FnIO G-Series:

GN-9251

BACnet/IP B-ASC Network Adapter

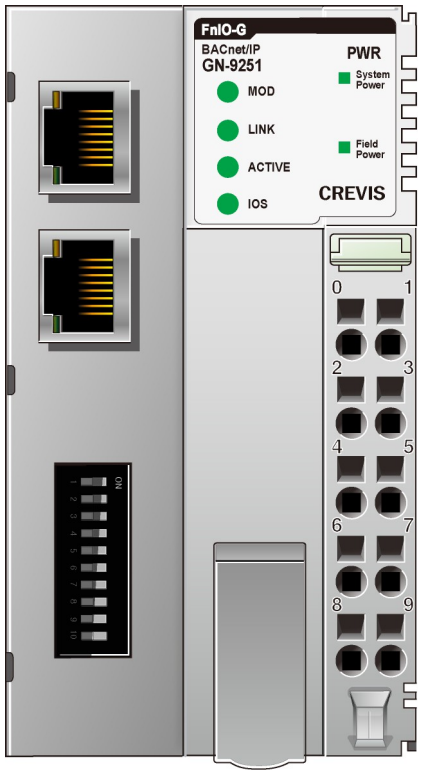


Table of Contents

Table of Contents....2
History....4
1.Environment Specification....5
2.GN-9251 (BACnet/IP Network Adapter)....6
2.1.GN-9251 Specification....6
2.2.GN-9251 Wiring Diagram....8
2.2.1.Power Connector (RTB)....8
2.3.GN-9251 LED Indicator....9
2.3.1.LED Indicator....9
2.3.2.MOD (Module Status LED)....9
2.3.3.LINK (Physical Connection LED)....9
2.3.4.ACTIVE (Exchange Data/Traffic Present LED)....9
2.3.5.IOS (Expansion Module Status LED)....10
2.3.6.Field Power, System Power LED (Field Power, System Power Status LED)....10
2.4.GN-9251 Electrical Interface....11
2.4.1.RJ-45 Socket....11
2.4.2.Dip Switch....11
2.4.3.USB 2.0 Port....11
2.5.Address Setup....12
2.5.1.IP-Address Setup using BOOTP/DHCP Sever....12
2.5.2.IP-Address Setupusing Dip Switch (Manual Function)....12
3.BACNET Interface....13
3.1.BACnet Devices Profiles....13
3.2.BACnet Application Specific Controller (B-ASC)....13
3.3.BACnet Interoperability Building Blocks (BIBBs)....14
3.4.BACnet Standard Object Types Supported....15
3.4.1.Device Object Type....15
3.4.2.Analog Input Object Type....16
3.4.3.Analog Output Object Type....16
3.4.4.Binary Input Object Type....16
3.4.5.Binary Output Object Type....17
3.4.6.GBUS Direct Object Type (Nonstandard Types)....17
3.4.7.GBUS Slot Object Type (Nonstandard Types)....17
3.5.Supported BACnet Function Codes....18

4.MODBUS Interface....19
4.1.I/O Process Image Map....19
4.1.1.MODBUS Interface Register/Bit Map....19
4.1.2.Example of Input Process Image (Input Register) Map....20
4.1.3.Example of Output Process Image (Output Register) Map....21
4.2.Supported MODBUS Function Codes....22
4.3.MODBUS Special Register Map....26
4.3.1.Adapter Identification Special Resgister (0x1000, 4096)....26
4.3.2.Adapter Watchdog Time, other Time Special Register (0x1020, 4128)....26
4.3.3.Adapter TCP/IP Special Register (0x1040, 4160)....27
4.3.4.Adapter Information Special Register (0x1100, 4352)....27
4.3.5.Expasion Slot Information Special Resister (0x2000, 8192)....29
4.4.Supported MODBUS Function Codes....30

History

Rev	Pages	Remarks	Date	Editor
1.00		Preliminary	May 6, 2020	Tae Eun, Kim
1.01			Jul 24, 2023	Tae Eun, Kim

1. Environment Specification

Environmental Specification	
Operation Temperature	60℃ ~ 70℃ : Power dissipation is limited to 0.8A. -40℃ ~ 60℃ : 1.5A full load is allowed.
UL Temperature	-20℃~60℃
Storage Temperature	-40℃ ~85℃
Relative Humidity	5%~90% Non-condensing
Mounting	DIN rail
General Specification	
Shock Operating	IEC 60068-2-27
Vibration Resistance	Based on IEC 60068-2-6, 4g
Industrial Emissions	EN61000-6-4/All : 2011
Industrial Immunity	EN 61000-6-2 : 2019
Installation Position	Vertical and horizontal installation is available
Product Certifications	CE, UL, UKCA

2. GN-9251 (BACnet/IP Network Adapter)

2.1. GN-9251 Specification

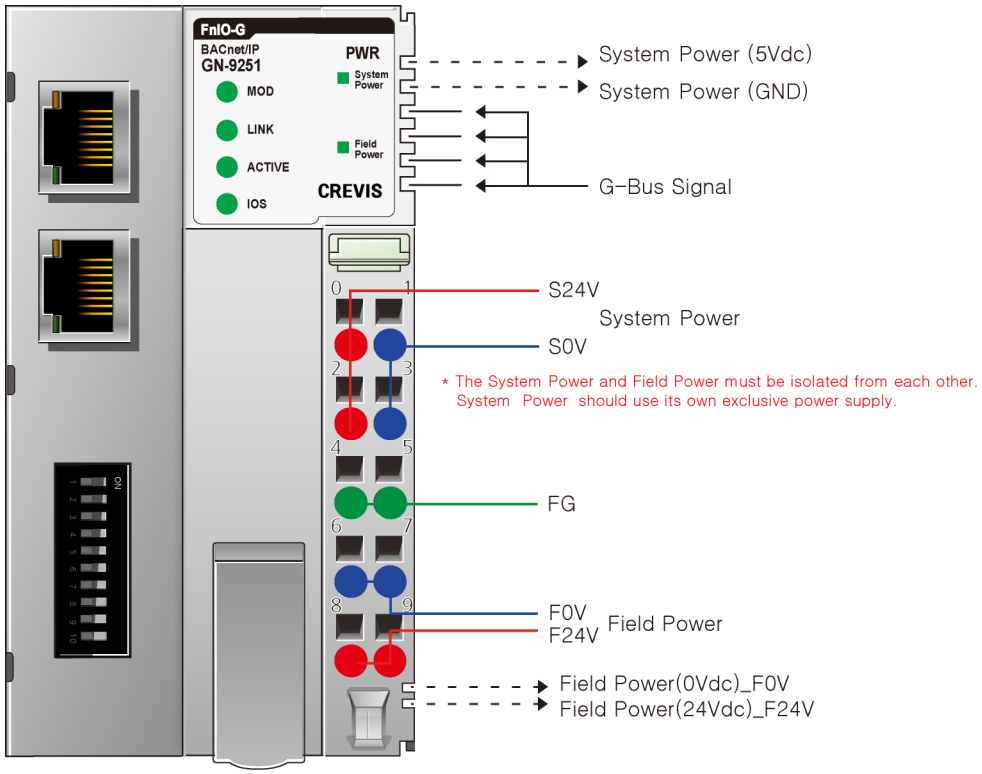
Items	Specification
Communication Interface Specification	
Adapter Type	Slave node (BACnet/IP)
BACnet Device Profile	BACnet Application Specific Controller (B-ASC)
Max. Supported Object Size	256 objects*
BACnet Protocol Version	1
BACnet Protocol Revision	12
Max. Expantsion Module	32 slots
Sub Protocol	MODBUS/TCP,MODBUS/UDP,HTTP,DHCP,10 TCP Connections
Max Length Bus Line	Up to 100m from Ethernet Hub/Switch with twisted CAT5 UTP/STP
Max. Nodes	Limited by Ethernet Specification.
Baud Rate	10/100Mbps, Auto-negotiation, Full duplex
Interface Connector	RJ-45 socket * 2pcs
IP-Address Setup	Via DHCP/BOOTP or IOGuide(Crevis Software)
IP-Address Range	xxx.xxx.xxx.1 ~ 253
IAP Mode	Set dip sw to 254. - IAP web-server IP: 192.168.0.100 - Using only Internet Explorer / recommended version 11
Serial Port	RS232 for MODBUS/RTU, Touch Pannel or IOGuide
Serial Configuration (RS232)	1 Node, 115200 baud, 8 data bits, no parity, 1 stop bit (Fixed)
Indicator *Refer to '2.3. LED Indicator'	6 LEDs - Window Lable MOD, LINK, ACTIVE, IOS, System Power, Field Power 2 LEDs - each RJ45 Connector Yellow(Link/Active), Green(Not used)
Module Location	Starter module left side of G-Series system
Field Power Detection	About 14Vdc

* Only analog I/O module(GT-3xxx,GT-4xxx) and digital I/O module(GT-1xxx,GT-2xxx) are supported.
* Analog modules are mapped to 1 Analog object per 1 word, and digital modules are mapped to 1 Binary object per 1 bit.

Items	Specification
General specification	
UL System Power	Supply voltage : 24Vdc nominal, Class2
System Power	Supply voltage : 24Vdc typical Supply voltage range : 15~30Vdc Protection : Output current limit (Min. 1.5A) - Reverse polarity protection
Power Dissipation	70mA typical @ 24Vdc
Current for I/O Module	1.5A @ 5Vdc**
Isolation	System power to internal logic : Non-isolation System power I/O driver : Isolation
UL Field Power	Supply voltage : 24Vdc nominal, Class2
Field Power	Supply voltage : 24Vdc typical (Max. 30Vdc)***
Max. Current Field Power Contact	DC 10A Max
Wiring	I/O Cable Max. 2.0mm^2(AWG 14)
Torque	0.8Nm(7 lb-in)
Weight	162g
Module Size	54mm x 99mm x 70mm
Environment Condition	**Refer to '1. Environment Specification'**

** When using in '60℃ ~ 70℃' temperature environment, the power dissipation is limited to 0.8A.
*** Field Power Range is different depending on IO Module series. Refer to IO Module`s Specification.

2.2. GN-9251 Wiring Diagram



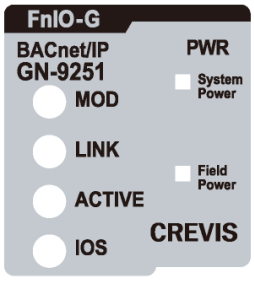
2.2.1. Power Connector (RTB)

Pin No.	Signal Description	Signal Description	Pin No.
0	System Power, 24V	System Power, Ground	1
2	System Power, 24V	System Power, Ground	3
4	F.G	F.G	5
6	Field Power, Ground	Field Power, Ground	7
8	Field Power, 24V	Field Power, 24V	9

* **Warning:** The system power must not be connected with field power. Use seperate voltage supplies.

2.3. GN-9251 LED Indicator

2.3.1. LED Indicator



LED No.	LED Description	LED Color
MOD	Module Status	Green/Red
LINK	Physical Connection	Green
ACTIVE	Exchange Data Traffic Present	Green/Red
IOS	Extension Module Status	Green/Red
System Power	System Power Enable	Green
Field Power	Field Power Enable	Green

2.3.2. MOD (Module Status LED)

Status	LED is	To indicate
Not Powered	OFF	Not power is supplied to the unit.
Device Operational	Green	The unit is operating in normal condition.
Devicce in Standby	Flashing Green	The device needs commissioning due to configuration missing, incomplete or incorrect.
IAP Mode	Green/Red Toggle	Availble for firmware download using FireFox.
Minor Fault	Flashing Red	Recoverable Fault. - EEPROM checksum fault.
Unrecoverable Fault	Red	The device has an unrecoverable fault. - Memory error or CPU watchdog error.

2.3.3. LINK (Physical Connection LED)

Status	LED is	To indicate
Not Linked	OFF	The device is not physically connected or may not be powered.
Adapter physical connected	Green	Adapter Ethernet Controller physically connected.

2.3.4. ACTIVE (Exchange Data/Traffic Present LED)

Status	LED is	To indicate
Physical network not ready	OFF	Device is idle or may not be powered.
Adapter exchange data	Flashing Green	Adapter(slave) exchange data/Traffic present. About 10msec flashing.
Data exchange is not possible.	Red	Bacnet initialization failed. - Too many object sizes.

2.3.5. IOS (Expansion Module Status LED)

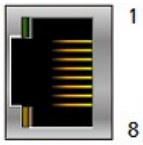
Status	LED is	To indicate
Not Powered	OFF	Device may not be powered.
No Expansion Module	Flashing Red	Adapter has no expansion module.
Internal Bus Connection, Run Exchanging I/O	Green	Exchanging I/O data.
Configuration Fault	Red	One or more expansion module occurred in fault state. - Detected invalid 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.

2.3.6. Field Power, System Power LED (Field Power, System Power Status LED)

Status	LED is	To indicate
No field, System power	OFF	Not supplied 24Vdc field power, 5Vdc system power.
Supplied field, System power	Green	Supplied 24Vdc field power, 5Vdc system power.

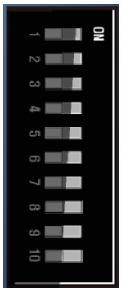
2.4. GN-9251 Electrical Interface

2.4.1. RJ-45 Socket



RJ-45	Signal Name	Description
1	TD+	Transmit +
2	TD-	Transmit -
3	RD+	Receive +
4	-	
5	-	
6	RD-	Receive -
7	-	
8	-	
Case	Shield	

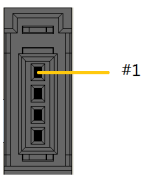
2.4.2. Dip Switch



DIP Pole#	Description	
1	IP_DIP bit#0	Lowest IP Address when Pole#10=ON ex) XXX.XXX.XXX.IP_DIP
2	IP_DIP bit#1	
3	IP_DIP bit#2	
4	IP_DIP bit#3	
5	IP_DIP bit#4	
6	IP_DIP bit#5	
7	IP_DIP bit#6	
8	IP_DIP bit#7	
18	= ON : Enable DHCP/BOOTP *	
19	= ON : Use Lowest IP Address with IP_DIP value	

* DHCP/BOOTP have to be set in special register 0x1045 (default 0 : BOOTP).
** Refer to '2.5.2. IP-Address Setup using Dip Switch(Manual Function)'

2.4.3. USB 2.0 Port



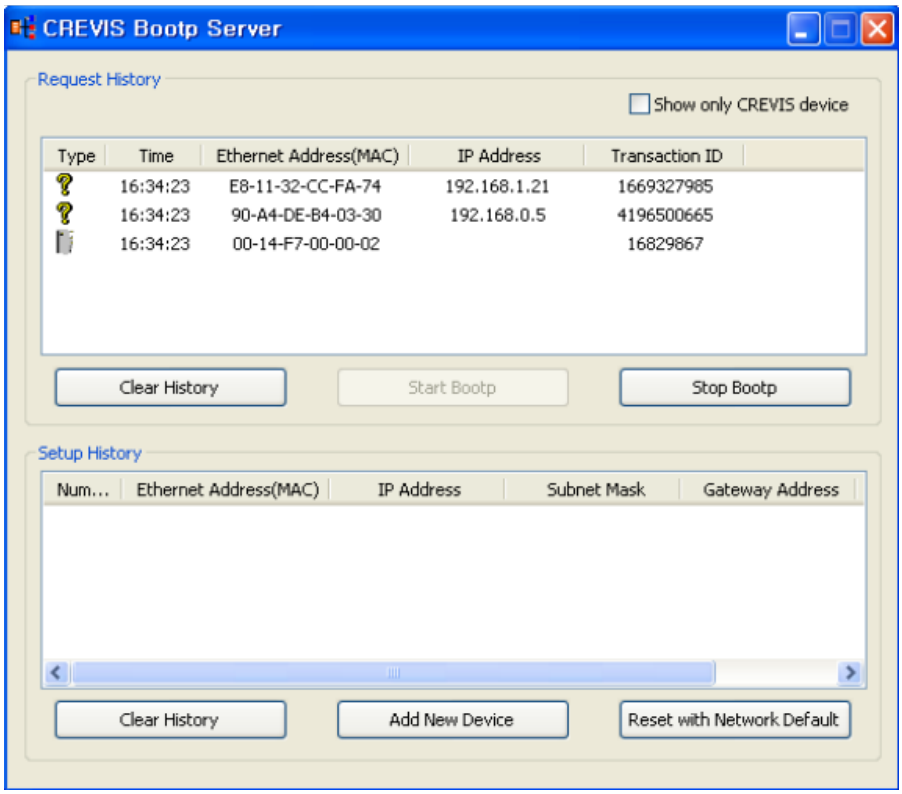
Pin#	Signal Name	Description
1	Reserved	----
2	TXD	RS232 TXD
3	RXD	RS232 RXD
4	GND	RS232 GND

2.5. Address Setup

2.5.1. IP-Address Setup using BOOTP/DHCP Sever

If the adapter BOOTP/DHCP enabled (DIP Pole#9 ON), the adapter sends BOOTP/DHCP request message of 20 times every 2sec. If BOOTP/DHCP sever does not response, the Adapter applies its IP Address with EEPROM (Latest saved IP Address). The following is an example of adapter IP-Address setup that can be used with a third party BOOTP/DHCP server.

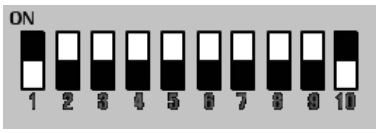
- CREVIS IO Guide Pro' s BOOTP server



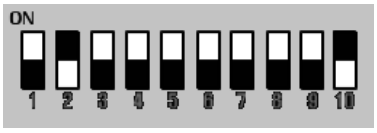
2.5.2. IP-Address Setupusing Dip Switch (Manual Function)

If the adapter DIP Pole#10 is ON, lowest IP address is set by DIP Pole#1~#8 manually. Refer to '2.4.2. Dip Switch' These are examples of adapter IP-Address setup by manual function.

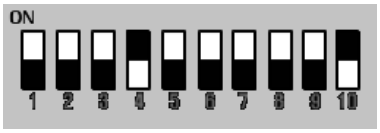
Ex) xxx . xxx . xxx . 1



Ex) xxx . xxx . xxx . 2



Ex) xxx . xxx . xxx . 8



Ex) xxx . xxx . xxx . 253

3. BACNET Interface

3.1. BACnet Devices Profiles

BACnet device profiles are categorized into families:

· Operator Interfaces. This family is composed of B-XAWS, B-AWS, B-OWS, and B-OD.
· Life Safety Operator Interfaces. This family is composed of B-ALSWS, B-LSWS, and B-LSAP.
· Access Control Operator Interfaces. This family is composed of B-XAWS, B-AACWS, B-ACWS, and B-ACSD.
· Controllers. This family is composed of B-BC, B-AAC, B-ASC, B-SA, and B-SS.
· Life Safety Controllers. This family is composed of B-ALSC and B-LSC.
· Access Control Controllers. This family is composed of B-AACC and B-ACC.
· Miscellaneous. This family is composed of B-RTR, B-GW, B-BBMD, B-ACDC, and B-ACCR.

Devices may claim to be multiple device types. For example, a device may claim to be both a B-BC and the B-RTR. Devices that claim multiple device profiles shall only combine capabilities from different device families, with the exception that multiple profiles may be selected from the Miscellaneous family. For example, a device may claim the B-BC, B-RTR, and BBBMD profiles, but a device may not claim both the B-BC and B-SS profiles.

The B-GENERAL device profile is not included in any of the profile families and is never claimed in conjunction with any other device profile, except those from the Miscellaneous family.

3.2. BACnet Application Specific Controller (B-ASC)

BACnet Application Specific Controller (B-ASC)

B-ASC is a controller with limited resources relative to a B-AAC. It is intended for use in a specific application and supports limited programmability. It enables specification of the following:

Data Sharing
- Ability to provide the values of any of its BACnet objects
- Ability to allow modification of some or all of its BACnet objects by another device

Alarm and Event Management
- No requirement

Scheduling
- No requirement

Trending
- No requirement

Device and Network Management
- Ability to respond to queries about its status
- Ability to respond to requests for information about any of its objects
- Ability to respond to communication control messages

3.3. BACnet Interoperability Building Blocks (BIBBs)

BACnet Interoperability Building Blocks (BIBBs) are collections of one or more BACnet services.
These are prescribed in terms of an "A" and a "B" device. Both of these devices are nodes on a BACnet internetwork.
In most cases, the "A" device will act as the user of data (client), and the "B" device will be the provider of this data (server). In addition, certain BIBBs may also be predicated on the support of certain, otherwise optional, BACnet objects or properties and may place constraints on the allowable values of specific properties or service parameters.

Data Sharing-ReadProperty-B (DS-RP-B)

• The B device is a provider of data to device A.

BACnet Service	Initiate	Execute
Read Property		x

Data Sharing-WriteProperty-B (DS-WP-B)

• The B device allows a value to be changed by device A.

BACnet Service	Initiate	Execute
Write Property		x

Device Management-Dynamic Device Binding-B (DM-DDB-B)

• The B device provides information about its device attributes and responds to requests to identify itself.

BACnet Service	Initiate	Execute
Who-Is		x
I-Am	x	

Device Management-Dynamic Object Binding-B (DM-DOB-B)

• The B device provides address information about its objects upon request.

BACnet Service	Initiate	Execute
Who-Has		x
I-Have	x	

Device Management-DeviceCommunicationControl-B (DM-DCC-B)

• The B device responds to communication control exercised by the A device.

BACnet Service	Initiate	Execute
DeviceCommunicationControl		x

3.4. BACnet Standard Object Types Supported.

If analog I / O module and digital I / O module are configured in BACnet NA, BACnet Object creation is automatically performed according to the number of channels. The object name consists of slot number, module name, number of channels, and object instance. The maximum number of object creation supported is 256.
ex) Channel 3 of the GT-4118 module, configured in the third slot. (Analog Input instance 19)
>> Slot3 GT4118_3 (AO19)

3.4.1. Device Object Type

The Device object type defines a standardized object whose properties represent the externally visible characteristics of a BACnet device. There shall be exactly one Device object in each BACnet device. A Device object is referenced by its Object_Identifier property, which is not only unique to the BACnet device that maintains this object but is also unique throughout the BACnet internetwork.

Property	Data Type	Default Value	Writable
Object_Identifier	BACnetObjectIdentifier	device, 741	O
Object_Name	CharacterString	G-Series BACnet/IP B-ASC Network Adapter: GN-9251	
Object_Type	BACnetObjectType	device (8)	
System_Status	BACnetDeviceStatus		
Location	CharacterString	Republic of Korea	
Vendor_Name	CharacterString	CREVIS CO,.LTD	
Vendor_Identifier	Unsigned16	1175	
Model_Name	CharacterString	GN-9251	
Firmware_Revision	CharacterString	1	
Application_Software_Version	CharacterString	--	
Protocol_Version	Unsigned	1	
Protocol_Revision	Unsigned	12	
Protocol_Services_Supported	BACnetServices Supported		
Protocol_Object_Types_Supported	BACnetObjectTypes Supported		
Object_List	BACnetARRAY[N] of BACnetObjectIdentifier		
APDU_Timeout	Unsigned	3000	
Number_Of_APDU_Retries	Unsigned	3	
Max_APDU_Length_Accepted	Unsigned	480	
Segmentation_Supported	BACnetSegmentation		
Device_Address_Binding	BACnetLIST of BACnetAddressBinding		
Database_Revision	Unsigned		
Release_Date (Nonstandard Types ID : 610)	CharacterString		

3.4.2. Analog Input Object Type

The Analog Input object type defines a standardized object whose properties represent the externally visible characteristics of an analog input. This object is connected with Analog Input IO (GT-3xxx) Module data.

Property	Data Type	Default Value	Writable
Object_Identifier	BACnetObjectIdentifier	analog-input 0	
Object_Name	CharacterString		
Object_Type	BACnetObjectType	analog-input (0)	
Present_Value	REAL		
Status_Flags	BACnetStatusFlags		
Event_State	BACnetEventState		
Reliability	BACnetReliability		
Out_Of_Service	BOOLEAN	FALSE	
Units	BACnetEngineeringUnits		
IO Parameter (Nonstandard Types ID : 700)	Octet String		O

3.4.3. Analog Output Object Type

The Analog Output object type defines a standardized object whose properties represent the externally visible characteristics of an analog output. This object is connected with Analog Output IO (GT-4xxx) Module data.

Property	Data Type	Default Value	Writable
Object_Identifier	BACnetObjectIdentifier	analog-output,0	
Object_Name	CharacterString		
Object_Type	BACnetObjectType	analog-output (1)	
Present_Value	REAL		O
Status_Flags	BACnetStatusFlags		
Event_State	BACnetEventState		
Out_Of_Service	BOOLEAN	FALSE	
Units	BACnetEngineeringUnits		
Priority_Array	BACnetPriorityArray		
Relinquish_Default	REAL		
IO Parameter (Nonstandard Types ID : 700)	Octet String		O

3.4.4. Binary Input Object Type

The Binary Input object type defines a standardized object whose properties represent the externally visible characteristics of a binary input. A "binary input" is a physical device or hardware input that can be in only one of two distinct states. In this description, those states are referred to as ACTIVE and INACTIVE. This object is connected with Digital Input IO (GT-1xxx) Module data.

Property	Data Type	Default Value	Writable
Object_Identifier	BACnetObjectIdentifier	binary input, 0	
Object_Name	CharacterString		
Object_Type	BACnetObjectType	binary-input (3)	
Present_Value	BOOLEAN		
Status_Flags	BACnetStatusFlags		
Event_State	BACnetEventState		
Out_Of_Service	BOOLEAN	FALSE	
Polarity	BACnetPolarity		O
IO Parameter (Nonstandard Types ID : 700)	Octet String		O

3.4.5. Binary Output Object Type

The Binary Output object type defines a standardized object whose properties represent the externally visible characteristics of a binary output. A "binary output" is a physical device or hardware output that can be in only one of two distinct states. In this description, those states are referred to as ACTIVE and INACTIVE. This object is connected with Digital Output IO (GT-2xxx) Module data.

Property	Data Type	Default Value	Writable
Object_Identifier	BACnetObjectIdentifier	binary-output,0	
Object_Name	CharacterString		
Object_Type	BACnetObjectType	binary-output (4)	
Present_Value	BOOLEAN		O
Status_Flags	BACnetStatusFlags		
Event_State	BACnetEventState		
Out_Of_Service	BOOLEAN	FALSE	
Polarity	BACnetPolarity		O
Priority_Array	BACnetPriorityArray		
Relinquish_Default	BACnetBinaryPV		
IO Parameter (Nonstandard Types ID : 700)	Octet String		O

3.4.6. GBUS Direct Object Type (Nonstandard Types)

GBUS Direct Object Type contains information of 'Network Adapter' and GBUS configuration value. The instance value is only 0.

Property	ID (741)	Data Type	Default Value	Writable
Object_Identifier		BACnetObjectIdentifier	741, 0	
Object_Name		CharacterString	GBUS Direct	
Object_Type		BACnetObjectType	741	
Slot Name	600	CharacterString		
Input Size (byte)	601	Unsigned		
Output Size (byte)	602	Unsigned		
Input Data	603	Octet String		
Output Data	604	Octet String		

3.4.7. GBUS Slot Object Type (Nonstandard Types)

GBUS Slot Object Type contains information of extension IO Module. The instance value is created by the number of IOs.

Property	ID (742)	Data Type	Default Value	Writable
Object_Identifier		BACnetObjectIdentifier	742, 0	
Object_Name		CharacterString	GBUS Slot x	
Object_Type		BACnetObjectType	742	
Firmware_Revision		Character String		
Slot Name	600	CharacterString		
Input Size (byte)	601	Unsigned		
Output Size (byte)	602	Unsigned		
Input Data	603	Octet String		
Output Data	604	Octet String		
Slot Type	605	CharacterString		
Release_Date	610	CharacterString		
IO Parameter	700	Octet String		O

3.5. Supported BACnet Function Codes

Latest information on BACnet
http://www.bacnet.org/

ASHRAE Website
https://www.ashrae.org/

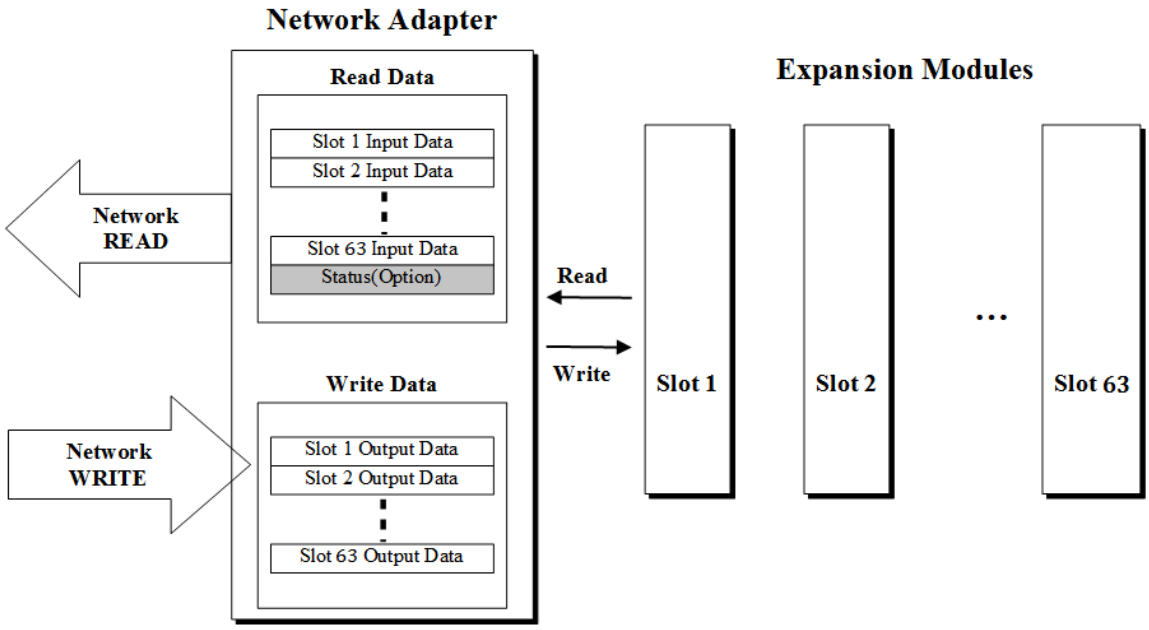
BACnet International / BACnet Testing Laboratories
https://www.bacnetinternational.org/default.aspx

Visual Test Shell for BACnet
https://sourceforge.net/projects/vts/

4. MODBUS Interface

4.1. I/O Process 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.



4.1.1. 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).

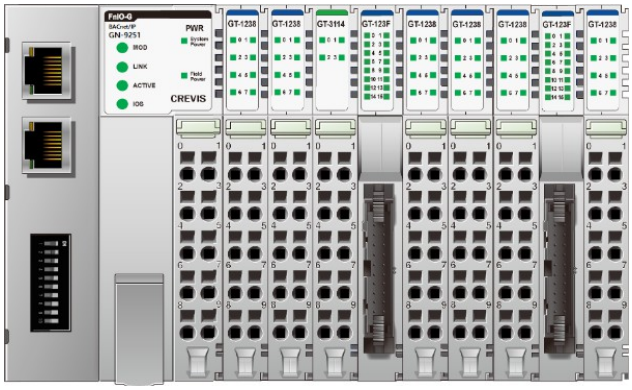
- **Bit Map**

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

4.1.2. 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 example slot configuration**



Slot No.	Module Description
#0	BACnet/IP 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

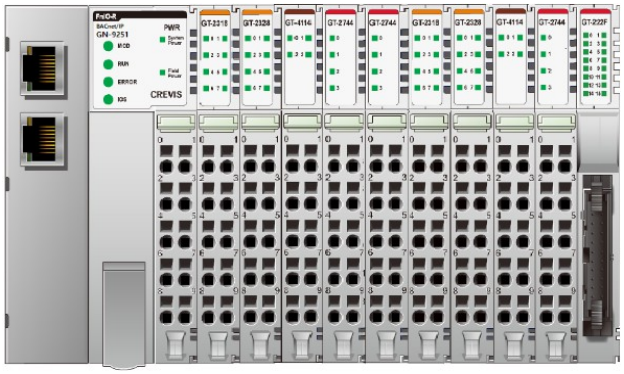
- **Input Process Image**

Address	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
0x0001	Discrete Input 8 pts (Slot#2)								Discrete Input 8 pts (Slot#1)							
0x0002	Analog Input Ch0 high byte (Slot#3)								Analog Input Ch0 low byte (Slot#3)							
0x0003	Analog Input Ch1 high byte (Slot#3)								Analog Input Ch1 low byte (Slot#3)							
0x0004	Analog Input Ch2 high byte (Slot#3)								Analog Input Ch2 low byte (Slot#3)							
0x0005	Analog Input Ch3 high byte (Slot#3)								Analog Input Ch3 low byte (Slot#3)							
0x0006	Discrete Input 8 pts (Slot#4)								Discrete Input 8 pts (Slot#4)							
0x0007	Discrete Input 8 pts (Slot#6)								Discrete Input 8 pts (Slot#5)							
0x0008	Discrete Input 8 pts (Slot#8)								Discrete Input 8 pts (Slot#7)							
0x0009	Discrete Input 8 pts (Slot#9)								Discrete Input 8 pts (Slot#8)							

4.1.3. 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 example slot configuration**



Slot No.	Module Description
#0	BACnet/IP 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 Output 8 pts (Slot#2)								Discrete Output 8 pts (Slot#1)							
0x0801	Analog Output Ch0 high byte (Slot#3)								Analog Output Ch0 low byte (Slot#3)							
0x0802	Analog Output Ch1 high byte (Slot#3)								Analog Output Ch1 low byte (Slot#3)							
0x0803	Analog Output Ch2 high byte (Slot#3)								Analog Output Ch2 low byte (Slot#3)							
0x0804	Analog Output Ch3 high byte (Slot#3)								Analog Output Ch3 low byte (Slot#3)							
0x0805	Empty, Don't Care				Discrete Out 4 pts (Slot#5)				Empty, Don't Care				Discrete Out 4 pts (Slot#4)			
0x0806	Discrete Output low 8 pts (Slot#7)								Discrete Output low 8 pts (Slot#6)							
0x0807	Analog Output Ch0 high byte (Slot#8)								Analog Output Ch0 low byte (Slot#8)							
0x0808	Analog Output Ch1 high byte (Slot#8)								Analog Output Ch1 low byte (Slot#8)							
0x0809	Analog Output Ch2 high byte (Slot#8)								Analog Output Ch2 low byte (Slot#8)							
0x080A	Analog Output Ch3 high byte (Slot#8)								Analog Output Ch3 low byte (Slot#8)							
0x080B	Discrete Output low 8 pts (Slot#10)								Empty, Don't Care				Discrete Out 4 pts (Slot#9)			
0x080C	Empty, Don't Care								Discrete Output high 8 pts (Slot#10)							

4.2. Supported MODBUS Function Codes

Function Code	Function	Description
1(0x01)	Read Coils (Read output bit)	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.
2(0x02)	Read Discrete Inputs (Read input bit)	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.
3(0x03)	Read Holding Registers (Read output word)	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.
4(0x04)	Read Input Registers (Read input word)	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.
5(0x05)	Write Single Coil (Write one bit output)	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.
6(0x06)	Write Single Register (Write one word output)	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.
8(0x08)	Diagnostics (Read diagnostic register) *Refer to the 4.2.1	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.
15(0x0F)	Write Multiple Coils (Write a number of output bits)	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.

16(0x10)	Write Multiple registers (Write a number of output words)	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.
23(0x17)	Read/Write Multiple registers (Read a number of input words /Write a number of output words)	Read a number of input words /Write a number of output words 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.

- Refer to MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1a

8 (0x08) Diagnostics

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) Watchdog time value, auto recovery, will be the factory defaults value.
2) Mac Address, IP Address, Subnet Mask Address, Gateway Address, BACnet Identifier, will be the factory defaults 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 1word of input process image.

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

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	

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 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.

- GN-9251 response exception code 01, 02, 03, 04 and 06.

4.3. 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).

4.3.1. Adapter Identification Special Resgister (0x1000, 4096)

Address	Access	Type, Size	Description
0x1000(4096)	Read	1word	Vendor ID = 0x0497(1175), Crevis. Co., Ltd.
0x1001(4097)	Read	1word	Device type = 0x000C, Network Adapter
0x1002(4098)	Read	1word	Product Code = 0x9050
0x1003(4099)	Read	1word	Firmware revision, if 0x0101, revision 1.01
0x1004(4100)	Read	2word	Product unique serial number
0x1005(4101)	Read	String upto 34byte	Product name string (ASCII) “GN-9251,BACnet/IP B-ASC,G-Series”
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 15word - 2word - 2word - 2word - 3word - 1word - 1word - 1word - 1word - 2word	Composite Id of following address * RTU mode 0x1100(4352), Modbus RS232 Node. (Fixed 0x0001) 0x1000(4096), Vendor ID 0x1001(4097), Device type 0x1002(4098), Product code 0x1003(4099), Firmware revision 0x1004(4100), Product serial number *TCP mode 0x1050(4176), IP address 0x1051(4177), Subnet mask 0x1052(4178), Gateway 0x1053(4179), Ethernet physical address (MAC ID) 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

4.3.2. 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)

4.3.3. Adapter TCP/IP Special Register (0x1040, 4160)

Address	Access	Type, Size	Description
0x1041(4161)	Read/Write	1word	MODBUS/TCP connection timeout time. (0.5sec unit) Maximum time of ModBus connection to stay to be opened without receiving a ModBus request. 0~3600 The 120 (60sec) is default value. The value 0 disables connection time out specially.
0x1042(4162)	Read	1word	Number of ModBus/TCP connected
0x1043(4163)	Read	1word	ModBus/TCP port, fixed 502
0x1044(4164)	Read	1word	Ethernet Interface Speed, 10(10Mbps) or 100(100Mbps)
0x1045(4165)*	Read/Write	1word	IP Setting Method. 0: BOOTP, 1:DHCP
0x1047(4167)	Read	1word	Status of DIP SW#9 DHCP/BOOTP(Enable/Disable). 0 : OFF, 1 : ON
0x1048(4168)	Read	1word	Enable/disable Lowest IP address via DIP Switch, 1:Enabled
0x1050(4176)	Read/Write	2word	IP address. If 192.168.123.1, then 0xA8C0, 0x017B. After update this value, IP address, Subnet mask and Gateway are applied as new one.
0x1051(4177)	Read/Write	2word	Subnet mask. If 255.255.255.0, then 0xFFFF, 0x00FF.
0x1052(4178)	Read/Write	2word	Gateway. If 192.168.123.254, then 0xA8C0, 0xFE7B.
0x1053(4179)	Read	3word	Ethernet physical address (MAC-ID). If 11-22-33-44-55-66, then 0x2211, 0x4433, 0x6655.

* Power off and then power on, this value is applied.

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

Address	Access	Type, Size	Description
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 and Field Power Status (MSB) ex) DHCP/Booth enable, Dip SW(0x01), Field Power On = 0x8101
0x110E(4366)	Read	upto 33word	Expansion slot's GT-number including GN First 1word is adapter's number, if GN-9251, then 0x9251
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'.

ModBus status	Internal bus status(G-Bus)
0x00 : No Error 0x01 : ERR_DIP_SWITCH 0x40 : ERR_CRC_LRC 0x80 : ERR_WATCHDOG	0x00 : OPERATING 0x02 : CONNECT_FAULT 0x03 : CONFIG_FAULT 0x04 : NO_EXPANSION 0x05 : INVALID_ATTR_VALUE 0x06 : TOO_MUCH_DATA 0x07 : VENDOR_ERROR 0x08 : NOT_EXPECTED_SLOT 0x09 : CRC_ERROR 0x80 : NO_FIELD_POWER

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).

4.3.5. Expasion 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)
Slot#17 0x2200(8704)~0x221F(8735)
Slot#18 0x2220(8736)~0x223F(8767)
Slot#19 0x2240(8768)~0x225F(8799)
Slot#20 0x2260(8800)~0x227F(8831)
Slot#21 0x2280(8832)~0x229F(8863)
Slot#22 0x22A0(8864)~0x22BF(8895)
Slot#23 0x22C0(8896)~0x22DF(8927)
Slot#24 0x22E0(8928)~0x22FF(8959)
Slot#25 0x2300(8960)~0x231F(8991)
Slot#26 0x2320(8992)~0x233F(9023)
Slot#27 0x2340(9024)~0x235F(9055)
Slot#28 0x2360(9056)~0x237F(9087)
Slot#29 0x2380(9088)~0x239F(9119)
Slot#30 0x23A0(9120)~0x23BF(9151)
Slot#31 0x23C0(9152)~0x23DF(9183)
Slot#32 0x23E0(9184)~0x23FF(9215)
Slot#33 0x2400(9216)~0x241F(9247)
Slot#34 0x2420(9248)~0x243F(9279)
…..
Slot#63 0x27C0(10176)~0x27DF(10207)

Address Offset	Expansion Slot#1	Expansion Slot#2	Expansion Slot#3	Expansion Slot#4	……..	Expansion Slot#63
+ 0x00(+0)	0x2000(8192)	0x2020(8224)	0x2040(8256)	0x2060(8288)	…….	0x27C0(10176)
+ 0x01(+1)	0x2001(8193)	0x2021(8225)	0x2041(8257)	0x2061(8289)	…….	0x27C1(10177)
+ 0x02(+2)	0x2002(8194)	0x2022(8226)	0x2042(8258)	0x2062(8290)	…….	0x27C2(10178)
+ 0x03(+3)	0x2003(8195)	0x2023(8227)	0x2043(8259)	0x2063(8291)	…….	0x27C3(10179)
+ 0x04(+4)	0x2004(8196)	0x2024(8228)	0x2044(8260)	0x2064(8292)	…….	0x27C4(10180)
+ 0x05(+5)	0x2005(8197)	0x2025(8229)	0x2045(8261)	0x2065(8293)	…….	0x27C5(10181)
+ 0x06(+6)	0x2006(8198)	0x2026(8230)	0x2046(8262)	0x2066(8294)	…….	0x27C6(10182)
+ 0x07(+7)	0x2007(8199)	0x2027(8231)	0x2047(8263)	0x2067(8295)	…….	0x27C7(10183)
+ 0x08(+8)	0x2008(8200)	0x2028(8232)	0x2048(8264)	0x2068(8296)	…….	0x27C8(10184)
+ 0x09(+9)	0x2009(8201)	0x2029(8233)	0x2049(8265)	0x2069(8297)	…….	0x27C9(10185)
+ 0x0A(+10)	0x200A(8202)	0x202A(8234)	0x204A(8266)	0x206A(8298)	…….	0x27CA(10186)
+ 0x0B(+11)	0x200B(8203)	0x202B(8235)	0x204B(8267)	0x206B(8299)	…….	0x27CB(10187)
+ 0x0C(+12)	0x200C(8204)	0x202C(8236)	0x204C(8268)	0x206C(8300)	…….	0x27CC(10188)
+ 0x0D(+13)	0x200D(8205)	0x202D(8237)	0x204D(8269)	0x206D(8301)	…….	0x27CD(10189)
+ 0x0E(+14)	0x200E(8206)	0x202E(8238)	0x204E(8270)	0x206E(8302)	…….	0x27CE(10190)
+ 0x0F(+15)	0x200F(8207)	0x202F(8239)	0x204F(8271)	0x206F(8303)	…….	0x27CF(10191)
+ 0x10(+16)	0x2010(8208)	0x2030(8240)	0x2050(8272)	0x2070(8304)	…….	0x27D0(10192)
+ 0x11(+17)	0x2011(8209)	0x2031(8241)	0x2051(8273)	0x2071(8305)	…….	0x27D1(10193)
+ 0x12(+18)	0x2012(8210)	0x2032(8242)	0x2052(8274)	0x2072(8306)	…….	0x27D2(10194)
+ 0x13(+19)	0x2013(8211)	0x2033(8243)	0x2053(8275)	0x2073(8307)	…….	0x27D3(10195)
+ 0x14(+20)	0x2014(8212)	0x2034(8244)	0x2054(8276)	0x2074(8308)	…….	0x27D4(10196)
+ 0x15(+21)	0x2015(8213)	0x2035(8245)	0x2055(8277)	0x2075(8309)	…….	0x27D5(10197)
+ 0x16(+22)	0x2016(8214)	0x2036(8246)	0x2056(8278)	0x2076(8310)	…….	0x27D6(10198)
+ 0x17(+23)	0x2017(8215)	0x2037(8247)	0x2057(8279)	0x2077(8311)	…….	0x27D7(10199)
+ 0x18(+24)	0x2018(8216)	0x2038(8248)	0x2058(8280)	0x2078(8312)	…….	0x27D8(10200)
+ 0x19(+25)	0x2018(8217)	0x2038(8249)	0x2058(8281)	0x2078(8313)	…….	0x27D9(10201)
+ 0x1A(+26)	0x201A(8218)	0x203A(8250)	0x205A(8282)	0x207A(8314)	…….	0x27DA(10202)
+ 0x1B(+27)	0x201B(8219)	0x203B(8251)	0x205B(8283)	0x207B(8315)	…….	0x27DB(10203)
+ 0x1C(+28)	0x201C(8220)	0x203C(8252)	0x205C(8284)	0x207C(8316)	…….	0x27DC(10204)
+ 0x1D(+29)	0x201D(8221)	0x203D(8253)	0x205D(8285)	0x207D(8317)	…….	0x27DD(10205)
+ 0x1E(+30)	0x201E(8222)	0x203E(8254)	0x205E(8286)	0x207E(8318)	…….	0x27DE(10206)
+ 0x1F(+31)	0x201F(8223)	0x203F(8255)	0x205F(8287)	0x207F(8319)	…….	0x27DF(10207)

Address Offset	Access	Type, Size	Description
+ 0x02(+2) **	Read	1word	Input start register address of input image word this slot.
+ 0x03(+3) **	Read	1word	Input word's bit offset of input image word this slot.
+ 0x04(+4) **	Read	1word	Output start register address of output image word this slot.
+ 0x05(+5) **	Read	1word	Output word's bit offset of output image word this slot.
+ 0x06(+6) **	Read	1word	Input bit start address of input image bit this slot.
+ 0x07(+7) **	Read	1word	Output bit start address of output image bit this slot.
+ 0x08(+8) **	Read	1word	Size of input bit this slot
+ 0x09(+9) **	Read	1word	Size of output bit this slot
+ 0x0A(+10)**	Read	n word	Read input data this slot
+ 0x0B(+11)**	Read/Write	n word	Read/write output data this slot
+ 0x0E(+14)	Read	1word	GT-number, if GT-1238, returns 0x1238
+ 0x0F(+15)	Read	String upto 72byte	First 1word 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	1word	Size of configuration parameter byte
+ 0x11(+17)**	Read/Write	n word	Read/write Configuration parameter data, Refer to each IO parameter Specification.
+ 0x17(+23)	Read	2word	Firmware Revision ex) 0x00010010 (Major revision 1 /Minor revision 1, Rev 1.001)
+ 0x19(+25)	Read	2word	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.

4.4. 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