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

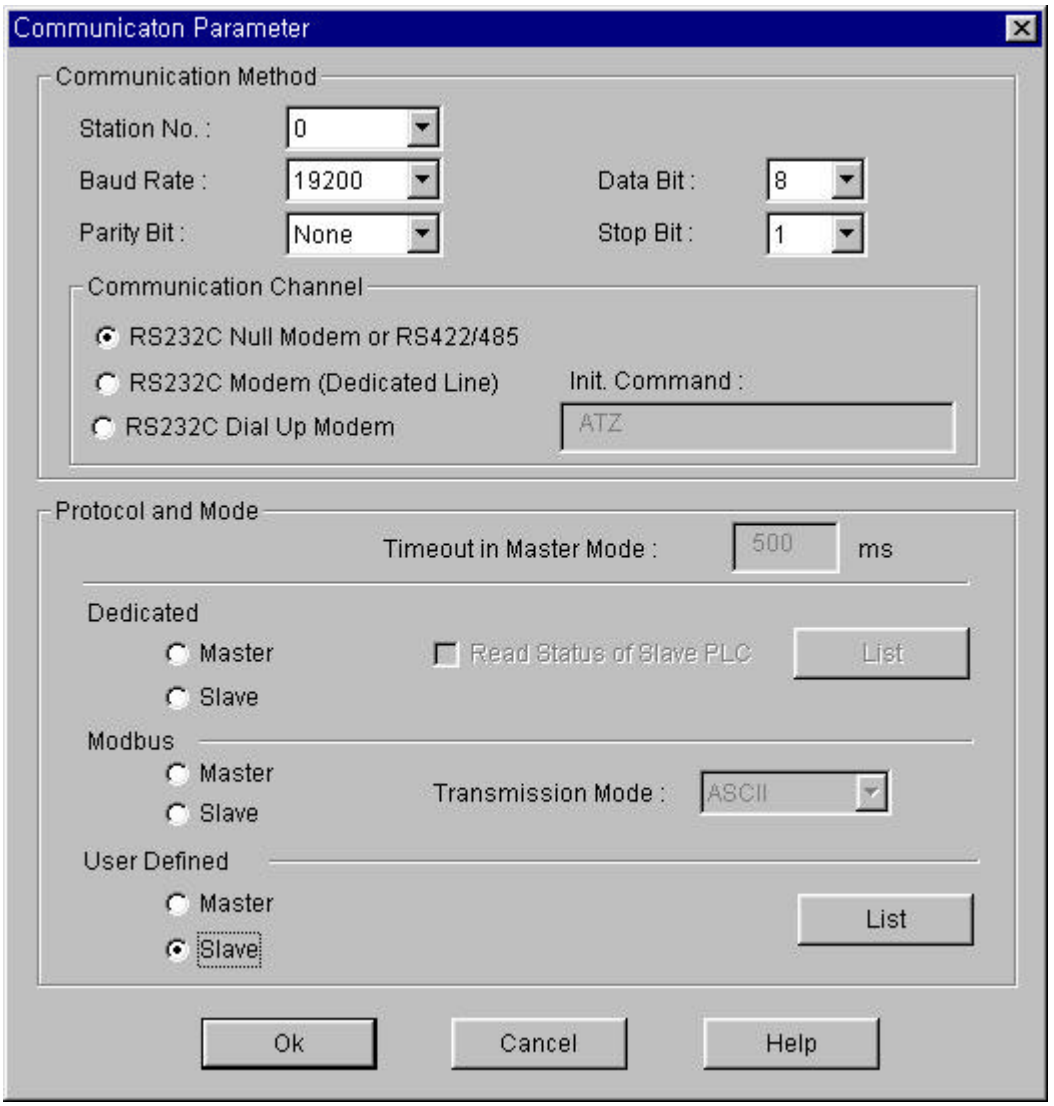
1) All numerical data can use hexadecimal, decimal, and binary type. If we convert decimal 7 and 10 into each type:
   Hexadecimal : H07, H0A or 16#07, 16#0A
   Decimal : 7, 10
   Binary : 2#0111, 2#1010

8.2.2 Parameter Setting

1) Setting Communications Parameter

(1) Open a new project file from GMWIN
   ① Select GM7 as PLC type
   ② 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.



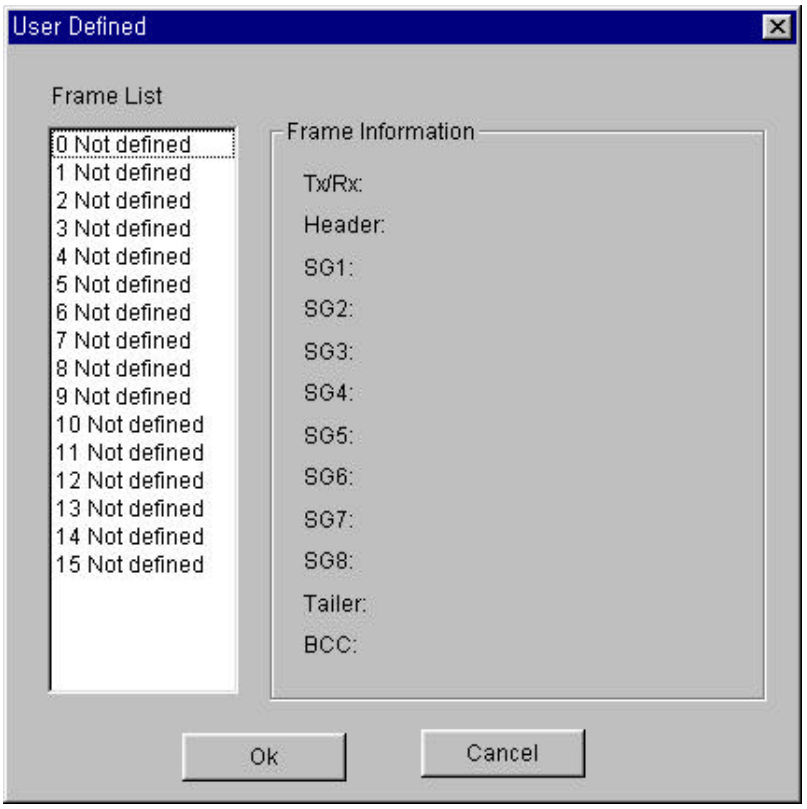
(3) Set according to the following table.

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	● 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). ● RS232C Modem (Dedicated Line) : It's to be selected for the communication, using an dedicated modem with Cnet I/F module (G7L-CUEB). ● 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). 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.
Timeout in Master Mode	● It's the time waiting a responding frame since the master GM7 base unit sends a request frame. ● The default value is 500ms. ● It must be set in consideration of the max. periodical time for sending/receiving of the master PLC. ● If it's set smaller than the max. send/receive periodical time, it may cause communication error.
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.

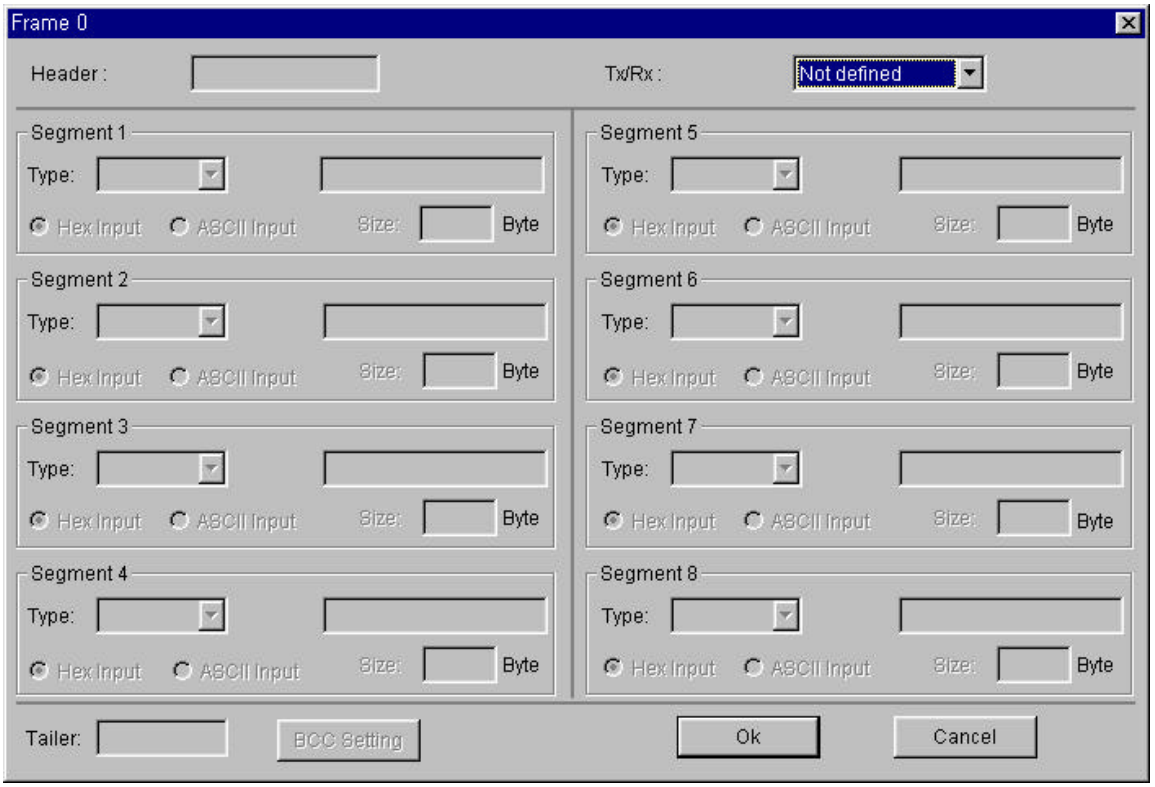
2) Setting frame

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

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



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



① Frame specification

- Header
  - Used in [Header] type.
  - 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 (← ×: [ ] isn't used)

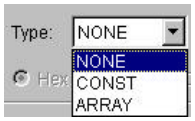
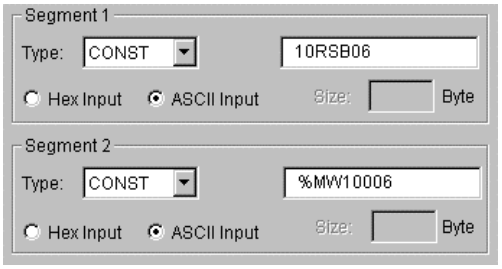
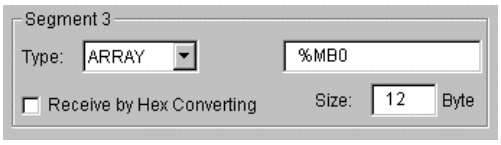
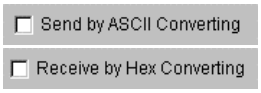
  - It is allowed to be only 3 consecutive characters.

    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.

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

Item	Contents
	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  If the segment is declared as ARRAY type, although word type data is declared in CONST type, the related area is to be set by byte. Ex2) As an dedicated protocol communication, 10RSB06%MW10006 is a frame to execute reading 6 word data from %MW100 at the slave station no. 16. At the moment, ARRAY must be set in 6 words that is 12 bytes, as the area to save the data that is read. 
	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
	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).
	If ARRAY is set, the size of area is to be set by byte. **The unit is a byte.**

- 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 (← ×: [ ] isn't used)

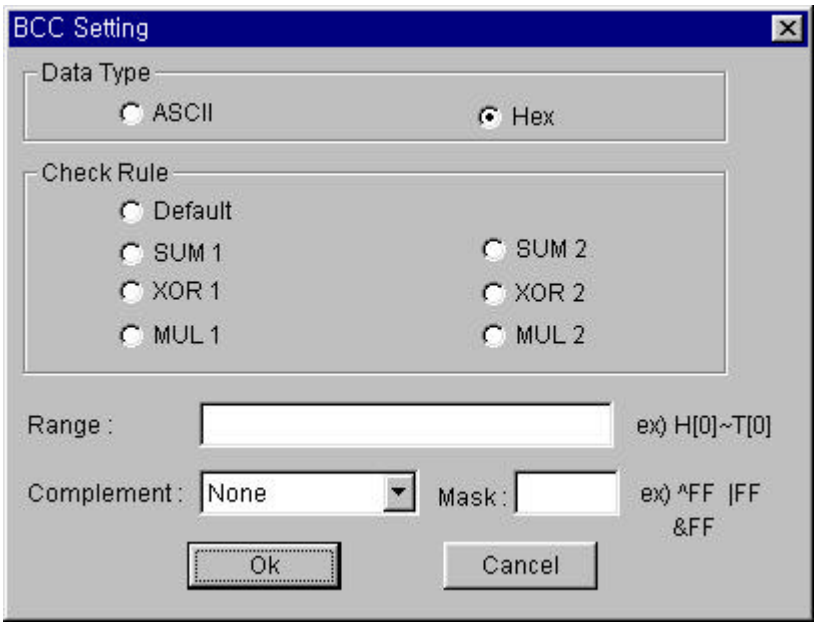
  - It is allowed to be only 3 consecutive characters.

    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)

- BCC setting: set BCC when it is needed.



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".
Check Rule	Default	It is that sum all the data from 2nd 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.
	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.
	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 →

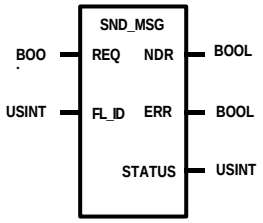
- Frame size
  - ASCII communication : max. 128 bytes
  - Hex communication : max. 256 bytes

- 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
	**Input** REQ : Execute function block at rising edge(0 → 1) FL_ID : Frame list field number to send. **Output** 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)

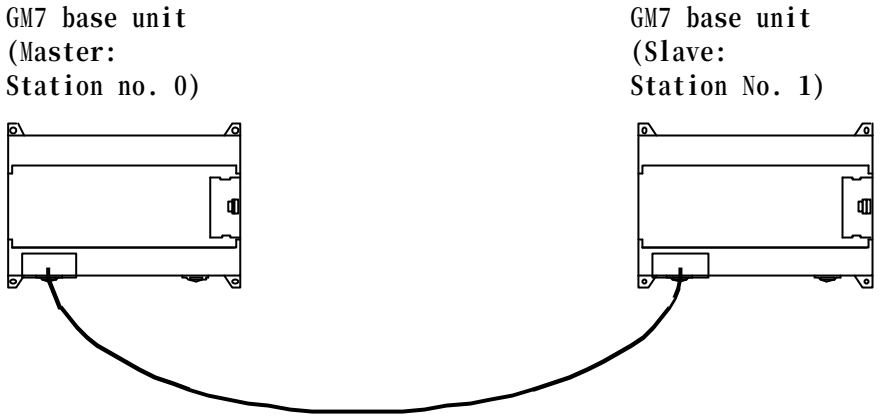
CODE	Error type	Meaning
06	Slave Device Busy	It's sending or waiting to receive
09	Parameter Error	Comm. Parameter setting, or communication enable setting error
10	Frame Type Error	The frame isn't for sending or set.

3) Error flag
Errors related with the frame list
① 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 sending part
Bit 1	Device Area Over	Error when I/Q area (128 bytes), or M area is exceeded.	
Bit 2	BCC Setting Error	BCC setting error	
Bit 8	BCC Check Error	Error when received BCC value is wrong.	Error flag on receiving part
Bit 9	Mismatch Error	Error when a received frame doesn't match with the registered frame.	
Bit 10	Hex Change Error	Error when converting to hexadecimal.	
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.



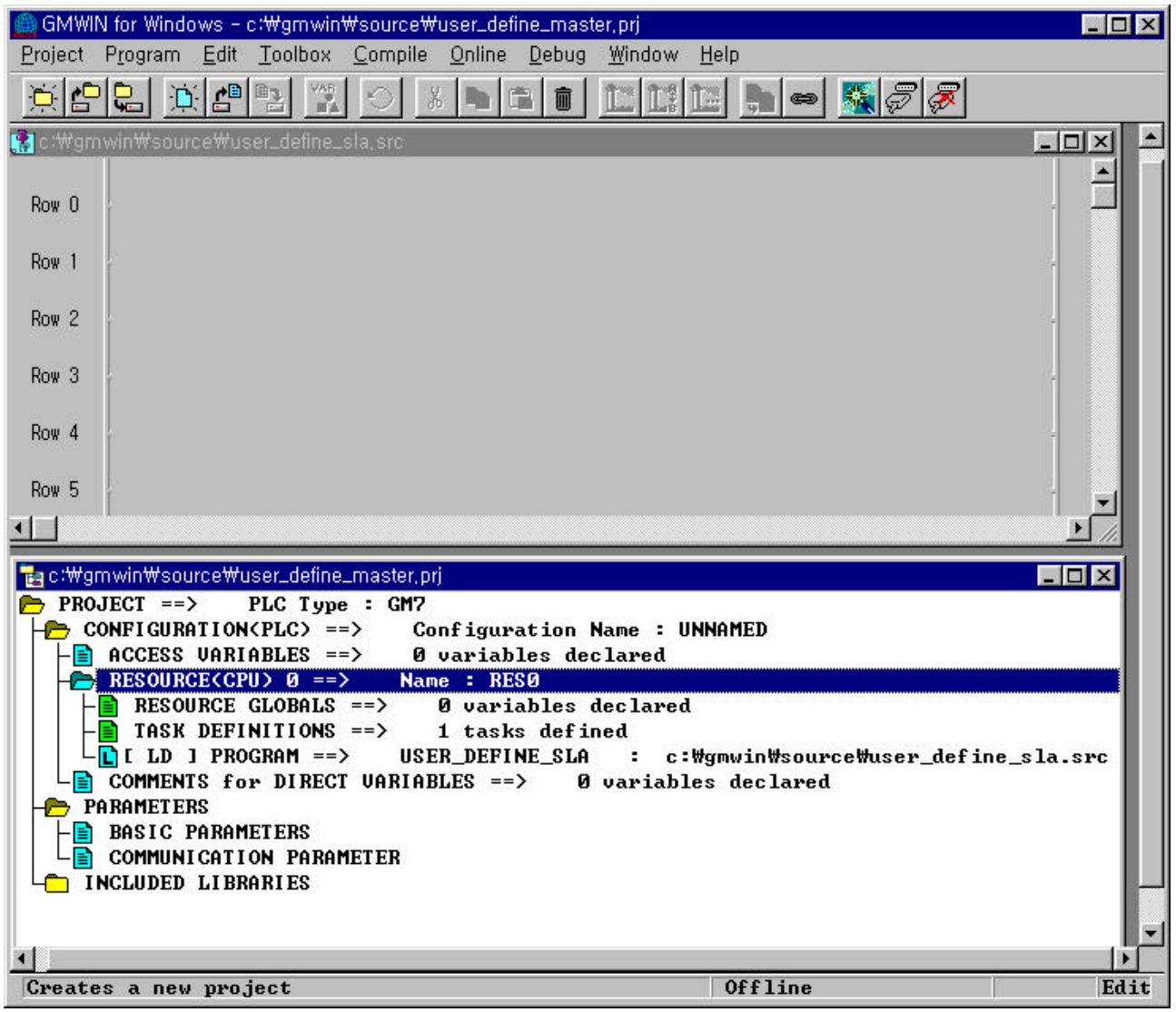
1:1 dedicated protocol communication between GM7's

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.

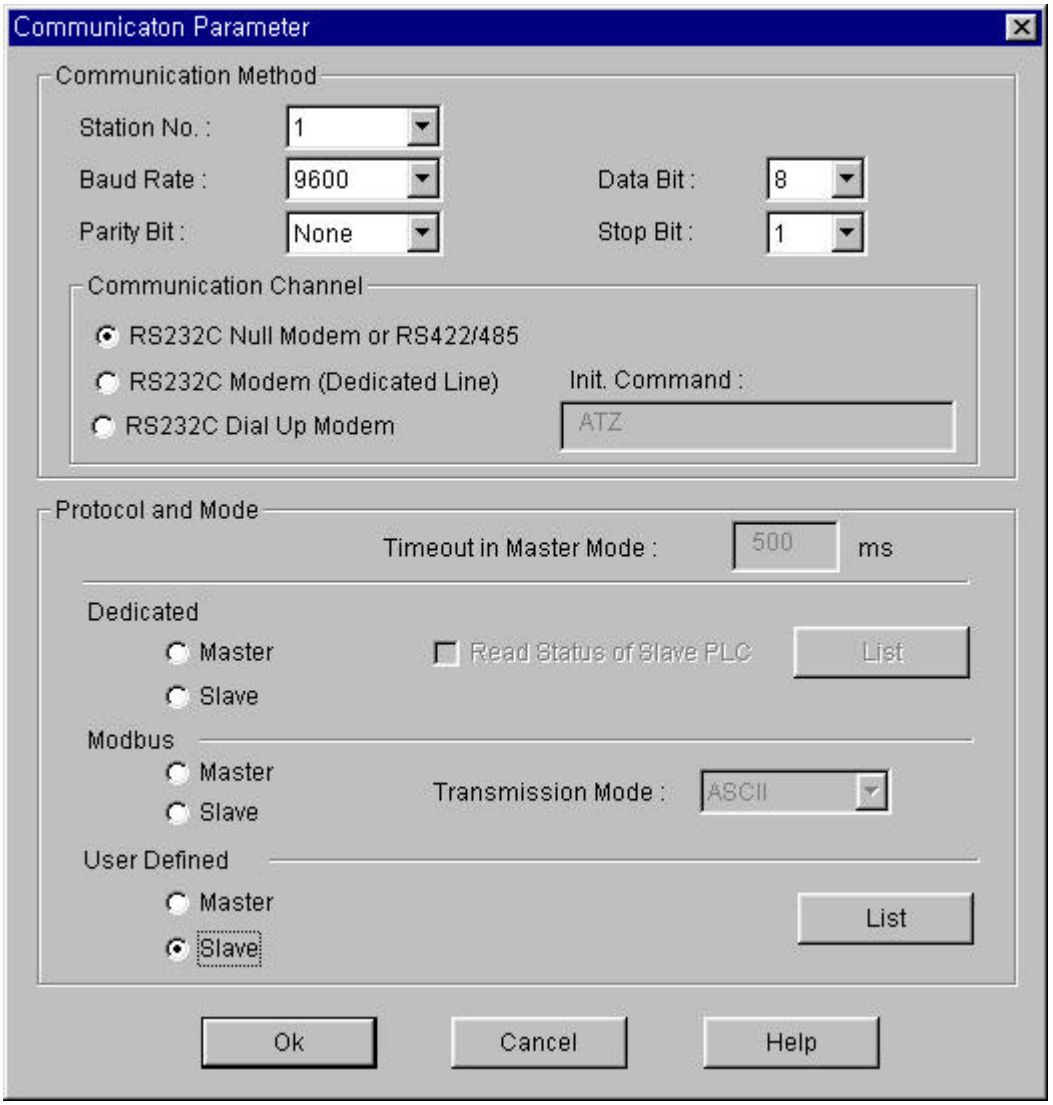
1) The Programming and setting communication parameter of the slave station

(1) Work at the slave station no. 1.

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



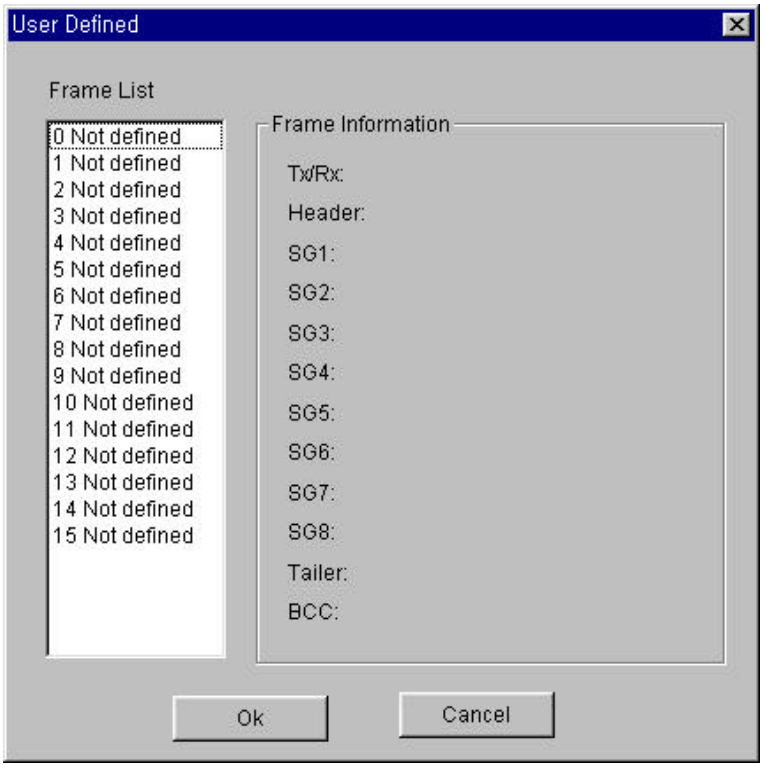
(3) Select communication parameter in GMWIN parameters and double click it, and then the window of communication parameter will be opens.



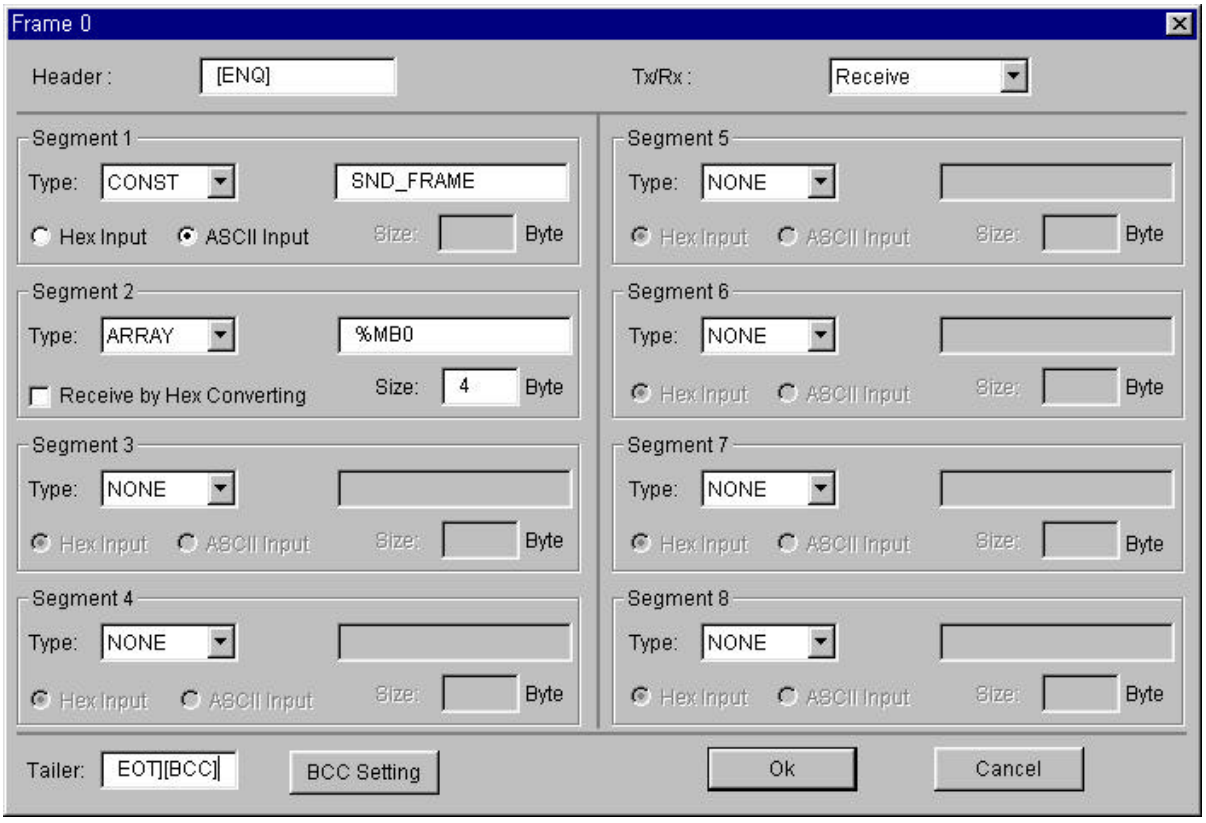
- Set parameters as the following table.

Communication Method						Protocol and Mode
Station 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.

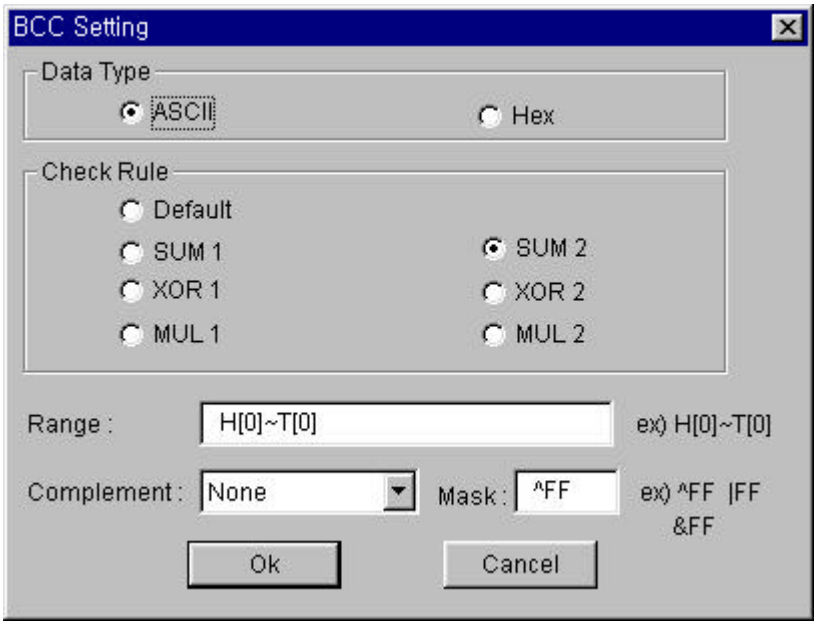


(5) Double click "Frame List 0" to activate "Frame 0" window and set as follows.



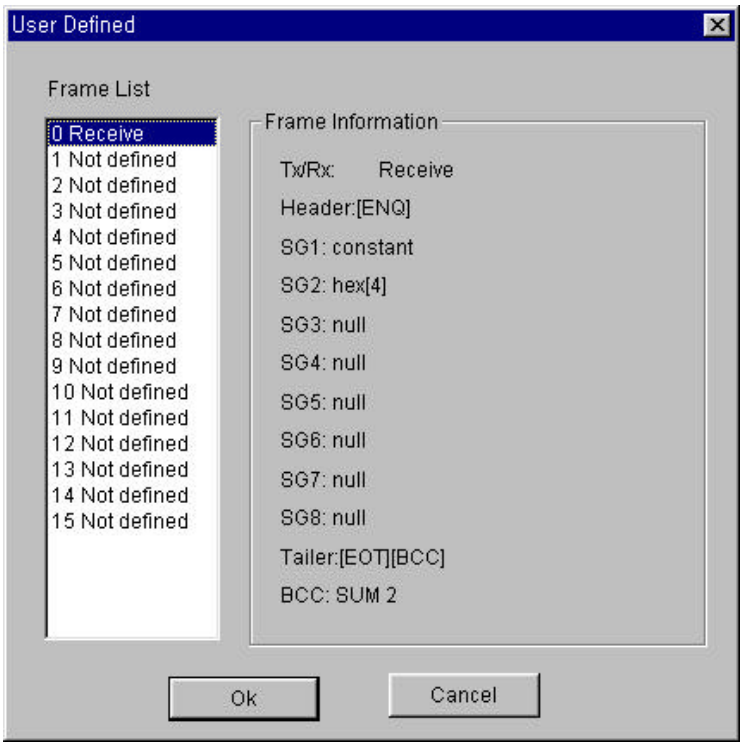
Item	Setting value
Header	[ENQ]
Tx/Rx	Receive
Segment 1	Type: CONST, field: SND_FRAME, select button for ASCII input
Segment 2	Type: ARRAY, field: %MB0, size: 4 bytes
Tailer	[EOT][BCC]

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

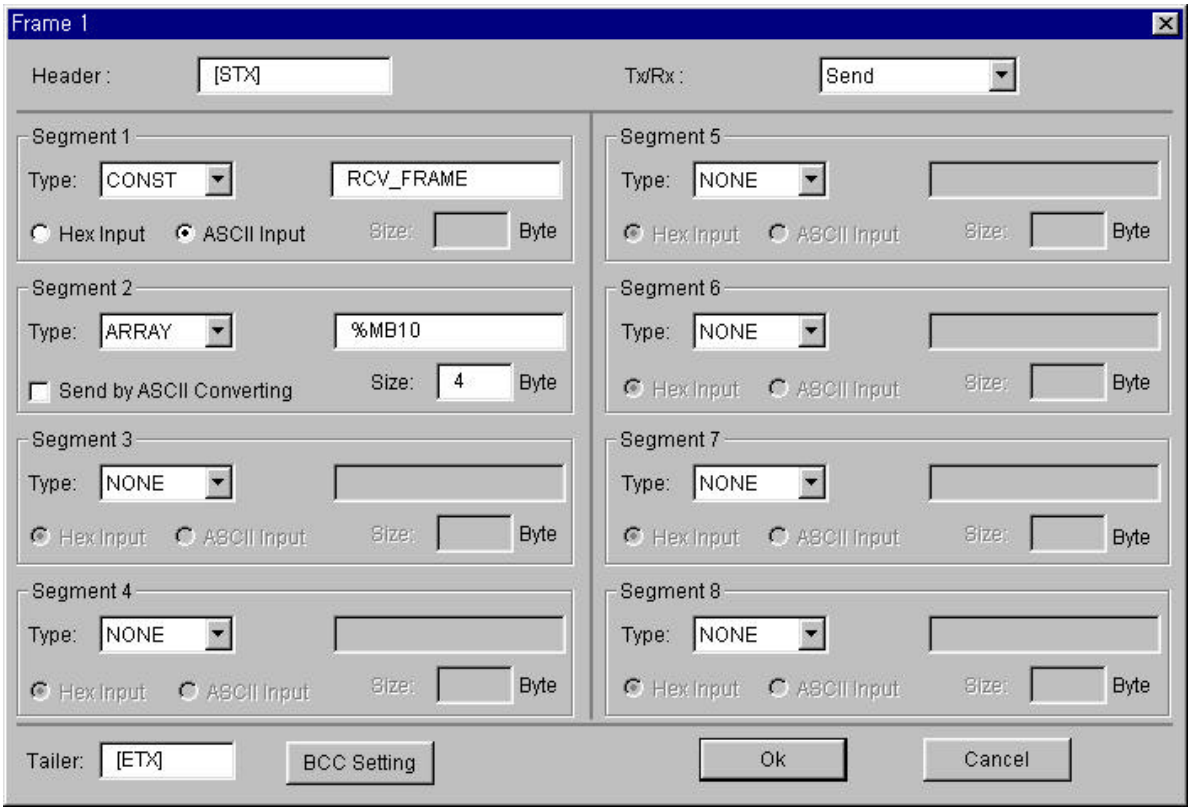


Item	Setting value
Data Type	ASCII
Check Rule	SUM 2
Range	H [0] ~ T [0] that is from header [ENQ] to tail [EOT].
Complement	None
Mask	Mask with HFF by XOR.

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

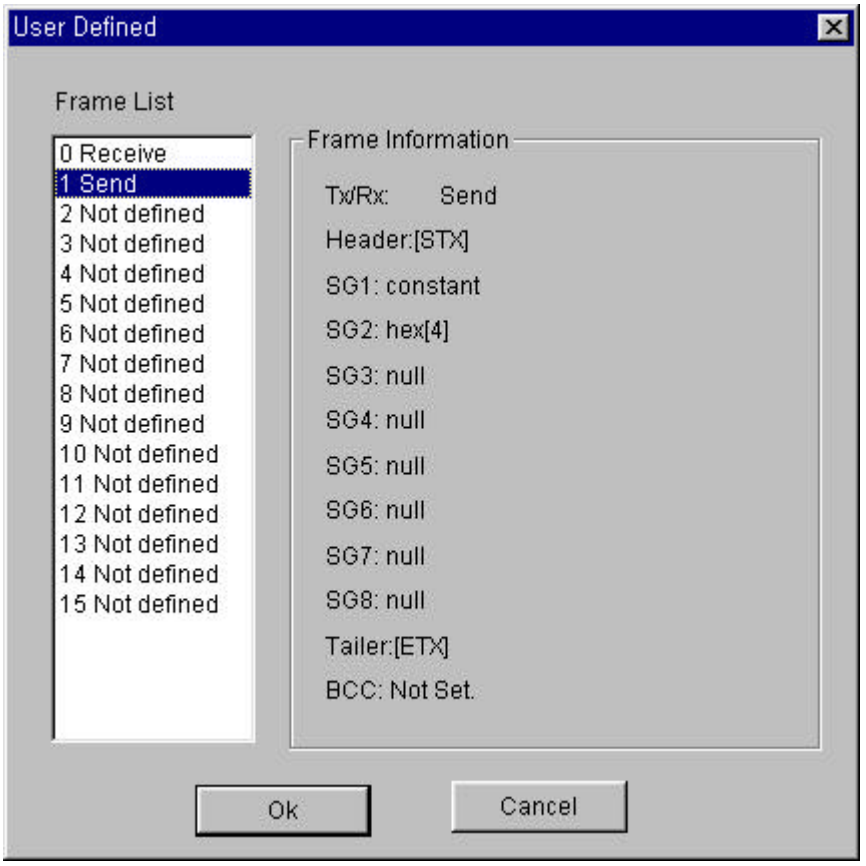


(8) Double click frame list 1 to activate Frame 1 window to set as follows.



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]

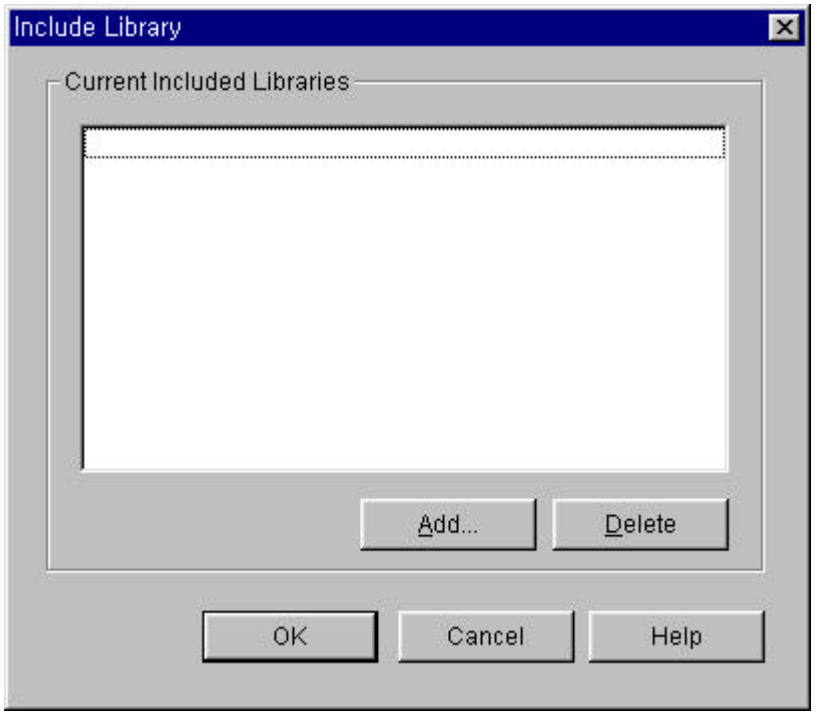
(9) After setting, click "OK" of Frame 1 to register as follows.

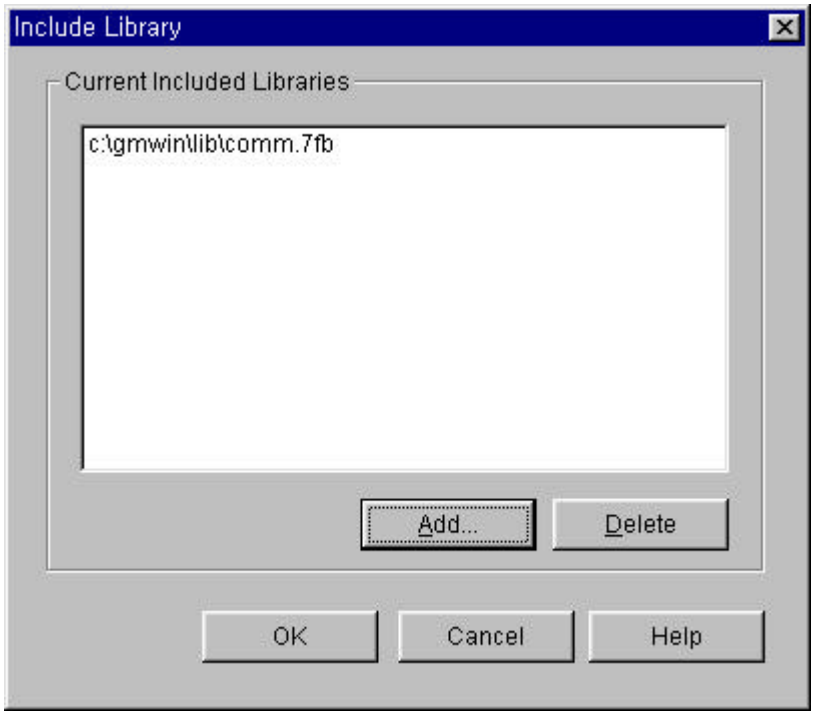


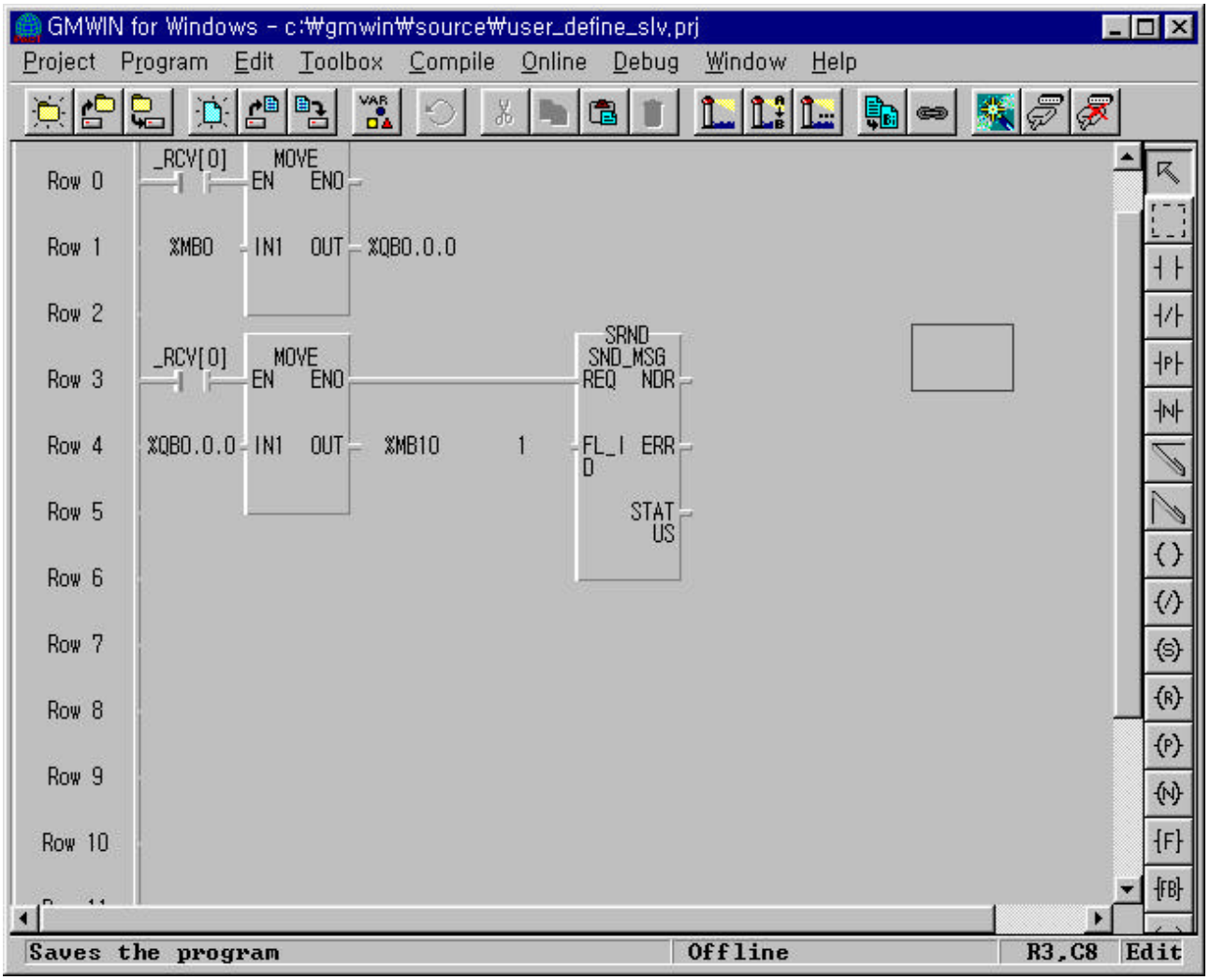
(10) Click "OK" to exit communication parameter setting and click "OK" to complete setting.

(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."





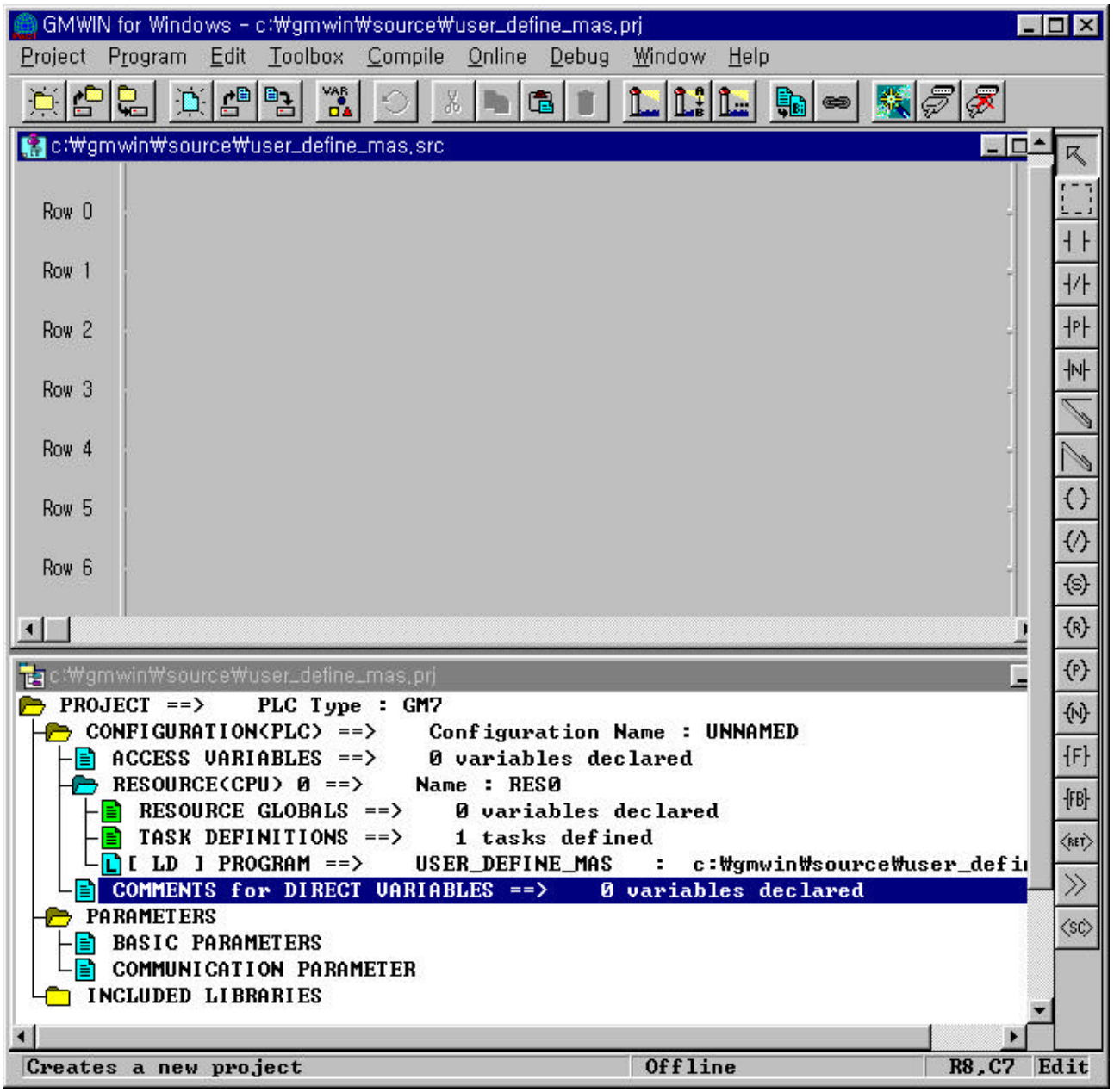


- If the frame 0 is received, save it in %MB0 of the slave (refer to frame 0 setting). Output value saved in %MB0 to direct variable %MB10 and again save the value in %QB0.0.0 in %MB10. If this execution is done without error, send frame 1 out of frame list from function block. Frame 1 is the frame that 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 "11 dedicated protocol communication."

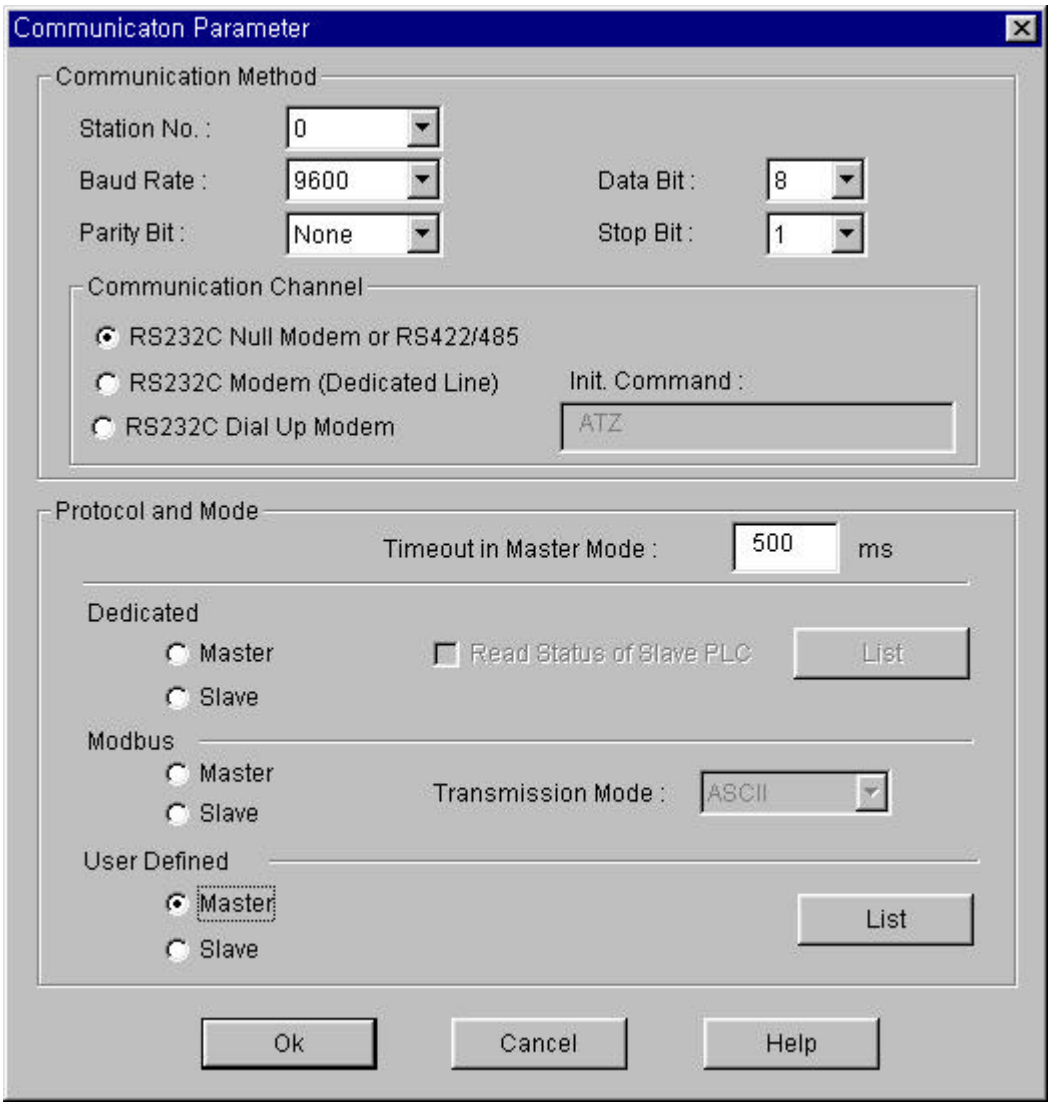
2) The programming and setting communication parameter of the master station

(1) Work at the master station no. 1.

(2) Open a new project file and set up a new program for the master station.



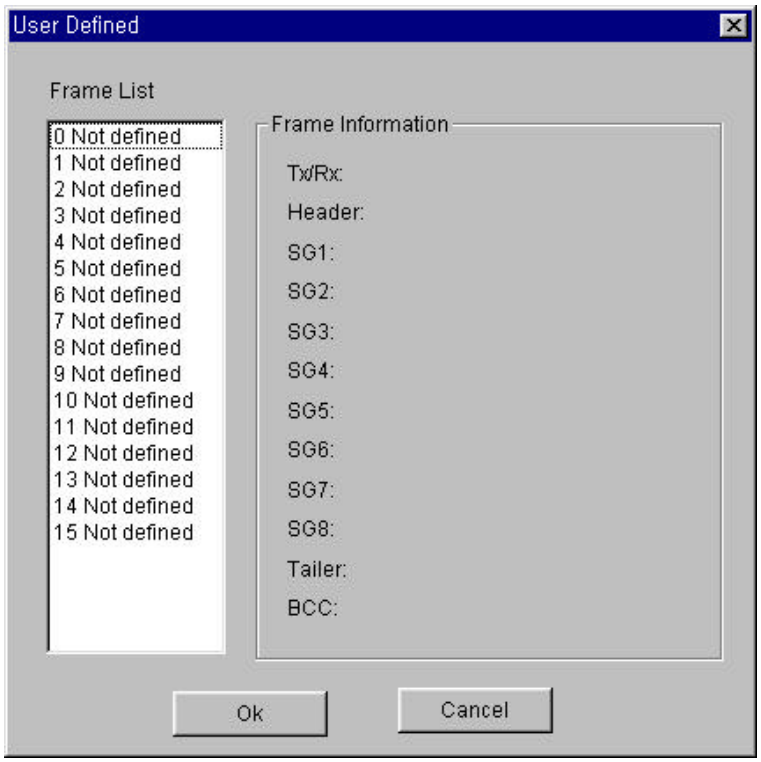
(3) If you double click the 'communication parameter' window in GMWIN, you can see the following window of the 'communication parameter.'



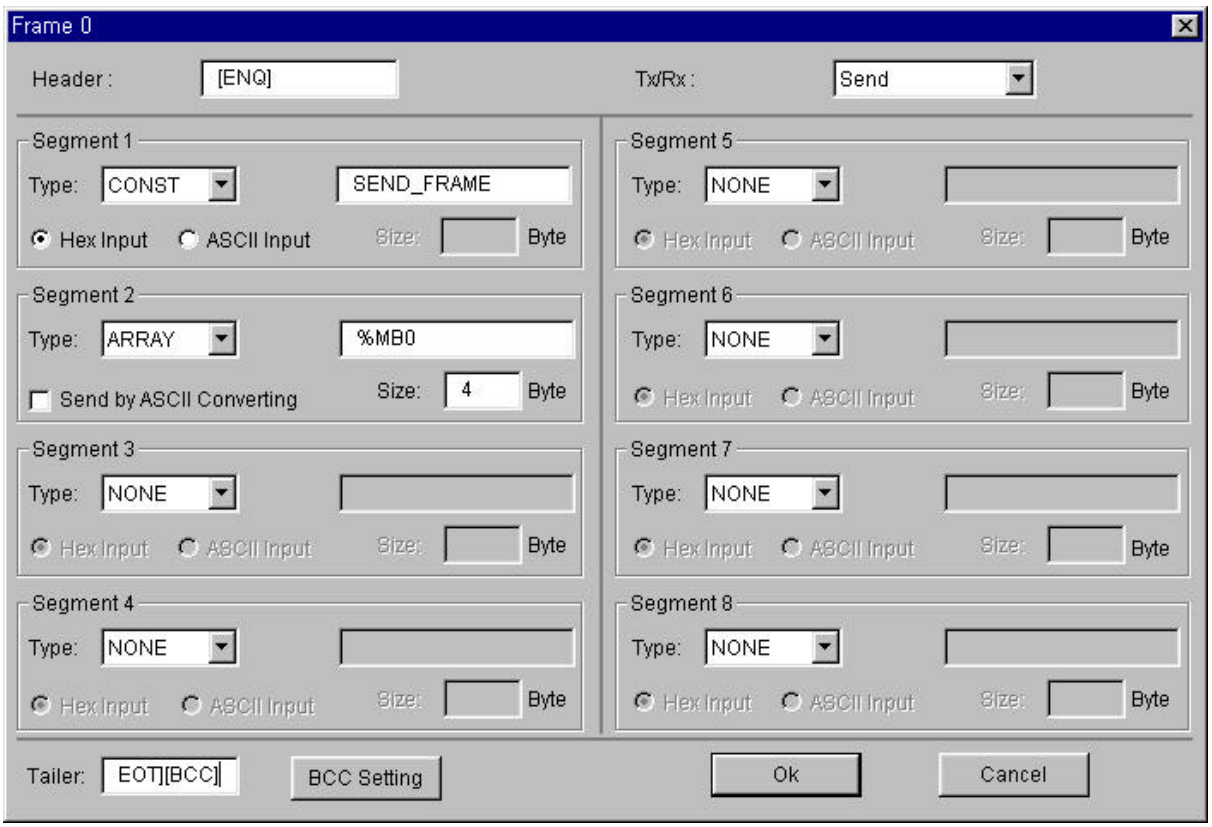
- Set parameters as the following table.

Communication Method						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.

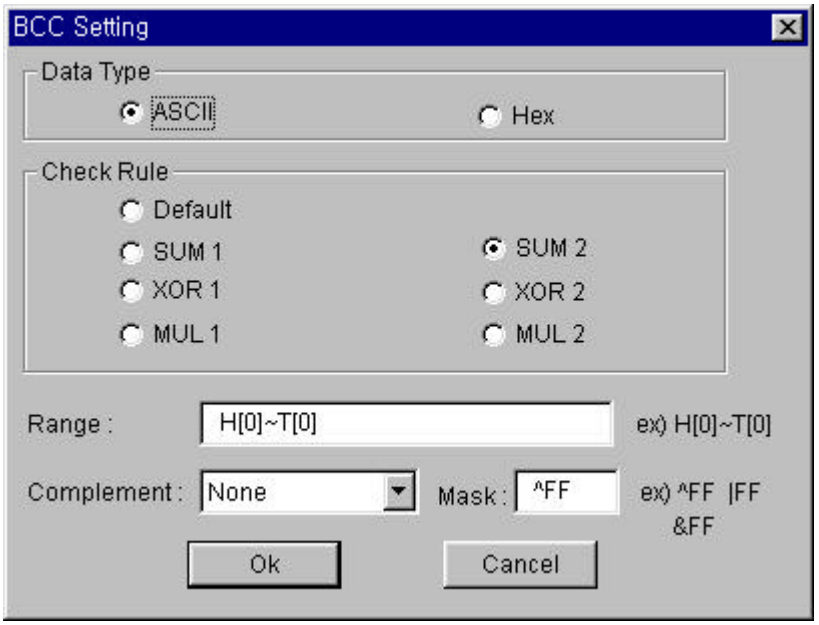


(5) Double click Frame list 0 to activate Frame 0 window and set as follows.



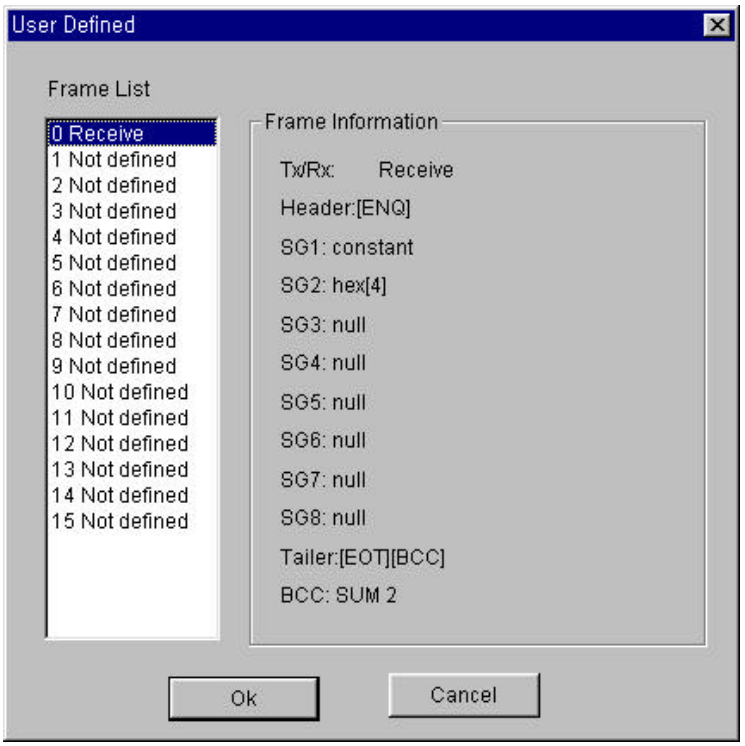
Item	Setting value
Header	[ENQ]
Tx/Rx	Send
Segment 1	Type: CONST, field: SND_FRAME, select button for ASCII input
Segment 2	Type: ARRAY, field: %MB0, size: 4 bytes
Tailer	[EOT][BCC]

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

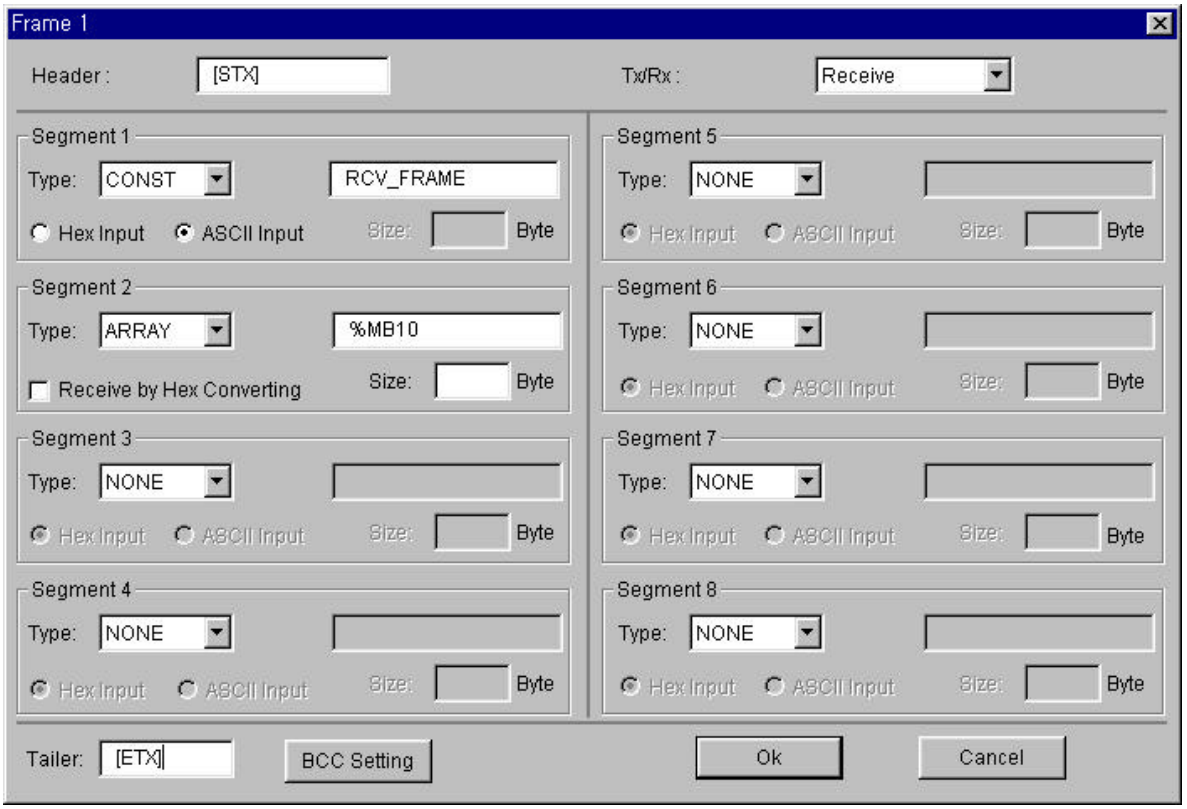


Item	Setting value
Data Type	ASCII
Check Rule	SUM 2
Range	H [0] ~ T [0] that is from header [ENQ] to tail [EOT].
Complement	None
Mask	Mask with HFF by XOR.

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

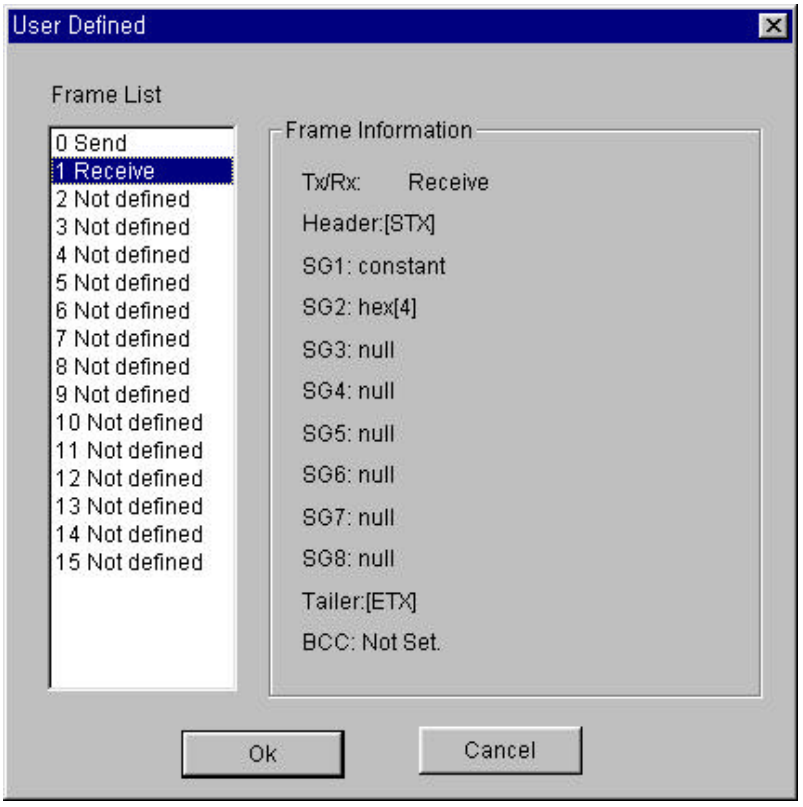


(8) Double click frame list 1 to activate Frame 1 window to set as follows.



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]

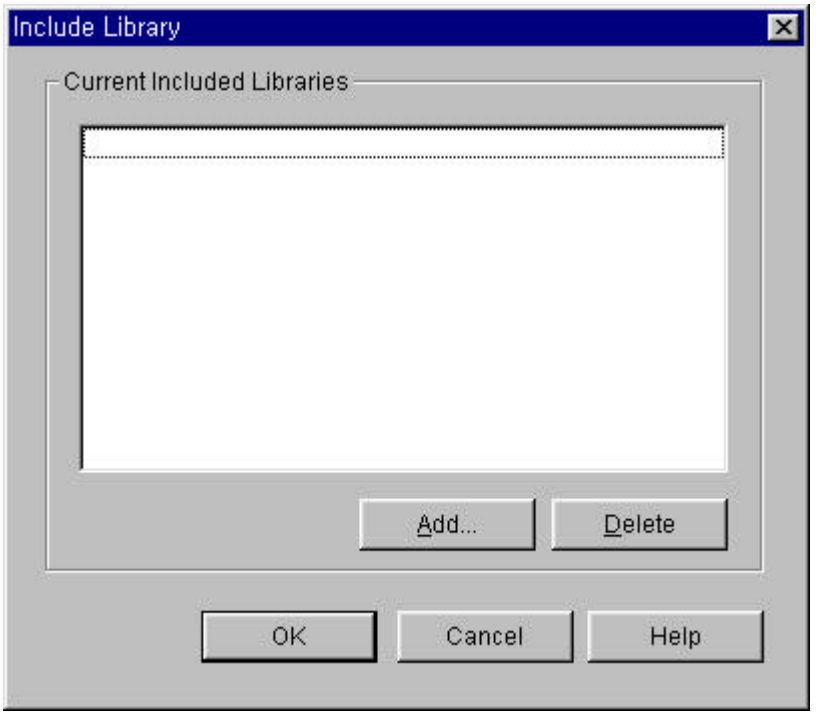
(9) After setting, click "OK" of Frame 1 to register as follows.

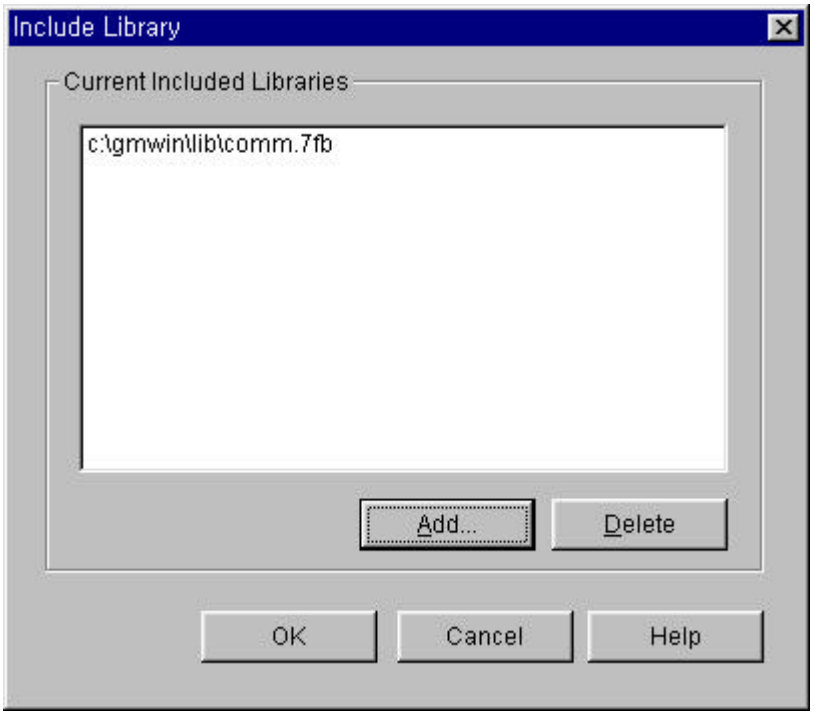


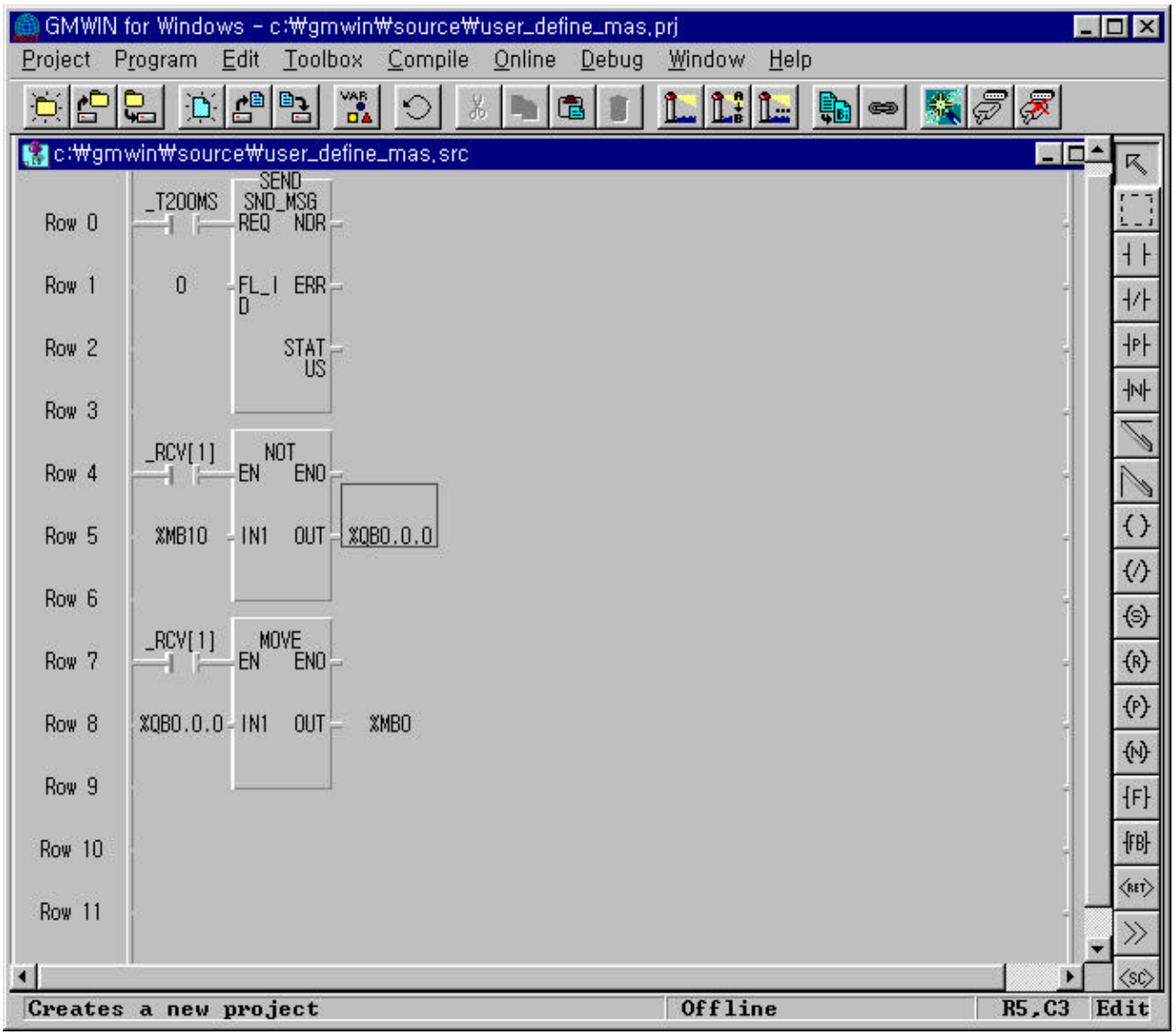
(10) Click "OK" to exit communication parameter setting and click "OK" to complete setting.

(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."



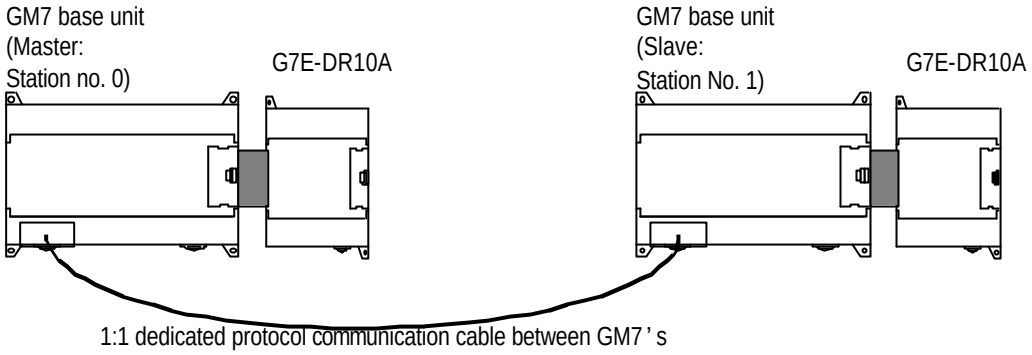




- It sends Frame 0 by operating function block every 200ms.
- Frame 0 sends 4-byte value from the master station in %MB0 to the slave station.
- The slave station saves Frame 1 that the master sent to %MB10. (Refer to frame setting of Frame 1) It logically reverses the value of %MB10 that is saved at 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 "11 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) The Program and communication parameter of the slave station

(1) Work at the slave station no. 1.

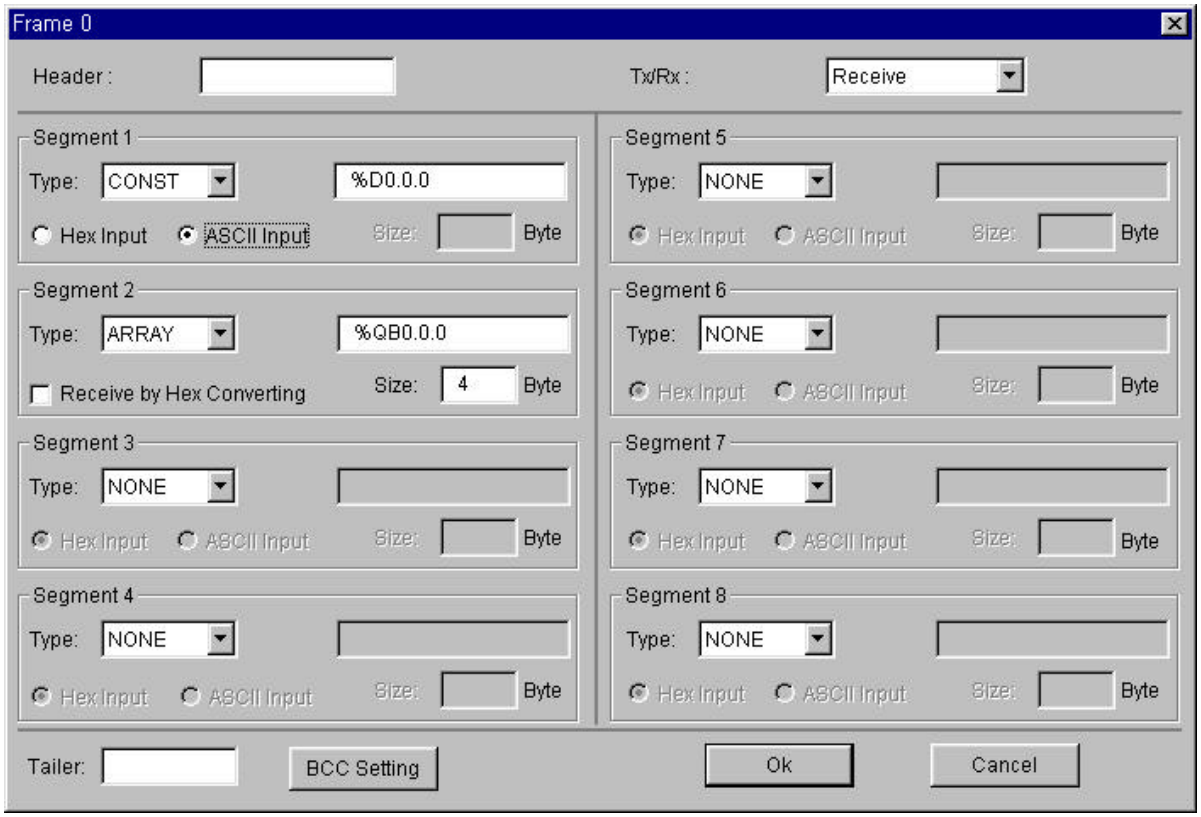
(2) Open a new project file and make a new program for the slave station.

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

- Set parameters as the following table.

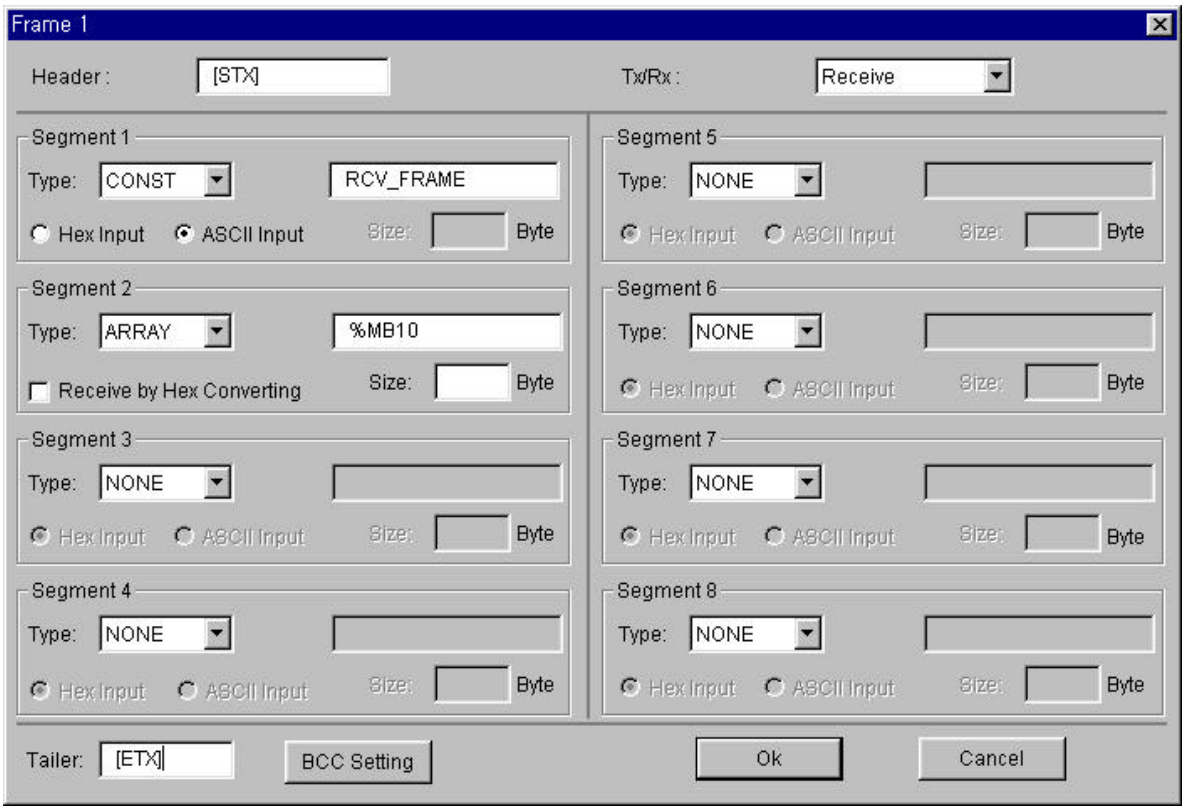
Communication Method						Protocol and Send mode
Station no.	Baud rate	Data bit	Parity bit	Stop bit	Communication channel	Dedicated
1	9600	8	None	1	RS232C null modem or RS422/485	Slave

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



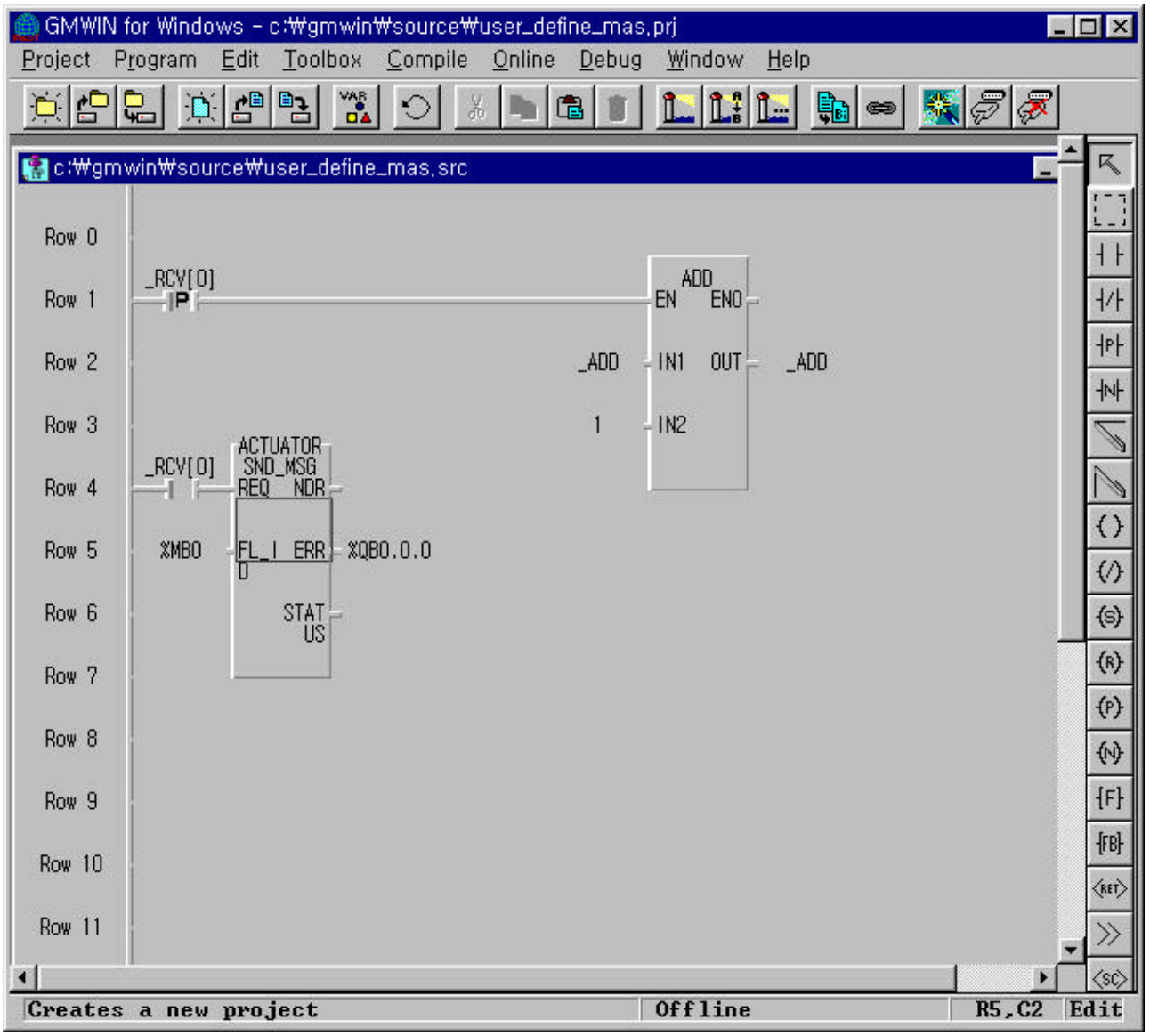
Item	Setting value
Header	None
Send/receive	Receive
Segment 1	Type: CONST, field: %QD0.0.0, select button for ASCII input
Segment 2	Type: ARRAY, field: %QB0.0.0, size: 4 bytes
Tail	None

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



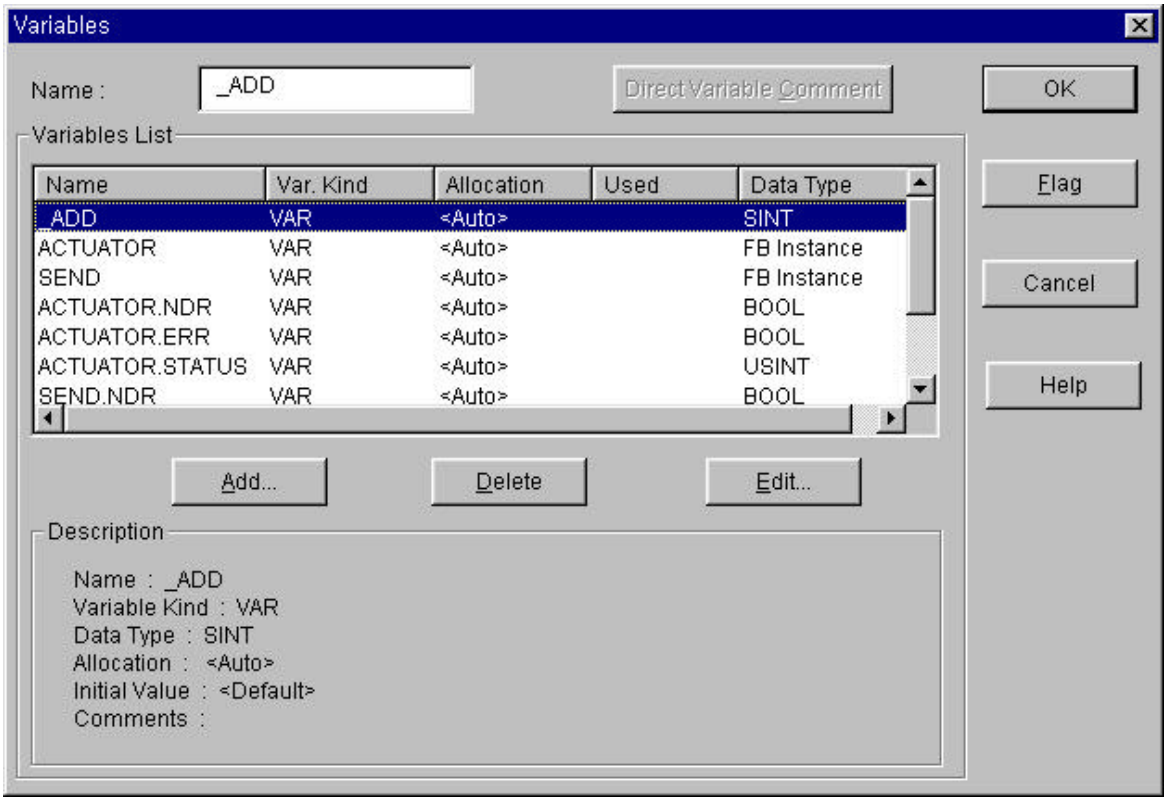
Item	Setting value
Header	None
Send/receive	Send
Segment 1	Type: CONST, field: %QB0.1.0, select button for ASCII input
Segment 2	Type: ARRAY, field: %QB0.1.0, size: 1 bytes
Tail	None

(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."**

- Variables window



2) The Program and communication parameter of the master station

(1) Work at the master station no. 0.

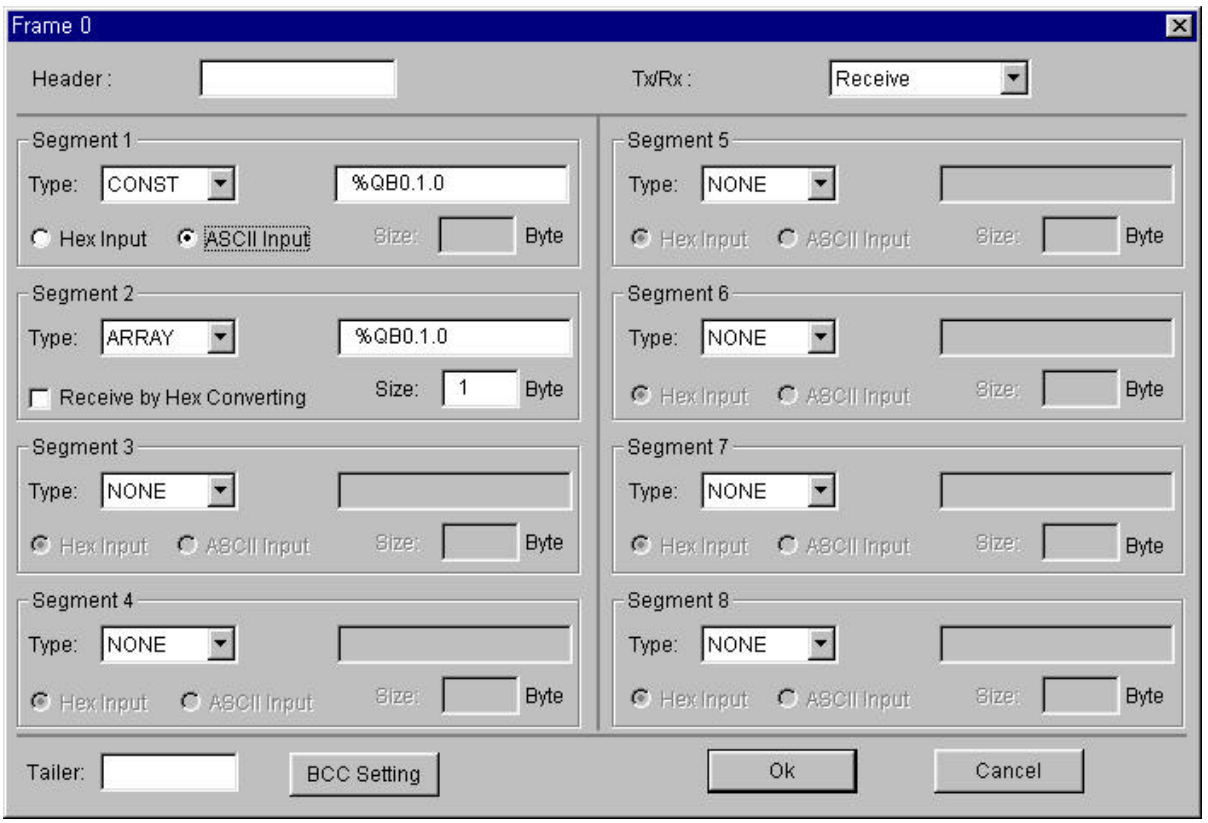
(2) Open a new project file and make a new program for the master station.

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

- Set parameters as the following table.

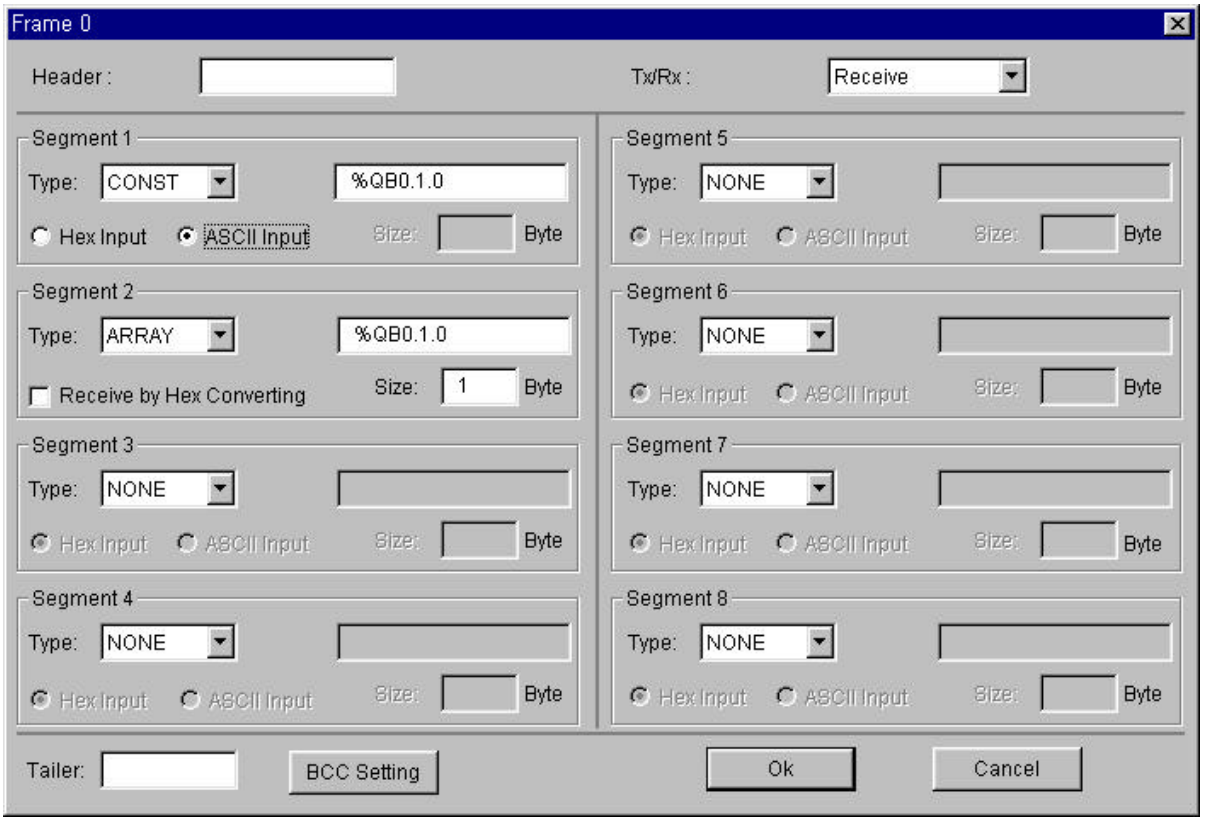
Communication Method						Protocol and Send mode
Station no.	Baud rate	Data bit	Parity bit	Stop bit	Communication channel	User-defined
0	9600	8	None	1	RS232C null modem or RS422/485	Master

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



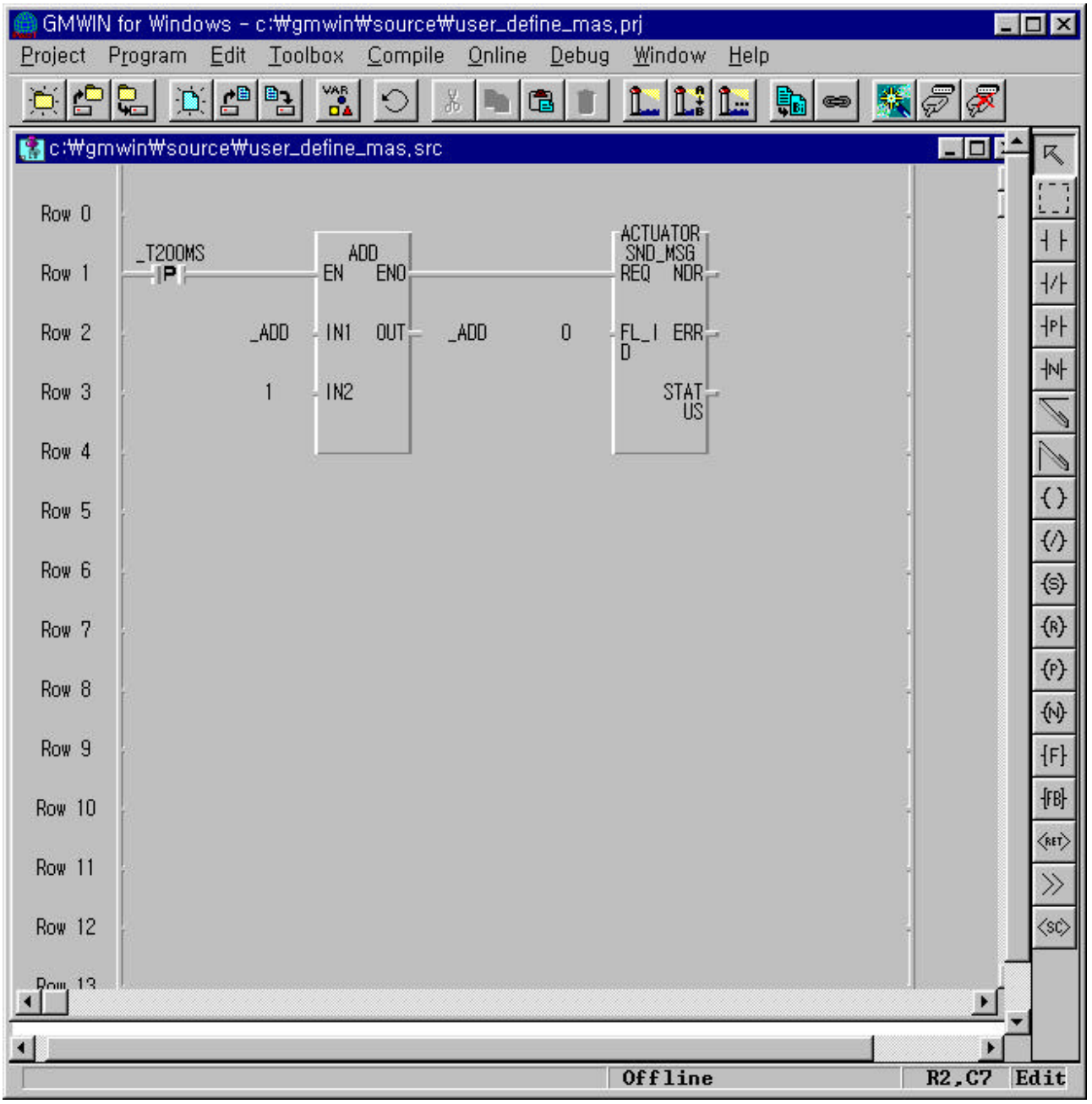
Item	Setting value
Header	None
Send/receive	Send
Segment 1	Type: CONST, field: %QB0.1.0, select button for ASCII input
Segment 2	Type: ARRAY, field: %QB0.1.0, size: 1 bytes
Tail	None

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



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.



- 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."**

- Variables window