
FnIO G - Series:

GL-9073

GL-9073 (MODBUS RS485 Network Adapter)

Table of Contents

1.Environment Specification.....	5
2.GL-9073 (Modbus RTU Network Adaptor).....	6
2.1.GL-9073 Specification.....	6
2.2.GL-9073 Wiring Diagram.....	7
2.3.GL-9073 LED Indicator.....	8
2.3.1.LED Indicator.....	8
2.3.2.MOD (Module Status LED).....	8
2.3.3.RXD (Receive Data LED).....	8
2.3.4.TXD (Transmit Data LED).....	8
2.3.5.IOS LED (Extension Module Status LED).....	8
2.4.GL-9073 Electrical Interface.....	9
2.4.1.RJ-45 Socket.....	9
2.4.2.Dip Switch.....	9
2.5.I/O Process Image Map.....	10
2.5.1.MODBUS Interface Register / Bit Map.....	10
2.5.2.Example of Input Process Image (Input Register) Map.....	11
2.5.3.Example of Output Process Image (Output Register) Map.....	12
3.MODBUS INTERFACE.....	13
3.1.MODBUS Interface Register/Bit Map.....	13
3.2.Supported MODBUS Function Codes.....	13
3.3.MODBUS Transmission Mode.....	14
3.3.1.RTU Transmission Mode.....	14
3.3.2.ASCII Transmission Mode.....	14
3.3.3.1 (0x01) Read Coils.....	14
3.3.3.2 (0x02) Read Discrete Inputs.....	15
3.3.3.3 (0x03) Read Holding Registers.....	15
3.3.3.4 (0x04) Read Input Registers.....	16
3.3.3.5 (0x05) Write Single Coil.....	16
3.3.3.6 (0x06) Write Single Register.....	17
3.3.3.8 (0x08) Diagnostics.....	17
3.3.10.15 (0x0F) Write Multiple Coils.....	19
3.3.11.16 (0x10) Write Multiple Registers.....	20
3.3.12.23 (0x17) Read/Write Multiple Registers.....	21
3.3.13.Error Response.....	22
3.4.MODBUS Special Register Map.....	23

3.4.1.Adapter Identification Special Register (0x1000, 4096).....	23
3.4.2.Adapter Watchdog Time, other Time Special Register (0x1020, 4128).....	23
3.4.3.Adapter Information Special Register (0x1100, 4352).....	24
3.4.4.Expansion Slot Information Special Resister (0x2000, 8192).....	25
3.5.Supported MODBUS Function Codes.....	26

History

Rev	Pages	Remarks	Date	Editor
1.00			Jan 6, 2020	Jy bae
		Image changed, UL Update	April 1, 2020	Joonho, Park
1.01	24	Modbus special register map Update(0x1119)	July 07, 2020	Joonho, Park
1.02	5, 7, 12, 13	Document Number Added General Specification Edited Wiring Diagram Updated Input/Output Process Image Address Edited	Jun. 19, 2023	Seonghyeon, Park

1. Environment Specification

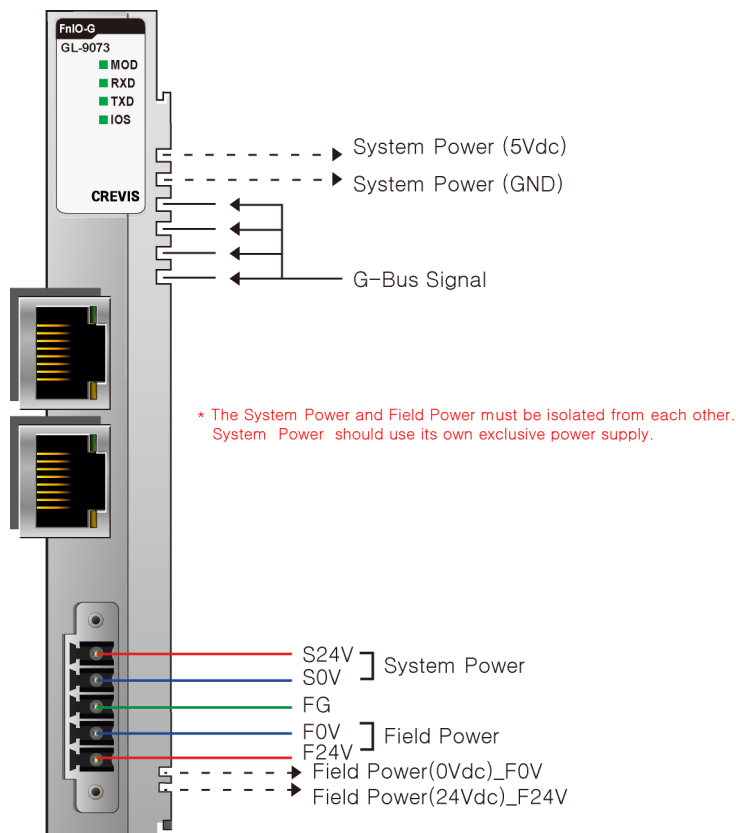
Environmental specification	
Operating Temperature	-20°C~60°C : 1.0A full load is allowed.
UL Temperature	-20°C~60°C
Storage Temperature	-40°C~85°C
Relative Humidity	5% ~ 90% non-condensing
Mounting	DIN rail
General specification	
Shock Operating	IEC 60068-2-27
Vibration Resistance	IEC 60068-2-6, 4g
Industrial Emissions	EN 61000-6-4/A11 : 2011
Industrial Immunity	EN 61000-6-2 : 2019
Installation Position	Vertical and horizontal installation is available.
Product Certifications	CE, UL

2. GL-9073 (Modbus RTU Network Adapter)

2.1. GL-9073 Specification

Items	Specification
Communication Specification	
Adapter Type	Slave node (MODBUS Serial RTU/ASCII Server)
Protocol	MODBUS RTU and ASCII
Max. Expansion Module	16 slots
Max. Input / Output Data Size	Max. Input 256 bytes / Output 256 bytes
Max Length Bus Line	500m
Max. Nodes	8 nodes
Baud Rate	2400, 4800, 9600, 19200, 38400, 57600, 115200 bps
Interface Connector	RJ-45 socket * 2pcs
Settable Node Address	Via Dip switch
Indicator	4 LEDs 1 Green/Red, Module Status (MOD) 1 Green, Receive Data (RXD) 1 Green, Transmit Data (TXD) 1 Green/Red, Expansion Module Status (IOS)
Module Location	Starter module left side of G-Series system
General specification	
UL System Power	Supply voltage : 24Vdc nominal, Class 2
System Power	Supply voltage : 24Vdc nominal Supply voltage range : 15~28.8Vdc Reverse polarity protection
Power Dissipation	20mA typical @ 24Vdc
Current for I/O Module	1.0A @ 5Vdc
Isolation	System power to internal logic : Non-isolation System power I/O driver : Isolation
UL Field Power	Supply voltage : 24Vdc nominal, Class 2
Field Power	Supply voltage : 24Vdc typical (Max. 28.8Vdc) * Field Power Range is different depending on IO Module series. Refer to IO Module's Specification.
Max. Current Field Power Contact	DC 8A Max
Wiring	I/O Cable Max. 2.0mm ² (AWG 14)
Weight	77g
Module size	22mm x 109mm x 70mm
Environment Condition	Refer to '1. Environment Specification'

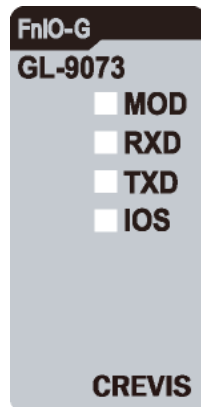
2.2. GL-9073 Wiring Diagram



Pin No.	Signal Description
1	System Power, 24V
2	System Power, Ground
3	F.G
4	Field Power, Ground
5	Field Power, 24V

2.3. GL-9073 LED Indicator

2.3.1. LED Indicator



LED No.	LED Function / Description	LED Color
MOD	Module Status	Green/Red
RXD	Receive Data	Green
TXD	Transmit Data	Green
IOS	Extension Module Status	Green/Red

2.3.2. MOD (Module Status LED)

Status	LED	To indicate
Not Powered	OFF	Power is not supplied to the unit.
Device Operational	Green	The unit is operating in normal condition.
Unrecoverable Fault	Red	The device has an unrecoverable fault. - Memory error or CPU watchdog error.

2.3.3. RXD (Receive Data LED)

Status	LED	To indicate
Not Powered or Not Linked	OFF	Device may not be powered
Adapter physical connected	Green	Adapter Ethernet Controller physically connected

2.3.4. TXD (Transmit Data LED)

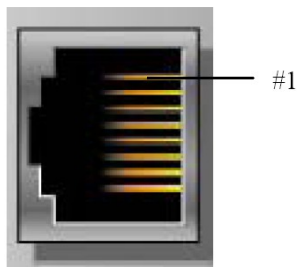
Status	LED	To indicate
Not Powered	OFF	Device is idle or may not be powered.
Adapter transmit frame	Flashing Green	Adapter(slave) transmit frame. About 20msec flashing.

2.3.5. IOS LED (Extension Module Status LED)

Status	LED	To indicate
Not Powered	OFF	Device may not be powered.
No Expansion Module	Flashing Red	Adapter has no expansion module
Internal Bus Connection, Run Exchanging I/O	Green	Exchanging I/O data.
Expansion Configuration Failed	Red	One or more expansion module occurred in fault state. - Detected invalid expansion module ID. - Overflowed Input/Output Size - Too many expansion module - Initialization failure - Communication failure. - Changed expansion module configuration. - Mismatch vendor code between adapter and expansion module.

2.4. GL-9073 Electrical Interface

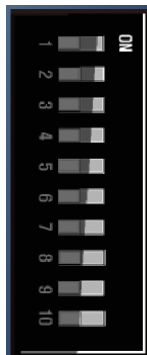
2.4.1. RJ-45 Socket



Shielded RJ-45 Socket

RJ-45	Signal Name	Description
1	RS485+	Transmit +
2	RS485-	Transmit -
3	GND	Communicaion Ground
4	-	
5	-	
6	-	
7	-	
8	FG	Frame Ground
Case	Shield	

2.4.2. Dip Switch



Item	Item set up	DIP Switch									
		#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Node ID	Eeprom Value*	OFF	OFF	OFF							
	Node ID 1	ON	OFF	OFF							
	Node ID 7	ON	ON	ON							
Baud rate	115200bps				OFF	OFF	OFF				
	2400bps				ON	OFF	OFF				
	4800bps				OFF	ON	OFF				
	9600bps				ON	ON	OFF				
	19200bps				OFF	OFF	ON				
	38400bps				ON	OFF	ON				
	57600bps				OFF	ON	ON				
Byte Format	115200bps				ON	ON	ON				
	8bit, No parity, 1Stop							OFF	OFF	OFF	
	8bit, Even Parity, 1Stop							ON	OFF	OFF	
	8bit, Odd Parity, 2Stop							OFF	ON	OFF	
	8bit, No Parity, 2Stop							ON	ON	OFF	
	7bit, Even Parity, 1Stop**							ON	OFF	ON	
	7bit, Odd Parity, 1Stop**							OFF	ON	ON	
RTU/ ASCII Mode	8bit, No Parity, 1Stop							ON	ON	ON	
	RTU Mode										OFF
	ASCII Mode										ON

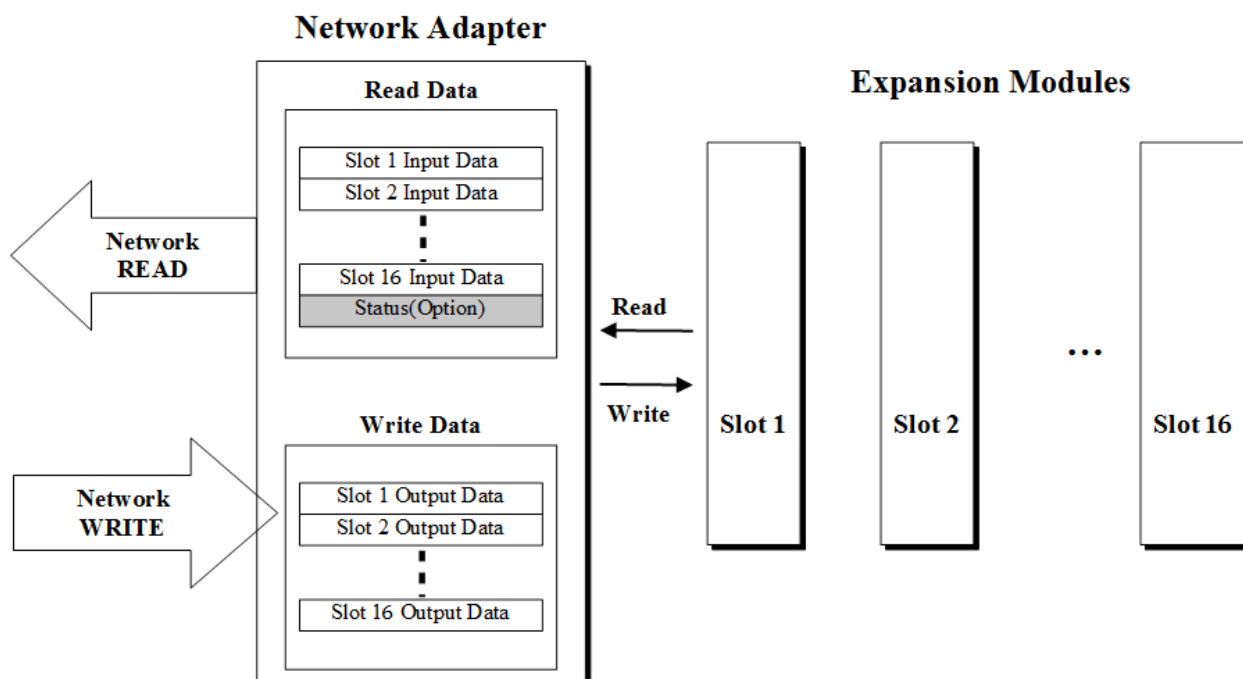
* Factory default value is 1.

** ASCII Mode is only available

2.5. I/O Processe Image Map

An expansion module may have 3 types of data as I/O data, configuration parameter and memory register.

The data exchange between network adapter and expansion modules is done via an I/O process image data by G-Series protocol. The following figure shows the data flow of process image between network adapter and expansion modules



2.5.1. MODBUS Interface Register / Bit Map

• Register Map

Start Address	Read / Write	Description	Func, Code
0x0000 ~	Read	Process input image registers (Real Input Register)	3,4,23
0x0800 ~	Read / Write	Process output image registers (Real Output Register)	3,16,23
0x1000 ~	Read	Adapter Identification special registers.	3,4,23
0x1020 ~	Read / Write	Adapter Watchdog, other time special register.	3,4,6,16,23
0x1100 ~	Read / Write	Adapter Information special registers.	3,4,6,16,23
0x2000 ~	Read / Write	Expansion Slot Information special registers.	3,4,6,16,23

* The special register map must be accessed by read/write of every each address (one address).

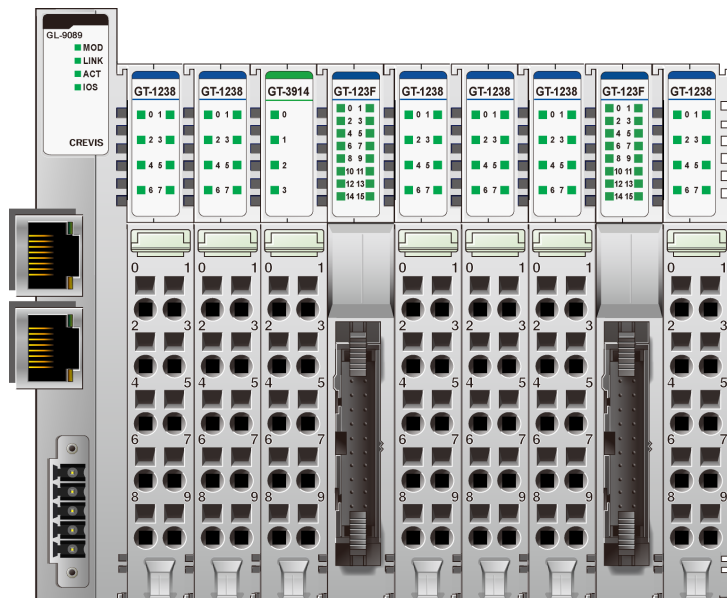
• Register Map

Start Address	Read / Write	Description	Func, Code
0x0000 ~	Read	Process input image bits All input registers area are addressable by bit address. Size of input image bit is size of input image register * 16.	2
0x1000 ~	Read / Write	Process output image bits All output registers area are addressable by bit address. Size of output image bit is size of output image register * 16.	1,5,15

2.5.2. Example of Input Process Image (Input Register) Map

Input image data depends on slot position and expansion slot data type. Input process image data is only ordered by expansion slot position

• For exmaple slot configuration



Slot No.	Module Description
#0	MODBUS RS485 Adapter
#1	8-discrete input
#2	8-discrete input
#3	4-analog input
#4	16-discrete input
#5	8-discrete input
#6	8-discrete input
#7	8-discrete input
#8	16-discrete input
#9	8-discrete input

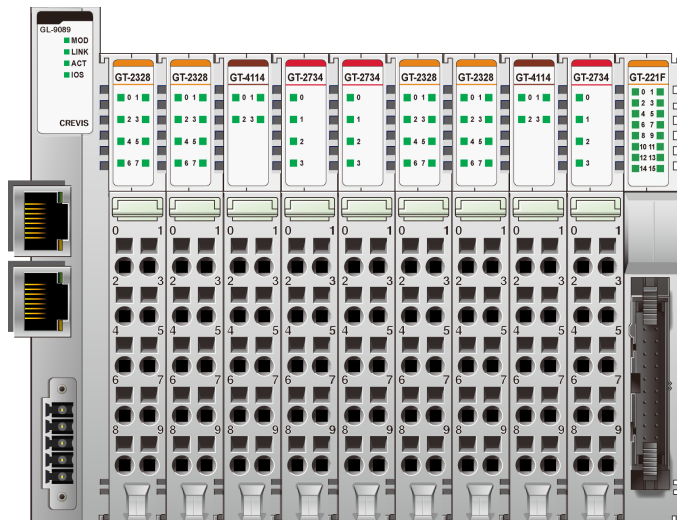
• Input Process Image

Address	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
0x0000	Discrete Input 8 pts (Slot#2)								Discrete Input 8 pts (Slot#1)							
0x0001	Analog Input Ch0 high byte (Slot#3)								Analog Input Ch0 low byte (Slot#3)							
0x0002	Analog Input Ch1 high byte (Slot#3)								Analog Input Ch1 low byte (Slot#3)							
0x0003	Analog Input Ch2 high byte (Slot#3)								Analog Input Ch2 low byte (Slot#3)							
0x0004	Analog Input Ch3 high byte (Slot#3)								Analog Input Ch3 low byte (Slot#3)							
0x0005	Discrete Input 8 pts (Slot#4)								Discrete Input 8 pts (Slot#4)							
0x0006	Discrete Input 8 pts (Slot#6)								Discrete Input 8 pts (Slot#5)							
0x0007	Discrete Input 8 pts (Slot#8)								Discrete Input 8 pts (Slot#7)							
0x0008	Discrete Input 8 pts (Slot#9)								Discrete Input 8 pts (Slot#8)							

2.5.3. Example of Output Process Image (Output Register) Map

Output image data depends on slot position and expansion slot data type. Output process image data is only ordered by expansion slot position.

• For exmaple slot configuration



Slot No.	Module Description
#0	MODBUS/TCP Adapter
#1	8-discrete output
#2	8-discrete output
#3	4-analog output
#4	4- relay output
#5	4-relay output
#6	8-discrete output
#7	8-discrete output
#8	4-analog output
#9	4-relay output
#10	16-discrete output

• Output Process Image

Address	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
0x0800	Discrete Input 8 pts (Slot#2)								Discrete Input 8 pts (Slot#1)							
0x0801	Analog Input Ch0 high byte (Slot#3)								Analog Input Ch0 low byte (Slot#3)							
0x0802	Analog Input Ch1 high byte (Slot#3)								Analog Input Ch1 low byte (Slot#3)							
0x0803	Analog Input Ch2 high byte (Slot#3)								Analog Input Ch2 low byte (Slot#3)							
0x0804	Analog Input Ch3 high byte (Slot#3)								Analog Input Ch3 low byte (Slot#3)							
0x0805	Discrete Input 8 pts (Slot#4)								Discrete Input 8 pts (Slot#4)							
0x0806	Discrete Input 8 pts (Slot#6)								Discrete Input 8 pts (Slot#5)							
0x0807	Discrete Input 8 pts (Slot#8)								Discrete Input 8 pts (Slot#7)							
0x0808	Discrete Input 8 pts (Slot#9)								Discrete Input 8 pts (Slot#8)							

3. MODBUS INTERFACE

3.1. MODBUS Interface Register/Bit Map

- Register Map

Start Address	Read/Write	Description	Func. Code
0x0000 ~	Read	Process input image registers (Real Input Register)	3,4,23
0x0800 ~	Read/Write	Process output image registers (Real Output Register)	3,16,23
0x1000 *	Read	Adapter Identification special registers.	3,4,23
0x1020 *	Read/Write	Adapter Watchdog, other time special register.	3,4,6,16,23
0x1100 *	Read/Write	Adapter Information special registers.	3,4,6,16,23
0x2000 *	Read/Write	Expansion Slot Information special registers.	3,4,6,16,23

* The special register map must be accessed by read/write of every each address (one address).

- Register Map

Start Address	Read/Write	Description	Func. Code
0x0000~	Read	Process input image bits All input registers area are addressable by bit address. Size of input image bit is size of input image register * 16.	2
0x1000~	Read/Write	Process output image bits All output registers area are addressable by bit address. Size of output image bit is size of output image register * 16.	1,5,15

3.2. Supported MODBUS Function Codes

Function Code	Function	Description
1(0x01)	Read Coils	Read output bit
2(0x02)	Read Discrete Inputs	Read input bit
3(0x03)	Read Holding Registers	Read output word
4(0x04)	Read Input Registers	Read input word
5(0x05)	Write Single Coil	Write one bit output
6(0x06)	Write Single Register	Write one word output
8(0x08)	Diagnostics	Read diagnostic register
15(0x0F)	Write Multiple Coils	Write a number of output bits
16(0x10)	Write Multiple registers	Write a number of output words
23(0x17)	Read/Write Multiple registers	Read a number of input words /Write a number of output words

- Refer to MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1a

3.3. MODBUS Transmission Mode

Two different serial transmission modes are defined : The RTU mode and the ASCII mode. It defines the bit contents of message fields transmitted serially on the line. It determines how information is packed into the message fields and decoded.

3.3.1. RTU Transmission Mode

When devices communicate on a MODBUS serial line using the RTU (Remote Terminal Unit) mode, each 8-bit byte in a message contains two 4-bit hexadecimal characters. The main advantage of this mode is that its greater character density allows better data throughput than ASCII mode for the same baud rate. Each message must be transmitted in a continuous stream of characters.

Start	Address	Function	Data	CRC Check	End
≥ 3.5 char	1 char	1 char	Up to 252 chars(s)	2 chars	≥ 3.5 char

3.3.2. ASCII Transmission Mode

When devices are setup to communicate on a MODBUS serial line using ASCII (American Standard Code for Information Interchange) mode, each 8-bit byte in a message is sent as two ASCII characters. This mode is used when the physical communication link or the capabilities of the device does not allow the conformance with RTU mode requirements regarding timers management.

Start	Address	Function	Data	LRC Check	End
1 char “,”	2 chars	2 chars	Up to 2x252 char(s)	2 chars	2 chars CR,LF

3.3.3. 1 (0x01) Read Coils

This function code is used to read from 1 to 2000 contiguous status of coils in a remote device. The Request PDU specifies the starting address, i.e. the address of the first coil specified, and the number of coils. In the PDU Coils are addressed starting at zero. Therefore coils numbered 1-16 are addressed as 0-15. The coils in the response message are packed as one coil per bit of the data field. Status is indicated as 1= ON and 0= OFF.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	-	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x01	0x01	“01”	0x30, 0x31
Starting Address Hi	0x10	0x10	“10”	0x31, 0x30
Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Outputs Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Outputs Lo	0x10	0x10	“10”	0x31, 0x30
Error Check (CRC/LRC)	-	0x31, 0x44	“7C”	0x37, 0x43
End of Frame	-	t1-t2-t3	CR, LF	0x0D, 0x0A

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x01	0x01	“01”	0x30, 0x31
Byte Count	0x02	0x02	“02”	0x30, 0x32
Output Status	0x00	0x00	“00”	0x30, 0x30
Output Status	0x00	0x00	“00”	0x30, 0x30
Error Check (CRC/LRC)	---	0x40, 0x34	“9A”	0x39, 0x41
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0x0A

3.3.4. 2 (0x02) Read Discrete Inputs

This function code is used to read from 1 to 2000 contiguous status of discrete inputs in a remote device. The Request PDU specifies the starting address, i.e. the address of the first input specified, and the number of inputs. In the PDU Discrete Inputs are addressed starting at zero. Therefore Discrete inputs numbered 1-16 are addressed as 0-15.

The discrete inputs in the response message are packed as one input per bit of the data field.

Status is indicated as 1= ON; 0= OFF.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“07”	0x36, 0x33
Function Code	0x02	0x02	“02”	0x30, 0x32
Starting Address Hi	0x00	0x00	“00”	0x30, 0x30
Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Inputs Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Inputs Lo	0x10	0x10	“0A”	0x31, 0x30
Error Check (CRC/LRC)	---	0x71, 0x84	“ED”	0x38, 0x42
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x02	0x02	“02”	0x30, 0x32
Byte Count	0x02	0x02	“02”	0x30, 0x32
Input Status	0x00	0x00	“00”	0x30, 0x30
Input Status	0x00	0x00	“00”	0x30, 0x30
Error Check (CRC/LRC)	---	0x40, 0x70	“99”	0x39, 0x39
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

3.3.5. 3 (0x03) Read Holding Registers

This function code is used to read the contents of a contiguous block of holding registers in a remote device. The Request PDU specifies the starting register address and the number of registers.

The register data in the response message are packed as two bytes per register, with the binary contents right justified within each byte. For each register, the first byte contains the high order bits and the second contains the low order bits.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x03	0x03	“03”	0x30, 0x33
Starting Address Hi	0x10	0x10	“10”	0x31, 0x30
Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Register Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Register Lo	0x01	0x01	“01”	0x30, 0x31
Error Check (CRC/LRC)	---	0x88, 0x88	“89”	0x38, 0x39
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0x0A

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x03	0x03	“03”	0x30, 0x33
Byte Count	0x02	0x02	“02”	0x30, 0x32
Output Register#0 Hi	0x02	0x02	“02”	0x30, 0x32
Output Register#0 Lo	0xE5	0xE5	“E5”	0x45, 0x35
Error Check (CRC/LRC)	---	0x81, 0x67	“B1”	0x42, 0x31
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0x0A

- In case of address 0x0800, 0x0801 output register value: 0x1122, 0x3344.

3.3.6. 4 (0x04) Read Input Registers

This function code is used to read from 1 to approx. 125 contiguous input registers in a remote device. The Request PDU specifies the starting register address and the number of registers. The register data in the response message are packed as two bytes per register, with the binary contents right justified within each byte. For each register, the first byte contains the high order bits and the second contains the low order bits.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x04	0x04	“04”	0x30, 0x34
Starting Address Hi	0x10	0x10	“10”	0x31, 0x30
Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Register Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Register Lo	0x01	0x01	“01”	0x30, 0x31
Error Check (CRC/LRC)	---	0x3D, 0x48	“88”	0x38, 0x38
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0x0A

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x04	0x04	“04”	0x30, 0x34
Byte Count	0x02	0x02	“02”	0x30, 0x32
Input Register#0 Hi	0x02	0x02	“02”	0x30, 0x32
Input Register#0 Lo	0xE5	0xE5	“E5”	0x45, 0x35
Error Check (CRC/LRC)	---	0x80, 0x13	“B0”	0x42, 0x30
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

- In case of address 0x0000, 0x0001 input register value: 0x0080, 0x0000.

3.3.7. 5 (0x05) Write Single Coil

This function code is used to write a single output to either ON or OFF in a remote device. The requested ON/OFF state is specified by a constant in the request data field. A value of FF 00 hex requests the output to be ON. A value of 00 00 requests it to be OFF. All other values are illegal and will not affect the output.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“36”	0x36, 0x33
Function Code	0x05	0x05	“05”	0x30, 0x35
Output Address Hi	0x10	0x10	“10”	0x31, 0x30
Output Address Lo	0x00	0x00	“00”	0x30, 0x30
Output Value Hi	0xFF	0xFF	“FF”	0x46, 0x46
Output Value Lo	0x00	0x00	“00”	0x30, 0x30
Error Check (CRC/LRC)	---	0x80, 0xB8	“8Y”	0x38, 0x59
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“36”	0x36, 0x33
Function Code	0x05	0x05	“05”	0x30, 0x35
Output Address Hi	0x10	0x10	“10”	0x31, 0x30
Output Address Lo	0x00	0x00	“00”	0x30, 0x30
Output Value Hi	0xFF	0xFF	“FF”	0x46, 0x46
Output Value Lo	0x00	0x00	“00”	0x30, 0x30
Error Check (CRC/LRC)	---	0x80, 0xB8	“8Y”	0x38, 0x59
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

- Output bit of address 0x1001 turns ON.

3.3.8. 6 (0x06) Write Single Register

This function code is used to write a single holding register in a remote device. Therefore register numbered 1 is addressed as 0. The normal response is an echo of the request, returned after the register contents have been written.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x06	0x06	“06”	0x30, 0x36
Register Address Hi	0x08	0x08	“08”	0x30, 0x38
Register Address Lo	0x00	0x00	“00”	0x30, 0x30
Register Value Hi	0x00	0x00	“00”	0x30, 0x30
Register Value Lo	0xFF	0xFF	“FF”	0x46, 0x46
Error Check (CRC/LRC)	---	0xC3, 0xA8	“90”	0x39, 0x30
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x06	0x06	“06”	0x30, 0x36
Register Address Hi	0x08	0x08	“08”	0x30, 0x38
Register Address Lo	0x00	0x00	“00”	0x30, 0x30
Register Value Hi	0x00	0x00	“00”	0x30, 0x30
Register Value Lo	0xFF	0xFF	“FF”	0x46, 0x46
Error Check (CRC/LRC)	---	0xC3, 0xA8	“90”	0x39, 0x30
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

- In case of address 0x0800 output register value: 0x0000 changes to 0x1122.

3.3.9. 8 (0x08) Diagnostics

MODBUS function code 08 provides a series of tests for checking the communication system between a client (Master) device and a server (Slave), or for checking various internal error conditions within a server.

The function uses a two-byte sub-function code field in the query to define the type of test to be performed. The server echoes both the function code and sub-function code in a normal response. Some of the diagnostics cause data to be returned from the remote device in the data field of a normal response.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x07	0x07	“07”	0x30, 0x37
Function Code	0x08	0x08	“08”	0x30, 0x38
Sub-Function Hi	0x00	0x00	“00”	0x30, 0x30
Sub-Function Lo	0x00	0x00	“00”	0x30, 0x30
Data Hi	0x11	0x11	“11”	0x31, 0x31
Data Lo	0x22	0x22	“22”	0x32, 0x32
Error Check (CRC/LRC)	---	0x6C, 0x24	“BE”	0x42, 0x45
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x07	0x07	“07”	0x30, 0x37
Function Code	0x08	0x08	“08”	0x30, 0x38
Sub-Function Hi	0x00	0x00	“00”	0x30, 0x30
Sub-Function Lo	0x00	0x00	“00”	0x30, 0x30
Data Hi	0x11	0x11	“11”	0x31, 0x31
Data Lo	0x22	0x22	“22”	0x32, 0x32
Error Check (CRC/LRC)	---	0x6C, 0x24	“BE”	0x42, 0x45
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

Sub-function 0x0000(0) Return Query Data

The data passed in the request data field is to be returned (looped back) in the response.
The entire response message should be identical to the request.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0000(0)	Any	Echo Request Data	

Sub-function 0x0001(1) Restart Communications Option

The remote device could be initialized and restarted, and all of its communications event counters are cleared.
Especially, data field 0x55AA make the remote device to restart with factory default setup of EEPROM.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0001(1)	0x0000 or 0xFF00	Echo Request Data	Reset
0x0001(1)	0x55AA+0xAB7B+Sumcheck	Echo Request Data	Reset with Factory default ¹⁾
0x0001(1)	0x55AA+0xAA55+Sumcheck	Echo Request Data	Reset with Factory default ²⁾

1),2) IP Address, Subnet Mask Address, Gateway Address will be the factory defaults value.

2) Mac Address will be the factory default value.

Sub-function 0x000A(10) Clear Counters and Diagnostic Register

The goal is to clear all counters and the diagnostic register. Counters are also cleared upon power-up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000A(10)	0x0000	Echo Request Data	

Sub-function 0x000B(11) Return Bus Message Count

The response data field returns the quantity of messages that the remote device has detected on the communications system since its last restart, clear counters operation, or power-up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000B(11)	0x0000	Total Message Count	

Sub-function 0x000C(12) Return Bus Communication Error Count

The response data field returns the quantity of CRC errors encountered by the remote device since its last restart, clear counters operation, or power-up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000C(12)	0x0000	CRC Error Count	

Sub-function 0x000D(13) Return Bus Exception Error Count

The response data field returns the quantity of MODBUS exception responses returned by the remote device since its last restart, clear counters operation, or power-up.

Exception responses are described and listed in section 3.2.11.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000D(13)	0x0000	Exception Error Count	

Sub-function 0x000E(14) Return Slave Message Count

The response data field returns the quantity of messages addressed to the remote device, or broadcast, that the remote device has processed since its last restart, clear counters operation, or power-up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000E(14)	0x0000	Slave Message Count	

Sub-function 0x000F(15) Return Slave No Response Count

The response data field returns the quantity of messages addressed to the remote device for which it has returned no response (neither a normal response nor an exception response), since its last restart, clear counters operation, or power-up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000F(15)	0x0000	Slave No Response Count	

Sub-function 0x0064(100) Return Slave ModBus, Internal Bus Status

The response data field returns the status of ModBus and Internal Bus addressed to the remote device.
This status values are identical with status 1 word of input process image.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0064(100)	0x0000	ModBus, Internal Bus Status	Same as status 1 word

Sub-function 0x0065(101) Return Slave Watchdog Error Count

The response data field returns the quantity of watchdog error addressed to the remote device since its last restart, clear counters operation, or power-up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0065(101)	0x0000	Watchdog Error Count	

3.3.10. 15 (0x0F) Write Multiple Coils

This function code is used to force each coil in a sequence of coils to either ON or OFF in a remote device. The Request PDU specifies the coil references to be forced. Coils are addressed starting at zero. A logical '1' in a bit position of the field requests the corresponding output to be ON. A logical '0' requests it to be OFF.

The normal response returns the function code, starting address, and quantity of coils forced.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x0F	0x0F	“0F”	0x30, 0x46
Starting Address Hi	0x10	0x10	“10”	0x31, 0x30
Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Outputs Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Outputs Lo	0x10	0x10	“10”	0x31, 0x30
Byte Count	0x02	0x02	“02”	0x30, 0x32
Output Value#0	0x0F	0x0F	“0F”	0x30, 0x46
Output Value#1	0x00	0x00	“00”	0x30, 0x30
Error Check (CRC/LRC)	---	0x47, 0x73	“5D”	0x35, 0x44
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“,”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x0F	0x0F	“0F”	0x30, 0x46
Starting Address Hi	0x10	0x10	“10”	0x31, 0x30
Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Outputs Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Outputs Lo	0x10	0x10	“10”	0x31, 0x30
Error Check (CRC/LRC)	---	0x58, 0x85	“6E”	0x36, 0x45
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

- In case of address 0x1015~0x1000 output bit value: 00000000_00000000 changes to 00000001_01010101.

3.3.11. 16 (0x10) Write Multiple Registers

This function code is used to write a block of contiguous registers (1 to approx. 120 registers) in a remote device. The requested written values are specified in the request data field. Data is packed as two bytes per register. The normal response returns the function code, starting address, and quantity of registers written.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x10	0x10	“10”	0x31, 0x30
Starting Address Hi	0x08	0x08	“08”	0x30, 0x38
Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Registers Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Registers Lo	0x01	0x01	“01”	0x30, 0x31
Byte Count	0x02	0x02	“02”	0x30, 0x32
Register Value#0 Hi	0x00	0x00	“00”	0x30, 0x30
Register Value#0 Lo	0xFF	0xFF	“FF”	0x46, 0x46
Error Check (CRC/LRC)	---	0xDE, 0xB2	“83”	0x38, 0x33
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x10	0x10	“10”	0x31, 0x30
Starting Address Hi	0x08	0x08	“08”	0x30, 0x38
Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Registers Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Registers Lo	0x01	0x01	“01”	0x30, 0x31
Error Check (CRC/LRC)	---	0x0B, 0xEB	“84”	0x38, 0x34
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

- In case of address 0x0800, 0x0801 output register value: 0x0000, 0x0000 changes to 0x1122, 0x3344.

3.3.12. 23 (0x17) Read/Write Multiple Registers

This function code performs a combination of one read operation and one write operation in a single MODBUS transaction. The write operation is performed before the read. The request specifies the starting address and number of holding registers to be read as well as the starting address, number of holding registers, and the data to be written. The byte count specifies the number of bytes to follow in the write data field.

The normal response contains the data from the group of registers that were read. The byte count field specifies the quantity of bytes to follow in the read data field.

• Request

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x17	0x17	“17”	0x31, 0x37
Read Starting Address Hi	0x00	0x00	“00”	0x30, 0x30
Read Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Read Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Read Lo	0x01	0x01	“01”	0x30, 0x31
Write Starting Address Hi	0x08	0x08	“08”	0x30, 0x38
Write Starting Address Lo	0x00	0x00	“00”	0x30, 0x30
Quantity of Write Hi	0x00	0x00	“00”	0x30, 0x30
Quantity of Write Lo	0x01	0x01	“01”	0x30, 0x31
Byte Count	0x02	0x02	“02”	0x30, 0x32
Write Reg. Value#0 Hi	0x00	0x00	“00”	0x30, 0x30
Write Reg. Value#0 Lo	0xFF	0xFF	“FF”	0x46, 0x46
Error Check (CRC/LRC)	---	0x1B, 0xCC	“7B”	0x37, 0x42
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0x0A

• Response

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x63	0x63	“63”	0x36, 0x33
Function Code	0x17	0x17	“17”	0x31, 0x37
Byte Count	0x02	0x02	“02”	0x30, 0x32
Read Reg. Value#0 Hi	0x00	0x00	“00”	0x30, 0x30
Read Reg. Value#0 Lo	0xFF	0xFF	“FF”	0x46, 0x46
Error Check (CRC/LRC)	---	0x04, 0x3C	“85”	0x38, 0x35
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0x0A

- In case of address 0x0800, 0x0801 output register value: 0x0000, 0x0000 changes to 0x1122, 0x3344.

3.3.13. Error Response

In an exception response, the server sets the MSB of the function code to 1. This makes the function code value in an exception response exactly 80 hexadecimal higher than the value would be for a normal response.

- **Exception Response Example**

Field name	Example	RTU	ASCII	ASCII (bus line)
Start of Frame	---	t1-t2-t3	“.”	0x3A
Slave Address	0x07	0x07	“07”	0x30, 0x37
Function Code	0x81	0x81	“81”	0x38, 0x31
Exception Code	0x02	0x02	“02”	0x30, 0x32
Error Check (CRC/LRC)	---	0x22, 0xC0	“76”	0x37, 0x36
End of Frame	---	t1-t2-t3	CR, LF	0x0D, 0xA

- **Exception Codes**

Exception Code	Name	Description
01	Illegal Function	The function code received in the query is not an allowable action for the server (or slave).
02	Illegal Data Address	The data address received in the query is not an allowable address for the server (or slave).
03	Illegal Data Value	A value contained in the query data field is not an allowable value for server (or slave).
04	Slave Device Failure	An unrecoverable error occurred while the server (or slave) was attempting to perform the requested action.
05	Acknowledge	The server (or slave) has accepted the request and is processing it, but a long duration of time will be required to do so.
06	Slave Device Busy	Specialized use in conjunction with programming commands. The server (or slave) is engaged in processing a long-duration program command. The client (or master) should retransmit the message later when the server (or slave) is free.
08	Memory Parity Error	The server (or slave) attempted to read record file, but detected a parity error in the memory. The client (or master) can retry the request, but service may be required on the server (or slave) device.
0A	Gateway Path Unavailable	Specialized use in conjunction with gateways, indicates that the gateway was unable to allocate an internal communication path from the input port to the output port for processing the request.

- GL-9073 response exception code 01, 02, 03, 04 and 06.

3.4. MODBUS Special Register Map

The special register map can be accessed by function code 3, 4, 6 and 16.
Also the special register map must be accessed by read/write of every each address (one address).

3.4.1. Adapter Identification Special Register (0x1000, 4096)

Address	Access	Type, Size	Description
0x1000(4096)	Read	1word	Vendor ID = 0x02E5(741), Crevis. Co., Ltd.
0x1001(4097)	Read	1word	Device type = 0x000C, Network Adapter
0x1002(4098)	Read	1word	Product Code = 0x91A0
0x1003(4099)	Read	1word	Firmware revision, if 0x0101, revision 1.01
0x1004(4100)	Read	2word	Product unique serial number
0x1005(4101)	Read	String upto 18byte	Product name string (ASCII) “GL-9073,Modbus/485 Adapter,GBUS”
0x1006(4102)	Read	1word	Sum check of EEPROM
0x1010(4112)	Read	2word	Firmware release date
0x1011(4113)	Read	2word	Product manufacturing inspection date
0x101E(4126)	Read	7word - 1word - 1word - 1word - 1word - 1word - 2word	Composite Id of following address * RTU mode 0x1100(4352), MSB : Modbus RS485 Node. LSB : RS232 Node 0x1000(4096), Vendor ID 0x1001(4097), Device type 0x1002(4098), Product code 0x1003(4099), Firmware revision 0x1004(4100), Product serial number

- String Type consists of valid string length (first 1word) and array of characters

3.4.2. Adapter Watchdog Time, other Time Special Register (0x1020, 4128)

A watchdog timer can be configured for timeout periods up to 65535(1unit=100msec). The Watchdog timer will timeout (timer decreased, reached 0) if ModBus operation to the slave node does not occur over the configured watchdog value, then the slave adapter forces that slot output value is automatically set to user-configured fault actions and values.

Address	Access	Type, Size	Description
0x1020(4128)	Read/ Write	1word	Watchdog time value 16bit unsigned. The time value is represented by multiples of 100msec. The 0 (watchdog timeout disabled) is default value. A changing of watchdog time value resets watchdog error and counter.
0x1021(4129)	Read	1word	Watchdog timer remain value This value is decreased every 100msec
0x1022(4130)	Read	1word	Watchdog error counter, it is cleared by writing address 0x1020
0x1023(4131)	Read/ Write	1word	Enable/disable auto recovery Watchdog error when receiving new frame. 0:Disable, 1:Enable(default). Its value is stored in EEPROM.
0x1028(4136)	Read	1word	IO update time, main loop time. (1usec unit)

3.4.3. Adapter Information Special Register (0x1100, 4352)

Address	Access	Type, Size	Description																			
0x1100(4354)	Read/ Write	1word	Node address - MSB : Modbus RS485 Node. LSB : RS232 Node(Fixed 1).																			
0x1102(4354)	Read	1word	Start address of input image word register. =0x0000																			
0x1103(4355)	Read	1word	Start address of output image word register. =0x0800																			
0x1104(4356)	Read	1word	Size of input image word register.																			
0x1105(4357)	Read	1word	Size of output image word register.																			
0x1106(4358)	Read	1word	Start address of input image bit. = 0x0000																			
0x1107(4359)	Read	1word	Start address of output image bit. =0x1000																			
0x1108(4360)	Read	1word	Size of input image bit.																			
0x1109(4361)	Read	1word	Size of output image bit.																			
0x110A(4362)	Read	1word	Update time for cyclic data change (same as 0x1028)																			
0x110D(4365)	Read	1word	Current Dip Switch State																			
0x110E(4366)	Read	upto 33word	Expansion slot's GT-number including GL First 1word is adapter's number, if GL-9073, then 0x9073																			
0x1110(4368)	Read	1word	Number of expansion slot																			
0x1113(4371)	Read	upto 33word	Expansion slot Module Id. First 1word is adapter's module id.																			
0x1119(4377)	Read	1word	Hi byte is ModBus status, low byte is internal bus status. Zero value means 'no error'.																			
			<table><tr><th>ModBus status</th><th>Internal bus status(G-Bus)</th></tr><tr><td>0x00 : No Error</td><td>0x00 : OPERATING</td></tr><tr><td>0x01 : ERR_DIP_SWITCH</td><td>0x01: COMMUNICATION_FAULT</td></tr><tr><td>0x40 : ERR_CRC_LRC</td><td>0x02 : CONNECT_FAULT</td></tr><tr><td>0x80 : ERR_WATCHDOG</td><td>0x03 : CONFIG_FAULT</td></tr><tr><td></td><td>0x04 : NO_EXPANSION</td></tr><tr><td></td><td>0x05 : INVALID_ATTR_VALUE</td></tr><tr><td></td><td>0x06 : TOO_MUCH_DATA</td></tr><tr><td></td><td>0x07 : VENDOR_ERROR</td></tr><tr><td></td><td>0x08 : NOT_EXPECTED_SLOT</td></tr><tr><td></td><td>0x09 : CRC_ERROR</td></tr></table>	ModBus status	Internal bus status(G-Bus)	0x00 : No Error	0x00 : OPERATING	0x01 : ERR_DIP_SWITCH	0x01: COMMUNICATION_FAULT	0x40 : ERR_CRC_LRC	0x02 : CONNECT_FAULT	0x80 : ERR_WATCHDOG	0x03 : CONFIG_FAULT		0x04 : NO_EXPANSION		0x05 : INVALID_ATTR_VALUE		0x06 : TOO_MUCH_DATA		0x07 : VENDOR_ERROR	
ModBus status	Internal bus status(G-Bus)																					
0x00 : No Error	0x00 : OPERATING																					
0x01 : ERR_DIP_SWITCH	0x01: COMMUNICATION_FAULT																					
0x40 : ERR_CRC_LRC	0x02 : CONNECT_FAULT																					
0x80 : ERR_WATCHDOG	0x03 : CONFIG_FAULT																					
	0x04 : NO_EXPANSION																					
	0x05 : INVALID_ATTR_VALUE																					
	0x06 : TOO_MUCH_DATA																					
	0x07 : VENDOR_ERROR																					
	0x08 : NOT_EXPECTED_SLOT																					
	0x09 : CRC_ERROR																					
0x111D(4381)	Read	1word	Adapter G-Series Revision.																			

* After the system is reset, the new "Set Value" action is applied.

** If the slot location is changed, set default value automatically (all expansion slot are live).

3.4.4. Expansion Slot Information Special Resister (0x2000, 8192)

Each expansion slot has 0x20(32) address offset and same information structure.

Slot#1	0x2000(8192)~0x201F(8223)	Slot#2	0x2020(8224)~0x203F(8255)
Slot#3	0x2040(8256)~0x205F(8287)	Slot#4	0x2060(8288)~0x207F(8319)
Slot#5	0x2080(8320)~0x209F(8351)	Slot#6	0x20A0(8352)~0x20BF(8383)
Slot#7	0x20C0(8384)~0x20DF(8415)	Slot#8	0x20E0(8416)~0x20FF(8447)
Slot#9	0x2100(8448)~0x211F(8479)	Slot#10	0x2120(8480)~0x213F(8511)
Slot#11	0x2140(8512)~0x215F(8543)	Slot#12	0x2160(8544)~0x217F(8575)
Slot#13	0x2180(8576)~0x219F(8607)	Slot#14	0x21A0(8608)~0x21BF(8639)
Slot#15	0x21C0(8640)~0x21DF(8671)	Slot#16	0x21E0(8672)~0x21FF(8703)

Address Offset	Expansion Slot#1	Expansion Slot#2	Expansion Slot#3	Expansion Slot#4	Expansion Slot#16
+ 0x00(+0)	0x2000(8192)	0x2020(8224)	0x2040(8256)	0x2060(8288)	0x21E0(8672)
+ 0x01(+1)	0x2001(8193)	0x2021(8225)	0x2041(8257)	0x2061(8289)	0x21E1(8673)
+ 0x02(+2)	0x2002(8194)	0x2022(8226)	0x2042(8258)	0x2062(8290)	0x21E2(8674)
+ 0x03(+3)	0x2003(8195)	0x2023(8227)	0x2043(8259)	0x2063(8291)	0x21E3(8675)
+ 0x04(+4)	0x2004(8196)	0x2024(8228)	0x2044(8260)	0x2064(8292)	0x21E4(8676)
+ 0x05(+5)	0x2005(8197)	0x2025(8229)	0x2045(8261)	0x2065(8293)	0x21E5(8677)
+ 0x06(+6)	0x2006(8198)	0x2026(8230)	0x2046(8262)	0x2066(8294)	0x21E6(8678)
+ 0x07(+7)	0x2007(8199)	0x2027(8231)	0x2047(8263)	0x2067(8295)	0x21E7(8679)
+ 0x08(+8)	0x2008(8200)	0x2028(8232)	0x2048(8264)	0x2068(8296)	0x21E8(8680)
+ 0x09(+9)	0x2009(8201)	0x2029(8233)	0x2049(8265)	0x2069(8297)	0x21E9(8681)
+ 0x0A(+10)	0x200A(8202)	0x202A(8234)	0x204A(8266)	0x206A(8298)	0x21EA(8682)
+ 0x0B(+11)	0x200B(8203)	0x202B(8235)	0x204B(8267)	0x206B(8299)	0x21EB(8683)
+ 0x0C(+12)	0x200C(8204)	0x202C(8236)	0x204C(8268)	0x206C(8300)	0x21EC(8684)
+ 0x0D(+13)	0x200D(8205)	0x202D(8237)	0x204D(8269)	0x206D(8301)	0x21ED(8685)
+ 0x0E(+14)	0x200E(8206)	0x202E(8238)	0x204E(8270)	0x206E(8302)	0x21EE(8686)
+ 0x0F(+15)	0x200F(8207)	0x202F(8239)	0x204F(8271)	0x206F(8303)	0x21EF(8687)
+ 0x10(+16)	0x2010(8208)	0x2030(8240)	0x2050(8272)	0x2070(8304)	0x21F0(8688)
+ 0x11(+17)	0x2011(8209)	0x2031(8241)	0x2051(8273)	0x2071(8305)	0x21F1(8689)
+ 0x12(+18)	0x2012(8210)	0x2032(8242)	0x2052(8274)	0x2072(8306)	0x21F2(8690)
+ 0x13(+19)	0x2013(8211)	0x2033(8243)	0x2053(8275)	0x2073(8307)	0x21F3(8691)
+ 0x14(+20)	0x2014(8212)	0x2034(8244)	0x2054(8276)	0x2074(8308)	0x21F4(8692)
+ 0x15(+21)	0x2015(8213)	0x2035(8245)	0x2055(8277)	0x2075(8309)	0x21F5(8693)
+ 0x16(+22)	0x2016(8214)	0x2036(8246)	0x2056(8278)	0x2076(8310)	0x21F6(8694)
+ 0x17(+23)	0x2017(8215)	0x2037(8247)	0x2057(8279)	0x2077(8311)	0x21F7(8695)
+ 0x18(+24)	0x2018(8216)	0x2038(8248)	0x2058(8280)	0x2078(8312)	0x21F8(8696)
+ 0x19(+25)	0x2019(8217)	0x2039(8249)	0x2059(8281)	0x2079(8313)	0x21F9(8697)
+ 0x1A(+26)	0x201A(8218)	0x203A(8250)	0x205A(8282)	0x207A(8314)	0x21FA(8698)
+ 0x1B(+27)	0x201B(8219)	0x203B(8251)	0x205B(8283)	0x207B(8315)	0x21FB(8699)
+ 0x1C(+28)	0x201C(8220)	0x203C(8252)	0x205C(8284)	0x207C(8316)	0x21FC(8700)
+ 0x1D(+29)	0x201D(8221)	0x203D(8253)	0x205D(8285)	0x207D(8317)	0x21FD(8701)
+ 0x1E(+30)	0x201E(8222)	0x203E(8254)	0x205E(8286)	0x207E(8318)	0x21FE(8702)
+ 0x1F(+31)	0x201F(8223)	0x203F(8255)	0x205F(8287)	0x207F(8319)	0x21FF(8703)

Address Offset	Access	Type, Size	Description
+ 0x02(+2) **	Read	1 word	Input start register address of input image word this slot.
+ 0x03(+3) **	Read	1 word	Input word's bit offset of input image word this slot.
+ 0x04(+4) **	Read	1 word	Output start register address of output image word this slot.
+ 0x05(+5) **	Read	1 word	Output word's bit offset of output image word this slot.
+ 0x06(+6) **	Read	1 word	Input bit start address of input image bit this slot.
+ 0x07(+7) **	Read	1 word	Output bit start address of output image bit this slot.
+ 0x08(+8) **	Read	1 word	Size of input bit this slot
+ 0x09(+9) **	Read	1 word	Size of output bit this slot
+ 0x0A(+10) **	Read	n words	Read input data this slot
+ 0x0B(+11) **	Read/Write	n words	Read/write output data this slot
+ 0x0E(+14)	Read	1 word	GT-number, if GT-1238, returns 0x1238
+ 0x0F(+15)	Read	String upto 72bytes	First 1 word is length of valid character string. If GT-1238, returns "00 1E 52 54 2D 31 32 33 38 2C 20 38 44 49 2C 20 32 34 56 64 63 2C 20 55 6E 69 76 65 72 73 61 6C 00 00" Valid character size = 0x001E =30 characters, "GT-1238, 8DI, 24Vdc, Universal"
+ 0x10(+16)	Read	1 word	Size of configuration parameter byte
+ 0x11(+17) **	Read/Write	n words	Read/write Configuration parameter data, Refer to each IO parameter Specification.
+ 0x17(+23)	Read	2words	Firmware Revision ex) 0x00010010 (Major revision 1 /Minor revision 16, Rev 1.016)
+ 0x19(+25)	Read	2words	Firmware release date.

* After the system is reset, the new "Set Value" action is applied.

** Nothing of output, input, memory or configuration parameter corresponding slot returns Exception 02.

3.5. Supported MODBUS Function Codes

MODBUS Reference Documents

<http://www.modbus.org>

MODBUS Tools

<http://www.modbustools.com>, modbus poll

<http://www.win-tech.com>, modscan32