FFFFh

B.7 Data Ranges for tM-DA1P1R1

The tM-DA1P1R1 module supports positive voltage/current and the following value ranges.

Type Code	Range	Data Format	Min.	Max.
0	0 mA to 20 mA	Engineering	0	+20000
1	4 mA to 20 mA		4000	20000
2	0 V to 10 V		0	+10000
4	0 V to 5 V		0	+5000

Note: If the Analog Format is set to "2's Complement", the data of any type code ranges from 0000h to FFFFh, i.e., 0-65535.

B.8 Data Ranges for tM-TH8

The tM-TH8 module supports the following type codes and value ranges for the thermistor.

Type Code	Thermistor Type (Range)	Data Format	Min.	Max.
		Engineering	-3000	24000
60	PreCon Type III 10K @ 25°C	2's Complement	F000	7FFF
60	(-30 °F to 240 °F)	% FSR	-012.50	+100.00
		Ohms	+000539.4	+173600.0
		Engineering	-5000	15000
61	Fenwell Type U 2K @ 25°C	2's Complement	D556	7FFF
01	(-50 °C to 150 °C)	% FSR	-033.33	+100.00
		Ohms	+000037.2	+134020.0
		Engineering	0	15000
62	Fenwell Type U 2K @ 25°C	2's Complement	0000	7FFF
62	(0 °C to 150 °C)	% FSR	+000.00	+100.00
		Ohms	+000037.2	+006530.0
		Engineering	-8000	10000
C 2	YSI L Mix 100 @ 25°C	2's Complement	999A	7FFF
63	(-80 °C to 100 °C)	% FSR	-080.00	+100.00
		Ohms	+000014.3	+014470.0
		Engineering	-8000	10000
C A	YSI L Mix 300 @ 25°C	2's Complement	999A	7FFF
64	(-80 °C to 100 °C)	% FSR	-080.00	+100.00
		Ohms	+000035.8	+067660.0
		Engineering	-7000	10000
CE	YSI L Mix 1000 @ 25°C	2's Complement	A667	7FFF
65	(-70 °C to 100 °C)	% FSR	-070.00	+100.00
		Ohms	+000106.4	+132600.0
		Engineering	-5000	15000
66	YSI B Mix 2252 @ 25°C	2's Complement	D556	7FFF
66	(-50 °C to 150 °C)	% FSR	-033.33	+100.00
		Ohms	+000041.8	+151000.0
		Engineering	-4000	15000
67	YSI B Mix 3000 @ 25°C	2's Complement	DDDE	7FFF
67	(-40 °C to 150 °C)	% FSR	-026.67	+100.00
		Ohms	+000055.6	+101000.0

Type Code	Thermistor Type (Range)	Data Format	Min.	Max.
		Engineering	-4000	15000
60	YSI B Mix 5000 @ 25°C	2's Complement	DDDE	7FFF
68	(-40 °C to 150 °C)	% FSR	-026.67	+100.00
		Ohms	+000092.7	+168300.0
		Engineering	-3000	15000
69	YSI B Mix 6000 @ 25°C	2's Complement	E667	7FFF
69	(-30 °C to 150 °C)	% FSR	-020.00	+100.00
		Ohms	+000111.5	+106200.0
		Engineering	-3000	15000
6A	YSI B Mix 10000 @ 25°C	2's Complement	E667	7FFF
bА	(-30 °C to 150 °C)	% FSR	-020.00	+100.00
		Ohms	+000185.9	+177000.0
		Engineering	-3000	15000
6B	YSI H Mix 10000 @ 25°C	2's Complement	E667	7FFF
0B	(-30 °C to 150 °C)	% FSR	-020.00	+100.00
		Ohms	+000237.0	+135200.0
		Engineering	-1000	20000
6C	YSI H Mix 30000 @ 25°C	2's Complement	F99A	7FFF
bC	(-10 °C to 200 °C)	% FSR	-005.00	+100.00
		Ohms	+000186.7	+158000.0
		Engineering	-5000	15000
70 to 77	User-defined	2's Complement	D556	7FFF
/010//	(-50 °C to 150 °C)	% FSR	-033.33	+100.00
		Ohms	+000000.0	+000000.0

Note: For "User-defined" types, if the resistance is larger than 204800 ohms, it will be treated as Under Range.

Analog Format	Under Range	Over Range	Protocol
Engineering	-9999.9	+9999.9	DCON
Engineering	-32768	32767	Modbus RTU/ASCII
2's Complement	8000	7FFF	DCON
2's Complement	8000		Modbus RTU/ASCII
Percent (% FSR)	-999.99	+999.99	DCON

B.8.1 Steinhart–Hart Resistance (User-defined Type)

The method to get the Steinhart–Hart coefficients of the thermistor:

- 1. To get Steinhart-Hart coefficients from the thermistor vendor.
- 2. Using the Resistance-Temperature table released by the thermistor vendor. Choose three pairs of resistance values and temperatures such as (R1, T1), (R2, T2), and (R3, T3) to calculate coefficients.

Refer to <u>Section 3.4.1 Configuration Page - User Defined Type</u> to calculate Steinhart–Hart coefficients and convert them to IEEE-754 form by using DCON Utility Pro.

For a typical thermistor, the relationship between resistance and temperature can be expressed by the Steinhart–Hart equation:

$1/T = A + B \ln R_T + C (\ln R_T)^3$

where \mathbf{R}_{T} is the resistance in ohms at temperature **T** in degrees Kelvin (K = °C + 273.15). The values of A, B, and C are called Steinhart Coefficients. The error of the equation is less than +/- 0.01°C in a 100°C span. The tM-TH8 supports user-defined types by specifying the Steinhart coefficients using the <u>@AASxTttC(data)</u> command.

Bit	Description			
31 (sign)	0: positive ; 1 : negative			
20.22 (overage of the second secon	The exponent base is 2. The actual exponent is calculated by			
30-23 (exponent)	subtracting 127 from the stored value.			
22.00 (manticea)	The mantissa is expressed as "1.f" where f is the fractional part			
22-00 (mantissa)	and is stored in this field			

The data sent is a 32-bit hexadecimal value in the IEEE-754 standard format:

For example, a hex. value C3694000h also equals to 1100 0011 0110 1001 0100 0000 0000 (2)

С	3	6	9	4	0	0	0
1100	0011	0110	1001	0100	0000	0000	0000

Bit 31 is **1**, indicating a negative number.

Bit 30 to 23 **10000110**(2) or 134(10), the exponent is 7 (i.e., 134 – 127 = 7).

Bit 22 to 00 The mantissa is **1.11010010100000000000**₍₂₎. After adjusting the mantissa with the exponent (7), the binary is 1<u>1101001</u>.0100000000000000. That is 233.25₍₂₎.

So, the floating-point number of "C3694000h" is -233.25.

Appendix C Modbus Register Mapping

Address (Base 1)	R/W	Description				
00257	R/W	The protocol to be used	0: DCON, 1: Modbus RTU			
00258		The protocol to be used	0: Determined by 00257; 1: Modbus ASCII			
00273	R	The module resets the status	0: Not yet rebooting 1: Ever reboot or first power-on			
40481		The firmware version (Low Word)	•			
40482		The firmware version (High Word)				
40483		The module name (Low Word)				
40484		The module name (High Word)				
40485	R/W	The module address	The valid range: 1 to 247			
40486		Bits 5: 0 Baud Rate: 0x03 to 0x0A Bits 7: 6 00: No Parity, 1 Stop Bit 01: No Parity, 2 Stop Bit 10: Even Parity, 1 Stop Bit 11: Odd Parity, 1 Stop Bit	CodeBaud Rate0x0312000x0424000x0548000x0696000x07192000x08384000x09576000x0A115200			
40488		The Modbus response delay (ms)	Valid range: 0 to 30			
00261		The Host watchdog	0: Disable, 1: Enable			
00270		The Host watchdog timeout status	Set to "1" to clear the status			
40489		The Host watchdog timeout value	Valid range: 0 to 255 (0.1s)			
40492		The Host watchdog timeout count	Set to "0" to clear the value			

For AI/AO Modules

Address (Base 1)	R/W	Description				
30001 - 30008 40001 - 40008	R	The AI values for channels 0 to 7				
40490		Enable/Disable the AI channels (n)	Valid range: 00h to (2 ⁿ -1)h			
00269	R/W	The Modbus data format	0: Hex., 1: Engineering			
00271		The sampling mode (0: Normal, 1: Fast)	Not for tM-TH8/DA1P1R1			
00129 - 00136 10129 - 10136	D	The over/under range status for the AI channels for +4 to +20 mA or 0 to +20	For tM-AD2/AD5C/AD8C			
00225 - 00228 10225 - 10228	- R	mA	For tM-AD4P2C2			
40257 - 40264	– R/W	The Type code for the AI channels	For tM-AD2/TH8/AD4P2C2			
40487		The type code for the Ar channels	For tM-AD5/AD8/AD5C/AD8C			
For tM-AD4P2C2						
40225 - 40226		The upper alarm limit for AI channels 0 a	nd 1			
40233 - 40234	R/W	The lower alarm limit of AI channels 0 an	d 1			
For tM-AD2						
40494	R/W	The lower threshold in +4 to +20 mA	Valid range: 0 to 40 (0.1mA)			
For tM-TH8						
40385 - 40392		The resistance offset for AI channels 0 to 7 at 0.1 ohms	Valid range: 0 to 255			
40449 - 40456		The temperature offset of AI channels 0 to 7 at 0.1 °C	Valid range: -128 to 127			
40769 - 40784	R/W	The Steinhart coefficient A for type codes 70 to 77				
40801 - 40816		The Steinhart coefficient A for type code	s 70 to 77			
40833 - 40848		The Steinhart coefficient A for type code	s 70 to 77			
00267		The temperature format	0: Fahrenheit, 1: Celsius			
00272	W	Set to "1" to load the factory calibration	parameters			
For tM-DA1P1R1						
40033	R/W	The AO value for channel 0				
30065, 40065	R	The Read-back value for the AO channel				
40097		The Safe value for the AO channel				
40193	R/W	The Power-on value for the AO channel				
40289		The Slew rate for the AO channel				
40417		The Type code for the AO channel				

For DI/DO Modules

Address (Base 1)	R/W	Description			
00033 - 00040 10033 - 10040	R	The DI values for channels 0 to 7			
00001 - 00008		The DO values for channels 0 to 7			
00265	R/W	The current DI state	0: Normal (ON/OFF returns 1/0)		
00266		The current DO state	1: Inverse (ON/OFF returns 0/1)		
00065 - 00072 10065 - 10072		The high latched value for the DI cha	nnels		
00097 - 00104 10097 - 10104	- R	The low latched value for the DI char	nels		
00073 - 00080 10073 - 10080		The high latched value for the DO ch	annels		
00105 - 00112 10105 - 10112		The low latched value for the DO cha	nnels		
00264	W	Set to "1" to clear the DIO latched va	lue		
30001 - 30008	_	The Counter values of the DI	For DI modules		
30129 - 30130 40129 - 40130	R		For tM-DA1P1R1/ AD4P2C2		
00193 - 00200	R/W	 The update mode for the edge-triggered Counter 0: The counter updates when there is a falling-edge signal in the DI channel. 1: The counter updates when there is a rising-edge signal in the DI channel. 			
00513 - 00520	W	Set to "1" to Clear the Counter Value			
00260	R/W	 The host watchdog mode 0: If a timeout occurs, the Timeout write the AO/DO value 1: If a timeout occurs, the AO/DO value Timeout status will be automatic 	alue can still be written and the		
00129 - 00136		The Safe value for the DO channels			
00161 - 00168		The Power-on value for the DO chan	nels		
For tM-AD4P2C2					
00289 - 00290		The low alarm status for channels 0 a Set to 1 to clear the low latched alarm	-		
00305 - 00306	R/W	The high alarm status for channels 0 and 1 Set to 1 to clear the high latched alarm			
00321 - 00322		Enable/Disable the alarm on channel	ls 0 and 1		
00337 - 00338		The alarm type for channels 0 and 1, which can be either momentary or latched			

Appendix D Modbus RTU Protocol Commands

Modbus is a communication protocol developed by MODICON Inc. in 1979. It's a standard, truly open, and the most widely used network communication protocol in the industrial automation field. SCADA and HMI software can easily integrate serial devices via Modbus protocol. Visit the website https://www.modbus.org for more information about Modbus.



Data structure of Modbus RTU:

Byte 00	Byte 01	Byte 02-03 Byte 04-05		CRC
Slave address	Function Code	Da	Checksum	
(Net ID, 1 to 247)	Function Code	Start Address	Channels	(2 Bytes)

<u>Note:</u> 1) The content of CRC will be omitted in the following section.

2) Refer to Appendix C Modbus Register Mapping to look up the start address.

The following Modbus RTU functions are supported by tM series module:

Function Code	Bit	Description	Reference Address	
01 (0x01)	1	Read Coil	Read the DO status	0x xxx
02 (0x02)	1	Read Discrete Input	Read the DI status	1xxxx
03 (0x03)	16	Read Multiple Registers	Read the AO value	4 xxxx
04 (0x04)	16	Read Multiple Input Registers	Read AI values	Зхххх
05 (0x05)	1	Force Single Coil	Write a DO status	0x xxx
06 (0x06)	1	Preset Single Register	Write an AO value	4 xxxx
15 (0x0F)	16	Force Multiple Coils	Write multiple DO status	Ox xxx
16 (0x10)	16	Preset Multiple Registers	Write multiple AO values	4 xxxx

Note:

Modbus address **0x**xxx can be used to read/write the status of **DO** channels. Modbus address **1**xxxx can be used to read the status of **DI** channels. Modbus address **3**xxxx can be used to read the value of **AI** channels. Modbus address **4**xxxx can be used to read/write the value of **AO** channels.

Exception Code

Exception Code (Hex.)	Name	Description	
01	Illegal Function	Not supported function	
02	Illegal Data Address	Invalid address	
03	Illegal Data Value	Invalid data	
04	Slave Device Failure	Fails to connect the slave device	
05	Acknowledge	Inform the command is processing	
06	Slave Device Busy	The slave device is busy	

DCON Utility Pro provides the **Command Line** function for testing the Modbus command.



Tool for Terminal (Command				×
COM Port	COM9	-	Protocol	Modbus RTU	
Baud Rate	9600	-	Format	N,8,1-None Parity	- Joand
Checksum	Disable	-	Address	2 tM Series	Send
Timeout	100 -	ms	Select ID	tC8	• •
Command	02 01 00 00 0	80 00			
Response	02 01 01 C3	l1 9D			
GET_MODULE GET_MODULE READ_DO OUTPUT_CHO OUTPUT_CH1 OUTPUT_CH2 OUTPUT_CH3 OUTPUT_CH3 OUTPUT_CH4 OUTPUT_CH5 OUTPUT_CH6 OUTPUT_CH7	_FIRMW _DO_BI1 _DO_BI1 _DO_BI1 _DO_BI1 _DO_BI1 _DO_BI1 _DO_BI1		4 :: [02 01 0 Clear		02 01 01 C3 11 9D];

<u>Command:</u> 02 01 00 00 08

02	01	00	00	00	08
module ID = 02	Read DO	Start address = 0000 (Base 0)		8-channe	l to be read

<u>Response:</u> 02 01 01 C3 11 9D

02	01	01	С3	11	9D
module ID = 02	Read DO	Data byte is 1	1100 0011 means DO 0, 1, 6, 7 = ON	CR	С

D.1 01 (0x01) Read Coil

The function code is used to read the DO or Coil status.

[Request]

Slave Address	Function Code	Start Address (<u>H</u> igh, <u>L</u> ow)	Channels (<u>H</u> igh, <u>L</u> ow)
1 Byte	1 Byte	2 Bytes	2 Bytes
1 to 247	0x01	See Appendix C Modbus Register Mapping	tM module: Max. 8-channel

[Response]

Slave Address	Function Code	Byte Count	Data
1 Byte	1 Byte	1 Byte	<mark>n</mark> Byte
1 to 247	0x01	<mark>n</mark> = (Channels+7)/8	n = 1; Byte 03 = bit 7 - 0 n = 2; Byte 04 = bit 15 - 8 n = m; Byte m+2 = bit (8m-1) - 8(m-1)

[Error Response]

Slave Address	Function Code	Exception Code
1 Byte	1 Byte	1 Byte
1 to 247	0x81 (80+ Function Code)	See " <u>Exception Code</u> "

Examples:

Request (Master): 01 01 00 00 00 02 (Hex.)

Slave Address	Function Code	Start Address		Cha	nnels
Byte 00	Byte 01	Byte 02 (H)	Byte 03 (L)	Byte 04 (H)	Byte 05 (L)
01	01	00	00	00	02
Module ID = 01	Read DO Status	Address = 0	0000 (Base 0)	Read two	o channels

<u>Note:</u> The base address for tM series module is **0** (i.e., base 0) so the start address must be set to the Modbus registers decrease 1, refer to Appendix C.

Response (Slave): 01 01 01 03 (Hex.)

Slave Address	Function Code	Byte Count	Data
Byte 00	Byte 01	Byte 02	Byte 03
01	01	01	03
Module ID = 01	Read DO Status	1 Byte	"0000 0011" means DO0, 1 = ON

D.2 02 (0x02) Read Discrete Inputs

The function code is used to read the DI status.

[Request]

Slave Address	Function Code	Start Address (<u>H</u> igh, <u>L</u> ow)	Channels (<u>H</u> igh, <u>L</u> ow)
1 Byte	1 Byte	2 Bytes	2 Bytes
1 to 247	0x02	See Appendix C Modbus Register Mapping	tM module: Max. 8-channel

[Response]

Slave Address	Function Code	Byte Count	Data
1 Byte	1 Byte	1 Byte	<mark>n</mark> Byte
1 to 247	0x02	n = (Channels+7)/8	n = 1; Byte 03 = bit 7 - 0 n = 2; Byte 04 = bit 15 - 8 n = m; Byte m+2 = bit (8m-1) - 8(m-1)

[Error Response]

Slave Address	Function Code	Exception Code
1 Byte	1 Byte	1 Byte
1 to 247	0x82 (80+ Function Code)	See " <u>Exception Code</u> "

Examples:

Request (Master): 01 02 00 32 00 02 (Hex.)

Slave Address	Function Code	Start Address		Chai	nnels
Byte 00	Byte 01	Byte 02 (H)	Byte 03 (L)	Byte 04 (H)	Byte 05 (L)
01	02	00	32	00	02
Module ID = 01	Read DI Status	Address = 1	0032 (Base 0)	Read two	channels

Note: The base address for tM series module is **0** (i.e., base 0) so the start address must be set to the Modbus registers decrease 1, refer to Appendix C.

Slave Address	Function Code	Byte Count	Data			
Byte 00	Byte 01	Byte 02	Byte 03			
01	02	01	03			
Module ID = 01	Read DI Status	1 Byte	0000 0011 means DO0, 1 = ON			

Response (Slave): 01 02 01 03 (Hex.)

D.3 03 (0x03) Read Multiple Registers

The function code is used to read multiple AO or Holding Registers values. <u>Note:</u> tM-DA1P1R1 supports a single analog output channel.

[Request]

Slave Address	Function Code	Start Address (<u>H</u> igh, <u>L</u> ow)	Channels (<u>H</u> igh, <u>L</u> ow)
1 Byte	1 Byte	2 Bytes	2 Bytes
1 to 247	0x03	See Appendix C Modbus Register Mapping	Word (16-bit Register)

[Response]

Slave Address	Function Code	Byte Count	Data
1 Byte	1 Byte	1 Byte	<mark>n</mark> Byte
1 to 247	0x03	<mark>n</mark> = Channels * 2 Bytes	n = m; Byte 03 = High Byte Byte 04 = Low Byte Byte m+1 = High Byte Byte m+2 = High Byte

[Error Response]

Slave Address	Function Code	Exception Code
1 Byte	1 Byte	1 Byte
1 to 247	0x83 (80+ Function Code)	See " <u>Exception Code</u> "

Examples:

Request (Master): 01 03 01 ED 00 01 (Hex.)

Slave Address	Function Code	Start A	ddress	Cha	nnels
Byte 00	Byte 01	Byte 02 (H)	Byte 03 (L)	Byte 04 (H)	Byte 05 (L)
01	03	01	ED	00	01
Module ID = 01	Read Holding Registers	Address = 4 0	0493 (Base 0)	Read on	e channel

<u>Note:</u> The base address for tM series module is **0** (i.e., base 0) so the start address must be set to the Modbus registers decrease 1, refer to Appendix C.

Response (Slave): 01 03 02 00 1E (Hex.)

Slave Address	Function Code	Byte Count		Data
Byte 00	Byte 01	Byte 02	Byte 03 (H)	Byte 04 (L)
01	03	02	00	1E
Module ID = 01	Read Holding Registers	2 Bytes		is less than 3mA, ed as Under Range

D.4 04 (0x04) Read Multiple Input Registers

The function code is used to read multiple AI, Input Registers, or Counter values.

[Request]

Slave Address	Function Code	Start Address (<u>H</u> igh, <u>L</u> ow)	Channels (<u>H</u> igh, <u>L</u> ow)
1 Byte	1 Byte	2 Bytes	2 Bytes
1 to 247	0x04	See Appendix C Modbus Register Mapping	Word (16-bit Register)

[Response]

Slave Address	Function Code	Byte Count	Data
1 Byte	1 Byte	1 Byte	<mark>n</mark> Byte
1 to 247	0x04	n = Channels * 2 Bytes	n = m; Byte 03 = High Byte Byte 04 = Low Byte Byte m+1 = High Byte Byte m+2 = Low Byte

[Error Response]

Slave Address	Function Code	Exception Code
1 Byte	1 Byte	1 Byte
1 to 247	0x84 (80+ Function Code)	See " <u>Exception Code</u> "

Examples:

Request (Master): 01 04 00 07 00 01 (Hex.)

Slave Address	Function Code	Start A	ddress	Cha	nnels
Byte 00	Byte 01	Byte 02 (H)	Byte 03 (L)	Byte 04 (H)	Byte 05 (L)
01	04	00	07	00	01
Module ID = 01	Read Input Registers	Address = 3 0	0007 (Base 0)	Read on	e channel

Note: The base address for tM series module is **0** (i.e., base 0) so the start address must be set to the Modbus registers decrease 1, refer to Appendix C.

Response (Slave):	01 04 02 00 05 (Hex.)			
Slave Address	Function Code	Byte Count		Data
Byte 00	Byte 01	Byte 02	Byte 03 (H)	Byte 04 (L)
01	04	02	00	05
Module ID = 01	Read Input Registers	2 Bytes	The counter	value of DI7 is 5

R

D.5 05 (0x05) Force Single Coil

The function code is used to write a single DO or Coil status.

[Request]

Slave Address	Function Code	Start Address (<u>H</u> igh, <u>L</u> ow)	Data (<u>H</u> igh, <u>L</u> ow)	
1 Byte	1 Byte	2 Bytes	2 Bytes	
		See Annon	See Appendix C	0x FF 00 means set to ON.
1 to 247 0x05	Modbus Register Mapping	0x 00 00 means set to OFF.		
			(All other values are invalid.)	

[Response]

Slave Address	Function Code	Start Address	Data
1 Byte	1 Byte	2 Bytes	2 Bytes
1 to 247	0x05	The same as the Byte 02-03 of the request	The same as the Byte 04-05 of the request

[Error Response]

Slave Address	Function Code	Exception Code
1 Byte	1 Byte	1 Byte
1 to 247	0x85 (80+ Function Code)	See " <u>Exception Code</u> "

Examples:

Request (Master): 02 05 00 03 FF 00 (Hex.)

Slave Address	Function Code	Start Address		D	ata
Byte 00	Byte 01	Byte 02 (H) Byte 03 (L)		Byte 04 (H)	Byte 05 (L)
02	05	00 03		FF	00
Module ID = 02	Write DO Status	Address = 0 0003 (Base 0)		Set	to ON

<u>Note:</u> The base address for tM series module is **0** (i.e., base 0) so the start address must be set to the Modbus registers decrease 1, refer to Appendix C.

Response (Slave): 02 05 00 03 FF 00 (Hex.)

Slave Address	Function Code	Start Address		D	ata
Byte 00	Byte 01	Byte 02 (H) Byte 03 (L)		Byte 04 (H)	Byte 05 (L)
02	05	00 03		FF	00
Module ID = 02	Write DO Status	0 0003 means DO3		Set	to ON

D.6 06 (0x06) Preset Single Register

The function code is used to write a single AO or Holding Registers value.

[Request]

Slave Address	Function Code	Start Address (<u>H</u> igh, <u>L</u> ow)	Data (<u>H</u> igh, <u>L</u> ow)
1 Byte	1 Byte	2 Bytes	2 Bytes
1 to 247	0x06	See Appendix C Modbus Register Mapping	Set the output value

[Response]

Slave Address	Function Code	Start Address	Data
1 Byte	1 Byte	2 Bytes	2 Bytes
1 to 247	0x06	The same as the Byte 02-03 of the request	The same as the Byte 04-05 of the request

[Error Response]

Slave Address	Function Code	Exception Code	
1 Byte	1 Byte	1 Byte	
1 to 247	0x86 (80+ Function Code)	See " <u>Exception Code</u> "	

Examples:

Request (Master): 01 06 01 E7 00 0A (Hex.)

Slave Address	Function Code	Start Address		D	ata
Byte 00	Byte 01	Byte 02 (H) Byte 03 (L)		Byte 04 (H)	Byte 05 (L)
01	06	01 E7		00	0A
Module ID = 01	Write Holding Register	Address = 4 0)487 (Base 0)	Set	to 10

<u>Note:</u> The base address for tM series module is **0** (i.e., base 0) so the start address must be set to the Modbus registers decrease 1, refer to Appendix C.

Response (Slave): 01 06 01 E7 00 0A (Hex.)

Slave Address	Function Code	Start Address		D	ata	
Byte 00	Byte 01	Byte 02 (H) Byte 03 (L)		Byte 04 (H)	Byte 05 (L)	
01	06	01 E7		00	0A	
Module ID = 01	Write Holding Register	Modbus Respond Delay is set to 10 (ms)				

D.7 15 (0x0F) Force Multiple Coils

The function code is used to write multiple DO or Coil values.

	ave dress	Function Code	Start Address	Channels	Byte Count	Data
1	Byte	1 Byte	2 Bytes	2 Bytes	1 Byte	<mark>n</mark> Byte
1 to	o 247	0x0F	Refer to Appendix C Byte 02 = High Byte Byte 03 = Low Byte	Byte 04 = High Byte Byte 05 = Low Byte	<mark>n</mark> = (Channels+7)/8	A bit corresponds to a channel. 1: ON ; 0: OFF. n = 1; Byte 07 = bit 7 - 0 n = 2; Byte 08 = bit 15 - 8 n = m; Byte m+6 = bit (8m-1) - 8(m-1)

[Request]

[Response]

Slave Address	Function Code	Start Address	Channels
1 Byte	1 Byte	2 Bytes	2 Bytes
1 to 247	0x0F	The same as the Byte 02-03	The same as the Byte 04-05 of
		of the request	the request

[Error Response]

Slave Address	Function Code	Exception Code		
1 Byte	1 Byte	1 Byte		
1 to 247	0x8F (80+ Function Code)	See " <u>Exception Code</u> "		

Examples:

Request (Master): 02 0F 00 03 00 05 01 1F (Hex.)

Slave Address	Function Code	Start Address		Channels		Byte Count	Data
Byte 00	Byte 01	Byte 02	Byte 03	Byte 04	Byte 05	Byte 06	Byte 07
02	OF	00	03	00	05	01	1F
Module $D = 02$ write multiple DQ. Set five bits to QN from the address 00002 (i.e., DQ2)							

Module ID = 02, write multiple DO. Set five bits to ON from the address **0**0003 (i.e., DO3).

<u>Note</u>: The base address for tM series module is **0** (i.e., base 0) so the start address must be set to the Modbus registers decrease 1, refer to Appendix C.

Response (Slave): 02 0F 00 03 00 05 (Hex.)

Slave Address	Function Code	Start Address		de Start Address Channels		nnels
Byte 00	Byte 01	Byte 02 Byte 03		Byte 04	Byte 05	
02	OF	00 03		00	05	
Module ID = 02	Write DO Status	Start from DO3		Five bit	s are set	

D.8 16 (0x10) Set Multiple Register

The function code is used to write multiple AO or Holding Registers values.

Slave Address	Function Code	Start Address	Channels	Byte Count	Data
1 Byte	1 Byte	2 Bytes	2 Bytes	1 Byte	<mark>n</mark> Byte
1 to 247	0x10	Byte 02 = High Byte Byte 03 = Low Byte	Byte 04 = High Byte Byte 05 = Low Byte	(Word number) n = Channels * 2 Bytes	n = m; Byte 07 = High Byte Byte 08 = Low Byte Byte m+5 = High Byte Byte m+6 = Low Byte

[Request]

[Response]

Slave Address	Function Code	Start Address	Channels		
1 Byte	1 Byte	2 Bytes	2 Bytes		
1 to 247	0×10	The same as the Byte 02-03	The same as the Byte 04-05		
1 to 247	0x10	of the request	of the request		

[Error Response]

Slave Address	Function Code	Exception Code		
1 Byte	1 Byte	1 Byte		
1 to 247	0x90 (80+ Function Code)	See " <u>Exception Code</u> "		

Examples:

Request (Master): 03 10 01 C0 00 02 04 00 0A 00 0A (Hex.)

Slave Address	Function Code	Start Address		Channels		Byte Count	Data
Byte 00	Byte 01	Byte 02	Byte 03	Byte 04	Byte 05	Byte 06	Byte 07 - 10
03	10	01	C0	00	02	04	000A 000A
Module ID = 03, write multiple Holding Registers.							

Set the temperature offset of two channels to 10 $(1^{\circ}C)$ from address **4**0448 (base 0).

<u>Note:</u> The base address for tM series module is **0** (i.e., base 0) so the start address must be set to the Modbus registers decrease 1, refer to Appendix C.

Response (Slave): 03 10 01 C0 00 02 (Hex.)

Slave Address	Function Code	Start Address		Channels		
Byte 00	Byte 01	Byte 02 (H)	Byte 03 (L)	Byte 04 (H)	Byte 05 (L)	
03	10	01	С0	00	02	
Module ID = 03	Write Holding Registers	The temperature offset of the AIO and AI1 of tM-TH8 are set to 1 (°C)				

Appendix E DCON Protocol Commands

All communication with tM modules consists of commands generated by the host and responses transmitted by the tM modules. Each module has a unique ID number that is used for addressing purposes and is stored in non-volatile memory.

The ID is 01 by default and can be changed using a user command. All commands to the modules contain the ID address, meaning that only the addressed module will respond. Except for both the ~** and #** commands that will be sent to all modules, in these cases, the modules do not reply to the command.

Request Command Format:

Leading Address	Command	[СНК]	CR
-----------------	---------	-------	----

Response Format:

	Leading	Address	Response	[СНК]	CR	
Leadin Addres	•	Character, like \sim , \$, #, @, %, !, ?, >.				
Addres	55	The slave address of a module that can be set by using DCON Utility Pro refer to Chapter 3.				
Comm	and/Response	Command with parameters or response with data.				
CHKSU	M	2-character checksum. If the Checksum is disabled, no [CHK] here.				2.
	(Reference: Section E.1 %AANNTTCCFF Command)					
CR		The end-of-command character, character return(0Dh)				

Calculate Checksum:

- 1. Sum the ASCII code of all the characters contained in the command in addition to the 'CR' terminator.
- 2. The Checksum is the last 2 digits of the sum value expressed in Hexadecimal format.

For example, if the string of request command is **\$012**(CR)

- 1. Sum = "\$"+ "0"+ "1"+ "2" = 24h + 30h + 31h + 32h = **B7h**
- 2. Checksum = "B7"
- 3. The request command with the checksum is \$012B7(CR)

If the string of response command is **!01200600**(CR)

- 1. Sum = "!" + "0" + "1" + "2" + "0" + "0" + "6" + "0" + "0" = 21h + 30h + 31h + 32h + 30h + 30h + 36h + 30h + 30h =**1AAh**
- 2. Checksum = "AA"
- 3. The response command with the checksum is = !01200600AA(CR)

Note: All characters should be in capital letters.

Users can test the DCON protocol command by using the **Command Line** function of DCON Utility Pro or the Terminal function of VxComm Utility.



DCON Utility Pro

Tool for Terminal	Fool for Terminal Command					
COM Port	COM1		•	Protocol	DCON -	ור
Baud Rate	9600		•	Format	N,8,1-None Parity	
Checksum	Disable		-	Address	3 • M Series • Send	
Timeout	100	•	ms	Select ID	tP4C4	
Command Response	@03 >000F [3	E 30 3	30 30	46]		
GET_MODULE GET_MODULE GET_MODULE SET_MODULE SET_MODULE READ_DO OUTPUT_DO OUTPUT_CHO OUTPUT_CHO OUTPUT_CHO	E_FIRM E_CONI E_PRO E_ONF E_CONF E_CONF E_DO_F L_DO_F 2_DO_F	±++	11:01	:: [@03]; [>000F]; [46 ms]==>OK	4
READ_DI READ_DI_HIG	GH_LAT -		(Clear	Save to \logger_report\	

VxComm Utility

🥩 VxComm Utility [v2 13.07A, Sep.06, 2017]	Configuration Setting	, • 💌
File Server Port Tools Restart Driver	COM Port TCP/IP Port	
Terminal	COM Port : COM1 Data Bits : 8 Baudrate : 9600 Parity Bit : None	•
Where remote series become part of your P	Baudrate : 9600 Parity Bit : None Open_COM Stop Bits : 1	• •
Add Serve Modbus TCP Master		
X Remove S		

- COM1,96	500 - Terminal V2.13.07 [Aug.	24, 2017]	
Send	\$012		CR (0x0D) Send
(Hex)	24 30 31 32		Interval (ms) 0
Receive	d: 20	Ma	ax. display lines 2000
	1 34 30 30 36 30 30 (1 34 30 30 36 30 30 (Clear Receiied
			Display
			● Hex/Text
			© Hex
			© Text
			© None
			- Exit
Status: OK			4

E.1 General Commands

Command	Response	Description
<u>\$AA2</u>	!AATTCCFF	Read the configuration of a module
%AANNTTCCFF	!AA	Set the communication parameters of a module
<u>\$AA8Ci</u>	!AACiRrr	Read the type code for an AI channel
<u>\$AA7CiRrr</u>	!AA	Set the type code for an AI channel
<u>\$AA9N</u>	!AATS	Read the type code and slew rate for an AO channel
<u>\$AA9NTS</u>	!AA	Set the type code and slew rate for an AO channel
<u>\$AA5</u>	!AAS	Read the reset status of a module
<u>\$AAF</u>	!AA(Data)	Read the firmware version of a module
<u>\$AAM</u>	!AA(Data)	Read the name of a module
<u>\$AAP</u>	!AASC	Read the communication protocol of a module
<u>\$AAPN</u>	!AA	Set the communication protocol of a module
<u>\$AAI</u>	!AAS	Read the INIT terminal status of a module
~AARD	!AAVV	Read the response delay time value of a module
~AARDVV	!AA	Set the response delay time value of a module
Only available for t	M-TH8	
<u>~AAI</u>	!AA	The Soft Init command
<u>~AATnn</u>	!AA	Set the Soft Init timeout

E.2 I/O Commands

Command	Response	Description
<u>\$AA6</u>	!AAVV	Read the status of AI/DI/DO channels (The format of response DIO data can be queried in this section)
<u>\$AA5VV</u>	!AA	Set the status of AI channels
<u>#**</u>	No Response	Notify all modules to synchronously access the DIO status
<u>\$AA4</u>	!S(Data)	Read the synchronous sampling DIO status
<u>@AA</u>	>(Data)	Read the status of DI/DO channels
<u>@AADI</u>	!AA00011	Read the status of DI/DO channels (for Multi-function module)
<u>@AA(Data)</u>	>	Set the status of DO channels
@AADODD	!AA	Set the status of DO channels (for Multi-function module)

Command	Response	Description
#AA00(Data)	>	Set the status of DO channels (DO0 to DO7)
#AA0A(Data)	>	Set the status of DO channels (DO0 to DO7)
#AA0B(Data)	>	Set the status of DO channels (DO8 to DO15)
#AA1cDD	>	Set the status of a DO channel (DO0 to DO7)
#AAAcDD	>	Set the status of a DO channel (DO0 to DO7)
#AABcDD	>	Set the status of a DO channel (DO8 to DO15)
<u>#AA</u>	>(Data)	Read data from all AI channels
<u>#AAN</u>	>(Data)	Read data from an Al channel
<u>\$AAA</u>	>(Data)	Read hexadecimal data from all AI channels
<u>\$AA8N</u>	!AA(Data)	Read data from an AO channel
<u>\$AA6N</u>	!AA(Data)	Read the last written data from an AO channel
<u>#AAN(Data)</u>	>	Set the value of an AO channel

E.3 Host Watchdog, Power-on Value, and Safe Value Commands

Command	Response	Description	
~**	No Response	Notify all modules that the Host works normally	
<u>~AA0</u>	!AASS	Read the host watchdog status of a module	
<u>~AA1</u>	!AA	Reset the host watchdog status of a module	
<u>~AA2</u>	!AAETT	Read the host watchdog timeout value of a module	
<u>~AA3EVV</u>	!AA	Set the host watchdog and a timeout value	
<u>~AA4V</u>	!AA(data)	Read the Power-on value or Safe value of the DO module	
<u>~AA5V</u>	!AA	Set the DO value as the Power-on value or Safe value	
<u>~AA4</u>	!AAPPSS	Read the Power-on value or Safe value (Multi-function-DO)	
~AA5PPSS	!AA	Set the Power-on value or Safe value (Multi-function-DO)	
<u>\$AA7N</u>	!AA	Read the Power-on value of the AO module	
<u>\$AA4N</u>	!AA	Set the AO value as the Power-on value	
<u>~AA4N</u>	!AA(data)	Read the Safe Value of the AO module	
<u>~AA5N</u>	!AA	Set the AO value as the Safe Value	

E.4 DI Latched and Counter Commands

Command	Response	Description	
<u>\$AALS</u>	!AA(data)	Read the latched status of DI channels	
<u>\$AAC</u>	!AA	Clear the latched status of DI channels	
<u>#AAN</u>	!AA(data)	Read a counter value of the DI channel	
<u>\$AACN</u>	!AA	Clear a counter value of the DI channel	
For Multi-function	For Multi-function Modules		
@AARECi	!AA(data)	Read a counter value of the DI channel for tM-AD4P2C2	
@AACECi	!AA	Clear a counter value of the DI channel for tM-AD4P2C2	
@AARECN	!AA(data)	Read a counter value of the DI channel for tM-DA1P1R1	
<u>@AACECN</u>	!AA	Clear a counter value of the DI channel for tM-DA1P1R1	

E.5 Alarm Commands

Note: The following commands only available for tM-AD4P2C2.

Command	Response	Description
<u>@AAEATCi</u>	!AA	Enable the alarm function for the AI channel
<u>@AADACi</u>	!AA	Disable the alarm function for the AI channel
@AARACi	!AAS	Read the setting of alarm type
@AARAO	!AAHHLL	Read the status of high/low alarm
@AARHCi	!AA(data)	Read the value of high alarm limit
@AARLCi	!AA(data)	Read the value of low alarm limit
@AAHI(data)Ci	!AA	Set the value of high alarm limit
@AALO(data)Ci	!AA	Set the value of low alarm limit
@AACHCi	!AA	Clear the status of the high latched alarm
@AACLCi	!AA	Clear the status of the low latched alarm

E.6 Module Calibration Commands

Command	Response	Description	
<u>~AAEV</u>	!AA	Enable/Disable the calibration (For analog input modules and tM-AD4P2C2)	
<u>\$AA1</u>	!AA	Perform zero calibration (V, A) (For analog input modules and tM-AD4P2C2)	
<u>\$AA1Ci</u>	!AA	Perform zero calibration (For tM-AD2)	
<u>\$AA0</u>	!AA	Perform span calibration (V, Ω) (For tM-AD5/AD8/TH8/AD4P2C2)	
<u>\$AA0Ci</u>	!AA	Perform span calibration (A) (For tM-AD2/AD5C/AD8C/AD4P2C2)	
Only for the tM-D	A1P1R1		
<u>\$AA0N</u>	!AA	Perform zero calibration	
<u>\$AA1N</u>	!AA	Perform span calibration	
<u>\$AA3NVV</u>	!AA	Adjust AO value for calibration	
Only for the tM-TH	Only for the tM-TH8		
<u>\$AAS1</u>	!AA	Reload the default calibration parameters	

E.7 User-defined Type Commands (for Thermistor)

<u>Note:</u> The following commands only support the tM-TH8.

Command	Response	Description
@AAA3Ci	!AA(data)	Read the temperature offset of a channel
@AAA2CiToo	!AA	Set the temperature offset of a channel
<u>@AAA7Ci</u>	!AA(data)	Read the resistance offset of a channel
@AAA6CiRrr	!AA	Set the resistance offset of a channel
<u>~AAD</u>	!AAT	Read the temperature scale (°C/°F)
<u>~AADT</u>	!AA	Set the temperature scale (°C/°F)
<u>@AAGxTtt</u>	!AA(data)	Read the Steinhart-Hart coefficient of a user-defined type
<u>@AASxTttC(data)</u>	!AA	Set the Steinhart-Hart coefficient of a user-defined type
@AARTTttR(Data)	!AA(data)	Read the scaling temperature based on input resistance of a user-defined type

\$AA2 Read the configuration of a module

Syntax:

\$AA2[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 2 Command to read module settings

Response:

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

Invalid Command: ?AA [CHKSUM] (CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **TT** Read the type code, refer to <u>Appendix B</u>.

Note: The Type Code for each channel of tM-AD2, tM-TH8, tM-DA1P1R1, and tM-AD4P2C2 modules must be set individually so the return value of the TT is **"00"**. For DIO modules, the return value is fixed to **"40**".

- **CC** The Parity and Baud Rate settings.
- **FF** The Checksum, Sample Mode, and Analog Format settings for analog modules. The Checksum and Counter Update Direction settings for digital modules.
 - **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	\$012	Read the configuration of module 01
		Module 01 settings:
Response:	!01 <u>40</u> 0600	Parity= N,8,1-no parity, Baud Rate=9600,
		Checksum=Disable, Sample Mode=Normal, Analog Format=Engineering.

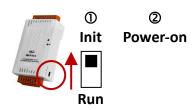
Command:	\$022	Read the configuration of module 02		
	Module 02 settings:			
Response: !02 <u>00</u> 0602		Parity= N,8,1-no parity, Baud Rate=9600,		
		Checksum=Disable, Analog Format= 2's Complement		

Related Commands: %AANNTTCCFF

Related Sections: 3.2 Configuration Page - Configuration

%AANNTTCCFF Set the communication parameters of a module

Note: The settings for parity, baud Rate, and checksum must be configured in **INIT mode**. And then, adjust the switch to Run position and reboot to apply the settings.



Syntax:

%AANNTTCCFF[CHKSUM](CR)

- % Delimiter Character
- **AA** The address of the module (Hex., 00 to FF)
- NN Set an address to the module (Hex., 00 to FF)
- TT The type code setting, refer to <u>Appendix B Type Codes and Ranges</u>
 <u>Note:</u> Using the <u>\$AA7CiRrr</u> command to set the type code for each channel when the model is tM-AD2, tM-TH8, or tM-AD4P2C2. Using the <u>\$AA9NTS</u> command when the model is tM-DA1P1R1. And, set "TT" as "**00**". When using DIO modules, set "TT" as "**40**".
- **CC** Set a parity and baud rate

(*), it stands for items that need to be configured in Init mode.

Bit	7	6	5	4	3	2	1	0	
[CC]	Parity		Baud Rate						
Parity	0		1	L	2 3				
(* Init)	N,8,1-no parity		N,8,2-no parity E,8,1-even p		en parity	n parity O,8,1-odd par			
Baud Rate	3	4	5	6	7	8	9	Α	
(* Init)	1200	2400	4800	9600	19200	38400	57600	115200	

FF <u>For using Analog module</u>: Set the Checksum, Sample Mode, and Analog Format.

Bit	7	6	5	4	3	2	1	0
[FF]	R	R C M			eserved,	F		
<u>C</u> hecksum	0: Disa	0: Disable						
(* Init)	1: Ena	1: Enable						
	0: Normal Mode (14 bits)							
Sample <u>M</u> ode	1: Fas ⁻	1: Fast Mode (12 bits)						
	0: Eng	: Engineering 2 : 2's Complement						
Analog <u>F</u> ormat	1 : Per	Percent (% of FSR)			Ohms			

Note: When using tM-TH8 or tM-DA1P1R1, set the reserved bit 5 as 0.

For using the Digital module: Set the Counter Update Direction and Checksum.

Bit	7	6	5	4	3	2	1	0
[FF]	CU	CU CS (<u>R</u> eserved, 0) C		CD				
Counter Update	0: Update the counter when there is a falling-edge signal							
	1: Update the counter when there is a rising-edge signal							
<u>C</u> heck <u>s</u> um	0: Disable							
(* Init)	1: Enable							
<u>C</u> o <u>d</u> e	tM-P4C4: 1 (read-only)							
	By default, the code is "0" and can be changed for other modules.							

Response:

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

Invalid Command: ?AA [CHKSUM] (CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command If the **Baud Rate** or **Checksum** setting is not configured in Init mode, the module will return an invalid command.
- **AA** The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: %0102 <u>40</u> 0600	Set the address of the digital module '01' to '02', the parity to 'N,8,1-no parity', the baud rate to '9600', the checksum to 'Disable', the sample mode to 'Normal' and the analog format to 'Engineering'				
Response: !02	Module 02 respond the command is valid				
Command: %010100 <u>0A</u> 00	Set the baud rate of module 01 as 115200 bps				
Response: ?01	The command is invalid because it is not set in Init mode				
Г					
Command: %010100 <u>0A</u> 00	In Init mode, set the baud rate of module 01 as 115200 bps				
Response: !01	Module 01 respond the command is valid				
Related Commands:	<u>\$AA2, \$AA7CiRrr, \$AA9NTS</u>				
Related Sections:	3.2 Configuration Page - Configuration				

\$AA8Ci Read the type code for an AI channel

Note: The command only available for tM-AD2, tM-TH8, and tM-AD4P2C2 modules.

Syntax:

\$AA8Ci[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 8 Command to read the type code of an analog input channel
- **Ci** i, the channel to be read (0 to 7)

Response:

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

Invalid Command: ?AA [CHKSUM] (CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- **AA** The address of the module (Hex., 00 to FF)
- **Ci** i, the channel to be read (0 to 7)
- Rrr rr, the type code of the specified channel
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$018C0	Read the type code for channel 0 of module 01, e.g., tM-AD2				
Response: !01C0R08	The type code of channel 0 is '08', i.e., 0 to 10 V				
Command: \$028C3	Read the type code for channel 3 of module 02, e.g., tM-AD4P2C2				
Response: !02C3R0D	The type code of channel 3 is '0D', i.e., -20 to +20 mA				
Command: 038C5	Read the type code for channel 5 of module 03, e.g.,tM-TH8				
Response: !03C5R6C	The type code of channel 5 is '6C', i.e., YSI H Mix 30000 @ 25°C -10°C to 200°C				

Related Commands:

\$AA7CiRrr, %AANNTTCCFF

Related Sections:

- 3.3.1 Configuration Page (tM-AD2), 3.3.3 Configuration Page (tM-TH8),
- 3.4.3 Configuration Page (tM-AD4P2C2)

\$AA7CiRrr Set the type code for an AI channel

Note: The command only available for tM-AD2, tM-TH8, and tM-AD4P2C2 modules. All channels of the modules can be configured individually.

Syntax:

\$AA7CiRrr[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 7 Command to set the type code of an analog input
- Ci i, the channel to be set (0 to 7)
- Rrr rr, the type code of the channel

Response:

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

Invalid Command: ?AA [CHKSUM] (CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$017C0R08	Set the type code of channel 0 of module 01 (e.g., tM-AD2) as '08' (i.e., 0 to 10 V)
Response 01	The command is valid
Command: \$027C3R0D	Set the type code of channel 3 of module 02 (e.g., tM-AD4P2C2) as '0D' (i.e., -20 to +20 mA)
Response: !02	The command is valid
Command, 6027CEDCC	Set the type code of channel 5 of module 03 (e.g., tM-TH8) as
Command: \$037C5R6C	'6C' (i.e., YSI H Mix 30000 @ 25°C -10°C to 200°C)
Response: !03	The command is valid
Command: \$037C1R30	Set the type code of channel 1 of module 03 (e.g., tM-TH8) as '30'
Response: ?03	The command is invalid

Related Commands:\$AA8Ci, %AANNTTCCFFRelated Sections:3.3.1 Configuration Page (tM-AD2), 3.3.3 Configuration Page (tM-TH8),
3.4.3 Configuration Page (tM-AD4P2C2)

\$AA9N Read the type code and slew rate for an AO channel

Note: The command is available for tM-DA1P1R1.

Syntax:

\$AA9N[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 9 Command to read the settings of an analog output channel
- **N** The channel to be read. It is '0' for the tM-DA1P1R1 module.

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- TThe type code of an analog output channelRefer to Appendix B.7 Data Ranges for tM-DA1P1R1
- S The slew rate of an analog output channel Refer to <u>Appendix A.4 Slew Rate Control</u>
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$0190	Read the settings for analog output channel 0 of module 01
Response: !0110	The type code is '1' (i.e., 4 to 20 mA); The slew rate is '0' (i.e., Immediate)

Related Commands:

<u>\$AA9NTS</u>

Related Sections:

3.4.2 Configuration Page - AO,Appendix A.4 Slew Rate Control,Appendix B.7 Data Ranges for tM-DA1P1R1

\$AA9NTS Set the type code and slew rate for an AO channel

Note: The command is available for tM-DA1P1R1.

Syntax:

\$AA9NTS[CHKSUM](CR)

- Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 9 Command to configure an analog output channel
- N The channel to be set. It is '0' for tM-DA1P1R1
- T The type code of an analog output channel
 Refer to <u>Appendix B.7 Data Ranges for tM-DA1P1R1</u>
- S The slew rate of an analog output channel Refer to Appendix A.4 Slew Rate Control

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$019021	Set the type code of analog output channel 0 as '2' (i.e., 0 to 10 V) and set the slew rate as '1' (i.e., 0.625 V/sec)		
Response: !01	The command is valid		

Related Commands:

<u>\$AA9N</u>

Related Sections:

3.4.2 Configuration Page - AO,Appendix A.4 Slew Rate Control,Appendix B.7 Data Ranges for tM-DA1P1R1

\$AA5 Read the reset status of a module

Syntax:

\$AA5[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 5 Command to read the reset status of a module

Response:

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

Invalid Command: ?AA [CHKSUM] (CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **S** Used to check if the module is ever rebooted

The first time to execute the command after the module is powered on, it will return a value '1'. Afterward, executing the command will get a value '0'. If you get a value "1" again which means the module has been rebooted.

- 1: The module has been rebooted (or just turned on).
- 0: The module is never rebooted after it is powered on.
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$015	Read the reset status of module 01			
Response: !011	Module 01 has been rebooted (or just turned on)			
Command: \$015	Read the reset status of module 01			
Response: !010	Module 01 is never rebooted after it is powered on			

\$AAF Read the firmware version of a module

Syntax:

\$AAF[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **F** Command to read the firmware version

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) A string to represent the firmware version

<u>Note</u>: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$01F	Read the firmware version of module 01
Response: !01A2.0	The version is "A2.0"

\$AAM Read the name of a module

Syntax:

\$AAM[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- M Command to read the module name

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Name) A string to represent the module name
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$03M	Read the name of module 03		
Response:!03tTH8The module name is "tM-TH8"			
Command: \$02M	Read the name of module 02		
Response: !02tAD4P2C2	The module name is "tM-AD4P2C2"		

Note: It is recommended NOT to change the name of tM series module.

\$AAP Read the communication protocol of a module

Syntax:

\$AAP[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- P Command to read the communication protocol

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **S** The protocol supported by the module
 - 0: Only DCON protocol is supported
 - 1: Both the DCON and Modbus RTU protocols are supported
 - 3: All of the DCON and Modbus RTU/ASCII protocols are supported
- **C** The Protocol setting of the module that is saved in EEPROM
 - 0: the protocol set in EEPROM is DCON
 - 1: the protocol set in EEPROM is Modbus RTU
 - 3: the protocol set in the EEPROM is Modbus ASCII
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command: \$01P	Read the Protocol setting of module 01				
Response: !0130	30 Module 01 supports both the DCON and Modbus RTU/ASCII protocol ar the Protocol setting is DCON				
Command: \$00P	Read the Protocol setting of the module in Init mode, that is the address is				
	00 and the protocol is DCON by default The address of the module is 02, it supports both the DCON and Modbus RTU/ASCII protocols and the protocol set in the EEPROM is Modbus ASCII				
Response: !0233	Note that the module settings are saved in EEPROM, , the user needs to set the switch to Run and reboot to apply the settings.				

Related Commands: **\$AAPN**

Related Sections: 3.2 Configuration Page - Configuration

\$AAPN Set the communication protocol of a module

<u>Note:</u> The command MUST be used in Init mode.

Syntax:

\$AAPN[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- P Command to set the communication protocol
- N 0: DCON protocol
 - 1: Modbus RTU protocol
 - 3: Modbus ASCII protocol

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.
- **Note:** After setting Protocol in Init mode (i.e., AA = **<u>00</u>**, Protocol = DCON), the settings will be saved in EEPROM, and it will become effective after rebooting in Run mode.

Command: \$00P1	Set the communication protocol of module to Modbus RTU in Init mode				
Response: ?01	The command is invalid because it is not set in Init mode				
Command: \$00P1	Set the communication protocol of module to Modbus RTU in Init mode				
Response: !02	The command is invalid for Module 02				
Command: \$00P	Read the Protocol setting of the module in Init mode, that is the address is 00 and the protocol is DCON by default				
Response: !0231	The address of the module is 02, it supports both the DCON and Modbus RTU/ASCII protocols and the protocol set in the EEPROM is Modbus RTU				
	Note that the module settings are saved in an EEPROM, the user needs to set the switch to Run and reboot to apply the settings.				

Examples:

Related Commands: <u>\$AAP</u>

Related Sections: 3.2 Configuration Page - Configuration



② Power-on

\$AAI Read the INIT terminal status of a module

Syntax:

\$AAI[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- I Command to read the module INIT status

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **S** The Init status of the module

0: The current position of the DIP switch is "Init"

- 1: The current position of the DIP switch is "Run"
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$01I	Read the Init status of module 01			
Response: !010 The DIP switch on the right side of the module is in the "Init" po				

~AARD Read the response delay time value of a module

Syntax:

~AARD[CHKSUM](CR)

- Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **RD** Command to read the response delay time value

Response:

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- VV Two hexadecimal digits to represent the response delay time value in milli-second, for example,01 denotes 1ms and 1E denotes 30ms. The max allowable value is 30 (1Eh)
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~01RD	Read the response delay time value of module 01
Response: !0102	The module returns 02, which means the response delay time value is 2ms

Related Commands:

<u>~AARDVV</u>

Related Sections:

3.2.2 to 3.2.4 Configure AI Items

~AARDVV Set the response delay time value of a module

Syntax:

~AARDVV[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **RD** Command to set the response delay time value
- **VV** Using two hexadecimal digits to represent the response delay time value in milli-second, for example, 01 denotes 1ms and 1E denotes 30ms. The max allowable value is 30 (1Eh)

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~01RD06	Set the response delay time value of module 01 to 6 ms				
Response: !01 A valid response returned by the module					
Command: ~01RD Read the response delay time value of module 01					
Response: 10106 The module returns 06, which means the response delay time value					

Related Commands:

~AARD

Related Sections:

3.2.2 to 3.2.4 Configure AI Items

~AAI The Soft INIT command

Description:

Normally, the Parity, Baud Rate, and Checksum setting of the module must be set in INIT mode. Alternatively, users can also configure them by using the software INIT command.



② Power-on

Note: The command is available for tM-TH8.

Make sure that the timeout has been set (<u>~AATnn</u>) before using this command.

Syntax:

~AAI[CHKSUM](CR)

- Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- I Command to set Soft Init

Response:

- Valid Command: **!AA**[CHKSUM](CR)
- Invalid Command: **?AA**[CHKSUM](CR)
 - ! Delimiter character for a valid command
 - ? Delimiter character for an invalid command
 - AA The address of the module (Hex., 00 to FF)
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~01T10	Set the Soft INIT timeout of module 01 to 16 seconds				
Response: !01	A valid response returned by the module				
Command: ~01I	Set the status of module 01 to Soft INIT				
Response: !01	A valid response returned by the module				
Command: %010100 <u>07</u> 00	Set Module 01 as follows: Parity = N,8,1-no parity, Baud Rate = 19200, Checksum = 0, and Analog Format = Engineering				
Response: !01	A valid response returned by the module				

Related Commands: <u>~AATnn</u>, <u>%AANNTTCCFF</u>

Related Sections: 2.5 Operating Switch and Parameters

~AATnn Set the Soft Init timeout

Note: The command is available for tM-TH8.

Syntax:

~AATnn[CHKSUM](CR)

- Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- T Command to set the soft INIT timeout
- **nn** Using two hexadecimal digits to represent the timeout value in seconds. The maximum timeout value is 60 seconds.
 - Note: After sending the Soft INIT command (i.e., ~ AAI), the %AANNTTCCFF command must be set before the Soft INIT timeout (i.e., the value =0), or the change of Parity, Baud Rate, and Checksum settings will be invalid. After rebooting, the Soft INIT timeout is 0.

Response:

Valid Command: !AA[CHKSUM](CR)

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command: ~01T10	Set the Soft INIT timeout of module 01 to 16 seconds				
Response: !01	A valid response returned by the module				
Command: ~01I	Set the status of module 01 to Soft INIT				
Response: !01	A valid response returned by the module				
	Set Module 01 as follows: Parity = N,8,1-no parity, Baud Rate = 19200,				
Command:	Checksum = 0, and Analog Format = Engineering				
%010100 <u>07</u> 00	Note: After completing the settings, it is recommended to set the Init				
	timeout to 0 to disable the Soft INIT				
Response: !01	A valid response returned by the module				

Examples:

Related Commands: <u>~AAI</u>, <u>%AANNTTCCFF</u>

Related Sections: 2.5 Operating Switch and Parameters

\$AA6 Read the status of AI/DI/DO channels

Description:

- The command is used to read the enabled/disabled status of each channel. It is available for analog input and tM-AD4P2C2 modules.
- The command is used to read the ON/OFF status of each channel. It is available for digital input/ output and tM-DA1P1R1 modules.

Syntax:

\$AA6[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 6 Command to read the channel status

Response:

Valid Command: [For AI modules] **!AAVV**[CHKSUM](CR) [For DIO module] **!(Data)**[CHKSUM](CR)

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- VV [For AI modules] Using two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 1, which means the channel is enabled, and 0 denotes that the channel is disabled.
- (Data) [For DIO modules] Using four-digit hexadecimal value followed by 00.

DIO Data Format Table:

- 1) The <u>\$AA4</u>, \$AA6, and <u>\$AALS</u> command: [the First Data] [The Second Data]00
- 2) The <u>@AA</u> command:

[the First Data] [The Second Data]

Module Name	DIO	The First Data		The Second Data		
tM-DA1P1R1	1 DO ; 1 DI	DO0	00 to 01	DI0	00 to 01	
tM-P3R3	3 DO ; 3 DI			00 to 07		00 to 07
tM-P3POR3		DO0 to DO2	00 to 07	DI0 to DI2	00 10 07	
tM-P4C4	4 DO ; 4 DI	DO0 to DO3	00 to 0F	DI0 to DI3 00 to 0F		
tM-P4A4				DIU to DI3	00 to UF	
tM-R5	5 DO	DO0 to DO4	00 to 1F			
tM-C8	8 DO	DO0 to DO7	00 to FF	00		
tM-P8	8 DI	DIO to DI7	00 to FF			

Note: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

[For AI and tM-AD4P2C2 modules]

Command: \$016	Read the enabled/disabled status of the AI module 01 (e.g., tM-AD8)							
	AIO to AI5 are enabled; AI6 and AI7 are disabled							
Response: !013F	AI7	AI6	AI5	AI4	AI3	AI2	Al1	AIO
Response. :01 <u>37</u>	0 0 1 1				1	1	1	1
	3						F	

Command: \$026	Read the enabled/disabled status of the AI module 02 (e.g., tM-AD4P2C2)						
	AI1 and AI3 are enabled ; AI0 and AI2 are disabled						
Response: 1020A	AI3	Al2	Al1	AI0			
	1	0	1	0			
		0	A				

[For DIO and tM-DA1P1R1 modules]

Command: \$016	Read the ON/OFF status of the DIO module 01 (e.g., tM-DA1P1R1)					
Response: ! <u>0101</u> 00	DO0 and DI0 are ON.DO0DI011					

Command: \$026	Read the ON/OFF status of the DIO module 02 (e.g., tM-C8)							
	DO0, DO2, DO4, and DO6 are ON; DO1, DO3, DO5, and DO7 are OFF							
Response: !D50000	D07	DO6	DO5	DO4	DO3	DO2	DO1	DO0
	1	1	0	1	0	1	0	1
	D				5			

Related Commands: <u>@AA</u>, <u>\$AA5VV</u>

Related Sections:

- 3.3 Configuration Page AI
- 3.4.3 Configuration Page AI/DO Alarm
- 3.4.4 Configuration Page DO
- 3.4.6 Configuration Page DI

\$AA5VV Set the status of AI channels

Description:

Specify the channel (s) to be enabled

<u>Note:</u> The command is available for AI and tM-AD4P2C2 modules. It is recommended that only the channels to be used are enabled.

Syntax:

\$AA5VV[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 5 Command to set the channels to enabled/disabled
- **VV** Using two-digit hexadecimal value, where bit 0 corresponds to channel 0, bit 1 corresponds to channel 1, etc. When the bit is 1, which means the channel is enabled, and 0 denotes that the channel is disabled.

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- **AA** The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

	Enable AI1, AI3 and AI4 channels of module 01 (e.g., tM-AD5)						
Command: \$015 <u>1A</u>	AI4 AI3 AI2 AI1 AI0						
	1 1 0 1 0						
Response: !01	The command is valid						
Command: \$016	Read the enabled/disabled status of module 01						
Response: !011A	AI1, AI3, and AI4 are enabled						

Related Commands: <u>\$AA6</u>

Related Sections: 3.3 Configuration Page - AI, 3.4.3 Configuration Page – AI/DO Alarm

#** Notify all modules to synchronously access the DIO status

Description:

All DIO modules with the same communication settings on one RS-485 bus will receive this command and record the current DIO status.

Note: 1. The command is available for DIO and Multi-function (tM-DA1P1R1, tM-AD4P2C2) modules.
 2. Test it with the <u>\$AA4</u> command by using VxComm Utility.

Syntax:

#**[CHKSUM](CR)

- # Delimiter Character
- ** Synchronized sampling command

Response:

No response.

Examples:

Command: #**	Notify all modules to synchronously read and store the DIO data				
Response:	No response.				

Command: \$014	Read the synchronized data of module 01 (e.g., tM-P8).							
Response: !1 <u>C300</u> 00	DI7	DI6	DI5	DI4	DI3	DI2	DI1	DIO
	1	1	0	0	0	0	1	1

Command: \$024	Read the synchronized data of module 01 (e.g., tM-AD4P2C2)						
Response: !1 <u>0200</u> 00	DO1 1	DO0 0	DI1 0	D10 0			
Command: \$024	Read the synchronized data of module 01 (e.g., tM-AD4P2C2)						
Response: !0 <u>0200</u> 00	0, means that it is not the first time to read the synchronized data. Users can execute the #** command to get the latest synchronized data.						

Related Commands: <u>\$AA4</u>

Related Sections: <u>\$AA6 Command – DIO Data Format Table</u>

\$AA4 Read the synchronous sampling DIO status

Note: 1. The command supports DIO and Multi-function (i.e., tM-DA1P1R1, tM-AD4P2C2) modules.

2. Test it with the $\frac{\#^{**}}{2}$ command by using VxComm Utility.

Syntax:

\$AA4[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 4 Command to read the synchronized data

Response:

Valid Com	mand:	!S(Data) [CHKSUM](CR)
Invalid Co	mmand:	?AA [CHKSUM](CR)
!	Delimiter	character for a valid command

- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **S** The status of the synchronized data
 - 1: The first time to read data after sampling
 - 0: Not the first time to read data, that is not the latest data.

Also, execute the $\frac{\#^{**}}{\#^{**}}$ command to synchronously record the current data.

Examples:

Command: #**	Notify all modules to synchronously read and store the DIO data
Response:	No response.

Command: \$014	Read the synchronized data of module 01 (e.g., tM-P8)						
Response: !1 <u>C300</u> 00	DI7 DI6 DI5 DI4 DI3 DI2 DI1 DI0						DIO
	1	1	0	0	0	0	1

	► СОМ1,9	600 - Terminal V2.13.07 [Aug.24, 2017]
	Send	\$014	CR (0x0D) Send
	(Hex)	24 30 31 34	Interval (ms) 0
	Receive	ed: 18	Max. display lines 2000
	21 31 4 30 0D	43 33 30 30 30 30	0D 21 30 43 33 30 30 30 11C30000.!0C3000 Clear Receided
xComm	n Utility	- Terminal	Display Hex/Text

Related Commands: #**

Related Sections: <u>\$AA6 Command – DIO Data Format Table</u>

@AA Read the status of DI/DO channels

Description:

Read the ON/OFF status of each channel.

Note: The command supports DIO and Multi-function (i.e., tM-DA1P1R1, tM-AD4P2C2) modules.

Syntax:

@AA[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)

Response:

Valid Command: >(Data)[CHKSUM](CR)

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

- > Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) Using four-digit hexadecimal to represent the DIO status
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command: @01	Read the DIO status of module 01 (e.g., tM-DA1P1R1)							
Response: >0101	DO0 and	DIO are C	DN					
	DO0	DIO						
	1	1						

Command: @02	Read the DIO status of module 02 (e.g., tM-AD4P2C2)									
Response: >0203	DO1, DI0 DO1	, and DI1 DO0	are ON; DI1	DO0 is C DI0)FF					
	1	0	1	1						

Command: @03	Read the DIO status of module 03 (e.g., tM-P3R3)								
	DO1, DI0, and DI2 are ON ; DO0 and DO2 are OFF								
Response: >0207	DO2	D01	DO0	DI2	DI1	DIO			
	0	1	0	1	1	1			
		02		07					

Related Commands: <u>\$AA6</u>

Related Sections: 3.4.4 Configuration Page - DO, 3.4.6 Configuration Page - DI

@AADI Read the status of DI/DO channels (for Multi-function module)

Description:

Read the DI and DO status of a module

Note: The command is available for tM-DA1P1R1 and tM-AD4P2C2 modules.

Syntax:

@AADI[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **DI** Command to read the digital input and digital output status

Response:

Valid Command: **!AA0OOII**[CHKSUM](CR)

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **OO** Using two-digit hexadecimal value to represent the DO status
- Using two-digit hexadecimal value to represent the DI status
 Bit 0 corresponds to DI0 (or DO0), Bit 1 corresponds to DI1 (or DO1), etc.
 When the bit is 1, indicates the DI/DO status is ON;
 When the bit is 0, indicates the DI/DO status is OFF.
- <u>Note</u>: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command: @01DI	Read the DIO status of module 01 (e.g., tM-DA1P1R1)						
Response: !010 <u>0101</u>	DO0 and DI0 are ON1						

Command: @02DI	Read the DIO status of module 02 (e.g., tM-AD4P2C2)								
Response: !020 <u>0203</u>	DO1, DI0	, and DI1	are ON;	DOO is OF	=F				
	DO1	DO0	DI1	DIO					
	1	0	1	1					

Related Commands: @AADODD

@AA(Data) Set the status of DO channels

Note: The command is available for DO and Multi-function (i.e., tM-DA1P1R1, tM-AD4P2C2) modules.

Syntax:

@AA(Data)[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- (Data) Data to be written to the DO channel

For DIO modules (e.g., tM-P4C4), using one-digit hexadecimal value to represent the DO status For DO module (e.g., tM-C8), using two-digit hexadecimal value to represent the DO status Bit 0 corresponds to DI0 (or DO0), Bit 1 corresponds to DI1 (or DO1), etc. When the bit is 1, indicates the DO status is ON; When the bit is 0, indicates the DO status is OFF.

Response:

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

- > Delimiter character for a valid command
- ? Delimiter character for an invalid command
- ! Delimiter character for an ignored command

When a host watchdog timeout occurs, the module will set all outputs to a Safe Value. In this case, the data received from DO channels will be ignored.

<u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command: @0307	For module 03 (e.g., tM-C8), set DO0, DO1, and DO2 to ON, and other channels to OFF.									
	D07	DO6	DO5	DO4	DO3	DO2	D01	DO0		
	0 0 0 0 0 1 1 1									
Response: > The command is valid										

Related Commands: <u>@AA</u>

Related Sections: 3.4.4 Configuration Page - DO

@AADODD Set the status of DO channels (for Multi-function module)

Description:

Set the DO status of a module

Note: The command is available for tM-DA1P1R1 and tM-AD4P2C2 modules.

Syntax:

@AADODD[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **DO** Command to set the DO status
- DD Using two-digit hexadecimal value,
 bit 0 corresponds to DO0, and bit 1 corresponds to DO1, etc.
 When the bit is 1, indicates the DO status is ON;
 When the bit is 0, indicates the DO status is OFF.

Response:

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command: @01DO01	For module 02 (e.g., tM-DA1P1R1), set DO0 to ON. DO0 1
Response: !01	The command is valid

Command: @02DO03	For modu	ule 02 (e. DO0	g., tM-AD4P2C2), set DO0 and DO1 to ON.
	1	1	
Response: !02	The com	mand is v	ralid

Related Commands: @AADI

Related Sections: 3.4.3 Configuration Page – AI/DO Alarm

#AA00(Data) Set the status of DO channels (DO0 to DO7)

Note: The command is available for DO modules, and it is the same as the #AA0A(Data) command.

Syntax:

#AA00(Data)[CHKSUM](CR)

- # Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 00 For more than 8-channel, only the lower channel (i.e., DO0 to DO7) can be set.
- (Data) Using the two-digit hexadecimal value, bit 0 corresponds to DO0 and bit 1 corresponds to DO1, etc. When the bit is 1, indicates the DO status is ON; When the bit is 0, indicates the DO status is OFF.

Response:

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

- Delimiter character for a valid command >
- ? Delimiter character for an invalid command
- ļ Delimiter character for an ignored command

When a host watchdog timeout occurs, the module will set all outputs to a Safe Value. In this case, the data received from DO channels will be ignored.

Note: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command:	and										
#010005	DO3	DO2	D01	DO0							
	0	1	0	1							
Response: >	The comm	nand is val	id								
Command:		For module 03 (e.g., tM-C8), set DO0, DO1, DO4, and DO5 to ON, and DO2, DO3, DO6, DO7 to OFF.									
#030033	DO7	DO6	DO5	DO4	DO3	DO2	D01	DO0			
	0	0	1	1	0	0	1	1			
Response: >	The comm	The command is valid									

Related Commands: #AA0A(Data), #AA0B(data), #AAAcDD, #AABcDD, \$AA6, @AA

#AA0A(Data) Set the status of DO channels (DO0 to DO7)

Note: The command is available for DO modules, and it is the same as the #AA00(Data) command.

Syntax:

#AA0A(Data)[CHKSUM](CR)

- # Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **OA** For more than 8-channel, only the lower channel (i.e., DO0 to DO7) can be set.

(Data) Using two-digit hexadecimal value,
 bit 0 corresponds to DO0, and bit 1 corresponds to DO1, etc.
 When the bit is 1, indicates the DO status is ON;
 When the bit is 0, indicates the DO status is OFF.

Response:

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

Ignored Command: ![CHKSUM](CR)

- > Delimiter character for a valid command
- ? Delimiter character for an invalid command
- Delimiter character for an ignored command
 When a host watchdog timeout occurs, the module will set all outputs to a Safe Value.
 In this case, the data received from DO channels will be ignored.
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command:	For modu	ule 01 (e.	g., tM-P3	POR3), set DO1, DO2 to ON, and DO0 to OFF					
#010A06	DO2	DO1	DO0						
TUTURU	1	1	0						
Response: >	The command is valid								

	For module 03 (e.g., tM-C8), set DO0, DO1, DO4, and DO5 to ON, and								
Command:	DO2, DO3, DO6, DO7 to OFF.								
#030A33	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0	
	0	0	1	1	0	0	1	1	
Response: >	The command is valid								

Related Commands: #AA00(Data), #AA0B(data), #AA1cDD, #AAAcDD, #AABcDD, \$AA6, @AA

#AA0B(Data) Set the status of DO channels (DO8 to DO15)

Note: The command is available for DO modules. tM modules support up to 8 DO channels (0 to 7).

Syntax:

#AA0B(Data)[CHKSUM](CR)

#	Delimiter Character
AA	The address of the module (Hex., 00 to FF)
0B	For more than 8-channel, only the higher channel (i.e., DO8 to DO15) can be set.
(Data)	Using two-digit hexadecimal value,
	bit 0 corresponds to DO8, and bit 1 corresponds to DO9, etc.
	When the bit is 1, indicates the DO status is ON;
	When the bit is 0, indicates the DO status is OFF.

Response:

Valid Corr	nmand:	>[CHKSUM](CR)
Invalid Command:		?[CHKSUM](CR)
Ignored C	ommand:	![CHKSUM](CR)
> Delimiter		haracter for a valid command
? Delimiter ch		haracter for an invalid command

- Delimiter character for an ignored command
 When a host watchdog timeout occurs, the module will set all outputs to a Safe Value.
 In this case, the data received from DO channels will be ignored
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

	For module 02, set DO8, DO9, DO12, and DO13 to ON, and								
Command:	DO10, DO11, DO14, DO15 to OFF								
#020B33	D015 D014 D013 D012 D011 D010 D09 D08								
	0 0 1 1 0 0 1 1								
Response: >	The command is valid								

Related Commands:

#AA00(Data), #AA0A(Data), #AA1cDD, #AAAcDD, #AABcDD, \$AA6, @AA

#AA1cDD Set the status of a DO channel (DO0 to DO7)

<u>Note</u>: The command is available for DO modules, and it is the same as the #AAAcDD command.

Syntax:

#AA1cDD[CHKSUM](CR)

#	Delimiter Character
AA	The address of the module (Hex., 00 to FF)
1	Command to set a single digital output channel.
	For more than 8-channel, only the lower channel (i.e., DO0 to DO7) can be set.
с	Specify the digital output channel to be set (0 to 7)
DD	00, set the DO channel to OFF
	01, set the DO channel to ON

Response:

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

- > Delimiter character for a valid command
- ? Delimiter character for an invalid command
- ! Delimiter character for an ignored command

When a host watchdog timeout occurs, the module will set all outputs to a Safe Value. In this case, the data received from DO channels will be ignored

Note: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command:	For module 01, set DO2 to ON.							
	DO7	DO6	DO5	DO4	DO3	DO2	D01	DO0
#011 <u>2</u> 01	0	0	0	0	0	1	0	0
Response: > The command is valid Command: For module 01, set DO4 to ON.								
Command:	DO7	DO6	D05	DO4	DO3	DO2	DO1	DO0
#011 <u>4</u> 01	0	0	0	1	0	1	0	0

Related Commands: #AA00(Data), #AA0A(Data), #AA0B(data), #AAAcDD, #AABcDD, \$AA6, @AA

#AAAcDD Set the status of a DO channel (DO0 to DO7)

Note: The command is available for DO modules, and it is the same as the #AA1cDD command.

Syntax:

#AAAcDD[CHKSUM](CR)

#	Delimiter Character
AA	The address of the module (Hex., 00 to FF)
Α	Command to set a single digital output channel
	For more than 8-channel, only the lower channel (i.e., DO0 to DO7) can be set
с	Specify the digital output channel to be set (0 to 7)
DD	00, set the DO channel to OFF
	01, set the DO channel to ON

Response:

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

- > Delimiter character for a valid command
- ? Delimiter character for an invalid command
- ! Delimiter character for an ignored command

When a host watchdog timeout occurs, the module will set all outputs to a Safe Value. In this case, the data received from DO channels will be ignored

Note: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Commandi	For module	e 01, set D	O2 to ON					
Command: #01A201	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0
#01A <u>2</u> 01	0	0	0	0	0	1	0	0
Response >	The comm	and is valio	k					

Commonde	For module 01, set DO4 to ON							
Command:	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0
#01A <u>4</u> 01	0	0	0	1	0	1	0	0
Response: >	The comm	and is valio	ł					

Related Commands: <u>#AA00(Data)</u>, <u>#AA0A(Data)</u>, <u>#AA0B(data)</u>, <u>#AA1cDD</u>, <u>#AABcDD</u>, <u>\$AA6</u>, <u>@AA</u>

#AABcDD Set the status of a DO channel (DO8 to DO15)

Note: The command is available for DO modules. tM modules support up to 8 DO channels (0 to 7).

Syntax:

#AA0B(Data)[CHKSUM](CR)

#	Delimiter Character
AA	The address of the module (Hex., 00 to FF)
В	Command to set a single digital output channel
	For more than 8-channel, only the higher channel (i.e., DO8 to DO15) can be set
С	Specify the digital output channel to be set (8 to 15)
DD	00, set the DO channel to OFF

01, set the DO channel to ON

Response:

Valid Command:	>[CHKSUM](CR)
Invalid Command:	?[CHKSUM](CR)
Ignored Command:	![CHKSUM](CR)
> Delimiter of	haracter for a valid command

- ? Delimiter character for an invalid command
- I Delimiter character for an ignored command When a host watchdog timeout occurs, the module will set all outputs to a Safe Value. In this case, the data received from DO channels will be ignored
- Note: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

	For module 02, set DO8, DO9, DO12, and DO13 to ON, and							
Command: DO10, DO11, DO14, DO15 to OFF								
#020B33	DO15	DO14	DO13	DO12	DO11	DO10	DO9	DO8
	0	0	1	1	0	0	1	1
Response: > The command is valid								

Related Commands:

#AA00(Data), #AA0A(Data), #AA0B(data), #AA1cDD, #AAAcDD, \$AA6, @AA

#AA Read data from all AI channels

<u>Note:</u> The command is available for AI and tM-AD4P2C2 modules.

Syntax:

#AA[CHKSUM]	(CR)
-------------	------

Delimiter CharacterAA The address of the module (Hex., 00 to FF)

Response:

Valid Com	nmand:	> (Data) [CHKSUM](CR)
Invalid Co	ommand:	?AA [CHKSUM](CR)
>	Delimiter	character for a valid command
?	Delimiter	character for an invalid command
(Data)	Data fror	n every analog input channels

- **AA** The address of the module (Hex., 00 to FF)
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: #01	Read data from all AI channels of module 01 (e.g., tM-AD2)			
Response: >+00.001+00.007				
Data in engineering format: Ch00 = +00.001, Ch01 = +00.007.				

Command: #02	Read data from all AI channels of module 02 (e.g., tM-AD4P2C2)				
Response: >+07.389+07.389+00.002+00.002 Data in engineering format: Ch00 = +07.389, Ch01 = +07.389, Ch02 = +00.002, Ch03 = +00.002.					

Command: #03	Read data from all AI channels of module 03 (e.g., tM-TH8)			
Response: >-9999.9-9999.9-9999.9-9999.9-9999.9-9999.9-9999.9-9999.9-9999.9				
Data in engineering format: Ch00 to Ch07 are -9999.9, which is under the range				

Related Commands: <u>#AAN</u>, <u>#AAA</u>

Related Sections:

3.3 Configuration Page - AI, 3.4.3 Configuration Page – AI/DO Alarm

#AAN Read data from an AI channel

Note: The command is available for AI and tM-AD4P2C2 modules.

Syntax:

#AAN[CHKSUM](CR)

Delimiter Character AA The address of the module (Hex., 00 to FF) Ν The channel to be read, zero-based

Response:

	-	
Valid Con	nmand: >(Data) [CHKSUM](CR)
Invalid Co	ommand: ?AA [CHKSUM](CR)	
>	Delimiter character for a vali	d command
?	Delimiter character for an in-	valid command
(Data)	Analog input data of the spe	cified channel
AA	The address of the module (I	-lex. <i>,</i> 00 to FF)

Note: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	#010	Read data from channel 0 of module 01 (e.g., tM-AD5)
Response:	>+06.251	It indicates Ch00 = +06.250
Command:	#013	Read data from channel 3 of module 01 (e.g., tM-AD2)
Response:	?02	The command is invalid due to no channel 3
Command:	#023	Read data from channel 3 of module 02 (e.g., tM-AD4P2C2)
Response:	>+00.002	It indicates Ch03 = +00.002

Related Commands: <u>#AA</u>, <u>#AAA</u>

Related Sections:

3.3 Configuration Page - AI, 3.4.3 Configuration Page – AI/DO Alarm

\$AAA Read hexadecimal data from all AI channels

<u>Note:</u> The command is available for AI and tM-AD4P2C2 modules.

Syntax:

\$AAA[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- A Command to read every analog input

Response:

Valid Cor	nmand:	>(Data)[CHKSUM](CR)
Invalid C	ommand:	?AA [CHKSUM](CR)
>	Delimiter	r character for a valid command

- ? Delimiter character for an invalid command
- (Data) Data from every analog input channel in hex. format
- AA The address of the module (Hex., 00 to FF)
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	\$01A Read data from all channels of module 01 (e.g., tM-AD2)				
Response:	>7FFA7FF7	Ch00 = +09.998, Ch01 = +09.998 (V)			
Command:	\$02A	Read data from all channels of module 02 (e.g., tM-AD4P2C2)			
Posponsor	>EE04EE040000000	Ch00 = +07.389, Ch01 = +07.389,			
Response:	>5E945E9400090008	Ch02 = +00.002, Ch03 = +00.002.			

Related Commands: <u>#AA</u>, <u>#AAN</u>

Related Sections:

3.3 Configuration Page - AI, 3.4.3 Configuration Page – AI/DO Alarm

\$AA8N Read data from an AO channel

Note: The command is available for tM-DA1P1R1.

Syntax:

\$AA8N[CHKSUM](CR)

- Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 8 Command to read the AO value
- **N** The channel to be read, 0 for tM-DA1P1R1

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) The analog output value
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Response: >	The command is valid
Command: #010+04.000	Set the AO value of channel 0 of module 01 to 4.0

Command: ~0180	Read the AO value of channel 0 of module 01
Response: !01+04.000	Module 01 returns 4.0 (Refer to B.7 Data Ranges for tM-DA1P1R1)

Related Commands:

#AAN(Data), \$AA6N

Related Sections:

3.4.2 Configuration Page - AO, Appendix B.7 Data Ranges for tM-DA1P1R1

\$AA6N Read the last written data from an AO channel

Note: The command is available for tM-DA1P1R1.

Syntax:

\$AA6N[CHKSUM](CR)

- Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **6** Command to read the last written analog output value
- N The channel to be read, 0 for tM-DA1P1R1

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) The last written analog output value.
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: #010+10.000	Set the AO value of channel 0 of module 01 to 10.0
Response: >	The command is valid
Command: \$0160	Read the last written AO value of channel 1 of module 01

Related Commands:

Response: !01+10.000

#AAN(Data), \$AA8N

Related Sections:

3.4.2 Configuration Page - AO, Appendix B.7 Data Ranges for tM-DA1P1R1

Module 01 returns 10.0

#AAN(Data) Set a value of an AO channel

Note: The command is available for tM-DA1P1R1.

Syntax:

#AAN(Data)[CHKSUM](CR)

- # Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **N** The analog output channel to be written, 0 for tM-DA1P1R1
- (Data) Data to be written to the analog output channel, refer to Appendix B.7

Response:

Valid Command:	>[CHKSUM](CR)

Out of Range: **?**[CHKSUM](CR)

Ignored Command: ![CHKSUM](CR)

- > Delimiter character for a valid command
- Pelimiter character indicates that the data is out of range.
 If it is over the range, then the output will be set to the maximum value of the range.
 If it is under range, then the output will be set to the minimum value of the range.
- Delimiter character for an ignored command
 When a host watchdog timeout occurs, the module will set all outputs to a Safe Value.
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command: \$0190	Read the AO configuration of channel 0 of module 01
Response: !0120	It returns: Type Code is 2 and Slew Rate is 0, which means output type is 0 to 10 V and output changing immediately.
Command: #010+05.000	Set the AO value of channel 0 of module 01 to 5.0
Response: >	The command is valid
Command: #010+15.000	Set the AO value of channel 0 of module 01 to 15.0.
Response: ?	The value is out of range, and it will be set to 10 V

Examples:

Related Commands: \$AA9N, %AANNTTCCFF

Related Sections: 3.4.2 Configuration Page - AO, Appendix B.7 Data Ranges for tM-DA1P1R1

~** Notify all modules that the Host works normally

Description:

Informs all modules that the host is OK.

Note: After sending the command, the delay time for the next command sending is 2ms.

Syntax:

~**[CHKSUM](CR)

- ~ Delimiter Character
- ** Host OK command.

Response:

No response.

Examples:

Command: ~**	Send a "Host OK" command to all modules
Response: No response	

Related Commands:

<u>~AA0, ~AA1, ~AA2, ~AA3EVV</u>

Related Sections:

~AA0 Read the host watchdog status of a module

Syntax:

~AA0[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **0** Command to read the host watchdog status

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **SS** Using two hexadecimal digits to indicates the status of the host watchdog.
 - Bit 7: 0, indicates that the host watchdog is disabled.
 - 1, indicates that the host watchdog is enabled.
 - Bit 2: 0, indicates that a host watchdog timeout has occurred.
 - 1, indicates that no host watchdog has occurred.
- **Note:** The host watchdog status is stored in EEPROM and can only be reset using the ~AA1 command.
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	~010	Read the host watchdog status of module 01							
Response:	!0100	Return 00, which means the host watchdog is disabled and no timeout occurs							
Command:	~020	Read the host watchdog status of module 02							
	Return 04, which means the host watchdog timeout has occurred								
Response:	!0204	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	:0204	0	0	0	0	0	1	0	0
			0				Z	1	

Related Commands: <u>~***</u>, <u>~AA0</u>, <u>~AA1</u>, <u>~AA2</u>, <u>~AA3EVV</u>

Related Sections: 3.4.5 Configuration Page – Host WDT, A.2 Dual Watchdog

~AA1 Reset the host watchdog status of a module

Syntax:

~AA1[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 1 Command to reset the host watchdog timeout status

Response:

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)

<u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	~010	Read the host watchdog status of module 01					
Response:	!0104	Return 04, which means the host watchdog timeout has occurred					
Command:	~011	Reset the host watchdog timeout status of module 01					
Response:	!01	The command is valid					
Command:	~010	Read the host watchdog status of module 01					
Response:	!0100	Return 00, which means the host watchdog is disabled and no timeout occurs					

Related Commands:

<u>~***</u>, <u>~AA0</u>, <u>~AA2</u>, <u>~AA3EVV</u>

Related Sections:

~AA2 Read the host watchdog timeout value of a module

Syntax:

~AA2[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 2 Command to read the host watchdog timeout value

Response:

Valid Command:	!AAEVV [CHKSUM](CR)
valid Command:	

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- E 0, means that the host watchdog is disabled1, means that the host watchdog is enabled
- **VV** Using two hexadecimal digits to represent the timeout value in tenths of a second, for example, 01 denotes 0.1 seconds and FF denotes 25.5 seconds
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~012	Read the host watchdog timeout value of module 01
Response: !01 <u>1FF</u>	Return 1FF, which means the host watchdog is enabled and the timeout value is 25.5 seconds

Related Commands:

<u>~***</u>, <u>~AA0</u>, <u>~AA1</u>, <u>~AA3EVV</u>

Related Sections:

~AA3EVV Set the host watchdog and a timeout value

Description:

To enable or disable the host watchdog and set the host watchdog timeout value of a module.

Syntax:

~AA3EVV[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **3** Command to set the host watchdog
- **E** 0, to disable the host watchdog.
 - 1, to enable the host watchdog.
- **VV** Using two hexadecimal digits to represent the 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 for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	~013164	Enable the host watchdog of module 01 and set the timeout value to 10.0 seconds. 64 $_{\rm (16)}$ = 100 $_{\rm (10)}$
Response:	!01	The command is valid
Command:	~012	Read the host watchdog timeout value of module 01.
Response:	!01 <u>164</u>	Return 164, which means the host watchdog is enabled and the timeout value is 10.0 seconds.

Related Commands:

<u>~**</u>, <u>~AA0</u>, <u>~AA1</u>, <u>~AA2</u>

Related Sections:

~AA4V Read the Power-on value or Safe value of the DO module

<u>Note:</u> tM-AD4P2C2 needs to use the <u>~AA4</u> command.

Syntax:

~AA4V[CHKSUM](CR)

- Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 4 Command to read the Power-on value or the Safe value
- V P: Read the Power-on value
 - S: Read the Safe Value

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) Using two hexadecimal digits followed by 00 to represent the Power-on value or the Safe value
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~014S	Read the Safe value of module 01 (e.g., tM-DA1P1R1)					
Response: !01 <u>01</u> 00	The Safe Value is 01 DO0 ON					

Command: ~024P	Read the	Read the Power-on value of module 02 (e.g., tM-C8)							
	The Pow	The Power-on Value is C3							
Response: !02 <u>C3</u> 00	DO7	D06	D05	DO4	DO3	DO2	DO1	DO0	
	ON	ON	OFF	OFF	OFF	OFF	ON	ON	

Related Commands: <u>~AA5V</u>

~AA5V Set the DO value as the Power-on value or Safe value

<u>Note:</u> tM-AD4P2C2 needs to use the <u>~AA5PPSS</u> command.

Syntax:

~AA5V[CHKSUM](CR)

- Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 5 Command to set the Power-on value or the Safe value
- V P: Set the Power-on value
 - S: Set the Safe value

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note</u>: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~015S	Set the Safe value of module 01 (e.g., tM-P4A4) to the current DO value
Response: !01	The command is valid

Command:	~014S	Read the Safe value of module 01						
		The Safe	Value is	0C				
Response:	!01 <u>0C</u> 00	DO3	DO2	DO1	DO0			
		ON	ON	OFF	OFF			

Related Commands: <u>~AA4V</u>

Related Sections: 3.4.3 Configuration Page – AI/DO Alarm, 3.4.4 Configuration Page - DO,

A.2 Dual Watchdog

~AA4 Read the Power-on value or Safe value (Multi-function-DO)

Note: The command is available for tM-DA1P1R1 and tM-AD4P2C2.

Syntax:

~AA4[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 4 Command to read the Power-on value and the Safe value

Response:

Valid Command: !A	APPSS[CHKSUM](CR)
-------------------	-------------------

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- PP Using two hexadecimal digits to represent the Power-on value
- SS Using two hexadecimal digits to represent the Safe value
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	~014	Read both the Power-on value and Safe value of module 01							
Response:	!010000	The Power-on value is 00, and the Safe Value is 00							
Command:	~024	Read both the Power-on value and Safe value of module 02							
	!020203	The Power-on value is 02, and the Safe Value is 03							
Deenenee		Power-on Value		Safe Value					
Response:		DO1	DO0	DO1	DO0				
		ON	OFF	ON	ON				

Related Commands: <u>~AA5PPSS</u>

Related Sections:3.4.3 Configuration Page – AI/DO Alarm, 3.4.4 Configuration Page - DO,
A.2 Dual Watchdog

~AA5PPSS Set the Power-on value or Safe value (Multi-function-DO)

Note: The command is available for tM-DA1P1R1 and tM-AD4P2C2. If the Alarm function of tM-AD4P2C2 is enabled, this command will be invalid.

Syntax:

~AA5PPSS[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **5** Command to set the Power-on value and the Safe value
- PP Using two hexadecimal digits to represent the Power-on value
- SS Using two hexadecimal digits to represent the Safe value

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~	°0150000	Set the power-on value to 00 and the safe value to 00 for module 01							
Response: !	01	The command is valid							
Command: ~	~014	Read both the Power-on value and Safe value of module 01							
	!010000	The Power-on value is 00, and the Safe Value is 00							
Beenense, 1		Power-c	on Value	Safe V	Value				
Response: !		DO1	DO0	DO1	DO0				
		OFF	OFF	OFF	OFF				

Related Commands: <u>~AA4</u>

Related Sections: 3.4.3 Configuration Page – AI/DO Alarm, 3.4.4 Configuration Page - DO,

A.2 Dual Watchdog

\$AA7N Read the Power-on value of the AO module

Description:

Read the Power-on value of the specified analog output channel.

Note: The command is available for tM-DA1P1R1.

Syntax:

\$AA7N[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 7 Command to read the Power-on AO value
- **N** The channel to be read, 0 for tM-DA1P1R1

Response:

- Valid Command: **!AA(Data)**[CHKSUM](CR) Invalid Command: **?AA**[CHKSUM](CR)
 - ! Delimiter character for a valid command
 - ? Delimiter character for an invalid command
 - AA The address of the module (Hex., 00 to FF)
 - (Data) The Power-on AO Value
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$0170	Read the Power-on AO value of channel 0 of module 01			
Response: !01+03.000	The Power-on value is 3.0			

Related Commands: \$AA4N

\$AA4N Set the AO value as the Power-on value

Description:

Set the Power-on value of the specified channel to an AO value.

Note: The command is available for tM-DA1P1R1.

Syntax:

\$AA4N[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 4 Command to set the Power-on value
- **N** The channel to be set, 0 for tM-DA1P1R1

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)

<u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	#010+05.000	Set the AO value of channel 0 of module 01 to 5.0				
Response:	>	The command is valid				
Command:	\$0140	Set the Power-on value of channel 0 of module 01 to the current AO value				
Response:	!01	The command is valid				
Command:	\$0170	Read the Power-on value of channel 0 of module 01				
Response:	!01+05.000	The Power-on value is 5.0				

Related Commands: #AAN(Data), \$AA7N

~AA4N Read the Safe Value of the AO module

Description:

Read the Safe value of the specified analog output channel.

<u>Note:</u> The command is available for tM-DA1P1R1.

Syntax:

- ~AA4N[CHKSUM](CR)
 - Delimiter Character
 - AA The address of the module (Hex., 00 to FF)
 - 4 Command to read the Safe AO value
 - **N** The channel to be read, 0 for tM-DA1P1R1

Response:

- Valid Command: **!AA(Data)**[CHKSUM](CR)
- Invalid Command: **?AA**[CHKSUM](CR)
 - ! Delimiter character for a valid command
 - ? Delimiter character for an invalid command
 - AA The address of the module (Hex., 00 to FF)
 - (Data) The Safe AO value
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~0140	Read the Safe value of AO channel 0 of module 01
Response: !01+02.5	D0 The Safe value is 2.5

Related Commands: <u>~AA5N</u>

~AA5N Set the AO value as the Safe Value

Description:

Set the Safe value of the specified channel to an AO value.

Note: The command is available for tM-DA1P1R1.

Syntax:

~AA5N[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **5** Command to set the Safe value
- **N** The channel to be set, 0 for tM-DA1P1R1

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: #010+03.500	Set the AO value of channel 0 of module 01 to 3.5.
Response: >	The command is valid

Command: ~0150	Set the Safe value of channel 0 of module 01 to the current AO value				
Response: !01	The command is valid				

Command: ~0140	Read the Safe value of channel 0 of module 01		
Response: !01+03.500	The Safe value is 3.5		

Related Commands: <u>#AAN(Data)</u>, <u>~AA4N</u>

\$AALS Read the latched status of DI channels

Syntax:

\$AALS[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- L Command to read the latched status
- **S** 0: Read the low latched status
 - 1: Read the high latched status

Response:

Valid Cor	nmand:	!AA(Data) [CHKSUM](CR)
Invalid Co	ommand:	?AA [CHKSUM](CR)
!	Delimite	r character for a valid command
?	Delimite	r character for an invalid command
AA	The addr	ress of the module (Hex., 00 to FF)
(Data)	Using fou	ur hexadecimal digits followed by 00 to represent the DI (or DO) latched status
	_	

Refer to the <u>\$AA6 Command – DIO Data Format Table</u> For DI/DO module: (DO or DI Latch) (00) 00

For DIO module: (DO Latch) (DI Latch) 00

<u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$01L0	Read the low latched status of module 01 (e.g., tM-P8)							
	For DI module: (DI Latch) (00) 00							
Response: <u> 3C</u> 0000	DI7	DI6	DI5	DI4	DI3	DI2	DI1	DIO
	0	0	1	1	1	1	0	0

Command: \$02L0	Read the low latched status of module 02, (e.g., tM-P3POR3)					
	For DIO m	odule: (DC	O Latch) (DI Latch)	00	
Response: <u>!0707</u> 00	DO2	D01	DO0	DI2	DI1	DI0
	1	1	1	1	1	1

Related Commands: <u>\$AAC</u>

\$AAC Clear the latched status of DI channels

Description:

Clear the high/low latched status of a module

Syntax:

\$AAC[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **C** Command to clear the latched status

Response:

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

- Invalid Command: **?AA**[CHKSUM](CR)
 - ! Delimiter character for a valid command
 - ? Delimiter character for an invalid command
 - AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$02C	Clear the latched status
Response: !02	The command is valid

Test the command:

When a DI signal changes from OFF to ON, it triggers a high latched status. When a DI signal changes from ON to OFF, it triggers a low latched status. If executing the **\$02C** command under the low latched status, the previous triggered high latched status can be cleared.

See <u>Chapter 4 Hardware Wiring for testing</u> to connect both the tM-P8 (8 DI) and tM-C8 (8 DO) modules, and test the command by using DCON Utility Pro on PC.

- 1. On the DO configuration page (see Section 3.4.4), check the DO0 to DO7 box to set them to ON. At this time, the status of DI0 to DI7 is from OFF to ON to trigger a high latch.
- 2. Uncheck the DO0 to DO7 box to set them to OFF.

At this time, the status of DIO to DI7 is from ON to ON to trigger a low latch.

3. Execute the **\$02C** command to clear the previous high latched statuses.

Related Commands: <u>\$AALS</u>

Related Sections: 3.4.4 Configuration Page - DO, 3.4.6 Configuration Page – DI, A.6 advanced DI Features (Latch and Counter)

#AAN Read a counter value of the DI channel

Syntax:

#AAN[CHKSUM](CR)

- # Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **N** The channel to be read (0 to F)

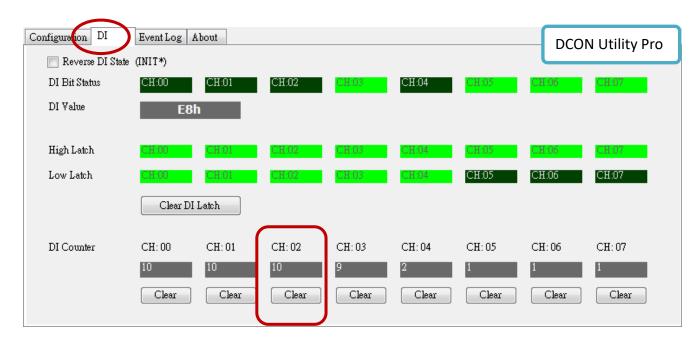
Response:

- Valid Command: **!(Data)**[CHKSUM](CR)
- Invalid Command: ?AA[CHKSUM](CR)
- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- (Data) Using five hexadecimal digits to represent the DI Counter value (00000 to 65535)
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: #012	Read the Counter value of channel 2 of module 01 (e.g., tM-P8)				
Response: !0100010	he return value is 10				
Command: #019	Read the Counter value of channel 9 of module 01				
Response: ?01	The command is invalid due to no channel 9				

Related Commands: <u>\$AACN</u>



\$AACN Clear a counter value of the DI channel

Syntax:

\$AACN[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **C** Command to clear the value of DI Counter
- **N** The channel to be cleared (0 to F)

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)

<u>Note</u>: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

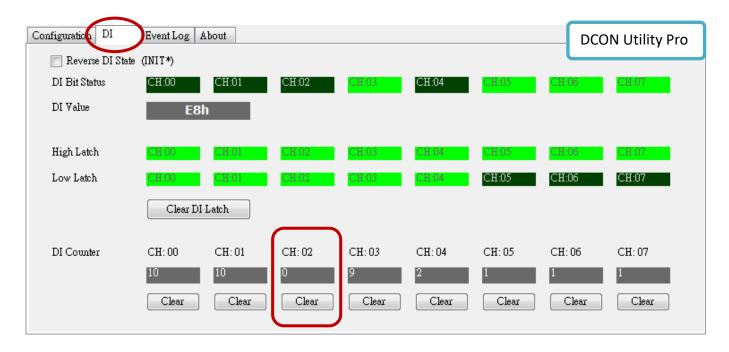
Command: #012	Read the Counter value of channel 2 of module 01 (e.g., tM-P8)			
Response: !01 <u>00010</u>	The return value is 10			
Command: \$01C2	Clear the Counter value of channel 2 of module 01 (e.g., tM-P8)			

Related Commands: <u>#AAN</u>

Response: !01

Related Sections: 3.4.6 Configuration Page – DI, A.6 advanced DI Features (Latch and Counter)

The command is valid



@AARECi Read a counter value of the DI channel for tM-AD4P2C2

Syntax:

@AARECi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- RE Command to read a DI Counter value
- **Ci** i, indicates the channel to be read (0 to 1)

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)

(Data) Using five hexadecimal digits to represent the DI Counter value (00000 to 65535)

<u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @02REC1	Read the Counter value of channel 1 of module 02
Response: !02 <u>00016</u>	The return value is 16
Command: @02REC9	Read the Counter value of channel 9 of module 02
Response: ?02	The command is invalid due to no channel 9

Related Commands: @AACECi

tAD4P2C2 Firmware[A1	05]	
Configuration AI/DO Al	ann DI Host WDT Event Log About	
📄 Reverse DI Sta	le (INIT*)	
DI Bit Status	CH:00 CH:01	
DI Value	00h	
High Latch	CH 00 CH 01	
Low Latch	CH.00 CH.01	
DI Counter	Clear DI Latch CH: 00 CH: 01	
Di Coulini	15 16	
	Clear Clear	DCON Utility Pro

@AACECi Clear a counter value of the DI channel for tM-AD4P2C2

Syntax:

@AACECi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- CE Command to clear the value of DI Counter
- **Ci** i, indicates the channel to be cleared (0 to 1)

Response:

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

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

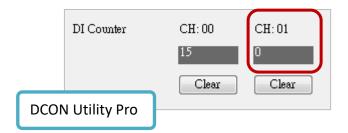
- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)

Note: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @02REC1	Read the Counter value of channel 1 of module 02	
Response: 10200016	The return value is 16	
Command: @02CEC1	Clear the Counter value of channel 1 of module 02	
Response: !02	The command is valid	
Command: @02REC1	Read the Counter value of channel 1 of module 02	
Response: 1020000	The return value is 0	

Related Commands: @AARECi



@AARECN Read a counter value of the DI channel for tM-DA1P1R1

Syntax:

@AARECN[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- RE Command to read a DI Counter value
- **CN** N, indicates the channel to be read, 0 for tM-DA1P1R1

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) Using five hexadecimal digits to represent the DI Counter value (00000 to 65535)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command @01REC0	Read the Counter value of channel 1 of module 01	
Response: !01 <u>00030</u>	The return value is 30	
Command: @01REC9	Read the Counter value of channel 9 of module 01	
Response: ?01	The command is invalid due to no channel 9	

Related Commands: @AACECN

tDA1P1R1 Firmware	[A103]	
Configuration AO	DO DI Host WDT Event Lo	g About
📝 Reverse DI	State (INIT*)	
DI Bit Status	CH.00	
High Latch	CH 00	
Low Latch	CH:00	
	Clear DI Latch	
DI Counter	30 Clear	DCON Utility Pro

@AACECN Clear a counter value of the DI channel for tM-DA1P1R1

Syntax:

@AACECN[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- CE Command to clear the value of DI Counter
- **CN** N, indicates the channel to be cleared (0)

Response:

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

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

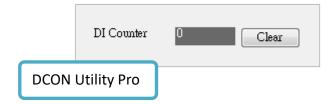
- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)

Note: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command @01REC0	Read the Counter value of channel 1 of module 01	
Response: !01 <u>00030</u>	The return value is 30	
Command: @01CEC0	Clear the Counter value of channel 1 of module 01	
Response: !01 The command is valid		
Command: @01REC0	Read the Counter value of channel 1 of module 01	
Response: 10100000	The return value is 0	

Related Commands: @AARECN



@AAEATCi Enable the alarm function for the AI channel

<u>Note:</u> The command is available for tM-AD4P2C2.

Syntax:

@AAEATCi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- T Alarm Type
 - M: Momentary; L: Latched
- Ci i, indicates the channel to enable alarm

Response:

Valid Command:	!AA [CHKSUM](CR)

- Invalid Command: **?AA**[CHKSUM](CR)
- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Response: !01

Command: @01EAMC0 Enable the alarm of channel 0 of module 01, and set the alarm type to Momentary		
Response: !01	The command is valid	
Command: @02EALC1	Enable alarm of channel 1 of module 02, and set the alarm type to Latch	

Related Commands: @AADACi, @AARACi

Related Sections: 3.4.3 Configuration Page - AI/DO Alarm

The command is valid

@AADACi Disable the alarm function for the AI channel

<u>Note:</u> The command is available for tM-AD4P2C2.

Syntax:

@AADACi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- DA Command to disable the alarm
- Ci i, indicates the channel to disable the alarm

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @02DAC0	Disable the alarm of channel 0 of module 02
Response: !01	The command is valid

Related Commands: @AAEATCi

Related Sections: 3.4.3 Configuration Page - AI/DO Alarm

@AARACi Read the setting of alarm type

Note: The command is available for tM-AD4P2C2.

Syntax:

@AARACi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **RA** Command to read the alarm setting
- Ci i, indicates the channel to read the alarm setting

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- S Alarm Type
 - 0: Disable; 1: Momentary; 2: Latch
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @01RAC0	Read the alarm setting of channel 0 of module 01
Response: !011	The alarm type is Momentary
Command: @01RAC1	Read the alarm setting of channel 1 of module 01

Command: @01RAC1	Read the alarm setting of channel 1 of module 01
Response: !012	The alarm type is Latch

Related Commands:

@AARAO, @AARHCi, @AARLCi, @AAHI(data)Ci, @AALO(data)Ci, @AACHCi, @AACLCi

Related Sections:

3.4.3 Configuration Page - AI/DO Alarm

@AARAO Read the status of high/low alarm

Note: The command is available for tM-AD4P2C2.

Syntax:

@AARAO[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- RAO Command to read the alarm setting

Response:

Valid Command: !AA	AHHLL[CHKSUM](CR)
--------------------	-------------------

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **HH** Using two hexadecimal digits to represent the currently activated high alarms.
- Using two hexadecimal digits to represent the currently activated low alarm.
 Bit 0 corresponds to channel 0, and bit 1 corresponds to channel 1, etc.
 When the bit is 1, means there is an activated high/low alarm associated with the channel.
 When the bit is 0, means there is no activated alarm.
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @01RAO	Read the currently activated alarms of module 01
Response: !0100 <u>03</u>	There are the activated low alarm on channel 0 and 1

Command: @02RAO	Read the currently activated alarms of module 02
Response: !02 <u>02</u> 00	There is an activated high alarm on channel 1

Related Commands:

@AARACi, @AARHCi, @AARLCi, @AAHI(data)Ci, @AALO(data)Ci, @AACHCi, @AACLCi

Related Sections:

3.4.3 Configuration Page - AI/DO Alarm

@AARHCi Read the value of high alarm limit

Note: The command is available for tM-AD4P2C2.

Syntax:

@AARHCi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **RH** Command to read the high alarm limit
- Ci i, indicates the channel to read the high alarm limit

Response:

- Valid Command: **!AA(Data)**[CHKSUM](CR) Invalid Command: **?AA**[CHKSUM](CR)
 - ! Delimiter character for a valid command
 - ? Delimiter character for an invalid command
 - AA The address of the module (Hex., 00 to FF)
 - (Data) The high alarm limit in engineering format
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	@028HC0	Read the high alarm limit of channel 0 of module 02
command.		
Response:	!02+09.000	The high alarm limit is + 9.0
	00000104	
Command:	@UZKHC1	Read the high alarm limit of channel 1 of module 02
Response:	102+08.000	The high alarm limit is +8.0

Related Commands:

@AARAO, @AARLCi, @AAHI(data)Ci, @AALO(data)Ci, @AACHCi, @AACLCi

Related Sections:

3.4.3 Configuration Page - AI/DO Alarm

@AARLCi Read the value of low alarm limit

Note: The command is available for tM-AD4P2C2.

Syntax:

@AARLCi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **RL** Command to read the low alarm limit
- Ci i, indicates the channel to read the low alarm limit

Response:

- Valid Command: **!AA(Data)**[CHKSUM](CR) Invalid Command: **?AA**[CHKSUM](CR) **!** Delimiter character for a valid command
 - ? Delimiter character for an invalid command
 - AA The address of the module (Hex., 00 to FF)
 - (Data) The low alarm limit in engineering format
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Response: !02-08.000

Command: @02RLC0	Read the low alarm limit of channel 0 of module 02
Response: !02-09.000	The low alarm limit is - 9.0
Command: @02RLC1	Read the low alarm limit of channel 1 of module 02

Related Commands: @AARAO, @AARHCi, @AAHI(data)Ci, @AALO(data)Ci, @AACHCi, @AACLCi

The low alarm limit is -8.0

Related Sections: 3.4.3 Configuration Page - AI/DO Alarm

@AAHI(data)Ci Set the value of high alarm limit

Note: The command is available for tM-AD4P2C2.

Syntax:

@AAHICi[CHKSUM](CR)

- Ø Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- HI Command to set the high alarm limit
- (data) The high alarm limit in engineering format
- Ci i, indicates the channel to set the high alarm limit

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @01HI+10.000C0	Set the high alarm limit of channel 0 of module 01 to + 10.0
Response: !01	The command is valid
Command: @01HI+09.000C1	Set the high alarm limit of channel 1 of module 01 to + 9.0
Response: !01	The command is valid

Related Commands: @AARAO, @AARHCi, @AARLCi, @AALO(data)Ci, @AACHCi, @AACLCi

Related Sections: 3.4.3 Configuration Page - AI/DO Alarm

@AALO(data)Ci Set the value of low alarm limit

Note: The command is available for tM-AD4P2C2.

Syntax:

@AALOCi[CHKSUM](CR)

- Ø Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- LO Command to set the low alarm limit
- (data) The low alarm limit in engineering format
- Ci i, indicates the channel to set the low alarm limit

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @01LO-05.000C0	Set the low alarm limit of channel 0 of module 01 to -5.0.
Response: !01	The command is valid

Command: @01HI+01.000C1	Set the low alarm limit of channel 1 of module 01 to + 1.0
Response: !01	The command is valid

Related Commands: @AARAO, @AARHCi, @AARLCi, @AAHI(data)Ci, @AACHCi, @AACLCi

Related Sections: 3.4.3 Configuration Page - AI/DO Alarm

@AACHCi Clear the status of a high latched alarm

Note: The command is available for tM-AD4P2C2.

Syntax:

@AACHCi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- CH Command to clear the high latch alarm
- Ci i, indicates the channel to clear the high latch alarm

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @01CHC1	Clear the high latched alarm of channel 0 of module 01
Response: !01	The command is valid

Related Commands: @AARAO, @AARHCi, @AARLCi, @AAHI(data)Ci, @AALO(data)Ci, @AACLCi

Related Sections: 3.4.3 Configuration Page - AI/DO Alarm

@AACLCi Clear the status of a low latched alarm

Note: The command is available for tM-AD4P2C2.

Syntax:

@AACLCi[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- CL Command to clear the low latch alarm
- Ci i, the channel to clear the low latch alarm

Response:

Valid Command:	!AA [CHKSUM](CR)	
vana communa.		

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @01CLC1	Clear the low latched alarm of channel 0 of module 01
Response: !01	The command is valid

Related Commands: @AARAO, @AARHCi, @AARLCi, @AAHI(data)Ci, @AALO(data)Ci, @AACHCi

Related Sections: 3.4.3 Configuration Page - AI/DO Alarm

~AAEV Enable/Disable the calibration

Note: The command is available for AI and tM-AD4P2C2 modules.

Syntax:

~AAEV[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **E** Command to enable/disable the calibration
- V 1: Enable calibration ; 0: Disable calibration

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	\$010	Perform a span calibration on module 01
Response:	?01	The command is invalid because the calibration is not yet enabled
Command:	~01E1	Enable the calibration on module 01
Response:	!01	The command is valid
Command:	\$010	Perform a span calibration on module 01
Response:	!01	The command is valid

Related Commands: \$AA0, \$AA1, \$AA0Ci, \$AA1Ci (Only available for tM-AD2)

Related Sections: 3.6 Calibration

\$AA1 Perform zero calibration (V, A)

Note: The command is available for AI and tM-AD4P2C2 modules. Before using it, execute the **~AAEV** command first.

Syntax:

\$AA1[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 1 Command to perform a zero calibration

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$011	Perform a zero calibration on module 01	
Response: ?01	The command is invalid because the calibration is not yet enabled	
Command: ~01E1	Enable the calibration on module 01	
Response: !01	The command is valid	
Command: \$011	Perform a zero calibration on module 01	
Response: !01	The command is valid	

Related Commands: <u>~AAEV</u>, <u>\$AA0</u>, <u>\$AA0Ci</u>

\$AA1Ci Perform zero calibration

<u>Note</u>: The command is available for tM-AD4P2C2. Before using it, executing the **~AAEV** command first.

Syntax:

\$AA1Ci[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 1 Command to perform a zero calibration
- Ci i, indicates the channel to be calibrated

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$011C0	Perform a zero calibration on channel 0 of module 01	
Response: ?01	The command is invalid because the calibration is not yet enabled	
Command: ~01E1	Enable the calibration on module 01	
Response: !01	The command is valid	
Command: \$011C0	Perform a zero calibration on channel 0 of module 01	
Response: !01	The command is valid	

Related Commands: <u>~AAEV</u>, <u>\$AA0Ci</u>

\$AA0 Perform span calibration (V, Ω)

Note: The command is available for tM-AD5, AD8, TH8, and tM-AD4P2C2. Before using it, executing the **~AAEV** command first.

Syntax:

\$AA0[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **0** Command to perform a span calibration

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	\$010	Perform a span calibration on module 01
Response:	?01	The command is invalid because the calibration is not yet enabled
Command:	~01E1	Enable the calibration on module 01
Response:	!01	The command is valid
Command:	\$010	Perform a span calibration on module 01
Response:	!01	The command is valid

Related Commands: <u>~AAEV</u>, <u>\$AA1</u>

\$AA0Ci Perform span calibration (A)

Note: The command is available for tM-AD2, AD5C, AD8C, and tM-AD4P2C2. Before using it, executing the **~AAEV** command first.

Syntax:

\$AA0Ci[CHKSUM](CR)

- **\$** Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **0** Command to perform a span calibration
- Ci i, indicates the channel to be calibrated

Response:

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~01E1	Enable the calibration on module 01
Response: !01	The command is valid

Command: \$010C0	Perform a span calibration on channel 0 of module 01
Response: !01	The command is valid

Related Commands: <u>~AAEV</u>, <u>\$AA1</u>, <u>\$AA1Ci</u> (Only available for tM-AD2)

\$AA0N Perform zero calibration

<u>Note:</u> The command is available for tM-AD1P1R1.

Syntax:

\$AA0[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **0** Command to perform a zero calibration
- **N** The channel to be calibrated, 0 for tM-AD1P1R1

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: 0100	Perform a zero calibration on channel 0 of module 01
Response: !01	The command is valid

Related Commands: <u>\$AA1N</u>, <u>\$AA3NVV</u>

\$AA1N Perform span calibration

<u>Note:</u> The command is available for tM-AD1P1R1.

Syntax:

\$AA1[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- 1 Command to perform a span calibration
- N The channel to be calibrated, 0 for tM-AD1P1R1

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$0110	Perform a span calibration on channel 0 of module 01
Response: !01	The command is valid

Related Commands: <u>\$AA0N</u>, <u>\$AA3NVV</u>

\$AA3NVV Adjust AO value for calibration

Note: The command is available for tM-AD1P1R1.

Syntax:

\$AA3NVV[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **3** Command to adjust the AO value
- **N** The channel to adjust the value, 0 for tM-AD1P1R1

VV Using two-digits hexadecimal value to adjust the analog output 00 to 5F, indicates to increase 0 to 95 counts ;
 FF to A1, indicates to decrease 1 to 95 counts
 Each count is about 4.88uA or 2.44mV

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$0	1301F	Set the value of channel 0 of module 01 to increase by 31 counts (It is about +151.28 uA or +75.64 mV)
Response: !01	1	The command is valid

Related Commands: <u>\$AA0N</u>, <u>\$AA1N</u>

\$AAS1 Reload the default calibration parameters

<u>Note:</u> The command is available for the tM-TH8 module.

Syntax:

\$AAS1[CHKSUM](CR)

- \$ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **S1** Command to reload the factory default calibration parameters

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: \$01S1	Reload the factory default calibration parameters for module 01
Response: !01	The command is valid

Related Commands: <u>~AAEV</u>

@AAA3Ci Read the temperature offset of a channel

<u>Note:</u> The command is available for the tM-TH8 module.

Syntax:

@AAA3Ci[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- A3 Command to read the temperature offset
- Ci i, indicates the channel to be read (0 to 7)

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) Using two-digits hexadecimal value to represent the temperature offset
 01: indicates 0.1 02: indicates 0.2 FF: indicates -0.1 FE: indicates -0.2
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	@03A3C2	Read the temperature offset of channel 02 of module 03
Response:	!030A	The offset value is 1.0
Command:	@03A3C5	Read the temperature offset of channel 05 of module 03
Response:	!03F0	The offset value is -1.6.

Related Commands: @AAA2CiToo

Related Sections: 3.3.3 Configuration Page 03 - AI (for tM-TH8)

@AAA2CiToo Set the temperature offset of a channel

<u>Note:</u> The command is available for the tM-TH8 module.

Syntax:

@AAA2CiToo[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- A2 Command to set the temperature offset
- **Ci** i, indicates the channel to be set (0 to 7).

Too oo, indicates the temperature offset in 0.1°C/°F, ranges from -12.8 to 12.7 Using the hexadecimal value, 01: indicates 0.1, 02: indicates 0.2,
 FF: indicates -0.1, FE: indicates -0.2

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)

<u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	@03A2C <u>2</u> T <u>0A</u>	Set the temperature offset of channel 02 of module 03 to 1.0
Response:	!03	The command is valid
Command:	@03A2C <u>5</u> T <u>F0</u>	Set the temperature offset of channel 05 of module 03 to -1.6
Response:	!03	The command is valid

Related Commands: <u>@AAA3Ci</u>

Related Sections: 3.3.3 Configuration Page 03 - AI (for tM-TH8)

@AAA7Ci Read the resistance offset of a channel

<u>Note:</u> The command is available for the tM-TH8 module.

Syntax:

@AAA7Ci[CHKSUM](CR)

- *@* Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- A7 Command to read the resistance offset
- Ci i, indicates the channel to be read (0 to 7)

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) Using two-digits hexadecimal value to represent the resistance offset
 01: indicates 0.1 02: indicates 0.2 FF: indicates 25.5 FE: indicates 25.4
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @	03A7C2	Read the resistance offset of channel 02 of module 03
Response: !0)30A	The offset value is 1.0

Command: @03A7C5	Read the resistance offset of channel 05 of module 03
Response: !03F0	The offset value is 24.0.

Related Commands: @AAA6CiRrr

Related Sections: 3.4.1 Configuration Page - User Defined Type

@AAA6CiRrr Set the resistance offset of a channel

<u>Note</u>: The command is available for the tM-TH8 module.

Syntax:

@AAA6CiRrr[CHKSUM](CR)

- Ø Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- A6 Command to set the resistance offset
- **Ci** i, indicates the channel to be set (0 to 7)
- Rrr rr, indicates the resistance offset in 0.1 ohms, ranges from 0.0 to 25.5 ohms.
 01: indicates 0.1 02: indicates 0.2 FF: indicates 25.5 FE: indicates 25.4

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	@03A6C <u>2</u> T <u>0A</u>	Set the resistance offset of channel 02 of module 03 to 1.0
Response:	!03	The command is valid
Command:	@03A6C <u>5</u> T <u>F0</u>	Set the resistance offset of channel 05 of module 03 to 24.0.
Response:	!03	The command is valid

Related Commands: @AAA7Ci

Related Sections: 3.4.1 Configuration Page - User Defined Type

~AAD Read the temperature scale ($^{\circ}C/^{\circ}F$)

<u>Note</u>: The command is available for the tM-TH8 module.

Syntax:

~AAD[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **D** Command to read the temperature scale

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- T Temperature scale

0: Celsius (by default); 1: Fahrenheit.

<u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: ~03D	Read the temperature scale of module 03
Response: !01 <u>0</u>	Return 0, stands for Celsius
Command: ~03D	Read the temperature scale of module 03
Response: !01 <u>1</u>	Return 1, stands for Fahrenheit

Related Commands: <u>~AADT</u>

~AADT Set the temperature scale ($^{\circ}C/^{\circ}F$)

<u>Note</u>: The command is available for the tM-TH8 module.

Syntax:

~AADT[CHKSUM](CR)

- ~ Delimiter Character
- AA The address of the module (Hex., 00 to FF)
- **D** Command to set the temperature scale
- T C: Celsius (by default) ; F: Fahrenheit.

Response:

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

Invalid Command: ?AA[CHKSUM](CR)

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	~03DC	Set the temperature scale of module 03 to Celsius
Response:	!01	The command is valid
Command [.]	~03DF	Set the temperature scale of module 03 to Eahrenheit

Command:	~03DF	Set the temperature scale of module 03 to Fahrenheit
Response:	!01	The command is valid

Related Commands: <u>~AAD</u>

@AAGxTtt Read the Steinhart-Hart coefficient of a user-defined type

<u>Note</u>: The command is available for the tM-TH8 module.

Syntax:

@AAGxTtt[CHKSUM](CR)

@	Delimiter Character
AA	The address of the module (Hex., 00 to FF)
G	Command to read the Steinhart-Hart coefficient
х	A: Read Steinhart-Hart coefficient A
	B: Read Steinhart-Hart coefficient B

- C: Read Steinhart-Hart coefficient C
- Ttt tt, the code of the user-defined type (70 to 77) for reading the Steinhart-Hart coefficient

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) Using eight-digits hexadecimal value to represent the Steinhart-Hart coefficient in IEEE-754 format
- **Note:** There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command: @03G <u>A</u> T70	Read the coefficient A of type code 70 of module 03		
Response: !03 <u>3A94030A</u>	The return value 3A94030A equals to 1.129241 x 10 ⁻³		
Command: @03G <u>B</u> T70	Read the coefficient B of type code 70 of module 03		
Response: !03 <u>39757ACF</u>	The return value 39757ACF equals to 2.341077 x 10 ⁻⁴		
	·		
Command: @03G <u>C</u> T70	Read the coefficient C of type code 70 of module 03		
Response: !03 <u>33BC73A</u> 5	The return value 33BC73A5 equals to 8.775468 x 10 ⁻⁸		

Related Commands: @AASxTttC(data), @AARTTttR(Data)

Related Sections: 3.4.1 Configuration Page - User Defined Type, B.8.1 Steinhart-Hart coefficient (User-defined Type)

@AASxTttC(data) Set the Steinhart-Hart coefficient of a user-defined type

<u>Note</u>: The command is available for the tM-TH8 module.

Syntax:

@AASxTttC(data)[CHKSUM](CR)

Delimiter Character
The address of the module (Hex., 00 to FF)
Command to set the Steinhart-Hart coefficient
A: Set Steinhart-Hart coefficient A
B: Set Steinhart-Hart coefficient B
C: Set Steinhart-Hart coefficient C
tt, the code of the user-defined type (70 to 77) for setting the Steinhart-Hart coefficient
(data), using eight-digits hexadecimal value to represent the Steinhart-Hart coefficient in

Response:

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

IEEE-754 format

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- <u>Note</u>: There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Command: @03S <u>A</u> T <u>70</u> C <u>3A94030A</u>	Set the coefficient A of type code 70 of module 03 to "3A94030A", also equals to 1.129241 x 10-3
Response: !03	The command is valid
Command: @03S <u>B</u> T <u>70</u> C <u>39757ACF</u>	Set the coefficient B of type code 70 of module 03 to "39757ACF", also equals to 2.341077 x $10^{\text{-4}}$
Response: !03	The command is valid
Command: @03S <u>C</u> T <u>70</u> C <u>33BC73A5</u>	Set the coefficient C of type code 70 of module 03 to 33BC73A5, which also equals 8.775468 x 10^{-8}
Response: !03	The command is valid

Examples:

Related Commands: @AAGxTtt, @AARTTttR(Data)

Related Sections: 3.4.1 Configuration Page - User Defined Type, B.8.1 Steinhart-Hart coefficient (User-defined Type)

@AARTTttR(Data) Read the scaling temperature based on input resistance

Description:

After configuring the Steinhart-Hart coefficients by using the <u>@AASxTttC(data)</u> command, also using this @AARTTttR(Data) command to read the temperature associated with the resistance.

<u>Note</u>: The command is available for the tM-TH8 module.

Syntax:

@AARTTttR(Data)[CHKSUM](CR)

Ø Delimiter Character
 AA The address of the module (Hex., 00 to FF)
 RT Command to read the temperature associated with the resistance
 Ttt tt, the user-defined type code (70 to 77, see Appendix B.8 Data Ranges for tM-TH8)
 R(Data) (Data), consists of seven decimal digits, or five digits, decimal point, and one additional digit, to represent the input resistance

Response:

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

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

- ! Delimiter character for a valid command
- ? Delimiter character for an invalid command
- AA The address of the module (Hex., 00 to FF)
- (Data) It consists of a +/- sign, three digits, a decimal point, and two additional digits to represent the temperature associated with the input resistance
- <u>Note:</u> There will be no response when the command syntax is incorrect, or a communication error occurs, or the specified address is not correct.

Examples:

Command:	Read the temperature associated with 104500 ohms of type 70 of	
@03 <u>RT</u> T <u>70</u> R <u>0104500</u>	module 03	
Response: !03-032.64	The return value is -32.64 degree	

Command:	Read the temperature associated with 801.2 ohms of type 70 of
@03 <u>RT</u> T <u>70</u> R <u>00801.2</u>	module 03
Response: !03+072.62	The return value is 72.62 degree

Related Commands: <u>@AAGxTtt</u>, <u>@AASxTttC(data)</u>

Related Sections:

3.4.1 Configuration Page - User Defined Type, B.8.1 Steinhart-Hart coefficient (User-defined Type)

Note: Using the <u>~AADT</u> command to set the temperature scale;

Using the <u>~AAD</u> command to read the settings.

Appendix F Quickly Save and Apply I/O Configurations

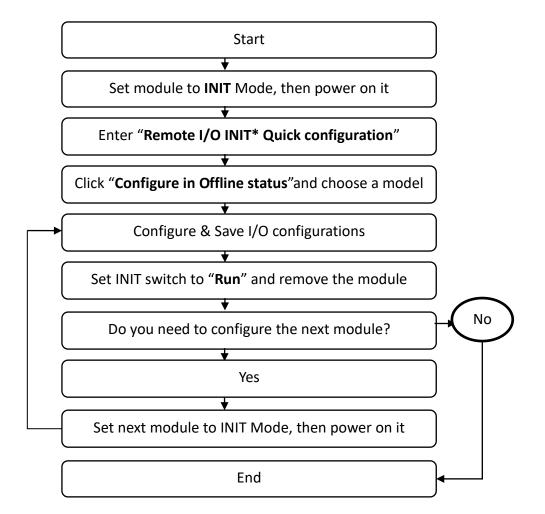
The Chapter will describe the "Remote I/O Configuration" function in DCON Utility Pro. If a case requires 250 modules with the same model to be configured, the user can only set one and then write configurations to the module one-by-one with a little bit of change. Also, save it as a file so that you can quickly write configurations to the module next time. This makes the setting to be easy.

Note:

- Configure one module at a time and turn on the module in INIT mode to allow writing settings.
- Support offline configuration. To configure and save the settings without connecting the module.
- When using a USB to RS-485 converter, make sure the driver has been installed.



Procedure of "Remote I/O INIT* Quick Configuration":



F.1 Set, Write, and Save I/O Configurations

Follow the steps below to set and write configurations to the module, then save it as a file.



Set I/O Configurations

1. Click the **Remote I/O INIT* Quick Configuration** button in DCON Utility Pro on PC.

DCON Utility Pro PC V 4.0.0.1 Beta Version						
₹ ▶ ॥ 🗄						
COM1:*	ID Address Baue Race Crieckson Format Status					

2. Choose a COM Port (Defaults = COM1), click the **Configure in Offline Status** button, and select **tM**, then choose a model.

Remote I/O INIT* Quick Configuration	About
COM Part	Please make sure the INIT* is connected to GND
Exit	tP4A4 tP4C4 tP8 tR5 tTH8

3. In the Offline Configuration window, the initial parameters of the module (e.g., tM-DA1P1R1) will be displayed on setting pages. You can change any settings.

tDA1P1R1 Firmware[0000] [Offline Configuration]						
Configuration AO	DO DI Ho	st WDT Commands Log	Summary	About		
Protocol (INIT*)	DCON	~				
Address	0 📮 [(00Н]				
Baud Rate (INIT*)	9600	~				
Parity (INIT*)	N,8,1	~				
Checksum (INIT*)	Disabled	\sim				
Analog Format	Engineering Forma	· 🗸				
Response Delay	0 ms					
Exit W	/rite Configurations f	to I/O Module	Save co	nfigurations to the file		

Note: In Offline mode, the hardware will not be changed by the settings.						
Configuration AO D	DO DI Host WDT Commands Log Summary About					
Bit Status						
CH:00						
DO Value	01h Set [Power On Value] Set [Safe Value]					
	Read DO					
	O Read Power ON Value					
	○ Read Safe Value					

Write I/O Configurations

After completing the setting, make sure that the module is connected with the PC under INIT mode and the COM Port is set (step 2). Then, continue the next step.

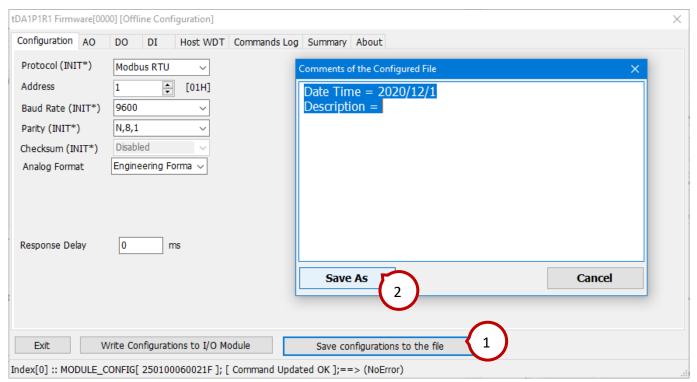
4. Click the **Write Configurations to I/O Module** button to write parameters to the I/O module and click the **OK** button.

tDA1P1R1 Firmware[000	0] [Offline Configu	uration]		
Configuration AO	DO DI H	Host WDT	Commands Log Summary About	
Protocol (INIT*) Address	Modbus RTU	∨ [01H]		
Baud Rate (INIT*)	9600	~		
Parity (INIT*)	N,8,1	~	Configure tDA1P1R1 OK	×
Checksum (INIT*)	Disabled	\sim		^
Analog Format	Engineering Form	na 🗸	"Satting Remote I/O Configurations OV:	
"Setting Remote I/O Configurations OK: Follow the instructions below: 1. Make sure the Dip Switch is set to Normal (or Run). 2. Re-power on the module to make new settings effective 3. Search for the module again and confirm the settings.				effective.
,				ОК 2
Exit Write Configurations to I/O Module Save configurations to the file				
Configure tDA1P1R1 OK==>Total Configure Command Co.				

- 5. Set the DIP switch to the **Run** position and remove the module. If you want to set the next one, keep the window open.
- 6. Install the module with the same model after setting the DIP switch to INIT and follow steps 3 to 6 to configure the module, e.g., set the address to "2".

Save I/O Configurations

7. After configuring all modules, click the **Save Configuration to the File** button and click the **Save As** button.

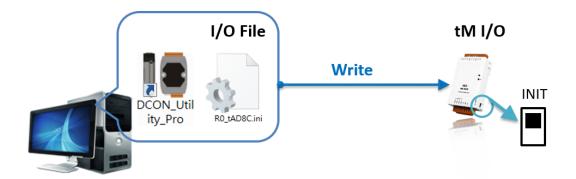


Also, create a folder according to the project name (e.g., App_01) and save it as a configuration file to be used for the next order.

Save As				×
← → ∽ ↑ 📙 « Desktop ♦ App_01	ڻ ۲	Search App_01		Q
Organize 🔻 New folder	Ň		== -	?
Name	Date modified	Туре	Size	^
No items	match your search.			~
<				>
File name R0_tDA1P1R1.ini				~
Save as type: INI ini files (*.ini)				~
∧ Hide Folders		Save	Cancel	

F.2 Configure the I/O Module by a Configuration File

The following will describe how to open the stored file and write configurations to the module.



<u>Note</u>: Only configure one module at a time under Init mode.

1. Click the **Remote I/O INIT* Quick Configuration** button in DCON Utility Pro on PC.

📜 DCON Utility Pro PC V 4.0.0.1 Beta Vers	sion	\sim
₹ ▶ ॥ ₽	СМД	
COM1:*	ID	Remote I/O INIT* Quick Configuration Address Bauu Kace Criecksum Format Status

2. In the "Remote I/O Configuration" window, choose a COM Port and click the **Configure as INIT* Status** button.

Remote I/O Configuration
Remote I/O INIT* Quick Configuration About
COM Port COM5 Please make sure the INIT* is connected to GND
Configure in Offline Status
Configure I/O by File

3. Select the configuration file (.ini) and click the **Open** button.

Open				×
$\leftrightarrow \rightarrow \cdot \uparrow$	Mesktop > App_01	ٽ v	Search App_01	Q
Organize 🔻	New folder			□ ?
ko_tDA1P1R	1.ini			
	File name: R0_tDA1P1R1.ini		Open	Cancel

Click the **Next** button under the expanded text box.

Remote I/O Configuration	
Remote I/O INIT* Quick Configuration About	
COM Port COM5 ~ Please make s	sure the INIT* is connected to GND
Configure in Offline Status	A1P1R1 ∨ 07к 087к ⊛ tм
Configure I/O by File	
View comments of the configuration	
Date Time = 2020/12/2 Description =	
Next Cancel	

4. Check or modify I/O settings and click the **Write Configurations to I/O Module** button to write configurations to the I/O module. After completing, adjust the DIP switch to the **Run** position and reboot the module to make new settings effective.

tDA1P1R1 Firmware[000	0] [Offline Configu	iration]				×
Configuration AO	DO DI H	lost WDT	Commands Log	Summary	About	
Protocol (INIT*)	Modbus RTU	~				
Address	1	01H				
Baud Rate (INIT*)	9600	~				
Parity (INIT*)	N,8,1	~				
Checksum (INIT*)	Disabled	\sim				
Analog Format	Engineering Form	na 🗸				
						_
			DO NOT char	nge the se	ettings in the ini file.	
Response Delay	0 ms		Users can mo	dify them	n in DCON Utility Pro.	
		(
Exit W	rite Configurations	s to I/O Mo	odule	Save co	nfigurations to the file	

Appendix G Save or Load I/O Project

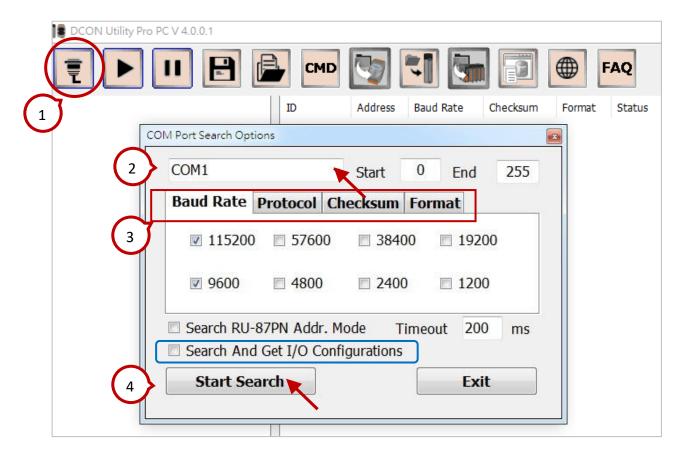
DCON Utility Pro v4.0 adds the **Save Project** and the **Load Project** functions. When many modules are required for a case, the user can search for the configured modules through different COM ports and save them as a project file, also encrypt the file by using any ZIP utility. Afterward, if the user needs to replace modules or build the same system, only load the project file to automatically compare the module settings, module names, and the COM port number to see if new modules meet the requirements of the original system for working normally.

G.1 Save Project

Follow the steps to save a project.

1. Search I/O modules.

Click the **COM Port** button and select a COM Port and conditions for searching, then click the **Start Search** button.



Note:

- Select the checkbox next to "Search and Get I/O Configurations" to search and load settings for modules. If NOT checked, also open the module configuration page to load settings.
- Using the same way to search again if there are some modules connected with other COM port.

If "Search and Get I/O Configurations" is not ticked, open the configuration page for specified modules

to load settings before saving the project.

	Address	Baud Rate	Checksum	Format	Status	Description	Comments
tTH8:04:A:0:N81:1	1[01h] 2[02h] 1H8 4[04h]	115200 115200 115200	Disabled Disabled Disabled	N,8,1 N,8,1 N,8,1	Remote I/O Remote I/O Remote I/O	[Modbus RTU]5*DO (Relay DO) [Modbus RTU]8*AI [Modbus RTU]8*AI (Universal Thermistor)	Supported Supported Supported
tR5 Firmware[A106]							×
Configuration DO	Host WDT Con	nmands Lo	g Summ	ary Ab	out		
Protocol	Modbus RTU	~					
Address	1	01H					
Baud Rate	115200	~					
Parity	N,8,1	~					
Checksum	Disabled	\sim					
Response Delay	0 ms						
						Set Module Configurations	
Exit							

2. Save the I/O Project.

Click the **Save Project** button, enter a file name, and click the **Save** button to save all modules and configurations used in the system.

₹ ▶	E 🖻 🚾 🛐	4 🔙 🗊 ●	FAQ	
COM4:* - tR5:01:A:0:N81:1 - tAD8C:02:A:0:N81:1 - tTH8:04:A:0:N81:1	tR5 1[01h] 115200 1 tAD8C 2[02h] 115200 1	Disabled N,8,1 Remote I/O Disabled N,8,1 Remote I/O	Description [Modbus RTU]5*DO (Relay DO) [Modbus RTU]8*AI [Modbus RTU]8*AI (Universal Thermist	Comments Supported Supported tor) Supported
	Save As			×
All searched COM ports will	$\leftarrow \rightarrow \checkmark \uparrow$. CCON_Utility	_Pro_PC > search > project	✓ ² Search project	م
be listed here.	Organize 🔻 New folder			EE 🔹 😮
	Name	Date modified	Type Size	
		No items match your se	arch.	
	File name: Case01			~
	Save as type:			~
	∧ Hide Folders		Save	Cancel

Next, the project comments window will be displayed. The user can add notes for the project and click OK.

Add some comments to project	x
Version = DCON Utility Pro PC V 4.0.0.1 Date Time = 10/16/2020 Description =	
ОК	Cancel

The dialog box will be displayed when saved successfully.

Saved successfully	×
Project D:\DCON_Utility_Pro_PC\search\project\Case01\ has been saved	
OK	Z

Notice:

The I/O project will be stored in the "..\DCON_Utility_Pro_PC\search\project" folder. It is recommended to back up the project on PC to avoid the user remove all older files when using a new version of DCON Utility Pro. Also, the project can be encrypted by using any ZIP utility.

Add to Archive					×
Archive: D:\DCON_Utility_P Case01.zip	ro_PC\search\proje	ct\		~	
Archive format:	zip	~	Update mode:	Add and replace files	~
Compression level:	Normal	\sim	Path mode:	Relative pathnames	~
Compression method:	Deflate	~	Options		
Dictionary size:	32 KB	~	Create SFX archive	8	
Word size:	32	\sim	Delete files after con	npression	
Solid Block size:		\sim	Encryption		
Number of CPU threads:	6 ~	/ 6	Enter password:		
Memory usage for Compressing:		195 MB	Reenter password:		
Memory usage for Decompressing	ng:	2 MB	****		
Split to volumes, bytes:		~	Show Password	ZipCrypto	~
Parameters:					
			OK Can	cel Help)

G.2 Load Project

Click the **Load Project** button and open the project file (.txt) in the project folder (e.g., Case01).

	2	СМД	I	4	P		F/
	ID	Address	Baud Rate	Checksum	Format	Status	
] © Open							×
	N_Utility_Pro_PC\	search\proje	ect\Case01 🗸 🗸	ට Search	Case01		<i>م</i>
Organize 🔻 New folder				-	• =	•	?
Name		Date mod	ified	Туре	Size		
📙 config		10/16/202	0 5:00 PM	File folder			
Case01.txt		10/16/202	0 5:00 PM	Text Document		1 KB	
File nan	ne: Case01.txt						~
				0	pen 🍗	Cance	

<u>Note:</u> After clicking the Open button, the project comments window will be displayed. Refer to the previous page.

<u>If the PC is connected with modules and the settings are correct</u>, you can see the following window after loading the I/O project.

- A. **COM Port I/O list**: Displays the COM port number and module names loaded from the project file, also comparison status for modules. The user can click each item to check the settings.
- B. **Configurations Loaded from a module**: Displays the current module settings.
- C. Configurations Loaded from file: Displays the module settings loaded from the project file.

Load Search Project file = D:\DCON_Utilit	oad Search Project file = D:\DCON_Utility_Pro_PC\search\project\Case01\Case01.txt X								
COM port I/O list	Configurations loaded from module	Configurations loaded from file							
	VERSION = A102 GET_COMMUNICATE_PARAMETER = Baud rate 115200, Format N,8,1 GET_MODBUS_MISC = Normal Mode, 60Hz Filter GET_MODBUS_DATAFORMAT = Engineering Format GET_RESPONSE_DELAY_TIME = 1ms CET_CHANNEL_EMARLE_STATUS = FFH = [62h]Type u Fenwell 0 ~ +150 °C = [62h]Type u Fenwell 0 ~ +150 °C	VERSION = A102 GET_COMMUNICATE_PARAMETER = Baud rate 115200, Format N,8,1 GET_MODBUS_MISC = Normal Mode, 60HZ Filter GET_MODBUS_DATAFORMAT = Engineering Format GET_RESPONSE_DELAY_TIME = Ims GET_CHAINNEL_ENANEE_STATUS = FFh GET_CH0_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH1_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH2_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH3_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH5_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH6_INPUF_RATURE_OFFSET = 0 GET_CH6_ITEMPERATURE_OFFSET = 0 GET_CH5_ITEMPERATURE_OFFSET = 0 GET_CH6_ITEMPERATURE_OFFSET = 0 GET_CH6_ITEMPERATURE_OFFSET = 0 GET_CH6_ITEMPERATURE_OFFSET = 0 GET_CH6_ITEMPERATURE_OFFSET = 0 GET_USER_DEFINED_TYPE_A = [70h]3386F00Bh GET_USER_DEFINED_TYPE_A = [70h]397A812Ch GET_USER_DEFINED_TYPE_A = [70h]397A812Ch GET_INPERATURE_DISPLAY_FORMAT = Celsius							
COM4;tTH8:[0,0,8,0,0,0];4[04h]];115200;Disabled;N,8,1;Remote I/O;[Modbus RTU]8*AI (Universal Thermistor);A102;3							

<u>If the PC is connected with modules but the settings are incorrect</u>, the "Settings unmatched" text will be displayed next to the module name and the error will be marked after loading the I/O project. The user can close the window and modify the settings.

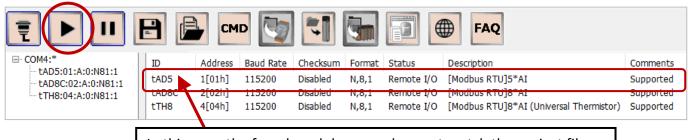
oad Search Project file = D:\DCON_Utility_Pro_PC\search\project\Case01\Case01.txt X								
COM port I/O list	Configurations loaded from module	Configurations loaded from file						
COM4	VERSION = A106 GET_COMMUNICATE_PARAMETER = Baud rate 115200, Format N,8,1 GET_DO_POWER_ON = 2h GET_DO_SAFE_VALUE = 6h GET_WDT_TIMER = 255ms GET_WDT_ENABLE = Disabled GET_WDT_ENABLE = Disabled ST_WDT_OVERWRITE = Enabled ST_GET_RESPONSE_DELAY_TIME = 2ms	VERSION = A106 GET_COMMUNICATE_PARAMETER = Baud rate 115200, Format N,8,1 GET_DO_POWER_ON = 2h GET_DO_SAFE_VALUE = 6h GET_WDT_TIMER = 255ms GET_WDT_ENABLE = Disabled GET_WDT_OVERWRITE = Enabled >> GET_RESPONSE_DELAY_TIME = 0ms						
C -tR5:01:A:0:N81:1:Set	ttings unmatched e I/O;[Modbus RTU]5*DO (Rela	y DO);A106;1						
-tAD8C:02:A:0:N81:1:	Matched							
-tTH8:04:A:0:N81:1:M	latched							

Change the settings to be the same with the project file, in this case, set the Response Delay to 0 ms.

₹ ▶ ॥ ₽		смд			9	FAC	2	
COM4:*	ID	Address	Baud Rate	Checksum	Format	Status	Description	Comments
-tR5:01:A:0:N81:1:Settings unn -tAD8C:02:A:0:N81:1:Matched	tR		115200	Disabled	N,8,1	Remote I/O	[Modbus RTU]5*DO (Relay DO)	Settings unmatche
-tTH8:04:A:0:N81:1:Matched		TH8 [04h]	115200	Disabled Disabled	N,8,1 N,8,1	Remote I/O	[Modbus RTU]8*AI [Modbus RTU]8*AI (Universal Thermistor)	Matched Matched
	>							
Clear								
tR5 Firmware[A106]								~
tKS Firmware[A100]								^
Configuration DO	Host W	DT Commar	nds Log	Summa	ry Ał	oout		
Protocol	Modbus	RTU 🕔	7					
Address	1	÷ 01	н					
Baud Rate	115200) ``	~					
Darity	N,8,1	```	7					
Parity	11,0,1							

Change it from 2ms to 0ms, and click the Set.... Button.
Response Delay

<u>If the PC is connected with modules but the communication settings or module names are incorrect</u>, the "Module not found" or "Module unmatched" text will be displayed next to the module name and only the loaded module settings can be reviewed.



In this case, the found module name does not match the project file.

COM port I/O list	Configurations loaded from module	Configurations loaded from file
COM4 -tAD8C:02:A:0:N81:1:(tAD5)Module unmaterial -tAD8C:02:A:0:N81:1:Matched -tTH8:04:A:0:N81:1:Matched	atched	VERSION = A106 GET_COMMUNICATE_PARAMETER = Baud rate 115200, Format N,8,1 GET_DO_POWER_ON = 2h GET_DO_SAFE_VALUE = 6h GET_WDT_TIMER = 255ms
	tM-R5 is not found.	GET_WDT_ENABLE = Disabled GET_WDT_OVERWRITE = Enabled GET_RESPONSE_DELAY_TIME = 0ms
COM4 	- Jhur DTUI	5*DO (Relay DO) The settings of the tM-R5

After closing the "Load..." window, the user can see the correct module name but cannot open the setting page. Please replace the correct module and set the communication parameters properly, then execute the Load Project to do the comparison.

DCON Utility Pro PC V 4.0.0.1 Load project I	D:\DCON_Utility_Pro_	PC\search\p	roject\Case0		1.txt	Q	
 tR5:01:A:0:N81:1:[tAD5]Module unm tAD8C:02:A:0:N81:1:Matched tTH8:04:A:0:N81:1:Matched 	ID Address 1[01h] 4[02h] 4[02h] 4[04h]	Baud Rate 115200 115200 115200	Checksum Disabled Disabled Disabled	Format N,8,1 N,8,1 N,8,1	Status Remote I/O Remote I/O Remote I/O	[Modbus RTU]8*AI	Comments [tAD5]Module unmatched Matched Matched
		Can not load tR5 form X Found [tAD5] Module unmatched				The correct module na prompted here, but the page cannot be opened	e setting
			ā	確定			

If the PC is not connected with the I/O module, the "Not exist" text will be displayed next to the module name and only the loaded module settings can be reviewed.

🛢 DCON Utility Pro PC V 4.0.0.1	
₹ ▶ ॥ 🗄	
	Load Project 10 Address Baud Rate Checksum Format Status

Also, click each module name to view the settings.

COM port I/O list	Configurations loaded from module	Configurations loaded from file
B-COM4:Not exist tR5:01:A:0:N81:1:Port 4 not exist tAD8C:02:A:0:N81:1:Port 4 not exist - tTH8:04:A:0:N81:1:Port 4 not exist COM4:Not exist - COM4:Not exist - tR5:01:A:0:N81:1:P - tR5:01:A:0:N81:1:P - tAD8C:02:A:0:N81:1:P - tTH8:04:A:0:N81:1:P	l:Port 4 not exist	VERSION = A102 GET_COMMUNICATE_PARAMETER = Baud rate 115200, Format N,8,3 GET_MODBUS_MISC = Normal Mode, 60Hz Filter GET_MODBUS_DATAFORMAT = Engineering Format GET_RESPONSE_DELAY_TIME = 1ms GET_CHA_INPLT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH1_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH3_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH4_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH6_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH7_INPUT_RANGE = [62h]Type u Fenwell 0 ~ +150 °C GET_CH6_INPUT_RANGE = [62h]Type u Fe
0M4;tTH8:[0,0,8,0,0,0];4[04h];11	5200;Disabled;N,8,1;Remote I/O;[Modbus F	The module settings in the project file.

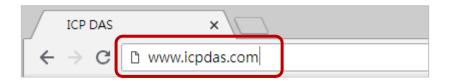
After closing the "Load..." window, the user can see the correct module names. Please connect modules and set the communication parameters properly, then execute the Load Project to do the comparison.

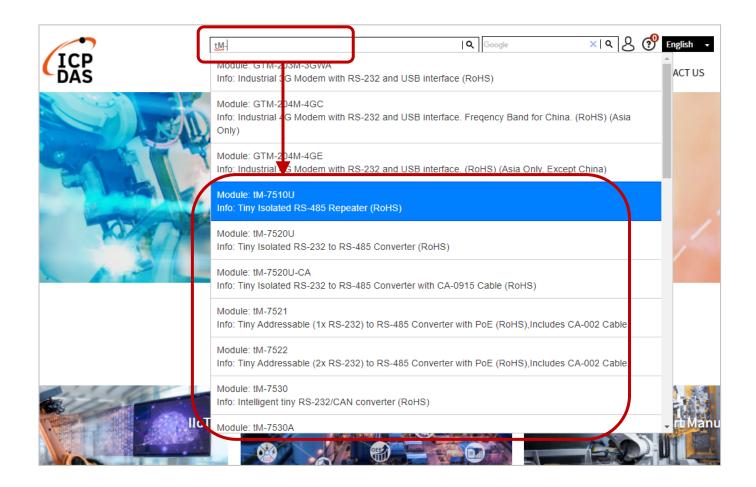
📜 DCON Utility Pro PC V 4.0.0.1 Load project 🛛	D:\DCON	_Utility_Pro	o_PC\search\	project\Cas	e01\Case	01.txt		×
🗧 🕨 🗉 🖻 🖨	СМІ	• 🔄			9	⊕ F4	AQ	
⊡ COM4:Not exist	ID	Addrose	Raud Rate	Checksum	Format	Statue	Description	Comments
	tR5 tAD8C tTH8	1[01h] 2[02h] 4[04h]	115200 115200 115200	Disabled Disabled Disabled	N,8,1 N,8,1 N,8,1	Remote I/O	[Modbus RTU]5*DO (Relay DO) [Modbus RTU]8*AI [Modbus RTU]8*AI (Universal Thermistor)	Port 4 not exist Port 4 not exist Port 4 not exist
Clear						ne will b not be c	e prompted here, opened.	

Appendix H Other Information

H.1 How do I Search the Product Webpage?

Enter the keyword in the Search box on the ICP DAS website (<u>www.icpdas.com</u>) to find out the product information.

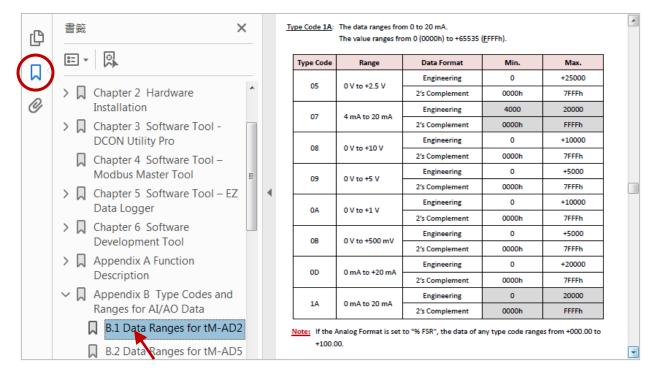




H.2 How do I Quickly Navigate a Specific Chapter?

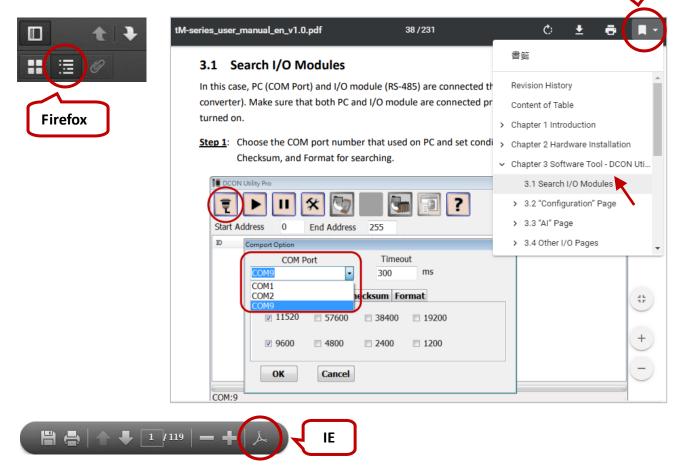
1) Read the PDF file on PC

Click the Bookmark icon on the left-side panel, and then click any heading with the chapter number.



2) Read the PDF file in a browser

Depending on the browser, find out the Bookmark icon at the top of the PDF file or just click on the page to show the toolbar.



Chrome