ETH_100 User Manual V1.1

January,2013

Content

1. Overview

ETH_100 offers an industrial-grade solution for remote monitoring, which manages simple and reliable network access. It can realize “seamless connection” with DWIN DGUS LCMs, and can operate with simple configuration without requirement for development of complex network protocol. Main features of ETH_100 are as follows:

- 10M/100M Ethernet physical layer connection; support Auto-MDIX;
- Automatically attempt to establish connection with server; support such functions as DHCP and DNS, etc.; support access of LAN or WAN;
- Provide RS232 and RS485 for the user, and carry out RTDX and non-instant data exchange through network interface and server host;
- Provide two versions, namely transparent proxy (ETH_100A) and black box (ETH_100B);
  1) The transparent proxy version (ETH_100A):
     - ✧ manage real-time transparent proxy of commands and data of RS232 or RS485 to the remote server host, which is sent by the remote server to the user;
  2) The black box version (ETH_100B)
     - ✧ manage real-time storage of data of RS232 or RS485 in internal SRAM (buffer cache for the user) and real-time uploading of data to the remote server host according to the setting.
     - ✧ offers onboard 1GB-4GB industrial SLC NAND Flash (effective P/E cycles: 100 thousand; data storage time: 10 years);
     - ✧ the user can regularly store images of buffer cache in NAND Flash with time stamp, to make export and playback of the black box data in the future more convenient. The black box version performs realtime transparent proxy of data sent by the remote server to the user;
- Debugging information output (RS232);
- Provide software for the user to configure the register, and helping the user set and changing the configuration parameters of ETH100.

2. List of Memories for Device Configuration

Every ETH_100 device uses 64-byte memory to describe the configuration information. The detailed definition of memories are shown in the following table. The variable address (Byte) 0x0B and 0x1A of ETH_100A of the transparent proxy version are 0x00 and 0x0000 respectively.

Variable address		Length （Bytes）	Definition	Specification	Default
Byte	DGUS				
0x00	0x00	10	Model and version of ETH_100		ETH100V11
0x0A	0x05:H	1	Enable DHCP	DHCP function 0x00: disable 0x01: enable	0x01
0x0B	0x05:L	1	Enable the storage	Storage function of the black box	0x00

			function of the black box	0x00: disable 0x01: enable	
0x0C	0x06:H	1	Storage time interval of the black box	Unit: 10s; 0x01 indicates 1 storage per 10s; 0x00: N/A.	0x06
0x0D	0x06:L	1	Baud rate of RS485 of ETH_100	The baud rate of RS232: 115200 bps; the baud rate of RS485 is configurable, and 0x00-0x0A is optional.	0x07
0x0E	0x07	4	IP address of ETH_100		192.168.1.100
0x12	0x09	4	IP address of router		192.168.1.1
0x16	0x0B	4	Subnet mask of ETH_100		255.255.255.0
0x1A	0x0D	2	Time interval of automatic data sending to Ethernet	Unit: 10s; 0x0001 indicates 1 storage per 10s; 0x0000 indicates data transparent proxy instead of storage in 56KB buffer cache.	0x0000
0x1C	0x0E:H	1	Size of automatic data sending to server	Unit: 1KB, with the value scope of 0-56; Maximum uploading value: 56KB; Default condition: start uploading from address 0 of buffer cache; size of buffer cache: 56KB. 0 indicates no data uploading.	0x01
0x1D	0x0E:L	1	Enable RS485 communication (high 4bit) and CRC check (low 4bit)	Enable RS485 communication: 0x0: disable, 0x1: enable. (after enabling the function, RS485 activated; otherwise, RS232 activated, which is effective during playback of the black box and transparent proxy data sent by Ethernet). Enable RS485 RCR check; 0x0: disable, 0x1: enable	0x00
0x1E	0x0F	2	Address of RS485	Configured address during RS485 communication	0x0001
0x20	0x10	2	Frame header of RS485	Configured frame header during RS485 communication	0x5AA5
0x22 - 0x3F	0x11	30	Reserve	Reserve	N/A
0x40	0x20	6	MAC address of		Unique

			ETH_100		factory numbe r
0x46	0x23	4	IP address of host server		192.16 8.1.2
0x4A	0x25	40	Domain name of host server		www.d win.co m.cn
0x72	0x39	2	Port number of host server		1000
0x74	0x3A: H	1	reserve	reserve	N/A
0x75	0x3A:L	1	Server connection way of ETH_100	0x00: connection through IP address of server 0x01: connection through domain name of server	0
0x76	0x3B	4	Address of DNS server	Set the address of DNS server when connection with domain name and closing of DHCP function with static IP address, so as to request the server for DNS.	192.16 8.1.1
0x7A	0x3D: H	1	Enable debugging information output	0x00: disable, no output of debugging information; 0x01: enable, output of partial ETH_100 operation information through RS232	0x01
0x7B	0x3D:L	1	Reserve	Reserve	
0x7C	0x3E	2	Feature check code	The code shall be 0xCCCC, which indicates that the device has been configurated correctly.	0xCCC C
0x7E	0x3F:H	1	Flash capacity version	1: 1GB, 2: 2GB 3: 3GB, 4: 4GB	2
0x7F	0x3F:L	1	Reserve	Reserve	

ETH_100 has 3 indicator lamps：1 power indicator lamp, 1 DHCP indicator lamp, and 1 indicator lamp for connection with the remote server. DHCP indicator lamp is off when DHCP function is disabled, flashes when the DHCP applies to the router for dynamic IP, address and always-on when DHCP successfully obtains dynamic IP address. The connection indicator lamp flashes when ETH_100 tries to establish connection with the server, and is always-on after ETH_100 establishes connection with the server successfully. Every time the user updates the memory space, ETH_100 will restart to make the new configuration effective.

- Model and version of ETH_100: describe ETH_100 with word string; default version: “ETH100V11”; the last byte is the end mark; total quantity of bytes: 10
- Enable DHCP: after the function is opened, ETH_100 will apply for dynamic IP (the router shall open DHCP function), and obtain router IP, subnet mask and DNS server address. Subsequent functions including DNS and connection etc. will not be realized until ETH_100 applies for dynamic IP

successfully. DHCP indicator lamp flashes when ETH_100 applies for dynamic IP, and is always-on when ETH_100 obtains dynamic IP successfully. DHCP has the automatic renewal function.

- Enable the storage function of the black box: ETH_100 supports storage of 56KB data of RAM in Flash according to time (actually occupying 64KB Flash space for convenient Flash storage). 2GB SLC NAND Flash can store 56KB data for 32768 times. Calculated according to backup per 60s, the total backup time exceeds 500h. If the storage function of the black box can be enabled after power-on, ETH_100 will automatically obtain current time from DGUS LCMs and update RTC clock setting.
- The baud rate of RS485 of ETH_100: the default is 0x07, 115200bps. The configurated scope: 0x00 - 0x0A.

Baud_485	0x00	0x01	0x02	0x03	0x04	0x05
baud rate	2400	2400	4800	9600	19200	38400
Baud_485	0x06	0x07	0x08	0x09	0x0A	
baud rate	57600	115200	28800	76800	62500	

- IP address of ETH_100: the IP address is used when DHCP function is disabled.
- IP address of router: the IP address is used when DHCP function is disabled.
- Subnet mask of ETH_100: the subnet mask is used when DHCP function is disabled.
- RS485 address: RS485 bus address; the default: 0x0001.
- MAC address of ETH_100: MAC address stores the unique factory number of 6 bytes. The user shall not change the address.
- IP address of host server: this IP address is used when LAN uses ETH_100 or WAN knows IP address of the server (connection with the server IP).
- Domain name of host server: use obtained IP for connection when the connection is carried out through the domain name of the server, ETH_100 applies for DNS, and the DNS is successful.
- DNS server address: this address is effective when the connection is carried out through the domain name, and is used to apply to the server for DNS. When DHCP is enabled, DNS server address is obtained by DHCP; when DHCP is disabled, the user shall appoint DNS server address.
- Hardware version: at present, ETH_100 plans to provide two hardware versions, namely transparent proxy version (ETH_100A) and 2GB Flash black box version (ETH_100B). The storage function of the black box shall be disabled in the transparent proxy version. In ETH_100B, the user can also enable the transparent proxy function; the difference is that the user can open the storage function of the black box and carry out data playback with the play function of the black box at the same time.
- RTC time synchronization function: if the storage function of the black box is enabled, ETH_100 will automatically obtain current time from DGUS LCMs and update RTC clock setting after power-on. If ETH_100 fails to obtain data from DGUS LCMs, ETH_100 will stop to prevent incorrect playback due to incorrect time stamp. Therefore, connect DGUS LCMs with ETH_100 after RTC time setting.

3. Communication Protocol

3.1 RS232 Communication Interface

Mode of serial port:

115200bps, N81 (1 start bit, 8 data bits, 1 stop bit, and no check bit)

Architecture of data frame:

Address	0x00	0x02	0x03	0x04
Definition	0x5AA5	Length	Command	DATA_Pack
Explanation	frame header	length of subsequent data	command	data

Specification of command set:

- **0x31 command (Command = 0x31) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x31) of configuration memory reading to ETH_100, and ETH_100 returns all configurations of 128 bytes during response to the command. The frame structure: frame header (0x5AA5) + Length (0x03) + Command (0x31) + ADR_H + ADR_L.

Note: ETH_100 neglects received ADR_H and ADR_L, and returns all configurations. ADR_H and ADR_L are returned to DGUS LCMs during response of ETH_100, which are used to write DGUS LCMs to this address. Response of ETH_100: frame header (0x5AA5) + Length (0x83) + Command (0x82) + ADR_H +ADR_L + DATA (128 bytes)

- **0x32 command (Command = 0x32) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x32) of configuration memory writing to ETH_100. The frame structure: frame header (0x5AA5) + Length (byte length) + Command (0x32) + 0x00 + ADR (0x00-0x1F) + WR_LEN (word length) + DATA.

Note: DGUS LCMs strictly forbids the user to operate areas after 32 words (64 bytes) (word address: 0x20-0x3F), while the upper computer software can operate all areas. If the user wants to change contents of 0x20 - 0x3F, the user shall use the upper computer software provided by DWIN. DGUS LCMs write information of any length into ETH_100. After sending command of memory writing, ETH_100 will immediately store configuration in internal Flash and restart.

- **0x33 command (Command = 0x33) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x33) of black box playback to ETH_100. The frame structure: frame header (0x5AA5) + Length (0x0F) + Command (0x33) + starting time of play by the black box (BCD coding, 6 bytes, YY:MM:DD HH:MM:SS) + ending time of play by the black box (the ending time shall be larger than the starting time, BCD coding, 6 bytes, YY:MM:DD HH:MM:SS) + size of played data of the black box (1 byte; unit: 2KB; value scope: 1-28; 1 indicates 2KB; the maximum uploading quantity: 56KB; start play from address 0 of the buffer cache; 0x00: N/A) + play time interval of the black box (1 byte; unit: 1s; 0x01 indicates one play of data of the black box per 1s; 0x00: N/A; value scope: 1s -255s). After receipt of the command, ETH_100 starts playback of data stored in NAND Flash once according to starting time and ending time.

- **0x34 command (Command = 0x34) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x34) of stopping playback of the black box to ETH_100. The frame structure: frame header (0x5AA5) + Length (0x01) + Command (0x34). After receipt of the command, ETH_100 stops the playback immediately.

- **0x82 command (Command = 0x82) (ETH_100→DGUS LCMs)**

ETH_100 sends the command (0x82) of variable memory writing to DGUS LCMs. The frame structure: frame header (0x5AA5) + Length + Command (0x82) + ADR_H +ADR_L + DATA. The command is effective during response of ETH_100 to 0x31 command and playback of data of the black box.

- **0x83 command (Command = 0x83) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x83) of variable memory writing to ETH_100. The frame structure: frame header (0x5AA5) + Length + Command (0x83) + ADR_H +ADR_L + WR_LEN (word

length) + DATA. ETH_100 writes received data into 56KB buffer cache space according to the address. Note: ADR is word address; ADR x 2 for address translation when ETH_100 writes it into the variable memory. Carry out transparent proxy of 0x83 command when the time interval of automatic data sending to Ethernet is 0s.

- **0x81 command (Command = 0x81) (DGUS LCMs ←→ETH_100)**

The command is effective only when ETH_100 enables the storage function of the black box. It's used for ETH_100 to read RTC clock of DGUS LCMs and power-on clock synchronization. After power-on, ETH_100 sends 0x81 command to RTC memory of DGUS LCMs; the frame structure: frame header (0x5AA5) + Length (0x03) + Command (0x81) + ADR (0x20) + RD_LEN (0x07, byte length). Response of DGUS LCMs to ETH_100: frame header (0x5AA5) + Length(0x0A) + Command (0x81) + ADR (0x20) + RD_LEN(0x07, byte length) + DATA. ETH_100 shall continuously read DGUS LCMs twice and compare if two results are consistent. If yes, write RTC information into own RTC memory.

- **other commands (DGUS LCMs→ETH_100→server host)**

Other commands are effective only when the time interval of automatic data sending to Ethernet is 0s (namely when the transparent proxy function is enabled). Note: transparent proxy commands shall not use the above commands (except for 0x83 command). After successful transparent proxy, ETH_100 sends response to DGUS LCMs; the frame structure: frame header (0x5AA5) + Length (0x02) + 0xC1 + 0x1C.

3.2 RS485 Communication Interface

Mode of serial port:

2400bps-115200bps; the baud rate is optional; N81 (1 start bit, 8 data bits, 1 stop bit, and no check bit). During RS485 bus communication, ETH_100 shall monitor the bus before active sending; if the bus is idle, ETH_100 can start active sending.

Architecture of data frame:

Address	0x00	0x02	0x03	0x04	
Definition	RS485 frame header is optional	Length	Command	DATA_Pack	CRC16
Explanation	frame header	length of subsequent data	command	data	check

In which: CRC16 is CRC checksum of command and data, with two bytes. (ANSI standard, X16+X15+X2+1).

Specification of command set:

Note: judge if CRC check of frame tail is effective according to RS485 CRC check enabling configuration in configuration memory.

- **0x31 command (Command = 0x31) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x31) of configuration memory reading to ETH_100, and ETH_100 returns all configurations of 128 bytes during response to the command. The frame structure: frame header (RS485 frame header is optional) + Length (0x03 or 0x05) + Command (0x31) + ADR_H + ADR_L + CRC. Note: ETH_100 will judge if the command is sent to itself according to RS485 bus address ADR485 (ADR_H, ADR_L= ADR485*64); if yes, ETH_100 shall return all contents of

configuration memory; ADR_H and ADR_L are returned to DGUS LCMs during response of ETH_100, to write DGUS LCMs to this address. Response of ETH_100: frame header (RS485 frame header is optional) + Length (0x83 or 0x85) + Command (0x82) + ADR_H +ADR_L + DATA (128 bytes) + CRC

- **0x32 command (Command = 0x32) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x32) of configuration memory writing to ETH_100. The frame structure: frame header (RS485 frame header is optional) + Length (byte length) + Command(0x32) + ADR_H + ADR_L + WR_LEN (word length) + DATA + CRC. ETH_100 will judge if the command is sent to itself according to RS485 bus address ADR485 (ADR_H, ADR_L between ADR485*64 and ADR485*64+0x20); if yes, after sending command of configuration memory writing, ETH_100 will immediately store configuration in internal Flash and restart. Note: RS485 strictly forbids the user to operate areas after 32 words (64 bytes) (word address: 0x20-0x3F).

- **0x33 command (Command = 0x33) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x33) of black box playback to ETH_100. The frame structure: frame header (0x5AA5) + Length (0x0F or 0x11) + Command (0x33) + starting time of play by the black box (BCD coding, 6 bytes, YY:MM:DD HH:MM:SS) + ending time of play by the black box (the ending time shall be larger than the starting time, BCD coding, 6 bytes, YY:MM:DD HH:MM:SS) + size of played data of the black box (1 byte; unit: 2KB; value scope: 1-28; 1 indicates 2KB; the maximum uploading quantity: 56KB; start play from address 0 of the buffer cache; 0x00: N/A) + play time interval of the black box (1 byte; unit: 1s; 0x01 indicates one play of data of the black box per 1s; 0x00: N/A; value scope: 1s -255s) + CRC. After receipt of the command, ETH_100 starts playback of data stored in NAND Flash once according to starting time and ending time.

- **0x34 command (Command = 0x34) (DGUS LCMs→ETH_100)**

DGUS LCMs sends the command (0x34) of stopping playback of the black box to ETH_100. The frame structure: frame header (0x5AA5) + Length (0x01 or 0x03) + Command (0x34) + CRC. After receipt of the command, ETH_100 stops the playback immediately.

- **0x82 command (Command = 0x82) (ETH_100→DGUS LCMs)**

ETH_100 sends the command (0x82) of variable memory writing to DGUS LCMs. The frame structure: frame header (RS485 frame header is optional) + Length + Command (0x82) + ADR_H +ADR_L + DATA+ CRC. The command is effective during response of ETH_100 to 0x31 command and playback of data of the black box.

- **0x82 command (Command = 0x82) (RS485 bus device→ETH_100)**

The user can use this command to write in internal 56KB buffer cache of ETH_100; when the command is used by external bus device to write DGUS LCMs, ETH_100 monitors the command and writes corresponding contents in own 56KB buffer cache. The frame structure: frame header (RS485 frame header is optional) + Length + Command (0x82) + ADR_H +ADR_L + DATA + CRC. ETH_100 writes the data in 56KB buffer cache according to the address. Note: ADR is word address; ADR x 2 for address translation when ETH_100 writes it into the variable memory. Carry out transparent proxy of 0x82 command when the time interval of automatic data sending to Ethernet is 0s.

- **0x83 command (Command = 0x83) (DGUS LCMs→ETH_100)**

The user can use this command for writing of internal 56KB buffer cache of ETH_100. This command can also be used for reading response of DGUS LCMs to external bus device; ETH_100 monitors this command and writes corresponding contents in own 56KB buffer cache. That means 0x82 command and 0x83 command can be written in the internal buffer cache of ETH_100. The frame structure: frame header (RS485 frame header is optional) + Length + Command (0x83) + ADR_H +ADR_L + WR_LEN + DATA + CRC. ETH_100 writes the data in 56KB buffer cache according to the address. Note: ADR is word address; ADR x 2 for address translation when ETH_100 writes it into the

variable memory. Besides, ETH_100 will judge the Length (if CRC is enabled, Length shall be larger than 6; if CRC is disabled, Length shall be larger than 4), so as to ensure that 0x83 command is the response of DGUS LCMs instead of reading command of external bus device to DGUS LCMs. Carry out transparent proxy of 0x83 command when the time interval of automatic data sending to Ethernet is 0s.

- **0x81 command (Command = 0x81) (DGUS LCMs ←→ETH_100)**

The command is effective only when ETH_100 enables the storage function of the black box. It's used for ETH_100 to read RTC clock of DGUS LCMs and power-on clock synchronization. After power-on, ETH_100 sends 0x81 command to RTC memory of DGUS LCMs; the frame structure: frame header (RS485 frame header is optional) + Length + Command (0x81) + ADR (0x20) + RD_LEN (0x07, byte length) + CRC. Response of DGUS LCMs to ETH_100: frame header (RS485 frame header is optional) + Length (0x0A or 0x0C) + Command (0x81) + ADR (0x20) + RD_LEN (0x07, byte length) + DATA + CRC. ETH_100 shall continuously read DGUS LCMs twice and compare if two results are consistent. If yes, write RTC information into own RTC memory.

- **other commands (DGUS LCMs→ETH_100→server host)**

Other commands are effective only when the time interval of automatic data sending to Ethernet is 0s (namely when the transparent proxy function is enabled). Note: transparent proxy commands shall not use the above commands (except for 0x82 command and 0x83 command). The frame structure: frame header (RS485 frame header is optional) + Length + Command + ADR_H +ADR_L + DATA + CRC. ADR_H, ADR_L shall be ADR485*64; otherwise, transparent proxy is not allowed. If CRC is enabled, transparent proxy of CRC data is also completed. After successful transparent proxy, ETH_100 sends response to DGUS LCMs; the frame structure: frame header (RS485 frame header is optional) + Length (0x02 or 0x04) + 0xC1 + 0x1C + CRC.

4. Network Communication Interface

Data structure of a typical Ethernet frame:

Ethernet header	IP header	TCP header	Application data	Ethernet tail
14 bytes	20 bytes	20 bytes		4 bytes

After power-on, ETH_100 will constantly try to establish TCP connection with the remote server host until the connection is successful. The remote server shall be TCP Server and monitor port 1000; ETH_100 shall be TCP Client and initiate connection with port 1000. If ETH_100 doesn't receive response during connection initialization, ETH_100 will automatically re-initialize the connection, and, in case of time-out, automatically exit and re-try connection with the server host. If the server host receives connection request from ETH_100, the server host shall release previous connection with ETH_100 and re-connect with ETH_100. The data exchange between ETH_100 and the remote server host is placed in the application data segment of Ethernet frame.

4.1 Communication from ETH_100 to Server

Architecture of data frame:

Address	0x00	0x02	0x04	0x05
Definition	0x5AA5	Length(2 Bytes)	Command	DATA_Pack
Explanation	frame header	length of subsequent data	Ccommand	data

Specification of command set:

- **0x82 command (Command = 0x82) (ETH_100→server host)**

ETH_100 can upload data segments regularly, with the data size regulated by corresponding configuration of the memory. The frame structure: frame header (0x5AA5) + Length (0x0402, 1026 bytes) + Command (0x82) + Sequence + DATA. Sequence
uploading, the serial number of the first data package is1. Every data package uploads data in 1KB buffer cache; for 8KB uploaded data, the serial number of the last data package is 8. If the server host continuously receives No. 1-8 data packages, 8KB data is uploaded completely.

- **0x31 command (Command = 0x31) (ETH_100→server host)**

The command is the response of ETH_100 to 0x31 command of the server host. The frame structure: frame header (0x5AA5) + Length (0x0082) + Command (0x31) + Sequence (0x01) + DATA (128 bytes).

- **0x35 command (Command = 0x35) (ETH_100→server host)**

ETH_100 uploads own MAC address to the server after receipt of MAC address reading command of the server host. The frame structure: frame header (0x5AA5) + Length (0x0007) + Command (0x35) + MAC_ADR(6 bytes).

4.2 Communication from Server to ETH_100

Architecture of data frame:

Address	0x00	0x02	0x03	0x04
Definition	0x5AA5	Length(1 Byte)	Command	DATA_Pack
Explanation	frame header	length of subsequent data	command	data

Specification of command set:

- **0x31 command (Command = 0x31) (server host→ETH_100)**

The server host sends the configuration memory reading command (0x31) to ETH_100, and ETH_100 returns all configurations of 128 bytes (0x80) during response to the command. The frame structure: frame header (0x5AA5) + Length (0x0082) + Command (0x31) + Sequence (0x01) + DATA (128 bytes).

- **0x35 command (Command = 0x35) (server host→ETH_100)**

The server host sends MAC address reading command to ETH_100, and ETH_100 uploads its MAC address to the server after receipt of the command. The frame structure: frame header (0x5AA5) + Length (0x01) + Command (0x35).

- **0x3B command (Command = 0x3B) (server host→ETH_100→DGUS LCMs)**

The server host ends downloading mode command (0x3B) to ETH_100, and ETH_100 sends the command to DGUS LCMs. After entering the downloading mode, carry out transparent proxy of data from all servers regardless of frame header. During the transparent proxy, ETH_100 automatically divides Ethernet data packages, so that the sizes of data packages sent to DGUS LCMs are within the allowable scope of DGUS LCMs. The frame structure: frame header (0x5AA5) + Length (0x01 or 0x03) + Command (0x3B) + CRC (the user judges the addition according to the the following situations: if RS485 is used and if RCR is enabled).

- **0x3C command (Command = 0x3C) (server host→ETH_100→DGUS LCMs)**

The server host sends the command of exiting the downloading mode (0x3C) to ETH_100, and ETH_100 sends the command to DGUS LCMs and returns to normal operation condition. The frame

structure: frame header (0x5AA5) + Length (0x01 or 0x03) + Command (0x3C) + CRC (the user judges the addition according to the the following situations: if RS485 is used and if RCR is enabled).

- **other commands (server host→ETH_100→DGUS LCMs)**

ETH_100 performs transparent proxy of data frames of the server to DGUS LCMs according to other commands. Note: the maximum data length for a frame of DGUS LCMs is 254 bytes (without CRC check) or 252 bytes (with CRC check). ETH_100 carries transparent proxy to RS232 serial port or RS485 serial port according to the setting (RS485 communication enabling parameters). Replace frame header of RS485 according to requirements during transparent proxy to RS485 serial port (CRC check is added by the server host during data sending).

5. Router Port Mapping

If the user's server is accessed to WAN through router, it's suggested to set IP address of the user's server in router as a static IP address, so as to prevent mapping failure due to change of IP address of the user's server after the port mapping function is set in the router. For example, if IP address of the user's server in router is 192.168.0.253, the monitoring port is 1000, and it's required to connect ETH_100 with the server port, map the port 1000 of intranet 192.168.0.253 to the internet in the router, so that ETH_100 can directly visit services provided by the intranet server through internet IP and port of the router.

6. Specification of Upper Computer Programming Configuration Software

DWIN provides an software for programming ETH_100, which can rapidly set various parameters of ETH_100. After setting the parameters, hit "set" button for parameter configuration of ETH_100. The user can store the current parameters into a document by clicking on "store the setting" button, and read the parameters later by hitting "import the setting" button. "Read the configuration" button can read and display the current configuration of ETH_100; "format" button is NAND Flash for ETH_100 formatting, to help the user erase data stored in Flash. On the page of "black box data playback test", the user can conveniently set starting and ending times of playback, data size and other parameters.