Chapter 13　The Applications for FBs-PLC CLink Function

As previously revealed in Chapter 12 that the FBs-PLC can support the "Ladder Program Control Interface" communication function for the applications of multi_drop FATEK CPU Link network or connecting with the intelligent peripherals through Port 1 ~ Port 4.

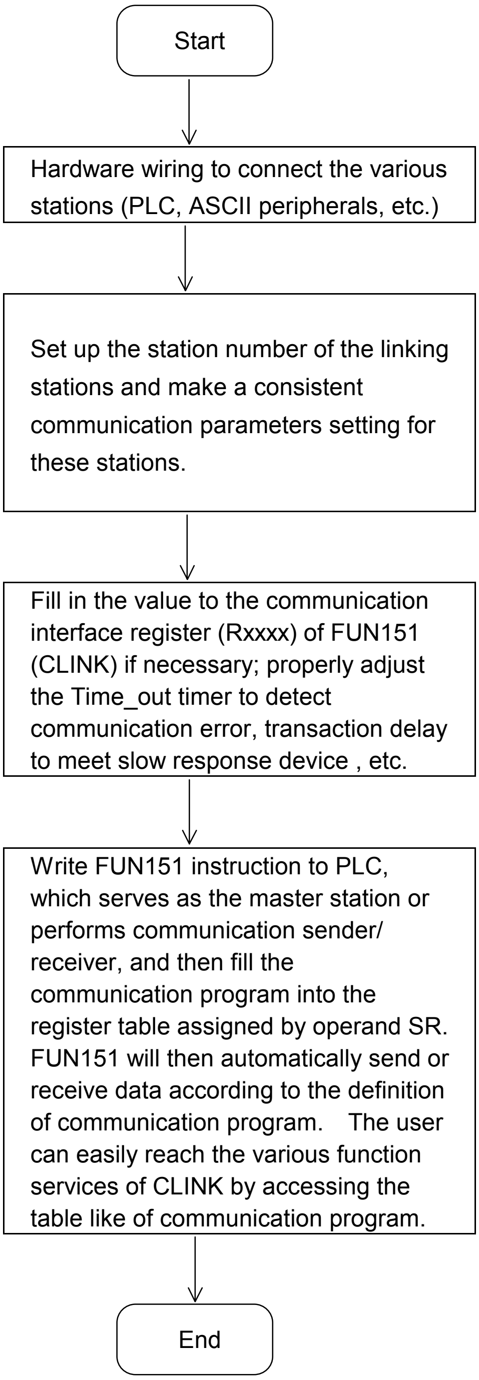
The RS-232 interface is for point to point connection, the RS-485 interface is for long distance connection or multi-drop communication network.

The FUN151 (CLINK) instruction provides MD0 to MD3 four kinds of instruction mode, that the MD3 mode is monopolized by Port 2 for “FATEK High Speed CPU Link Network” ,　the others are for “Ordinary Communication Link”. The following list enlisted the description for the difference on various instruction modes for the CLINK instruction.

Category	Item	Baud Rate	Data Length	Transmitting code	Error detection	Command processing speed
FUN151 (CLINK)	High Speed LINK （MD3） * Port 2 only	38.4K bps | 921.6K bps	8-bit	Binary code	CRC-16	Immediately
	Ordinary LINK （MD0～MD2）	4.8K bps | 921.6K bps	7-bit or 8-bit Adjustable	ASCII code	Checksum	Processing during Housekeeping

13.1 Application for FUN151 instruction

13.1.1 Procedures for usage



Start

Hardware wiring to connect the various stations (PLC, ASCII peripherals, etc.)

Set up the station number of the linking stations and make a consistent communication parameters setting for these stations.

Fill in the value to the communication interface register (Rxxxx) of FUN151 (CLINK) if necessary; properly adjust the Time_out timer to detect communication error, transaction delay to meet slow response device , etc.

Write FUN151 instruction to PLC, which serves as the master station or performs communication sender/ receiver, and then fill the communication program into the register table assigned by operand SR. FUN151 will then automatically send or receive data according to the definition of communication program. The user can easily reach the various function services of CLINK by accessing the table like of communication program.

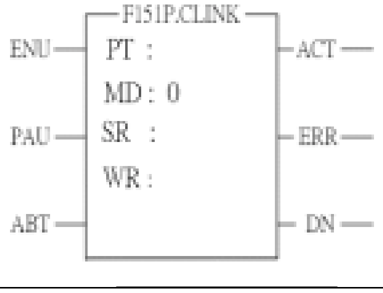
End

- Station number can be set to any one between 1 to 254 without replication.
- For communication parameters, please refer to the description of "Communication Related Setting".

13.1.2 Explanation of respective modes and application program for FUN151

This section will base on the four instruction modes (MD0 to MD3) of FUN151 (CLINK) instruction to explain their usages, with respective practical application program examples.

FUN 151 CLINK	Convenient instruction of FUN151: MD0 (Which makes PLC act as the master station in CPU LINK network through Port 1~4)	FUN 151 CLINK



Pt : Assign the port, 1～4

MD : 0, serves as the master station of Fatek CPU Link (employs Fatek communication protocol)

SR : Starting register of communication program (see example for its explanation)

WR: Starting register for instruction operation (see example for its explanation). It controls 8 registers, the other programs can not repeat in using.

Range Operand	HR	ROR	DR	K
	R0 ｜ R3839	R5000 ｜ R8071	D0 ｜ D3999	
Pt				1~4
MD				0~3
SR	○	○	○	
WR	○	○*	○	

Descriptions

1. FUN151: MD 0, it makes PLC act as the master of FATEK CPU Link Network through Port 1~ 4.
2. The master PLC may connect with 254 slave stations through the RS-485 interface.
3. Only the master PLC needs to use FUN151 instruction, the slave doesn't need.
4. It employs the program coding method or table filling method to plan for the data flow controls; i.e. from which one of the slave station to get which type of data and save them to the master PLC, or from the master PLC to write which type of data to the assigned slave station. It needs only seven registries to make definition; every seven registers define one packet of data transaction.
5. When execution control 〝EN↑〞changes from 0→1 and both inputs "PAU" and "ABT" are 0, and if Port 1/2/3/4 hasn't been controlled by other communication instructions [i.e. M1960 (Port1) / M1962 (Port2) / M1936 (Port3) / M1938 (Port4) = 1], this instruction will control the Port 1/2/3/4 immediately and set the M1960/M1962/M1936/M1938 to be 0 (which means it is being occupied), then going on a packet of data transaction immediately. If Port 1/2/3/4 has been controlled (M1960/M1962/M1936/M1938 = 0), then this instruction will enter into the standby status until the controlling communication instruction completes its transaction or pause/abort its operation to release the control right (M1960/M1962/M1936/M1938 =1), and then this instruction will become enactive, set M1960/M1962/M1936/M1938 to be 0, and going on the data transaction immediately.
6. While in transaction processing, if operation control "PAU" becomes 1, this instruction will release the control right (M1960/M1962/M1936/M1938 = 1) after this transaction. Next time, when this instruction takes over the transmission right again, it will restart from the next packet of data transaction.
7. While in transaction processing, if operation control "ABT" becomes 1, this instruction will abort this transaction immediately and release the control right (M1960/M1962/M1936/M1938 = 1). Next time, when this instruction takes over the transmission right again, it will restart from the first packet of data transaction.
8. While it is in the data transaction, the output indication "ACT" will be ON.
9. If there is error occurred when it finishes a packet of data transaction, the output indication "DN" & "ERR" will be ON.
10. If there is no error occurred when it finishes a packet of data transaction, the output indication "DN" will be ON.

【 Interface Signals】

. Dedicated Relays and Registers for corresponding port :

	Port 1	Port 2	Port 3	Port 4
Port Busy Indicator	M1960	M1962	M1936	M1938
Port Finished Indicator	M1961	M1963	M1937	M1939
Port Communication Parameters	R4146	R4158	R4043	R4044
TX Delay & RX Time-out Span	R4147	R4159	R4045	R4048

Port Busy Indicator : This signal is generated from CPU.
ON, it represents that port is free and ready.
OFF, it represents that port is busy, data transaction is going.

Port Finished Indicator : This signal is generated from CPU.
When the communication program completed the last packet of data transaction, this signal will be ON for one scan time (for successive data transaction).
When the communication program completed the last packet of data transaction, this signal will be still ON (for single packet of data transmission).

Port Communication Parameters :

The register is for communication parameters setting of corresponding port. (please refer to the chapter of communication parameters setting)

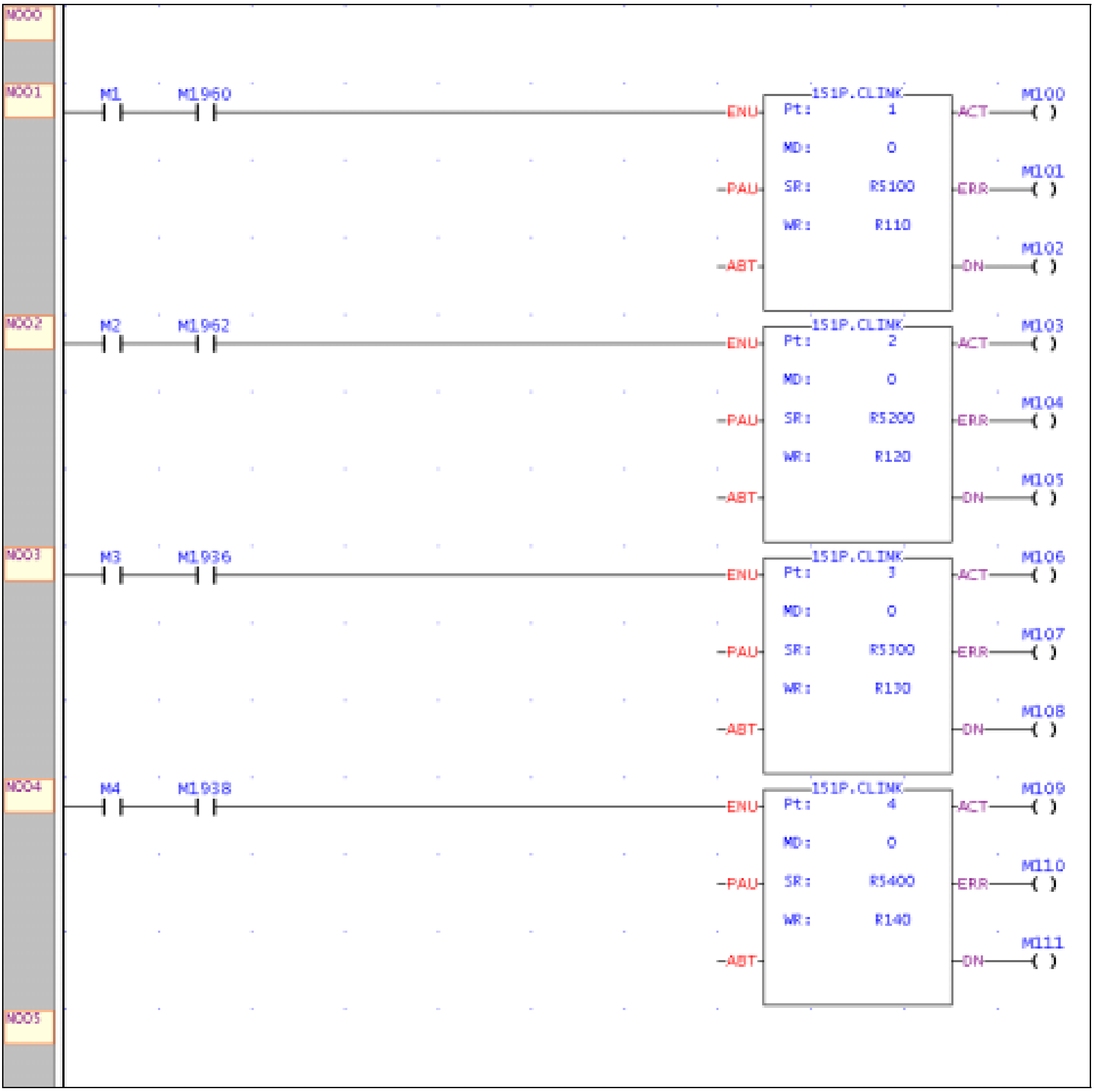
TX Delay & RX Time-out Span :

The content of Low Byte defines the receive Time-out span of CLINK instruction; its unit is 0.01 second (the default is 50, which means 0.5 second)
The CLINK instruction employs receive Time-out span to judge whether the slave station on line or not. When the master PLC sent out the read/write command to the slave station, the slave station didn't reply within this period means that there is abnormal event in communication called Time-out. When there are multi-drop linking, properly adjust this value (greater than 1 scan time of the slave station with the longest scann time) to shorten the communication response time among the active linking stations if there are many slave stations power off (The time-out cases will happen).

The content of High Byte defines the transmission delay time between two packets of data transaction for CLINK instruction; its unit is 0.01 second (the default is 0).

Making PLC act as the master station in CPU LINK network through Port 1~4

Program example Automatic cycling transmission

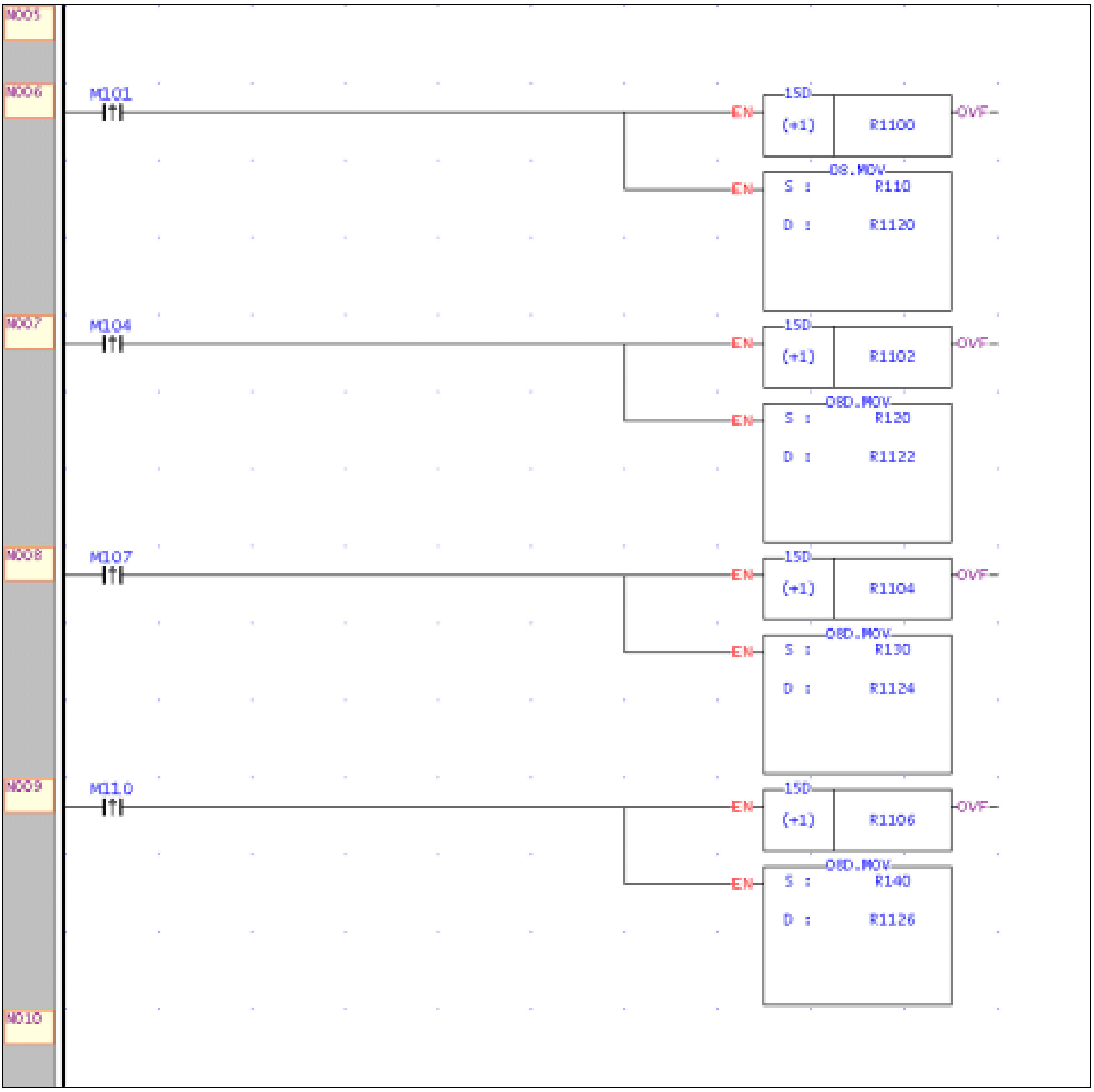


Explanation on program example

- When execution control M1/M2/M3/M4=1, and corresponding port is not occupied by other communication instruction (M1960/M1962/M1936/M1938=1), CLINK instruction will start the data transaction. The M1960/M1962/M1936/M1938 is OFF during data transaction, and when the transaction is finished, the M1960/M1962/M1936/M1938 becomes ON. Employ the OFF↔ON change of M1960/M1962/M1936/M1938 (FUN151 execution control “EN↑” = 0→1 means starting) may automatically starts for every packet of data transaction successively (when the last packet of transaction is completed, it will automatically return to the first packet of transaction to obtain the automatic cycling transmission).

Program example Error Logging

. When there is communication error, gets and stores the error message, it would be helpful for error analysis or logging.



Description

・Explanation for the operand SR of FUN151: MD0

SR：Starting register for communication program of CLINK instruction

SR+0	Total transactions	• Low Byte is valid; one transaction takes 7 registers to describe, which means 7 registers define a packet of data transaction.
SR+1	Slave station No. which is about to transact with	• Low Byte is valid, 0～254 (0 means that master PLC broadcasts the data to all slave PLC, the slave PLC does not reply)
SR+2	Command code	• Low Byte is valid; =1, means reading data from slave PLC; =2, means writing data to slave PLC.
SR+3	Data length of this transaction	• Low Byte is valid; the range is 1～64.
SR+4	Data type of Master PLC	• Low Byte is valid, and its range is 0 to 13; it defines the data type of master PLC (see next page).
SR+5	Starting reference of Master PLC	• Word is valid; it defines the starting address of data (master).
SR+6	Data type of slave PLC	• Low Byte is valid, and its range is 0 to 13; it defines the data type of slave PLC (see next page).
SR+7	Starting reference of Slave PLC	• Word is valid; it defines the starting address of data (slave).
SR+8	Slave station No. which is about to transact with	Description of the 2_{nd} packet of transaction
SR+9	Command Code	
SR+10	Data length of this transaction	
SR+11	Data type of Master PLC	
SR+12	Starting reference of Master PLC	
SR+13	Data type of slave PLC	
SR+14	Starting reference of Slave PLC	
	⋮	

Making PLC act as the master station in CPU LINK network through Port 1~4

・ Master/Slave data type, code and reference number

Data code	Data type	Reference number
0	X (discrete input)	0～255
1	Y (discrete output)	0～255
2	M (internal relay M)	0～1911
3	S (step relay S)	0～999
4	T (timer contact)	0～255
5	C (counter contact)	0～255
6	WX (word of discrete input ,16 bits)	0～240, it must be the multiple of 8.
7	WY (word of discrete output ,16 bits)	0～240, it must be the multiple of 8.
8	WM (word of internal relay,16 bits)	0～1896, it must be the multiple of 8.
9	W S (word of step relay,16 bits)	0～984, it must be the multiple of 8.
10	TR (timer register)	0～255
11	CR (counter register)	0～199
12	R (data register Rxxxx)	0～3839
13	D (data register Dxxxx)	0～4095

Note: The data type for master and slave must be consistent. i.e. if the master station is any value between 0 to 5, the slave station must also be any value between 0 to 5; if the master station is any value between 6 to 13, the slave station must also be any value between 6 to 13.

・ Explanation for the operand WR of FUN151:MD0

WR：Start of working register

	High Byte	Low Byte
WR+0	Result code	Transaction No.
WR+1	Station number	Command code
WR+2	For internal operation	
WR+3	For internal operation	
WR+4	For internal operation	
WR+5	For internal operation	
WR+6	For internal operation	
WR+7	For internal operation	

- Result code indicates the transaction result; 0= normal, other value =abnormal.
- Transaction No. indicates which one is in processing.
- Station number, the slave station No. which is in transaction.

  Command code

  =44H, reading successive discrete status from slave PLC.

  =45H, writing successive discrete status to slave PLC.

  =46H, reading successive registers from slave PLC.

  =47H, writing successive registers to slave PLC.
- WR+4's b0=1, Port has been occupied and this instruction is waiting to acquire the transmission right for data transaction.

  b4=1 , This instruction is not first time performing.
  b12 , Output indication for "ACT"
  b13 , Output indication for "ERR".
  b14 , Output indication for "DN".

Result code: 0, this transaction is successful.
2, data length error (data length is 0 or greater than 64 in one transaction).
3, command code error (command code is greater than 2).
4, data type error (data type is greater than 13, please refer to data type code).
5, reference number error (please refer to reference number).
6, inconsistence in data type (e.g. master station is 0～5 while slave is 6～13).
A, no response from slave station (Time-out error).
B, communication error (received error data).

- For easy programming and trouble shooting, the Winproladder provides the table editing environment to edit the communication table of FUN151 instruction; Key in the complete FUN151 instruction first and then move the cursor to the position of it, depressing the "Z" key, now comes the table editing environment. The user can create the new communication table or display the existed table under this friendly user interface operation.

Communication Table for FUN151:MD0

Sequence	Command	Slave	Master Data	Slave Data	Length
0 ~ nnn	Read (=1) Write (=2)	Describing the station number of slave PLC which is about to transact with. Station number=0, The master PLC broadcasts the data to all slave PLCs and slave PLCs will not reply Station number=N, it means the station number of the slave PLC which is about to transact with the master PLC N=1~ 254	Describing the data type & reference number of this packet of transaction for the master PLC. X0 ~ X255 Y0 ~ Y255 M0 ~ M1911 S0 ~ S999 T0 ~ T255 C0 ~ C255 WX0 ~ WX240 WY0 ~ WY240 WM0 ~ WM1896 WS0 ~ WS984 TR0 ~ TR255 CR0 ~ CR199 R0 ~ R3839 D0 ~ D4095	Describing the data type & reference number of this packet of transaction for the slave PLC. X0 ~ X255 Y0 ~ Y255 M0 ~ M1911 S0 ~ S999 T0 ~ T255 C0 ~ C255 WX0 ~ WX240 WY0 ~ WY240 WM0 ~ WM1896 WS0 ~ WS984 TR0 ~ TR255 CR0 ~ CR199 R0 ~ R3839 D0 ~ D4095	Data length of this transaction. 1 ~ 64

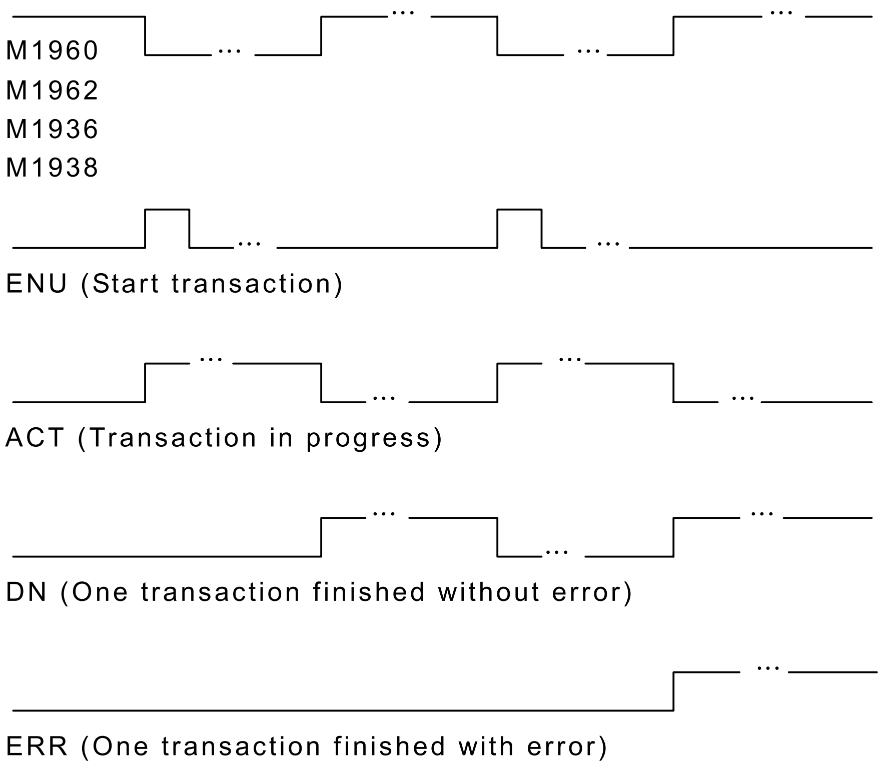
※ Win-Proladder provides the user friendly table edit for CLINK master：

Sequence No.	Command	Slave	Data of Master		Data of Slave	Data length
000	Read	1～254	X0～X255		X0～X255	1～64
			Y0～Y255		Y0～Y255	
			M0～M1911	←	M0～M1911	
			S0～S999		S0～S999	
			R0～R3839		R0～R3839	
			D0～D3999		D0～D3999	
001	Write	0～254	X0～X255		X0～X255	
			Y0～Y255		Y0～Y255	
			M0～M1911		M0～M1911	
			S0～S999	→	S0～S999	
			R0～R3839		R0～R3839	
			D0～D3999		D0～D3999	
002						
.						
.						

- Output Indicators

〝ACT〞 ON : Transaction is in progress
〝ERR〞 ON : Error occurred (Refer to the result code)
〝DN〞 ON : One transaction finished

- Waveform of Input and Output signals

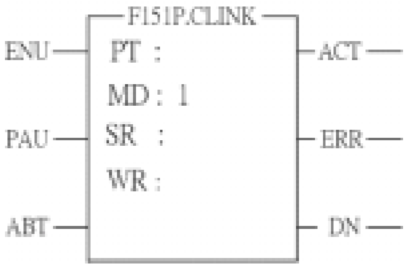

M1960
M1962
M1936
M1938
ENU (Start transaction)
ACT (Transaction in progress)
DN (One transaction finished without error)
ERR (One transaction finished with error)

Note : 1. Only "DN" will be ON if one transaction finished without error

2. "ERR" & "DN" will be ON at the same time if one transaction finished with error

3. M1961/M1963/M1937/M1939 will be ON one scan time while the last packet of transaction finished

FUN151 CLINK	Convenient instruction of FUN151: MD1 (Which makes PLC act as the communication sender through Port 1~4)	FUN151 CLINK



Pt : Assign the port, 1～4

MD: 1, link with intelligent peripherals that equipped with communication interface

SR : Starting register for data transmission table

WR: Starting register for instruction operation (see example for explanation). It controls 8 registers, the other programs cannot repeat in use.

Range Operand	HR	ROR	DR	K
	R0 | R3839	R5000 | R8071	D0 | D3999	
Pt				1～4
MD				0～3
SR	○	○	○	
WR	○	○*	○	

Descriptions

1. FUN151:MD1, it makes PLC act as the communication sender to link with the intelligent peripherals that equipped with communication interface.

2. A master PLC may connect to multi sets of peripherals that have identical communication protocol through the RS-485 interface.

3. The communication protocol/format is written with LADDER program, which must be consistent with the linked peripherals.

4. When execution control 〝EN↑〞 changes from 0→1 and both inputs “PAU” and “ABT” are 0, and if Port 1/2/3/4 hasn’t been controlled by other communication instructions [i.e. M1960 (Port1) / M1962 (Port2) / M1936 (Port3) / M1938 (Port4) = 1], this instruction will control the Port 1/2/3/4 immediately and set the M1960/M1962/M1936/M1938 to be 0 (which means it is being occupied), then going on a packet of data transaction immediately. If Port 1/2/3/4 has been controlled (M1960/M1962/M1936/M1938 = 0), then this instruction will enter into the standby status until the controlling communication instruction completes its transaction or pause/abort its operation to release the control right (M1960/M1962/M1936/M1938 =1), and then this instruction will become enactive, set M1960/M1962/M1936/M1938 to be 0, and going on the data transaction immediately.

5. During transaction, if the “PAU” input becomes 1, this instruction will pause and release the control right (set M1960/M1962/M1936/M1938 = 1) after it completed the transmission of the on-going data.

6. During transaction, if the “ABT” input becomes 1, this instruction will abort the transmission and release the control right immediately (set M1960/M1962/M1936/M1938 = 1).

7. While transaction is going, the output indication “ACT” will be ON.

8 When a packet of data transaction is finished (transmission finished or "transmit then receive" completed), if there is error occurred, the output indication “DN” & “ERR” will be ON.

9 When a packet of data transaction is finished (transmission finished or "transmit then receive" completed), if there is no error occurred, the output indication “DN” will be ON.

Convenient instruction of FUN151: MD1
(Which makes PLC act as the communication sender through Port 1~4)

【 Interface Signals】

. Dedicated Relays and Registers for corresponding port :

	Port 1	Port 2	Port 3	Port 4
Port Busy Indicator	M1960	M1962	M1936	M1938
Port Finished Indicator	M1961	M1963	M1937	M1939
Port Communication Parameters	R4146	R4158	R4043	R4044
RX Time-out Span	R4147	R4159	R4045	R4048
New Message Detection Time Interval	R4148	R4148	R4148	R4148

Port Busy Indicator : This signal is generated from CPU.
ON, it represents that port is free and ready.
OFF, it represents that port is busy, data transaction is going.

Port Finished Indicator : This signal is generated from CPU.
ON, it means data transaction has been completed.

Port Communication Parameters :
The register is for communication parameters setting of corresponding port. (please refer to the chapter of communication parameters setting)

RX Time-out Span :

The content of Low Byte defines the receive Time-out span of CLINK instruction; its unit is 0.01 second (the default is 50, which means 0.5 second)
The CLINK instruction employs receive Time-out span to judge whether the slave station on line or not. When the master PLC sent out the read/write command to the slave station, the slave station didn't reply within this period means that there is abnormal event in communication called Time-out. When there are multi-drop linking, properly adjust this value (greater than 1 scan time of the slave station with the longest scann time) to shorten the communication response time among the active linking stations if there are many slave stations power off (The time-out cases will happen).

The content of High Byte makes no sense at this mode.

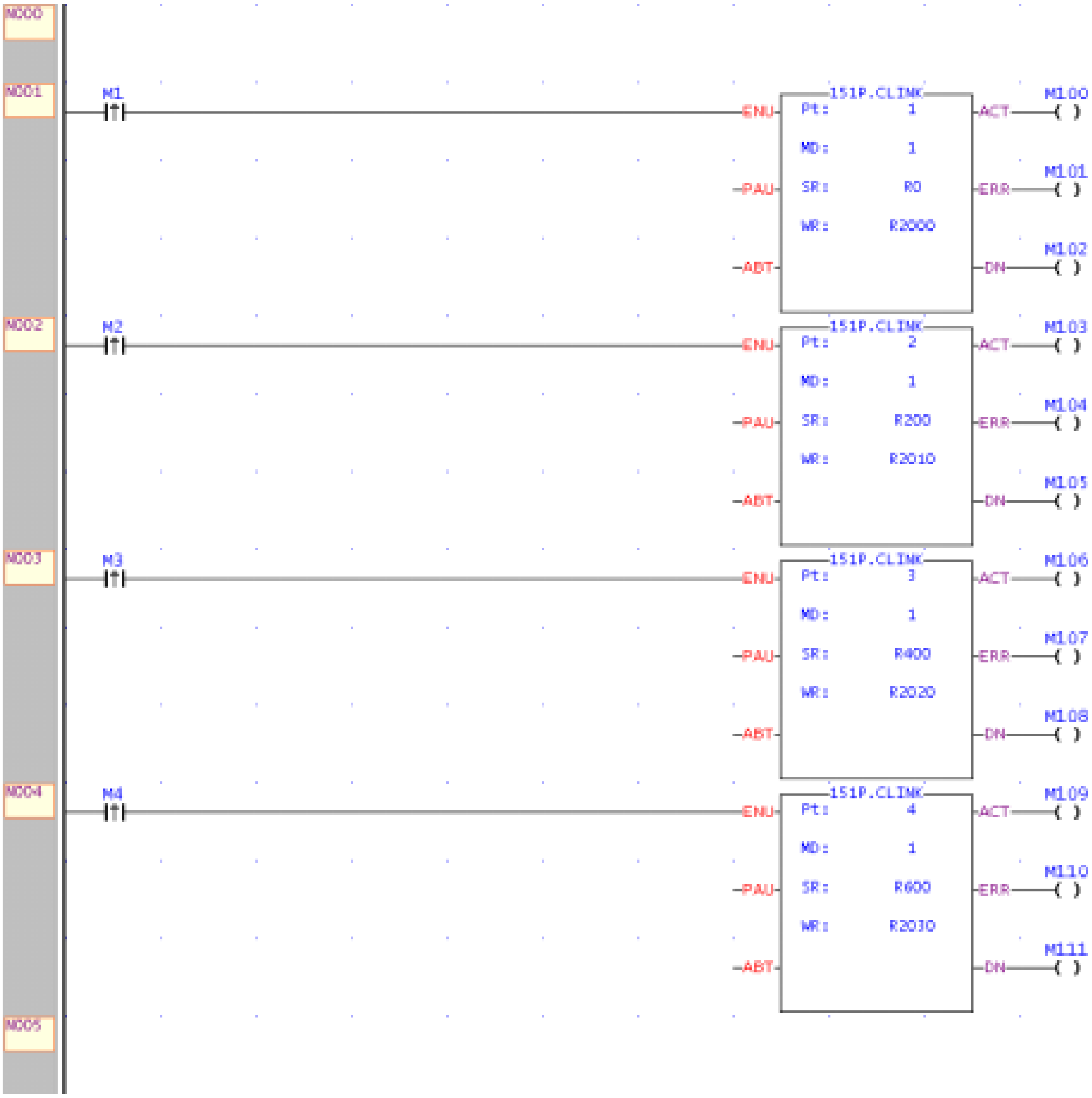
New Message Detection Time Interval :
While the communication port being used to communicate with the intelligent peripherals through the FUN151 convenient instruction, if the communication protocol without the end of text to tell the last character of message frame, it needs message detection time interval to judge the end of receiving packet. High byte of R4148 is used for this setting.

※High Byte of R4148 : New message detection time interval setting for Port 1~ 4 (Unit in mS)

Convenient instruction of FUN151: MD1

(Which makes PLC act as the communication sender through Port 1~4)

Program example Making the PLC act as the master to link with non-statndard communication protocol peripheral devices thorugh Port 1,2,3,4



Convenient instruction of FUN151: MD1

(Which makes PLC act as the communication sender through Port 1~4)

● Explanation for the operand SR of FUN151: MD1

SR : Starting register of data transmission table

SR+0	Transmit only or Transmit then Receive	• Low byte is valid, 0: transmit only, no response from the slave 1: transmit then receive the responding message.
SR+1	Starting & Ending code for receiving	• High byte : Start of text for receiving Low byte : End of text for receiving
SR+2	Length of Transmission	• The maximum length of data to be transmitted is 511
SR+3	Data 1	• Low byte is valid
SR+4	Data 2	• Low byte is valid
SR+5	Data 3	• Low byte is valid
SR+6	Data 4	• Low byte is valid
• • •		
	Data N	• Low byte is valid

Note 1: When selecting the transmit-only mode, the Starting /Ending code of receiving is meaningless.

2: When it is in the "transmit then receive" mode, before the starting of transmission, it must first to estimate the starting and ending code of responding message from communication partner and write them into the receiving starting/ending code register (e.g. SR+1=0203H, 02H stands for starting code and 03H for ending code), so as to ensure the correct message frame receiving. The communication protocol with starting/ending code makes the identifying of every packet of messages easy, and the communication program is simple and efficient.

3: When it is in the "transmit then receive" mode, fills the high byte of starting/ending code register with 0 if no starting code in responding message; if no ending code in responding message, fills 0 to the low byte of starting/ending code register. Adjusts the high byte of R4148 (message detection time interval) to judge whether a packet of data has been received completely; the unit is 0.001 second (the default is 0CH, 12mS).

The communication protocol without ending code depends on message detection time interval to tell whether it has received completely a packet of data (the setting of message detection time interval must be greater than the maximum response delay time between data bytes when communication partner is replying), thus it may ensure the receiving of the whole packet to be complete. Generally speaking, the data in transmitting is transmitted one byte after another continuously; therefore, if there is pause (greater than message detection time interval), it means the packet of message is transmitted completely.

Convenient instruction of FUN151: MD1

(Which makes PLC act as the communication sender through Port 1~4)

● Explanation for the operand WR of FUN151:MD1

WR : Start of working register

	High Byte	Low Byte	
WR+0	Result code	0	• Result code =0, OK ; = other values, abnormal.
WR+1	For internal operation use		• Working registers for CLINK instruction
WR+2	For internal operation use		
WR+3	For internal operation use		

WR+4	For internal operation use		• WR+4 : b0=1, Pending
			b12= 〝ACT〞 output indication
			b13= 〝ERR〞 output indication
			b14= 〝DN〞 output indication

WR+5	For internal operation use		
WR+6	For internal operation use		
WR+7	For internal operation use		
WR+8	Total amount of data received		• The total amount of data byte being received (the register for received data length; it includes the starting and ending code).
WR+9	Data 1		• The first byte of data received (if there is the starting code, it is the starting code); High byte =0.
•	Data 2		• The second byte of data received; High byte =0.
•	Data 3		• The third byte of data received; High byte =0.
• •			
•	Data N		• The N_th byte of data received (if there is the ending code, it is the ending code); High byte =0.

Result code : 0, transaction is successful.

2, data length error (the value is 0, or the packet of transaction is greater than 511)

A, no response from the slave

B, communication abnormal (received error data)

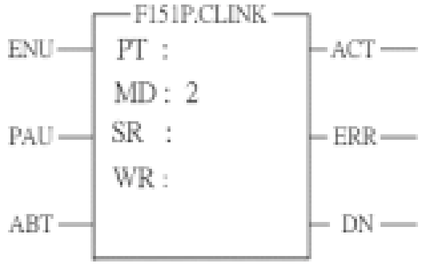
● Output Indicator

〝ACT〞 ON : Transaction is in progress

〝ERR〞 ON : Error occurred

〝DN〞 ON : One transaction finished

FUN151 CLINK	Convenient instruction of FUN151: MD2 (Which makes PLC act as the communication receiver through Port 1~4)	FUN151 CLINK



Pt : Assign the port, 1～4

MD: 2, PLC waiting to receive the message sent by intelligent peripherals

SR : Starting register for data transmission table

WR: Starting register for instruction operation (see example for explanation). It controls 8 registers, the other programs cannot repeat in use.

Range Operand	HR	ROR	DR	K
	R0 | R3839	R5000 | R8071	D0 | D3999	
Pt				1～4
MD				0～3
SR	○	○	○	
WR	○	○*	○	

Descriptions

1. FUN151: MD2 instruction provides Fatek PLC with ability to receive message sent by peripherals with communication interface at any time.

2. The communication protocol is written with LADDER program, which must be consistent to the peripheral device.

3. When execution control 〝EN↑〞 changes from 0→1 and both inputs “PAU” and “ABT” are 0, and if Port 1/2/3/4 hasn’t been controlled by other communication instructions [i.e. M1960 (Port1) / M1962 (Port2) / M1936 (Port3) / M1938 (Port4) = 1], this instruction will control the Port 1/2/3/4 immediately and set the M1960/M1962/M1936/M1938 to be 0 (which means it is being occupied). If Port 1/2/3/4 has been controlled (M1960/M1962/M1936/M1938 = 0), then this instruction will enter into the standby status until the controlling communication instruction completes its transaction or pause/abort its operation to release the control right, and then this instruction will become enactive.

4. When the input “PAU” or “ABT” becomes 1, it gives up the receiving immediately (M1960/M1962/M1936/M1938 = 1).

5. While it is in the receiving state, the output indication “ACT” is ON.

6. When a packet of data transaction finished (receive finished or receive then transmit completed), if there is error occurred, the output indication “DN” & “ERR” will be ON for one scan time.

7. When a packet of data transaction finished (receive finished or receive then transmit completed), if there is no error occurred, the output indication “DN” will be ON for one scan time.

【Interface Signals】

. Dedicated Relays and Registers for corresponding port :

	Port 1	Port 2	Port 3	Port 4
Port Busy Indicator	M1960	M1962	M1936	M1938
Port Finished Indicator	M1961	M1963	M1937	M1939
Port Communication Parameters	R4146	R4158	R4043	R4044
TX Time-out Span	R4147	R4159	R4045	R4048
New Message Detection Time Interval	R4148	R4148	R4148	R4148

Port Busy Indicator : This signal is generated from CPU.
ON, it represents that port is free and ready.
OFF, it represents that port is busy, data transaction is going.

Port Finished Indicator : This signal is generated from CPU.
ON, it means data transaction has been completed.

Port Communication Parameters :
The register is for communication parameters setting of corresponding port. (please refer to the chapter of communication parameters setting)

TX Time-out Span :
The Low Byte defines the Time-out span of FUN151:MD2 instruction; its unit is 0.01 second (the default is 32H). When the PLC received the message and must respond to it (receive then transmit mode), but the LADDER program is unable to process and send out the responding message during this period of time, the CPU will give up response this time and automatically restore back to receiving state. When FUN151:MD2 is set to be "receive only" mode, this value is meaningless.

The content of High Byte makes no sense at this mode.

New Message Detection Time Interval :
While the communication port being used to communicate with the intelligent peripherals through the FUN151 convenient instruction, if the communication protocol without the end of text to tell the last character of message frame, it needs message detection time interval to judge the end of receiving packet. High byte of R4148 is used for this setting.

※High Byte of R4148：New message detection time interval setting for Port 1~ 4 (Unit in mS)

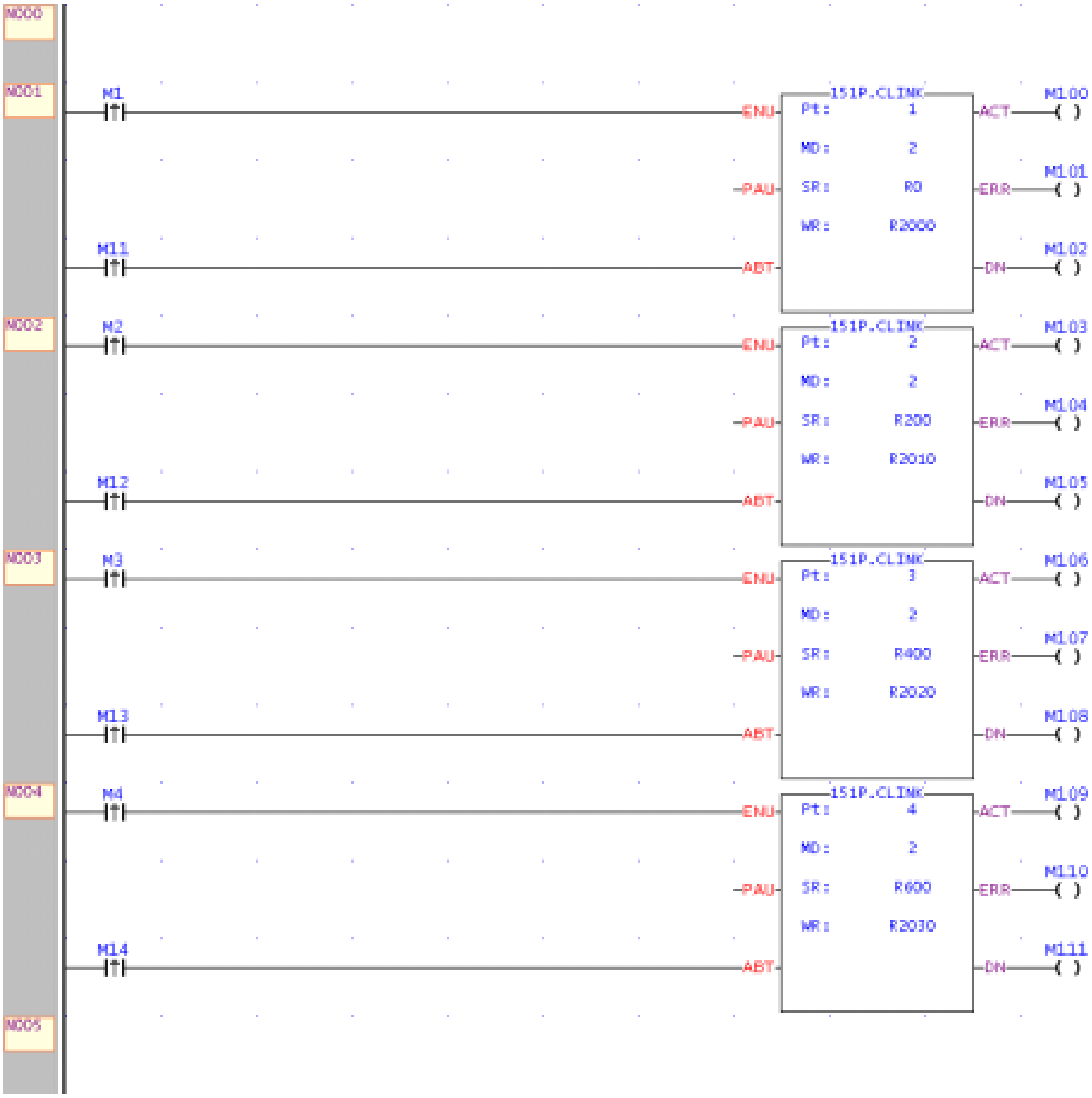
Note 1: Once FUN151:MD2 activated, it will stay in receiving state all the time; unless the input signal of PAU” or “ABT” becomes ON, then it will escape from receiving state and stop receiving and waiting for next time it will be activated again.

2: When there is change on Starting/Ending code for receiving, it must make the input signal of PAU” or “ABT” becomes ON once, and re-activate the receive control “EN ↑ ” from 0→1 to start message receiving

Convenient instruction of FUN151: MD2

(Which makes PLC act as the communication receiver through Port 1~4)

Program example Making the PLC act as the slave to receive the data from non-statndard communication protocol peripherals thorugh Port 1,2,3,4



Convenient instruction of FUN151: MD2

(Which makes PLC act as the communication receiver through Port 1~4)

● Explanation for the operand SR of FUN151: MD2

SR : Starting register of data receiving table

SR+0	Receive only or Receive then Transmit	• Low Byte is valid, =0, "receive only" mode ; =1, "receive then transmit" mode
SR+1	Starting/Ending code of receiving	• High Byte : Describing the starting code of receiving Low Byte : Describing the ending code of receiving.
SR+2	Length of reply data	• Maximum of length is 511. It will satrt to transmitte the reply data as long as the length is not 0
SR+3	Reply data 1	• Low Byte is valid
SR+4	Reply data 2	• Low Byte is valid
• • •		
	Reply data N	• Low Byte is valid

Note 1: When selecting the "receive only" mode, CPU fills the received data into the receiving registers and set the length after it has received a packet of message, and starts to receive the next packet of message immediately.

2: When selecting the "receive then transmit" mode, CPU fills the received data into the receiving registers and set the length after it has received a packet of message; then it starts to wait for the reply data length which is not zero to start transmitting reply data (therefore when select this mode, it must control the reply data length to be zero before the reply data completely filled into the reply registers; when the reply data fills into the reply registers finished, it may then set the length of reply data).

3: It must fills the starting code and ending code into the starting/ending code register before the starting of receiving (e.g. SR+1=0A0DH, 0AH stands for starting code and 0DH for ending code), so as to ensure it to be free from receiving error.

The communication protocol with starting/ending code makes the identifying of every packet of messages easy, and the communication program is simple and efficient.

4: If the receiving message without starting code, fills the high byte of starting/ending code with 0; if the receiving message without ending code, fills the low byte of starting/ending code with 0. Adjusting High Byte of R4148 (new message detection time interval) to detect whether a packet of message has been received completely, the unit is 0.001 second (default is 0CH, 12 mS). The communication protocol without ending code depends on new message detection time interval to tell whether it has received completely for a packet of data (the setting of new message detection time interval must be greater than the maximum delay time between data bytes to be received), thus it may ensure the receiving of the whole packet to be completed. Generally speaking, the data in transmitting is transmitted one byte after another continuously; therefore, if there is pause (greater than new message detection time interval), it means that the packet of message is transmitted completely.

5: When selecting "receive only" mode, if the receiving message has no ending code, the interval between every packet of data sent by the sender must be greater than the receiver's new message detection time interval, otherwise the receiver won't be able to distinguish between each packet of data correctly.

Convenient instruction of FUN151: MD2

(Which makes PLC act as the communication receiver through Port 1~4)

● Explanation for the operand WR of FUN151:MD2

WR : Start of working register

	High Byte	Low Byte	
WR+0	Result code	0	Result code =0, OK ; = other values, abnormal.
WR+1	For internal operation use		• Working registers for CLINK instruction
WR+2	For internal operation use		
WR+3	For internal operation use		

WR+4	For internal operation use		• WR+4 : b0=1, Pending
			b12= 〝ACT〞 output indication
			b13= 〝ERR〞 output indication
			b14= 〝DN〞 output indication

WR+5	For internal operation use		
WR+6	For internal operation use		
WR+7	For internal operation use		
WR+8	Total amount of data received		• The total amount of data byte being received (the register for received data length; it includes the starting and ending code).
WR+9	Data 1		• The first byte of data received (if there is the starting code, it is the starting code); High byte =0.
•	Data 2		• The second byte of data received; High byte =0.
•			
•	Data N		• The N_th byte of data received (if there is the ending code, it is the ending code); High byte =0.

Note: When CPU received a packet of message, it filled the data to receiving registers and set up the received data length. Before the LADDER program starts to receive, you may clear the register of received data length to be 0; it means the receiving of a new packet of message when compared and found that the received data length is not zero. After the LADDER program gets the received data, it clears the received data length register to be 0. Just compare to see the received data length register is not zero means the receiving of a packet of new message, and so it may easily to process the receiving action.

Result code: 0, data transaction is successful.
2, the data length is error (the value is 0, or the transaction is greater than 511)
A, unable to reply message within Time-out span ("receive then transmit" mode).
B, communication abnormal (received error data)

Convenient instruction of FUN151: MD2

(Which makes PLC act as the communication receiver through Port 1~4)

● Explanation of input control for program example

1. When the execution control input M1//M2/M3/M4 change from 0→1, if assigned port is not controlled by other communication instruction (M1960/M1962/M1936/M1938=1) and it enters into the receiving state immediately (M1960/M1962/M1936/M1938 keeping OFF all the time).

2. When "ABT" input M11/M12/M13/M14 changes from 0→1, it escapes from receiving state immediately (M1960/M1962/M1936/M1938=1)

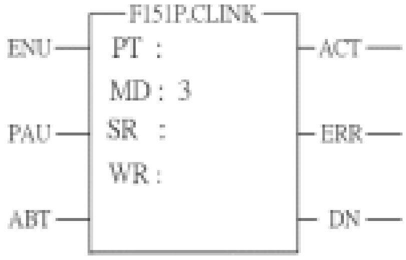
● Output indication

"ACT" ON : In receiving state

"ERR" ON : Error occurred in previous packet of transaction, it will be ON for a scan time

"DN" ON : The previous packet of transaction completed without error, ON for a scan time.

FUN151 CLINK	Convenient instruction of FUN151: MD3 (it makes the PLC serve as the master of “Fatek high speed CPU Link network” through Port2)	FUN151 CLINK



Pt : Only port 2 is valid

MD : 3, serves as the master station of Fatek High Speed CPU Link network

SR : Starting register of communication program (see example for its explanation)

Pt : Starting register for instruction operation (see example for its explanation). It controls 8 registers, the other programs can not repeat in using.

Range / Operand	HR	ROR	DR	K
	R0 | R3839	R5000 | R8071	D0 | D3999	
Pt				1～4
MD				0～3
SR	○	○	○	
WR	○	○*	○	

Descriptions

1. FUN151: MD3, it provides high speed data sharing between Fatek's PLC (data response time will not be influenced by the scan time of PLC).
2. A master PLC can link with 254 slave PLCs at the most to share data through the RS-485 interface.
3. FUN151: MD3 is required only by master PLC, not by the slave PLC.
4. The station number of master PLC must be No.1, or it should be assigned by R4054 register if which is not No.1 but need to be as the master.
5. The setting of M1958 for slave PLC must be ON (M1958 OFF is for non-high speed link), but it’s not necessary for master PLC.
6. In high speed linking, the maximum Baud Rate is 921.6K bps and minimum is 38.4K bps (adjustable); the data length is fixed at 8 Bits. Data is transmitted with binary code (which is twice time as fast as ASCII Code), and the error checking is adopting CRC-16, which is more reliable than Checksum.
7. The principle of high speed linking data transmission is based upon the COMMON DATA MEMORY concept to design; e.g. as the master PLC sent out the content of R0 to R31, .the contents of R0～R31 for all the slave PLCs will be the same as the master’s; when slave PLC no.2 sent out the contents of R32～R47, the R32～R47 contents of master PLC and other slave PLCs will be the same as PLC station no.2’s, etc.
8. When PLC is in STOP mode, the Port 2 enters into the standard interface mode that it can connect to WinProladder, MMI, or graphic supervisor (the communication parameter is set by R4158).
9. It employs the program coding or table filling method to plan for data flow control; i.e. for what kind of data being sent from which PLC station to all the PLC on line, it takes only 7 registers (5 of which is being physically used, and 2 reserved) to define; every 7 registers define once communication transaction.
10. When execution control “EN ↑ ” changes from 0→1 and both pause “PAU” and abort “ABT” are 0, this instruction will control Port 2 and set M1962 to be “0” (being controlled) and processing the data transaction immediately, suppose the Port 2 is not controlled by other communication instruction (M1962=1). If Port 2 is being controlled (M1962=0), this instruction will enter into wait state until the controlling instruction completes the transmission or pause/abort the operation to release the control right (M1962=1); then it enacts from wait state, engages in the transmitting transaction and sets M1962 to be “0”.
11. When pause “PAU” or abort “ABT” of input is 1, it escapes from high speed data link immediately (M1962 ON).
12. Within the high speed linking, the output indication “ACT” is ON; Port 2 is occupied.
13. When there is error occurred while it is starting the high speed linking, the output indication “ERR” will be ON, and the high speed linking will not be performed.

【Interface signals】

M1958 : While in the PLC high speed data linking, slave PLC must set M1958 ON (not necessary for master PLC) For non high speed data linking of PLC, the slave PLC must set M1958 OFF.

M1962 : The signal is generated from CPU.
ON represents the Port 2 is available.
OFF represents the Port 2 is occupied.

M1963 : The signal is generated from CPU.
When M1967 is ON (this signal is controlled by the user program) and after the last packet of communication transaction is completed, the CPU sets M1962 and M1963 ON, and the high speed data transmission will be stopped; it must control "ABT" (transmission abort) to be ON, and then restart execution control "EN ↑ " to change from 0→1 before the high speed linking can restart.
When M1967 is OFF (this signal is controlled by the user program), the high speed data transmission will automatically restart a new transmission from the first packet of communication transaction (M1962 and M1963 is keeping OFF state) after the last packet of communication transaction is completed.

M1967 : One-time or cycling control (controlled by the user program)
ON, one cycle, it will stop after the last packet of data transaction is performed completely.
OFF, successive cycles, it will restart from first packet of transaction when it has finished the last packet of transaction.

R4054 : It assigns the PLC station which is not no.1 to act as the master of high speed linking.

	High byte	Low byte	
R4054	55	Station number.	H

When the station number of the PLC is not number 1, fills its station number (low byte of R4055 stores the station number) into the low byte of R4054 and writes to high byte of R4054 with 55H, and then controls the execution control input "EN ↑ " from 0→1; even though the PLC station which is not no.1, it can still be the master station for high speed linking.

R4055 : When high byte of R4055 is not 55H,Low byte of R4055 shows the station number of PLC.
When high byte of R4055 is 55H,Low byte of R4055 defines the station number of PLC.

R4058 : Showing the station number of slave PLC which is abnormal while high speed linking (0: Represents normal; if many slave PLC were abnormal in the mean time, it is possible to see only one number; after the debugging of abnormal and clear R4058 to be 0 until the value of R4058 keeping to be 0, it will then network works normal). In communication transaction program or table, it must exist the case for slave station to send data to other stations then can the master PLC detect whether the slave station is online without error; if in the communication transaction program or table, there is only the master station sending data to slave stations, the master PLC can't detect whether slave PLC is on line without error. The user must employ programming skill to add abnormal detecting program to the master PLC and slave PLC to do the error checking (as a matter of fact, the program is very simple; just makes the PLC, which is sending data, to create an ON↔OFF variation signal. Once the receiving PLC does not detect the ON↔OFF variation signal in a period of time, it means that there is communication error).

Convenient instruction of FUN151: MD3

(it makes the PLC serve as the master of “Fatek high speed CPU Link network” through Port 2)

R4059 : Error logging of abnormal slave PLC while high speed linking.

	High byte	Low byte	
R4059	Abnormal code	Abnormal count	H

Low byte: Abnormal count summation

High byte: Abnormal code

OAH, No response from slave station
OBH, Error data (CRC Error)
01H, Framing Error
02H, Over_Run Error
04H, Parity Error
08H, CRC error

Explanation for the checking method for abnormal communication is the same as that for R4058.

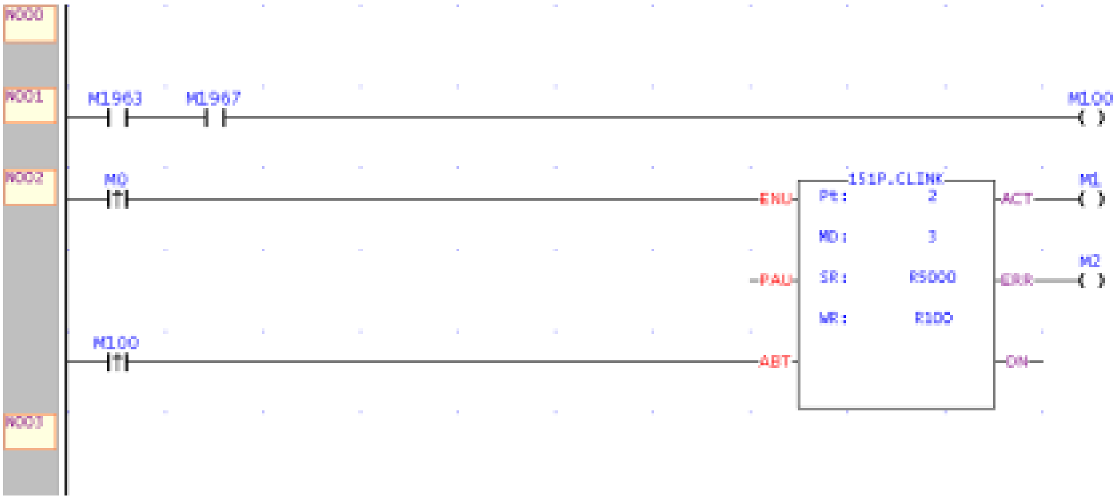
R4160 : Port 2 Rx/Tx Time-out setting (in high speed linking). The system will base on the setting of R4161 communication parameter to produce pertaining set point if high byte of R4160 is not 56H, the user need not to set it. If high byte of R4160 is 56H, the low byte of R4160 is reserved for manual setting.

R4161 : Communication parameter setting register for Port 2 High Speed CPU Link.

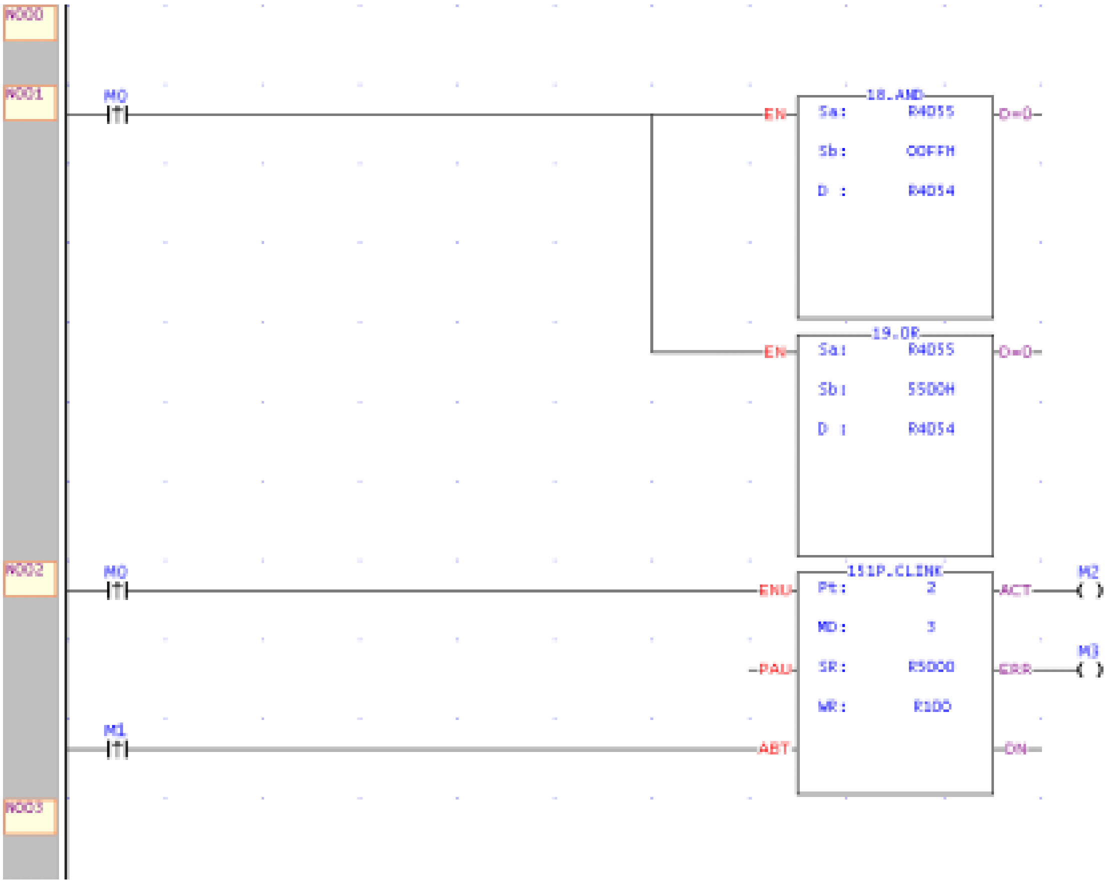
Convenient instruction of FUN151: MD3

(it makes the PLC serve as the master of “Fatek high speed CPU Link network” through Port 2)

Program example 1 PLC no. 1 serves as the master of high speed data linking



Program example 2 PLC which station number is not no.1 serves as the master of high speed data linking.



Program example 3 The same machine sets or equipments (with same LADDER program) perform multi-station data collection or distributed control through RS-485 high speed linking.

The principle for high speed data linking is based on COMMON DATA MEMORY concept to design; while designing, it must devise a successive data block and evenly distributed to respective PLCs to do data exchange among PLCs. e.g.:

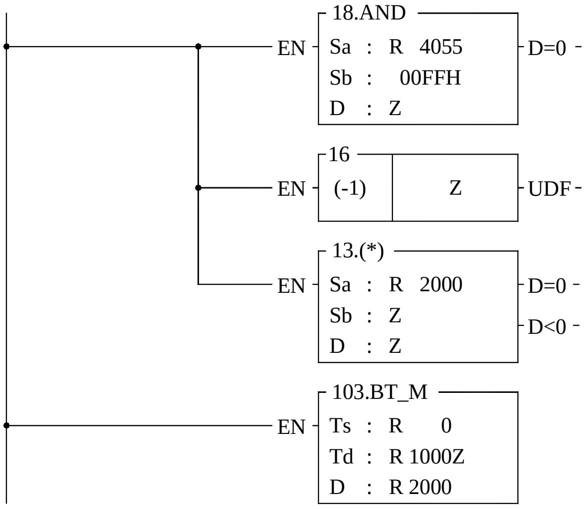
R1000～R1031: The data block of PLC no. 1 (through high speed linking, the other PLCs' content of R1000～R1031 become the same as that of PLC no.1).

R1032～R1063: The data block of PLC no. 2 (through high speed linking, the other PLCs' content of R1032～R1063 become the same as that of PLC no.2).

● ●
● ●
● ●

For example, get the production data (stored at R0～R31) from each machine set, and collectively gathering R1000～R1639 (suppose there are 20 sets linking) stored in master PLC through RS-485 high speed data linking; it needs merely the master PLC of high speed linking to connect to MMI or graphic supervisor, then it can monitor and store, for follow up processing, the production data of respective machine sets with real time effect.

Note: If it is simply for data collection and monitoring and no need to do real time control, employs the FUN151: MD0 can easily and concisely accomplish the assignment; when requiring real time control or supervisoring , it must employ FUN96: MD3 to accomplish a speedy, precisely controlling demand.



- Get PLC station number and write it in pointer Z
- Station number deducts 1
- R2000 = Length of data to be sent from each station (e.g. 32)
- data length * (station number–1): Directing to the apportioned data block of this station.
- Move production data from respective stations to the apportioned data block of respective stations, and transmitting it to all other PLCs on line through high speed data linking.

Convenient instruction of FUN151: MD3

(it makes the PLC serve as the master of “Fatek high speed CPU Link network” through Port 2)

Description

- Explanation for operand SR of FUN151: MD0
  SR：Starting register for communication program of CLINK instruction

SR+0	Packets of data transaction	● Low Byte is valid. A packet of transmission demands 7 registers to describe; i.e. 7 registers define a packet of data.
SR+1	Station number to transmitt	● Low Byte is valid. 1～254
SR+2	Command code	● Low Byte is valid, it can only be 4 (high speed linking command).
SR+3	Length of this packet of data	● Low Byte is valid. 1～32, defines the data length of one transaction.
SR+4	Data type	● Low Byte is valid. 12＝R; 13＝D.
SR+5	Data starting reference	● Word is valid. Defines starting number of working data.
SR+6	Reserved	● Code for data type 12: R data register
SR+7	Reserved	13: D data register
SR+8	Station number to transmitt	Describing for the 2_nd packet of transaction
SR+9	04	
	Length of data	
•	Data type	
•	Data starting reference	
•	Reserved	
•	Reserved	

Code for data type	Data starting reference
12: R data register	0～3839
13: D data register	0～3999

- Explanation for operand WR of FUN151:MD3
  WR：Starting address or working register

	High Byte	Low Byte
WR+0	Result code	
WR+1	For internal operation	
WR+7	For internal operation	

Result code: 0: Correct format
2: Data length error (Length is 0 or greater than 32)
3: Command code error (Command is not equal to 4)
4: Data type error (Data type is not 12 nor 13)
5: Data reference error

Convenient instruction of FUN151: MD3

(it makes the PLC serve as the master of “Fatek high speed CPU Link network” through Port 2)

- For easy programming and trouble shooting, the Winproladder provides the table editing environment to edit the communication table of FUN151 instruction; Key in the complete FUN151 instruction first and then move the cursor to the position of it, depressing the "Z" key, now comes the table editing environment. The user can create the new communication table or display the existed table under this friendly user interface operation.

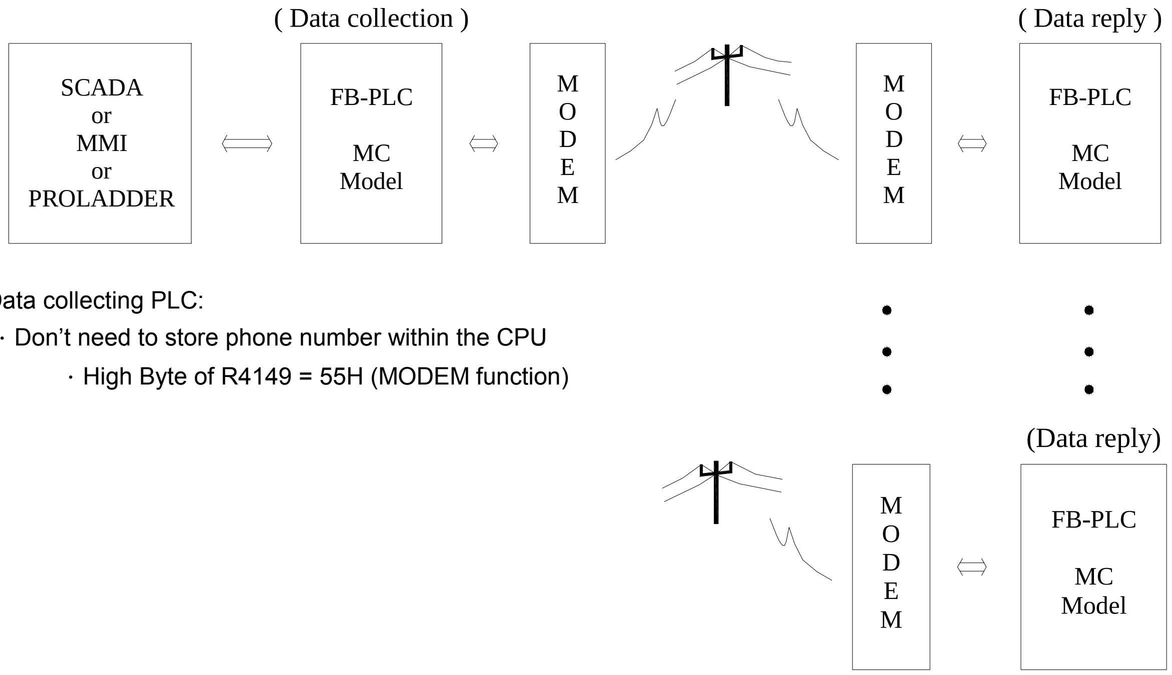
Communication Table for FUN151:MD3

※Only Port 2 is valid for FUN151：MD 3

Sequence	Command	Station No.	Data	All Station	Length
0～nnn	High Speed Link =4	Station number to transmitt the data 1～254	The data will be transmtted R0～R3839 D0～D3999	The data will be received R0～R3839 D0～D3999	Data length of this transaction 1～32

CPU Link by way of Port 1 to connect to Modem

- PLC can connect to MODEM through communication port 1, and by way of telecommunication network to link and share data with remote PLC. Its application is as follows:
  - · Perform automatic data collection from the remote end.
  - · Automatically report for alarm and abnormal conditions
  - · Associate with current available graphic supervisoring software or MMI etc. standard products to constitute a wide area network automatic monitoring system. It doesn't need to develop specific designing, so as to reduce the development risk and time limit.
- Hardware configuration, and setting:



Data collecting PLC:

· Don't need to store phone number within the CPU

· High Byte of R4149 = 55H (MODEM function)

Data reply PLC:

· High Byte of R4149 = 55H (MODEM function)

· R4140～R4145 sets the phone number for general data collecting PLC end (extension phone function allowed).
e.g. Phone number is 02-28082192, then R4140=8220H, R4141=1280H, and R4142=0E29H.
If phone number is: 02-28082192 ext 100, then R4140=2A20H, R4141=2808H, R4142=A291H, R4143=AAAAH, R4144=001AH, R4145=000EH.

· Explanation: R4140～R4145 is telephone number register for dialing;
"E" is the ending character of phone number; "A" is the dial delaying character (usually the dialing of extension number or international long distance call can be reached by making use of dial delaying, the delayed time for a delaying character is based on MODEM setting, which is about 2 second). "B" stands for "#" character (can dial B. B. CALL), and "C" stands for "*" character.

· It employs CLINK (FUN151:MD0) instruction to write data to the general data collecting PLC or to read data from general data collecting PLC (refer to FUN151:MD0 Instruction user guide).

· The maximum communication Baud Rate can reach 115200 bps
(both of the communication ends must be consistent in setting)

· Let the communication parameters be 8-bit and Non-parity will be better for almost Modem

CPU Link by way of Port 1 to connect to Modem

【MODEM dialing interface signal】

M1959: OFF, dialing by "Tone" ; ON, dialing by "Pulse"

M1964: OFF→ON, dial up ; ON→OFF, hang up

R4163: The Low Byte of R4163 is used to control the application of X instruction while MODEM dialing.

=1, it does not detect dial tone nor busy tone while MODEM dialing.

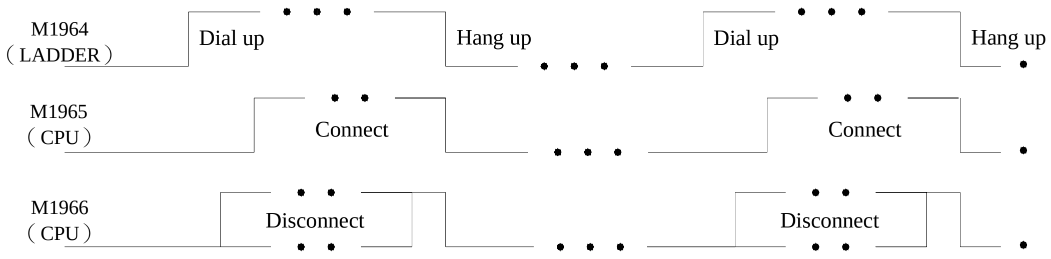
=2, it detects only dial tone but does not detect busy tone while MODEM dialing.

=3, it dials directly without detecting dial tone, but will detect busy tone after MODEM dialing.

=4, it detects both dial tone and busy tone for MODEM dialing.

For other values, it works as 4; different country system needs to adjust the setting pertaining to the country.

The High Byte of R4163 is used to set the ring count for auto answer mode of Modem.



Note 1: Of M1965 and M1966, there will be only one ON, not both to be ON at the same time.

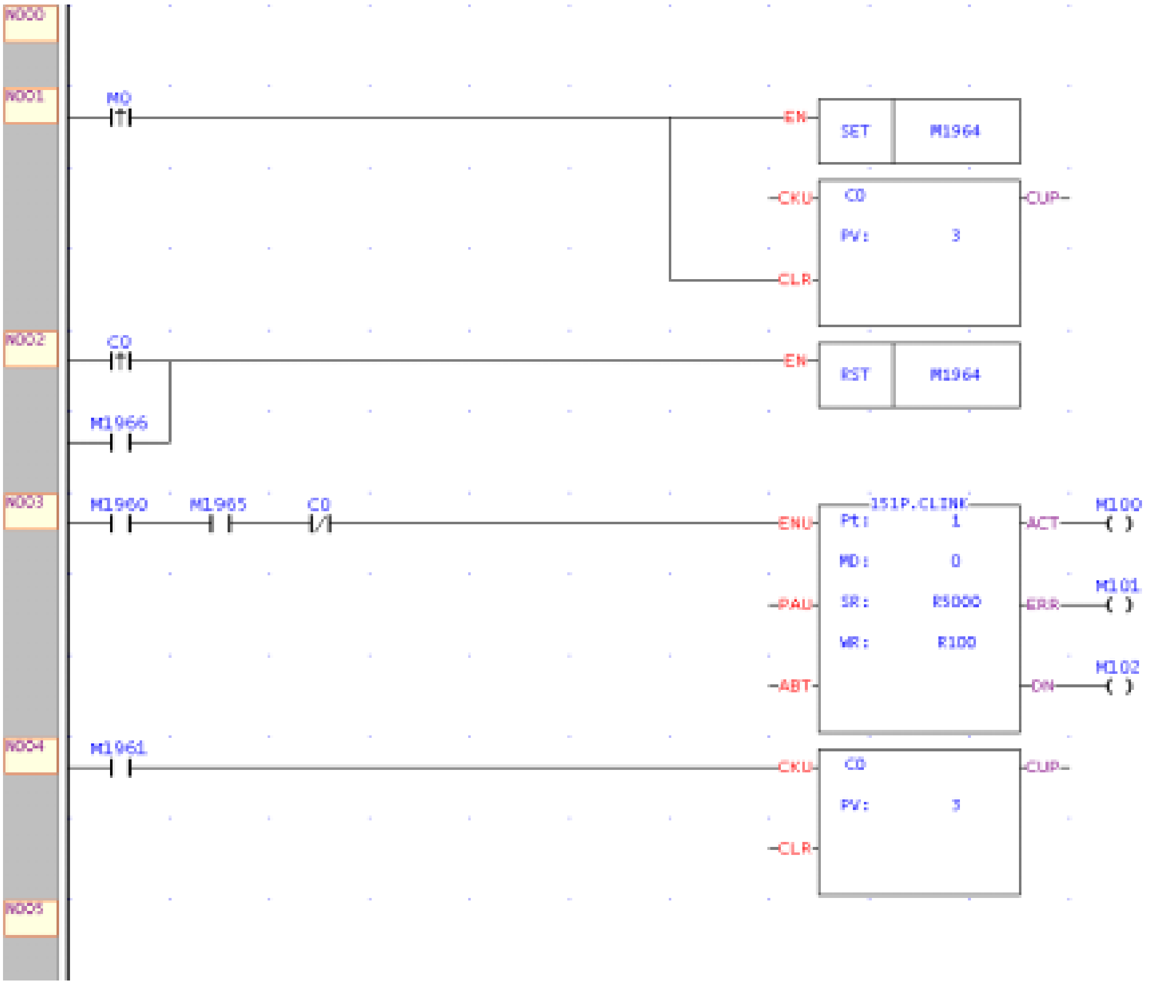
2: The waiting time for dial connection is 1 minute; if unable to connect, it will redial twice (totally 3 times). If all of the dial connection tries failed, CPU will set M1966 to be ON (connection failed).

3: When the quality of communication is not stable and easy to disconnect, you may employ the abnormal detecting function of CLINK instruction to control M1964 redials for connection (delay time of redial must be more than 10 seconds).

4: When PLC change from RUN to STOP, the CPU will automatically change MODEM to be receiving state, which could accept the remote side dial connection.

5: When PLC is not in dialing or MODEM connection states, CPU will automatically change MODEM to be receiving state, which could accept the remote side dial connection.

Example program


N000
N001
M0
EN
SET
M1964
CKU
C0
CUP
PV:
3
CLR
N002
C0
EN
RST
M1964
M1966
N003
M1960
M1965
C0
151P.CLINK
ENU
Pt:
1
ACT
M100
MD:
0
PAU
SR:
R5000
ERR
M101
WR:
R100
ABT
DN
M102
N004
M1961
CKU
C0
CUP
PV:
3
CLR
N005