# 8.2 User Defined Protocol Communication

### 8.2.1 Introduction

User Defined Protocol Communication allows users who do communication between GM7 Basic Unit and other kind of device to define the other company's protocol at GLOFA PLC. There re a number of kinds of protocols made by many companies, that it s difficult to have all protocols in it. So if a user defines a protocol that can be applied to his/her purpose, GM7 Basic Unit executes the communication with the other kind of device through the defined protocol. For this, protocol frame must be defined in GMWIN (Version 3.3 or higher). And exact knowledge about the contents of the protocol defined by the user is vital in making the communication possible. GMWIN (Version 3.3 or higher) can download a user defined protocol frame into GM7 Basic Unit and it is saved(it is not erased by power's off/on). But protocol frames are damaged to download with changes of parameter, or to fail to back up the data, caused by lower back-up battery voltage than the standard. For using user-defined mode, he/she should program with function block controlling sending/receiving of PLC as well as edit frames. This section explains UDPC setting & usage.

#### Remark

 All numerical data can use hexadecimal, decimal, and binary type. If we convert decimal 7 and 10 into each type: Hexadecimal : H07, H0A or 16#07, 16#0A Decimal : 7, 10 Binary : 2#0111, 2#1010

# 8.2.2 Parameter Setting

- 1) Setting Communications Parameter
  - Open a new project file from GMWIN Select GM7 as PLC type Make a separate project file for each of the master and slave.
  - (2) After setting communication parameter at GMWIN. Double click it to activate this window.

	neter
Communication M	ethod
Station No. :	0 💌
Baud Rate :	19200 💌 Data Bit : 8 💌
Parity Bit :	None 🔻 Stop Bit : 1 💌
_ Communication	n Channel
C RS232C Nu	JII Modem or RS422/485
C RS232C Mc	odem (Dedicated Line) Init. Command :
C RS232C Dia	al Up Modem
Dedicated	
Dedicated C Mast	er 🗖 Read Status of Slave PLC List
Dedicated C Mast C Slave Modbus ——	er E Read Status of Slave PLC List
Dedicated C Mast C Slave Modbus C Mast C Slave	er Eransmission Mode : ASCII
Dedicated C Mast C Slave Modbus C Mast C Slave User Defined	er Read Status of Slave PLC List er Transmission Mode : ASCI
Dedicated C Mast C Slave Modbus C Mast C Slave User Defined C Mast	er Read Status of Slave PLC List er Transmission Mode : ASCII

(3) Set according to the following table.

Item	Setting range
Station No.	Station no. from 0 to 31.
Baud Rate	1200, 2400, 4800, 9600, 19200, 38400, 57600 bps
Data Bit	7 or 8 bits
Parity Bit	0, Even or Odd
Stop Bit	1 or 2 bit(s)
Communication Channel	<ul> <li>RS232C Null Modem or RS422/485 : It's a communication channel for the communication, using GM7 base unit's built-in communication and Cnet I/F module (G7L-CUEC).</li> <li>RS232C Modem (Dedicated Line) : It's to be selected for the communication, using an dedicated modem with Cnet I/F module (G7L-CUEB).</li> <li>RS232C Dial Up Modem : It's to be selected for the general communication connecting through the telephone line by dial up modem and Cnet I/F module (G7L-CUEB).</li> <li>Footnote) Using Cnet I/F module (G7L-CUEB) supporting RS232C, RS232C dedicated or dial-up modem communication can be done, but not through Cnet I/F module (G7L-CUEC) supporting RS422/485.</li> </ul>
Timeout in Master Mode	<ul> <li>It's the time waiting a responding frame since the master GM7 base unit sends a request frame.</li> <li>The default value is 500ms.</li> <li>It must be set in consideration of the max. periodical time for sending/receiving of the master PLC.</li> <li>If it's set smaller than the max. send/receive periodical time, it may cause communication error.</li> </ul>
User Define Master / Slave	If it is set as the master, it s the subject in the communication system. If it s set as the slave, it only responds to the request frame of the master.

2) Setting frame

(1) Select one out of user defined terms of protocol and mode in communication parameter, registration "List' button is activated.

User Defined	
C Master	Liet
<ul> <li>Slave</li> </ul>	List

0 Not defined 1 Not defined 2 Not defined 3 Not defined 4 Not defined 5 Not defined 6 Not defined 8 Not defined 10 Not defined 11 Not defined 12 Not defined 13 Not defined 14 Not defined 15 Not defined	Tame Information Tx/Rx: Header: SG1: SG2: SG3: SG4: SG5: SG6: SG6: SG7: SG8: Tailer: BCC:	
--	--	--

(2) Click "List" button to activate the following window.

(3) Select one of 1 15 in frame list to open the following window.

Frame O					×
Header :	Ţ			Tx/Rx : Not define	ed 🔽
Segment 1 —			1	Segment 5	
Туре:	7			Type:	
C Hexinput	C ASCII Input	Size:	Byte	Hex Input     C ASCII Input	Size: Byte
- Segment 2				Segment 6	
Туре:	-	ļ		Type:	
C Hexinput	C ASCII Input	Size:	Byte	Hex input     C     ASCII input	Size: Byte
Segment 3				Segment 7	
Туре:	7			Type:	
C Hexinput	C ASCII Input	Size:	Byte	Hex Input     C     ABCII Input	Size: Byte
Segment 4				Segment 8	
Туре:	7			Type:	
Hex Input	${f C}$ ABCII Input	Size:	Byte	C Hex Input C ASCII Input	Size: Byte
Tailer:	BC	C Setting		Ok	Cancel

Frame specification

- Header
  - Used in [Header] type.
    Possible characters as h
    - Possible characters as headers are 1 alphabet letter, 1 numeric number, or control characters like "NUL (H00)", "STX (H02)", "ETX(H03)", "EOT(H04)", "ACK(H06)", "NAK(H15)", "SOH(H01)", "ENQ(H05)", "BEL(H07)", "BS(H08)", "HT(H09)", "LF(H0A)", "VT(H0B)", "FF(H0C)", "CR(H0D)", "SO(H0E)", "S1(H0F)", "DLE(H10)", "DC1(H11)", "DC2(H12)", "DC3(H13)", "DC4(H14)", "SYN(H16)", "ETB(H17)", "CAN(H18)", "EM(H19)", "SUB(H1A)", "ESC(H1B)", "FS(H1C)", "GS(H1D)", "RS(H1E)", "US(H1F)" and "DEL(H7F)."

Ex1) [NUL], [ENQ]	(	)	
Ex2) NUL, ENQ	(	<b>x</b> :[	] isn' t used)

- It is allowed to be only 3 consecutive characters.

Ex1) [ENQ][STX][NUL]	(	)
Ex2) [A][NUL][ENQ][STX]	(	×: 4 consecutive characters are used)

- Tx/Rx
  - Not defined : It is the initial value that doesn' t declare a frame format.
  - Send : It is that declares send frame.
  - Receive : It is that declares receive frame.
  - When Frame 0 window is activated, Tx/Rx term is set as "Not defined," and all the segments are not in activation.

Item	Contents
Type: NONE  NONE Fiex NONE CONST ARRAY	To set a segment type, there re NONE (not defined), CONST (fixed data area), ARRAY (variable data area). CONST declares commands and fixed data that are used for communication frame and ARRAY is used to input and save the data needed for interactive communication. ARRAY type must be always set by byte. Ex1) %MB0, %QB0.0.0 () Ex2) %MX0, %MW0, %MD0, %QX0.0.0, %QW0.0. ()
	This field is to declare commands and fixed data that will be used in communication frame and constant data to be declared by inputting. ASCII input must be done within 10 characters and hex within 20 characters. If the number exceeds the limit, set the next segment as the same type and continue to input there. Ex1) 10RSB06%MW10006  Segment1 Type: CONST
O Hex Input       ASCII Input	It is a radio button to select the input type of commands. There re 2 kinds as hex or ASCII value. Ex1) ASCII : 1 0 R S B 0 6 % M W 1 0 0 Ex2) Hex : 31 30 52 53 42 30 36 25 57 44 31 30 30
<ul> <li>Send by ASCII Converting</li> <li>Receive by Hex Converting</li> </ul>	If ARRAY (variable data area) is set, it asks whether it would convert data to ASCII to send (at send frame), or convert to hexadecimal to receive(at receive frame).
Size: Byte	If ARRAY is set, the size of area is to be set by byte. The unit is a byte.

- Segment (1-8): Enter segment by segment to separate fixed sending data area (CONSTANT) and variable data area (Array).

- Tail
  - Used in [Tail] type.
  - Possible characters as tails are 1 alphabet letter, 1 numeric number or control characters like "NUL (H00)", "STX (H02)", "ETX (H03)", "EOT (H04)", "ACK (H06)", "NAK (H15)", "SOH (H01)", "ENQ (H05)", "BEL (H07)", "BS (H08)", "HT (H09)", "LF (H0A)", "VT (H0B)", "FF (H0C)", "CR (H0D)", "SO (H0E)", "S1 (H0F)", "DLE (H10)", "DC1 (H11)", "DC2 (H12)", "DC3 (H13)", "DC4 (H14)", "SYN (H16)", "ETB (H17)", "CAN (H18)", "EM (H19)", "SUB (H1A)", "ESC (H1B)", "FS (H1C)", "GS (H1D)", "RS (H1E)", "US (H1F)", "DEL (H7F)", and "BCC(HFE)."

Ex1) [NUL], [EOT]	(	)	
Ex2) NUL, EOT	(	<b>x</b> :[	] isn' t used)

- It is allowed to be only 3 consecutive characters.

Ex3) [ENQ][STX][NUL]	(	)
Ex4) [A][NUL][ENQ][STX]	(	×: 4 consecutive characters are used)

- It s possible to use BCC that can detect errors. BCC must be set as [BCC] to be used. To set BCC contents, click "BCC Setting" button on the right side.

Ex5)

Tailer:	EOTJ (BCC	BCC Setting
---------	-----------	-------------

BCC setting: set BCC when it is needed.

Data Type	271 MARSA	
C ASCII	Hex	
Check Rule		
C Default		
C SUM 1	C SUM 2	
C XOR 1	C XOR 2	
C MUL 1	O MUL 2	
ange :	ex)	H[0]~T[0]
omplement : None	Mask: ex	) ^FF  FF &FF

Item		Contents
Data Type		ASCII adds 2 bytes BCC value in ASCII type to frame. Hex adds 1 byte BCC value in Hex type to frame. For the detailed setting BCC, refer to 8.1.6 " Execution of Commands".
	Default	It is that sum all the data from 2 <sup>nd</sup> data to the data before the data marked as [BCC] and input the result to the [BCC] area
	SUM 1	BCC method uses sum like defaults, but the user can set the BCC area.
lle	SUM 2	BCC method is the same with SUM 1, but it's used when the user masks any value to the last BCC value.
Check Ru	XOR 1	BCC method is OR (EXCLUSIVE OR).
	XOR 2	BCC method is the same with XOR 1, but it's used when the user masks any value to the last BCC value.
	MUL 1	BCC method is MULTIPLY that is, multiplication.
	MUL 2	BCC method is the same with MUL 1, but it's used when the user masks any value to the last BCC value.
Range		H signifies header, S is for segment, and T is for tail. Ex1) When header is set as [ENQ][STX], tail is set as [EOT][ETX], and the range of setting BCC is to be from [STX] to [ETX], then set as H [1]~T [1].
Complement		It is to set whether not to take complement number or to take the complement number of 1 or 2 at [BCC] value. If mask setting is done after taking a complement number, the user can set any value to do masking.
Mask		Sets any value and method of masking. Ex1) When masking by XOR method, using a value, HFF : ^FF Ex2) When masking by OR method, using a value, HFF :  FF When masking by AND method, using a value, HFF : &FF

Keys on Keyboard, for setting masking method



- Frame size
  - ASCII communication : max. 128 bytes
  - Hex communication : max. 256 bytes
- Flag (\_RCV [n]: n is a frame list no.)
  - It's a flag to indicate whether a user defined frame is received in the order set by the user.
  - It's BOOL type and ARRAY type in the size of 6.
  - If the received frame is matched with the declared frame in frame list number 3, \_RCV [3] starts blinking. (0 1 0)
- When frame receiving is done, GM7 base unit check if there's any match between the received frame and the declared frame in frame list. If there is, let the \_RCV [n] flag blink and save the received data in the assigned area.

# 8.2.3 Function Block

1) SND\_MSG

Function block	Explanation
BOO - REQ NDR - BOOL USINT - FL_ID ERR - BOOL	Input         REQ       : Execute function block at rising edge(0 1)         FL_ID       : Frame list field number to send.         Output
STATUS USINT	NDR: When ends without error, this is set to 1 and keeps till the next request for function block.ERR: When an error occurs, this is set to 1 and keeps till the next request for function block.STATUS: When an error occurs, output error code.

(1) Function

This function block is to send the frame registered in the frame list.

(2) Error

Output error codes to the output STATUS. For the detailed information, refer to the error codes.

2) Error codes (Status of the function block)

CODE	Error type	Meaning
06	Slave Device Busy	It's sending or waiting to receive
09	Parameter Error	Comm. Parameter setting, or communication enable setting error
10	Frame Type Error	The frame isn' t for sending or set.

3) Error flag

Errors related with the frame list Save in \_UD\_ERR [n].

Bit no.	Error type	Meaning	Remark
Bit 0	Frame Length Over	Error when a frame is bigger than 128 bytes (hexadecimal: 256 bytes).	Error flag on
Bit 1	Device Area Over	Error when I/Q area (128 bytes), or M area is exceeded.	sending part
Bit 2	BCC Setting Error	BCC setting error	
Bit 8	BCC Check Error	Error when received BCC value is wrong.	
Bit 9	Mismatch Error	Error when a received frame doesn't match with the registered frame.	Error flag on receiving
Bit 10	Hex Change Error	Error when converting to hexadecimal.	part
Bit 11	Frame List Error	Frame setting or communication enable setting error	

## 8.2.4 Example of Use 1

This example is supposed that there's a communication between GM7's by the user-defined protocol. The system configuration is as follows and the cable is the same with the one of 1:1 dedicated protocol communication.



1:1 dedicated protocol communication between GM7's

The data in M area of the master station is sent to the slave station and the slave station saves the received data in M area outputs as direct variable, and sends the data back to the master. This process repeats between the master and the slave.

- 1) The Programming and setting communication parameter of the slave station
  - (1) Work at the slave station no. 1.
  - (2) Open a new project file and make a new program for the slave station.

GMWIN for Windows - c:₩gmwin₩source₩user_define_master,	prj 📃 🖂 🗙
Project Program Edit Toolbox Compile Online Debug	⊻indow <u>H</u> elp
🚼 c:\#gmwin\#source\#user_define_sla, src	
Row 0 Row 1 Row 2 Row 3 Row 4	
Dem E	
<pre>Complete complete complet</pre>	_ □ × me : UNNAMED ared clared d : c:₩gmwin₩source₩user_define_sla.src ariables declared
Creates a new project	Offline Edit

(3) Select communication parameter in GMWIN parameters and double click it, and then the window of communication parameter will be opens.

municaton Para	ameter
Communication	Method
Station No. :	1
Baud Rate :	9600 🔽 Data Bit : 8 💌
Parity Bit :	None 💌 Stop Bit : 1 💌
Communicati	on Channel
C RS232C	Null Modem or RS422/485
C RS232C M	Nodem (Dedicated Line) Init. Command :
C RS232C D	Dial Up Modern ATZ
Dedicated C Mas	ster 🗖 Read Status of Slave PLC List
C Sla	ve
Modbus —	
C Mas C Sla	ster Transmission Mode : ASCII 🗾
User Defined	
C Mas	ster List
🖲 Sla	
	OK Cancel Help

• Set parameters as the following table.

Communication Method Proto						Protocol and Mode
Statio n No.	Baud Rate	Data Bit	Parity Bit	Stop Bit	Communication channel	User Defined
1	9600	8	None	1	RS232C null modem or RS422/485	Slave

(4) Click "List" to open the following window.

0 Not defined 1 Not defined 2 Not defined 3 Not defined 4 Not defined 5 Not defined 6 Not defined 9 Not defined 10 Not defined 11 Not defined 12 Not defined 13 Not defined 14 Not defined 15 Not defined	Frame Information Tx/Rx: Header: SG1: SG2: SG3: SG4: SG5: SG6: SG6: SG7: SG8: Tailer: BCC:	
--	---	--

(5) Double click "Frame List 0" to activate "Frame 0" window and set as follows.

Frame 0	×
Header : [ENQ]	Tx/Rx : Receive
Segment 1	Segment 5
Type: CONST  SND_FRAME	Type: NONE 🔽
C Hex Input C ASCII Input Size: Byte	C Hex input C ASCII input Size: Byte
Segment 2	Segment 6
Type: ARRAY Y %MB0	Type: NONE
F Receive by Hex Converting Size: 4 Byte	Hex input     ASCII input     Size:     Byte
Segment 3	Segment 7
Type: NONE	Type: NONE
C Hex Input C ASCII Input Size: Byte	👁 Hex Input 🗢 ASCII Input Size: 🗾 Byte
Segment 4	Segment 8
Type: NONE	Type: NONE 💌
C Hexinput C ASCII input Size: Byte	C Hex Input C ASCII Input Size: Byte
Tailer: EOT][BCC] BCC Setting	Ok Cancel

Item	Setting value
Header	[ENQ]
Tx/Rx	Receive
Segment 1	Type: CONST, field: SND_FRAME, select button for ASCII input
Segment 2	Type: ARRAY, field: %MB0, size: 4 bytes
Tailer	[EOT][BCC]

(6) "[BCC]" is set in tail after setting, click "BCC setting" to activate BCC setting window, set as follows and click "OK" to close.

<ul> <li>ASCII</li> </ul>	C Hex	
Check Rule		
C Default		
C SUM 1	SUM 2	
C XOR 1	C XOR 2	
C MUL1	O MUL 2	
Range :	e	x) H[0]~T[0]
Complement : None	Mask: ^FF e	x) ^FF  FF &FF

Item	Setting value
Data Type	ASCII
Check Rule	SUM 2
Range	H [0] ~ T [0] that is from header [ENQ] to tail [EOT].
Complement	None
Mask	Mask with HFF by XOR.

(7) After BCC setting, click "OK" of Frame 0 window to register the frame as follows.

	- Frame Information	
1 Not defined 2 Not defined 3 Not defined 4 Not defined 5 Not defined 6 Not defined 7 Not defined 9 Not defined 10 Not defined 11 Not defined 12 Not defined 13 Not defined 13 Not defined 14 Not defined	Tx/Rx: Receive Header:[ENQ] SG1: constant SG2: hex[4] SG3: null SG4: null SG5: null SG6: null SG7: null SG8: null Tailer:[EOT][BCC] BCC: SUM 2	

(8) Double click frame list 1 to activate Frame 1 window to set as follows.

Frame 1	×
Header: [STX]	Tx/Rx : Send
Segment 1	Segment 5
Type: CONST  RCV_FRAME	Type: NONE 💌
C Hex Input C ASCII Input Size: Byte	C Hex Input O ASCII Input Size: Byte
Segment 2	Segment 6
Type: ARRAY  MB10	Type: NONE
Send by ASCII Converting Size: 4 Byte	C Hex Input C ASCII Input Size: Byte
Segment 3	Segment 7
Type: NONE	Type: NONE
C Hex Input C ASCII Input Size: Byte	C Hex Input C ABCII Input Size: Byte
Segment 4	Segment 8
Type: NONE	Type: NONE
C Hex Input C ASCII Input Size: Byte	C Hex Input C ASCII Input Size: Byte
Tailer: [ETX] BCC Setting	Ok Cancel

Item	Setting value
Header	[STX]
Tx/Rx	Send
Segment 1	Type: CONST, field: RCV_FRAME, select button for ASCII input
Segment 2	Type: ARRAY, field: %MB10, size: 4 bytes
Tailer	[ETX]

(9) After setting, click "OK" of Frame 1 to register as follows.



(10) Click "OK" to exit communication parameter setting and click "OK" to complete setting.

- (11) Setup a program like the following figure and download to the slave station GM7. For the detailed program setting and downloading, refer to GMWIN manual.
  - Function block is used in the program. Before using function block, double click "Current included Libraries" to open the following window. Click "Add" to add COMM.7FB and click "OK."

Current li	ncluded Librarie	95	
- Content in		ю.	
4		r in	
		L bhA	Delete I

🚔 GMWIN	for Windo	ws - (	∷₩gmwir	i₩source	#user_def	ine_slv,p	rj				_ 🗆 🗵
Project	P <u>r</u> ogram	<u>E</u> dit	<u>T</u> oolbox	<u>C</u> ompile	<u>O</u> nline	<u>D</u> ebug	<u>W</u> indow	<u>H</u> elp			
<u>Þ</u>		r and a state of the state of t			* 🖿 0	1		<b>1</b>	<b>.</b>	<u> </u>	₹
Row O	_RCV[0]	EN MO	DVE EN0								- R
Row 1	XMB0	- IN1	OUT - XQ	BO.O.O							<u>L_i</u> ++
Row 2						SRND			·		1/}
Row 3	_RCV[0]	EN M	ENO		R	ND_MSG Eq NDR -					니아 - NH
Row 4	%QB0.0.	0 - IN1	OUT :	(MB10	1 - F D	L_I ERR-	,				
Row 5	8					STAT US					
Row 6					877						$\langle \rangle$
Row 7											(\$)
Row 8											(R)
Row 9	8										() ()
Row 10											{F}
											<b>→</b> {FB}
Saves 1	the prog	yram					Offline			R3,C	8 Edit

- If the frame 0 is received, save it in %MB0 of the slave (refer to frame 0 setting). Output value saved in %MB0 to direct variable %MB10 and again save the value in %QB0.0.0 in %MB10. If this execution is done without error, send frame 1 out of frame list from function block. Frame 1 is the frame that sends 4 bytes data saved in %MB10.
- If Frame 0 isn' t received, this program doesn' t work.
- To open communication, set through "communication enable setting."
- For "communication enable setting" method, refer to 3) Communication enable setting of 8.1.7 "1:1 dedicated protocol communication."

- 2) The programming and setting communication parameter of the master station
  - (1) Work at the master station no. 1.
  - (2) Open a new project file and set up a new program for the master station.

GMWIN for Windows - c:\#gmwin\#source\#user_define_mas.prj	_ 🗆 ×
Project Program Edit Toolbox Compile Online Debug Window Help	
😭 c:\#gmwin\#source\#user_define_mas,src 📃 🗖	
Row 0	
Row 1	+ F +/F
Row 2	-{P}-
Row 3	
Row 4	
Row 5	{}
Row 6	() (3)
	(8) (8)
rac:₩gmwin₩source₩user_define_mas.prj	(9)
PROJECT ==> PLC Type : GM7	(1)
CONFIGURATION(PLC) ==> Configuration Name : UNNAMED	1F1
= RESOURCE(CPU)  0 == Name : RESO	01
	-{FB}-
- TASK DEFINITIONS ==> 1 tasks defined	<ret></ret>
L LD J PROGRAM ==> USER_DEFINE_MAS : c:#gmwin#source#user_defi	
PARAMETERS	
BASIC PARAMETERS	(30)
	-
Creates a new project Offline R8,C7	Edit

(3) If you double click the 'communication parameter' window in GMWIN, you can see the following window of the 'communication parameter.'

Communicaton Parame	eter			×
Communication Met	hod			
Station No. :	0 💌			
Baud Rate :	9600 💌	Data Bit :	8 💌	
Parity Bit :	None 💌	Stop Bit :	1 💌	
Communication (	Channel			
C RS232C Null	Modem or RS422/485			
C RS232C Mod	em (Dedicated Line)	Init. Command	:	
C RS232C Dial	Up Modem	ATZ		
Protocol and Mode – Dedicated	Timeout in M	aster Mode :	500 ms	
C Slave				
Modbus C Master C Slave	Transmi	ssion Mode : 🛛 🛛 🗛	CII	
User Defined Master Slave			List	
	Ok Ca	incel	Help	

• Set parameters as the following table.

	Protocol and Mode					
Station No.	Baud Rate	Data Bit	Parity Bit	Stop Bit	Communication Channel	User Defined
0	9600	8	None	1	RS232C null modem or RS422/485	Master

(4) Click "List" to activate frame list window.

0 Not defined 1 Not defined 2 Not defined 3 Not defined 4 Not defined 5 Not defined 6 Not defined 9 Not defined 10 Not defined 11 Not defined 12 Not defined 13 Not defined 14 Not defined 15 Not defined	Frame Information Tx/Rx: Header: SG1: SG2: SG3: SG4: SG5: SG6: SG6: SG6: SG7: SG8: Tailer: BCC:	
--	---	--

(5) Double click Frame list 0 to activate Frame 0 window and set as follows.

Frame 0	×
Header: [ENQ]	Tx/Rx: Send
Segment 1	Segment 5
Type: CONST  SEND_FRAME	Type: NONE
Hex Input C ASCII Input Size: Byte	Hex Input     O ASCII Input     Size:     Byte
Segment 2	Segment 6
Type: ARRAY 💌 %MB0	Type: NONE
Size: 4 Byte	C Hex Input C ASCII Input Size: Byte
Segment 3	Segment 7
Type: NONE	Type: NONE
C Hex Input C ASCII Input Size: Byte	G Hex Input C ASCII Input Size: Byte
Segment 4	Segment 8
Type: NONE	Type: NONE
C Hex Input C ASCII Input Size: Byte	C Hex Input C ASCII Input Size: Byte
Tailer: EOT][BCC] BCC Setting	Ok Cancel

Item	Setting value
Header	[ENQ]
Tx/Rx	Send
Segment 1	Type: CONST, field: SND_FRAME, select button for ASCII input
Segment 2	Type: ARRAY, field: %MB0, size: 4 bytes
Tailer	[EOT][BCC]

(6) "[BCC]" is set in tail after setting, click "BCC Setting" to activate BCC setting window, set as follows and click "OK" to close.

• ASCII	C Hex	
Check Rule		
C Default		
C SUM 1	SUM 2	
C XOR 1	C XOR 2	
C MUL1	O MUL 2	
Range : H[0]~T[0]		ex) H[0]~T[0]
Complement : None	Mask: ^FF	ex) ^FF  FF &FF

Item	Setting value
Data Type	ASCII
Check Rule	SUM 2
Range	H [0] ~ T [0] that is from header [ENQ] to tail [EOT].
Complement	None
Mask	Mask with HFF by XOR.

(7) After BCC setting, click "OK" of Frame 0 window to register the frame as follows.

	- Frame Information			
0 Receive				
1 Not defined	Tx/Rx: Receive			
2 Not defined	Header/ENO			
3 Not defined	Header.[ENG]			
4 Not defined	SG1: constant			
6 Not defined	SG2: hex[4]			
7 Not defined	SG3: null			
9 Not defined	SG4: null			
10 Not defined	SG5: null			
12 Not defined	SG6: null			
13 Not defined 14 Not defined	SG7: null			
15 Not defined	SG8: null			
	Tailer:[EOT][BCC]			
	BCC: SUM 2			

(8) Double click frame list 1 to activate Frame 1 window to set as follows.

Frame 1	×
Header: [STX]	Tx/Rx : Receive
Segment 1	Segment 5
Type: CONST T RCV_FRAME	Type: NONE 💌
C Hex Input C ASCII Input Size: Byte	C Hex Input C ASCII Input Size: Byte
Segment 2	Segment 6
Type: ARRAY 💌 %MB10	Type: NONE
TReceive by Hex Converting Size: Byte	O Hex input O ASCII input Size: Byte
Segment 3	Segment 7
Type: NONE	Type: NONE
C Hex Input C ASCII Input Size: Byte	C Hex Input C ASCII Input Size: Byte
Segment 4	Segment 8
Type: NONE	Type: NONE
C Hex Input C ABCII Input Size: Byte	C Hex Input C ASCII Input Size: Byte
Tailer: [ETX] BCC Setting	Ok Cancel

Item	Setting value
Header	[STX]
Tx/Rx	Receive
Segment 1	Type: CONST, field: RCV_FRAME, select button for ASCII input
Segment 2	Type: ARRAY, field: %MB10, size: 4 bytes
Tailer	[ETX]

(9) After setting, click "OK" of Frame 1 to register as follows.

and the second	Eromo Information	
0 Send	Frame mormation	
1 Receive	Tx/Rx: Receive	
3 Not defined	Header:[STX]	
4 Not defined	SG1: constant	
6 Not defined	8G2: hex[4]	
7 Not defined	SG3: null	
9 Not defined	SG4: null	
10 Not defined	SG5: null	
12 Not defined	SG6: null	
13 Not defined	SG7: null	
15 Not defined	SG8: null	
	Tailer:[ETX]	
	BCC: Not Set.	

(10) Click "OK" to exit communication parameter setting and click "OK" to complete setting.

- (12) Setup a program like the following figure and download to the slave station GM7. For the detailed program setting and downloading, refer to GMWIN manual.
  - Function block is used in the program. Before using function block, double click "Current included Libraries" to open the following window. Click "Add" to add COMM.7FB and click "OK."

Ż				
		Add	Delet	e [

c:Igmwinilib	0100mm.710	

🙆 GMWIN	for Windov	vs – c	:₩gmwin	₩source₩	user_def	ine_mas,	prj				
Project F	<sup>p</sup> rogram j	<u>E</u> dit	Toolbox	<u>C</u> ompile	Online	Debug	Window	Help	 In the second se		
	<mark>9</mark> . <u>))</u>	<b>*</b>		53	6 🖿 C			Î	<u> </u>	<b>7</b>	34 13
C:₩gm	win₩sour	ce₩u	ser_define	_mas,src						_ 0	
Row O	_T200MS	SE SND, REQ	_MSG NDR								
Row 1	0	- FL_1 D	ERR							-	
Row 2			STAT								-{P}-
Row 3	DCVI 11		 от							÷	741
Row 4		-EN	ENO							1	
Row 5	%MB10	- IN1	OUT	0.0.0						-	$\langle \rangle$
Row 6	_RCV[1]	MO	)VE							1	\$
Row 7		EN	ENO							Ē	(R) (P)
Row 8	XQB0.0.0	- IN1		(MBO						1	0
Row 9											{F}
Kow 10											(8EL)
Row 11									4	<u> </u>	• >>
Creates	a new j	proj	ect				0ffli	ne	R	5,03	Edit

- It sends Frame 0 by operating function block every 200ms.
- Frame 0 sends 4-byte value from the master station in %MB0 to the slave station.
- The slave station saves Frame 1 that the master sent to %MB10. (Refer to frame setting of Frame 1) It logically reverses the value of %MB10 that is saved at this moment, outputs to the direct variable, %QB0.0.0 and saves again the value output to the direct variable %QB0.0.0. to %MB0.
- Conclusively, when 8 LED of the master are on, then 8 LED of the slave are off, and vice versa.
- To open communication, set through "communication enable setting."
- For "communication enable setting" method, refer to 3) Communication enable setting of 8.1.7 "1:1 dedicated protocol communication."

## 8.2.5 Example of Use 2

This example is supposed to activate direct variables with the help of the user-defined protocol. It s ideal for the system configuration to connect digital I/O module to GM7 base unit. If digital I/O module isn't connected, only a part of the operation is possibly checked.



1:1 dedicated protocol communication cable between GM7' s

#### The Program and communication parameter of the slave station 1) (1) Work at the slave station no. 1.

(2) Open a new project file and make a new program for the slave station.

(3) Select communication parameter in GMWIN parameter and double-click it, the window of comm. parameter menu opens.

> Set parameters as the following table. •

	Protocol and Send mode					
Station no.	Baud rate	Data bit	Parity bit	Stop bit	Communicati on channel	Dedicated
1	9600	8	None	1	RS232C null modem or RS422/485	Slave

Frame O			×
Header:		Tx/Rx : Receive	•
Segment 1		Segment 5	
Type: CONST 💌	%D0.0.0	Type: NONE	
C Hex Input C ASCII Input	Size: Byte	Hex Input     O ASCII Input	Size: Byte
Segment 2	Y	Segment 6	
Type: ARRAY	%QB0.0.0	Type: NONE	
F Receive by Hex Converting	Size: 4 Byte	O Hexinput O ASCII Input	Size: Byte
Segment 3		Segment 7	
Type: NONE		Type: NONE	
C Hex Input C ASCII Input	Size: Byte	Hexinput     O ASCII Input	Size: Byte
Segment 4		Segment 8	
Type: NONE		Type: NONE	
C Hexinput C ASCII Input	Size: Byte	C Hex Input C ASCII Input	Size: Byte
Tailer: B	CC Setting	Ok	Cancel

(4) Double "frame list 0" to activate "Frame 0" window and set as follows.

Item	Setting value
Header	None
Send/receive	Receive
Segment 1	Type: CONST, field: %QD0.0.0, select button for ASCII input
Segment 2	Type: ARRAY, field: %QB0.0.0, size: 4 bytes
Tail	None

Frame 1		×
Header : [STX]	Tx/Rx : Receive	•
Segment 1	Segment 5	
Type: CONST  RCV_FRAME	Type: NONE 💌 🗍	
C Hex Input C ASCII Input Size: Byte	O Hex Input O ASCII Input	Size: Byte
Segment 2	Segment 6	
Type: ARRAY  MB10	Type: NONE	
🗖 Receive by Hex Converting Size: 🗾 Byte	O Hex Input O ASCII Input	Size: Byte
Segment 3	Segment 7	
Type: NONE	Type: NONE	
C Hex Input C ASCII Input Size: Byte	Hex Input     C ASCII Input	Size: Byte
Segment 4	Segment 8	
Type: NONE	Type: NONE	
Hex Input     O     ASCII Input     Size:     Byte	C Hex Input C ASCII Input	Size: Byte
Tailer: [ETX] BCC Setting	Ok	Cancel

(5) Double click frame list 1 to activate Frame 1 window to set as follows.

Item	Setting value
Header	None
Send/receive	Send
Segment 1	Type: CONST, field: %QB0.1.0, select button for ASCII input
Segment 2	Type: ARRAY, field: %QB0.1.0, size: 1 bytes
Tail	None

(6) Set up a program like the following figure and download to the slave station GM7. For the detailed program setting and downloading, refer to GMWIN manual.



- When the frame 0 is received, it outputs the value to %QB0.0.0 of the slave station. (Refer to the Frame 0 setting.)
- When Frame 0 is received, it operates ADD function to increase the value of the symbolic variable, \_ADD 1 by 1, and because the symbolic variable \_ADD has been assigned a memory at the direct variable, %QB0.1.0, outputs the result of the ADD function to the digital I/O module.
- The slave reads the direct variable, %QB0.1.0 of the digital I/O and sends it again to the master. (This
  function enables to check whether the communication is being done without the digital I/O module by
  the direct variable outputs of the GM7 base unit.)
- To open communication, set through "communication enable setting."

riables					
Name :	D		Direct V	ariable <u>C</u> omment	ОК
Name	Var. Kind	Allocation	Used	Data Type	Flag
ADD	VAR	<auto></auto>		SINT	
ACTUATOR	VAR	<auto></auto>		FB Instance	
SEND	VAR	<auto></auto>		FB Instance	Cancel
ACTUATOR.NDR	VAR	<auto></auto>		BOOL	
ACTUATOR.ERR	VAR	<auto></auto>		BOOL	
ACTUATOR.STATUS	VAR	<auto></auto>		USINT	Liele
SEND.NDR	VAR	<auto></auto>		BOOL	
Ado	i	<u>D</u> elete	1	<u>E</u> dit	
Description	10	- 75 <del>-</del> 2			B
Nama : ADD					
Variable Kind 1 V	AR				
Data Type : SINT					
Allocation : <auto< td=""><td>)&gt;</td><td></td><td></td><td></td><td></td></auto<>	)>				
Initial Value : «De	efault≻				
million value					

#### The Program and communication parameter of the master station (1) Work at the master station no. 0.

- (2) Open a new project file and make a new program for the master station.
- (3) Select communication parameter in GMWIN parameter and double-click it, the window of comm. parameter menu opens.
  - Set parameters as the following table.

Communication Method						Protocol and Send mode
Station no.	Baud rate	Data bit	Parity bit	Stop bit	Communication channel	User-defined
0	9600	8	None	1	RS232C null modem or RS422/485	Master

(4) Double "frame list 0" to activate "Frame 0" window and set as follows.

Frame 0	×
Header:	Tx/Rx : Receive
Segment 1	Segment 5
Type: CONST  VQB0.1.0	Type: NONE
C Hex Input C ASCII Input Size: Byte	Hex Input     C ASCII Input     Size:     Byte
Segment 2	Segment 6
Type: ARRAY T %QB0.1.0	Type: NONE
F Receive by Hex Converting Size: 1 Byte	Hex Input     C ASCII Input     Size:     Byte
Segment 3	Segment 7
Type: NONE	Type: NONE
C Hex Input C ASCII Input Size: Byte	Hex Input     C ASCII Input     Size:     Byte
Segment 4	Segment 8
Type: NONE	Type: NONE
C Hex Input C ASCII Input Size: Byte	C Hex Input C ASCII Input Size: Byte
Tailer: BCC Setting	Ok Cancel

Item	Setting value
Header	None
Send/receive	Send
Segment 1	Type: CONST, field: %QB0.1.0, select button for ASCII input
Segment 2	Type: ARRAY, field: %QB0.1.0, size: 1 bytes
Tail	None

(5) Double click frame list 1 to activate Frame 1 window to set as follows.

Frame 0		×
Header :	Tx/Rx : Receive	•
Segment 1	Segment 5	
Type: CONST 💌 %QB0.1.0	Type: NONE	
C Hex Input C ASCII Input Size: Byte	Hex Input     C ASCII Input	Size: Byte
Segment 2	Segment 6	
Type: ARRAY Y %QB0.1.0	Type: NONE	
Receive by Hex Converting Size: 1 Byte	O Hex Input C ASCII Input	Size: Byte
Segment 3	Segment 7	
Type: NONE	Type: NONE	
C Hex Input C ASCII Input Size: Byte	Hex Input     C ASCII Input	Size: Byte
Segment 4	Segment 8	
Type: NONE	Type: NONE	
Hex Input     O     ASCII Input     Size:     Byte	C Hex Input C ASCII Input	Size: Byte
Tailer: BCC Setting	Ok	Cancel

Item	Setting value
Header	None
Send/receive	Receive
Segment 1	Type: CONST, field: %QB0.1.0, select button for ASCII input
Segment 2	Type: ARRAY, field: %QB0.1.0, size: 1 bytes
Tail	None

(6)	Set up a program like the following figure and download to the slave station GM7. For the detailed program
	setting and downloading, refer to GMWIN manual.

👜 GMWIN	N for Windows – c:\gmwin\source\user_define_mas,prj 📃	
Project	P <u>r</u> ogram <u>E</u> dit <u>T</u> oolbox <u>C</u> ompile <u>O</u> nline <u>D</u> ebug <u>W</u> indow <u>H</u> elp	
😭 c :\#gm	mwin\%source\%user_define_mas,src	R
Row O		
Row 1	_T200MS ADD SND_MSG	4 +
		1/1
Row 2	_ADD - IN1 OUTADD O - FL_I ERR	
Row 3	1 - IN2 STAT US	
Row 4		1
Row 5		()
Row 6		()
		(s) (v)
Row 7		(8) (0)
Row 8		(N)
Row 9		{F}
Row 10		<del>{</del> FB}
Row 11		<re>ket&gt;</re>
D 10		$\gg$
Row 12		
₽ош 13 ◀		
•		8
	Offline R2,C7 H	Edit

- It operates ADD function to increase the value of the symbolic variable, \_ADD 1 by 1 at every 20ms, and because the symbolic variable \_ADD has been assigned a memory at the direct variable, %QB0.0.0, outputs the result of the output contact point of the GM7 base unit.
- The master reads the direct variable, %QB0.0.0 of the GM7 base unit and sends it again to the master. (Refer to Frame 0 setting.)
- To open communication, set through "communication enable setting."

Name :AD	D		Direct V	ariable <u>C</u> omment	OK
/ariables List					
Name	Var. Kind	Allocation	Used	Data Type 🔺	<u>E</u> lag
ADD	VAR	<auto></auto>		SINT	
ACTUATOR	VAR	<auto></auto>		FB Instance	
SEND	VAR	<auto></auto>		FB Instance	Cancel
ACTUATOR.NDR	VAR	<auto></auto>		BOOL	<u> </u>
ACTUATOR.ERR	VAR	<auto></auto>		BOOL	
ACTUATOR.STATUS	VAR	<auto></auto>		USINT	
SĘND.NDR	VAR	<auto></auto>		BOOL 🚬 革	Heip
Ado	l	<u>D</u> elete	1	<u>E</u> dit	
Description Name : _ADD Variable Kind : VA Data Type : SINT Allocation : <auto Initial Value : <de Comments :</de </auto 	AR )> :fault>			22	

• Variables window