

TSN/TSB TRAINING (PLC)

LS ELECTRIC

Automation CIC) Tech. Center (PE Part)

JONGHA PARK

CONTENTS

1. Line-Up

<u>1. PLC</u>	03
<u>2. HMI</u>	05
<u>3. GMC (General Motion Controller)</u>	07
<u>4. Special Module</u>	09
<u>5. Communication Module</u>	11

2. PLC

<u>1. XG5000 Basic</u>	13
<u>2. Data Structure</u>	73
<u>3. Devices</u>	83
<u>4. Instructions</u>	96
<u>5. Communication (Ethernet)</u>	154
<u>6. Communication (Cnet)</u>	170
<u>7. Communication (User Defined Protocol)</u>	178

3. XGI-CPUZ

<u>1. Overview</u>	187
<u>2. Specification</u>	190

1. Line-Up

1. PLC

1. PLC

	Large • Medium		Small • Micro		
	Redundancy system	High performance	Compact	Slim	Distributed I/O
Product					
	XGR	XGK/XGI	XBC/XEC	XBM-H	Smart I/O
Spec	- Processing speed: 42ns/step - I/O Device point: 131,072 - IEC 61131-3 standard language - Switching over time: 4.3 ~ 22 ms	- Processing speed: 8.5ns/step - I/O Device point: 131,072 - Dedicated/IEC 61131-3 standard language	- Processing speed: 60ns/step - I/O point: 352 - SD card (logging) - Dual Port Ethernet - Dedicated/IEC 61131-3 Standard language	- Processing speed: 40ns/step - I/O point: 256 6 axis Servo Control (Pulse) - Dedicated language	- Block type: 32 - Expansion type: 256 - RAPIDnet+ (EtherNet/IP, RAPIDnet, Modbus TCP/IP), EtherCAT - Profibus-DP, Rnet, DeviceNet, Modbus
Application	Process Automation - Water&Wastewater treatment - Power plant - Chemical plant - Oil & Gas, etc.	Factory Automation - Semiconductor - Secondary battery - Display, Home appliance - Automobile - Packaging, logistics	Machine - Press machine - Packaging machine - Refrigeration machine, etc.	Machine - Press machine - Packaging machine - Refrigeration machine, etc.	Distributed control (remote control) - Semiconductor - Secondary battery - Display, home appliance - Automobile, logistics, etc..

1. Line-Up

2. HMI

2. HMI

HMI Panel / iXP2 Series & eXP2 Series						HMI S/W
	iXP2-1500	iXP2-1200	iXP2-1000	iXP2-0800	eXP2	XGT InfoU, wXP
 Product						
	iXP2-1500	iXP2-1200	iXP2-1000	iXP2-0800	eXP2-04/05/07/10	
 Spec	15" TFT - XGA(1,024x768) 16,777,218 colors - Ethernet I/F (2ch) - Serial I/F - USB I/F - SD Card I/F	12.1" TFT - XGA(1,024x768) 16,777,216 colors - Ethernet I/F (2ch) - Serial I/F - USB I/F - SD Card I/F	10.4" TFT - XGA(1,024x768) 16,777,216 colors - Ethernet I/F (2ch) - Serial I/F - USB I/F - SD Card I/F	8.4" TFT - SVGA(800x600) 16,777,216 colors - Ethernet I/F (2ch) - Serial I/F - USB I/F - SD Card I/F	10.1,7,5.6 4.3" TFT - eXP2-07/10 24bits(16.7M)colors - Micro SD card I/F - Ethernet I/F - eXP2-05 18bits(262,144)colors - Ethernet I/F - eXP2-04 24bits(16.7M)colors - Ethernet I/F	 - Intelligent & Integrated Development Environment - Graphic Library & Object Support for Screen Creation - Visual Basic Script for Creation of Flexible Application
	New HMI developed with advanced technologies of LS ELECTRIC to optimize the user experience					
XGT Panel						

1. Line-Up

3. GMC (General Motion Controller)

3. GMC



Product



XMC-E08A/EI6A/E32A/E32C



MXP 3.0



L7P/L7C
(Pulse Type Drive)



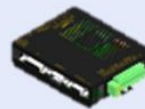
L7NH
(Network type Drive)



Pegasus
(All in One servo)



EtherCAT I/O
Expansion



DC Servo



Linear Motor
(Quadrature Enc.)



Gearbox
(Helical gear)



Rotary Motor
(BiSS-C/Quad. Enc.)



DD Motor
(BiSS-B Enc.)



3rd Party Motor
(EnDat2.2/Tamagawa/Sinusoidal)



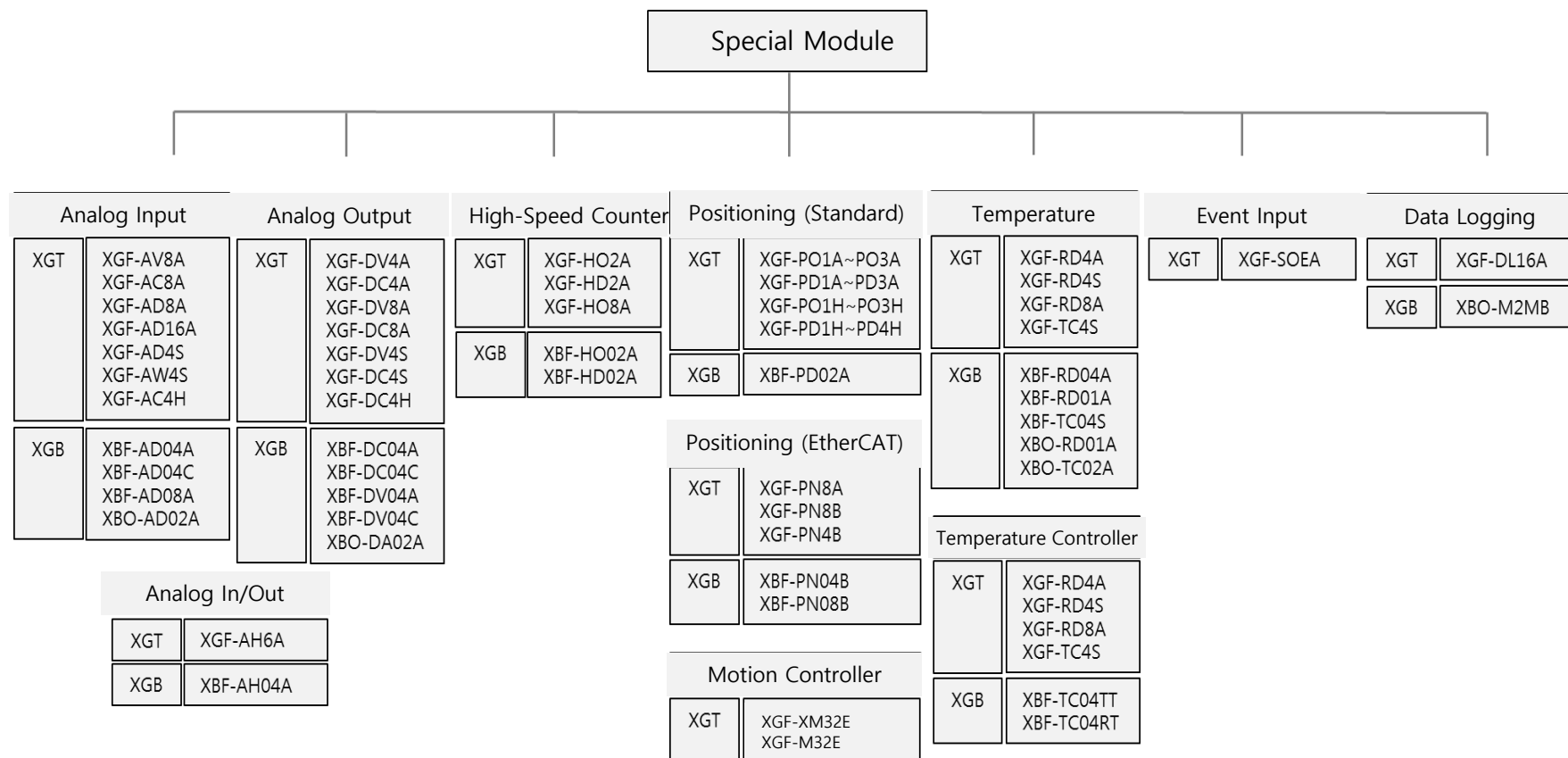
Application

Robotics, Conveyor Belts, Printing, Machine Tool, Packaging, Metal Cutting & Metal Forming Machines, Woodworking/CNC, Textiles, Printing Presses/Printers, Automatic Door Openers

1. Line-Up

4. Special Module

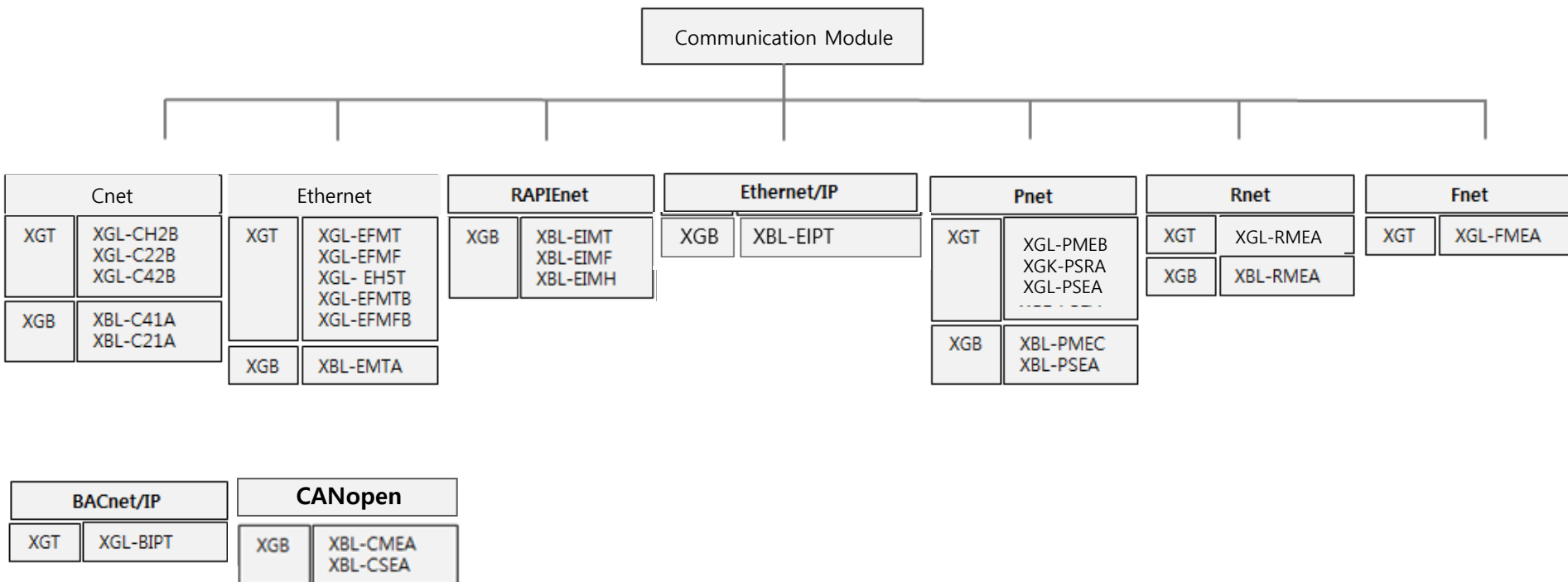
4. Special Module



1. Line-Up

5. Communication Module

5. Communication Module



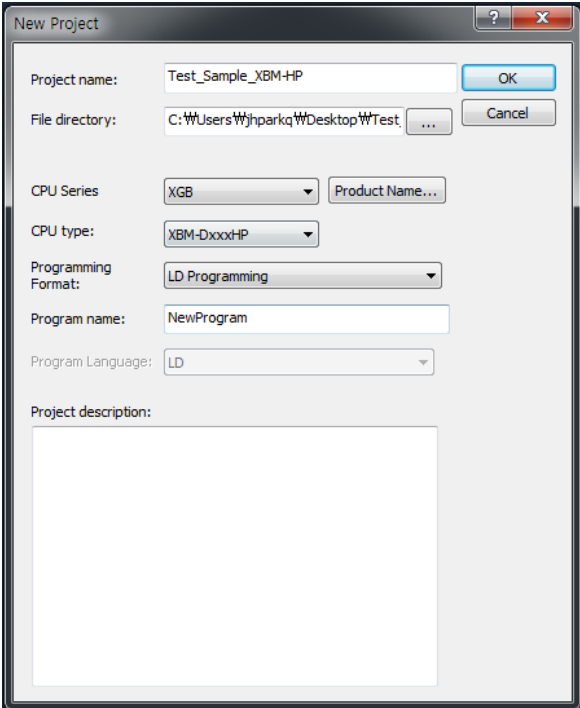
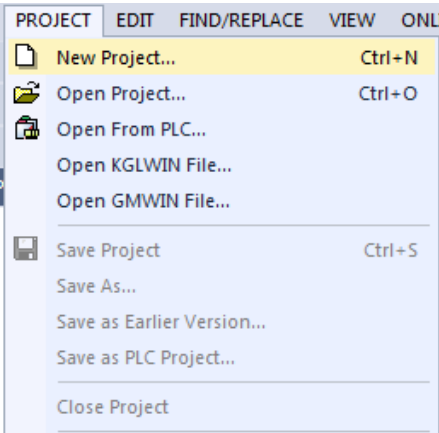
2. PLC

1. XG5000 Basic

1. XG5000 Basic

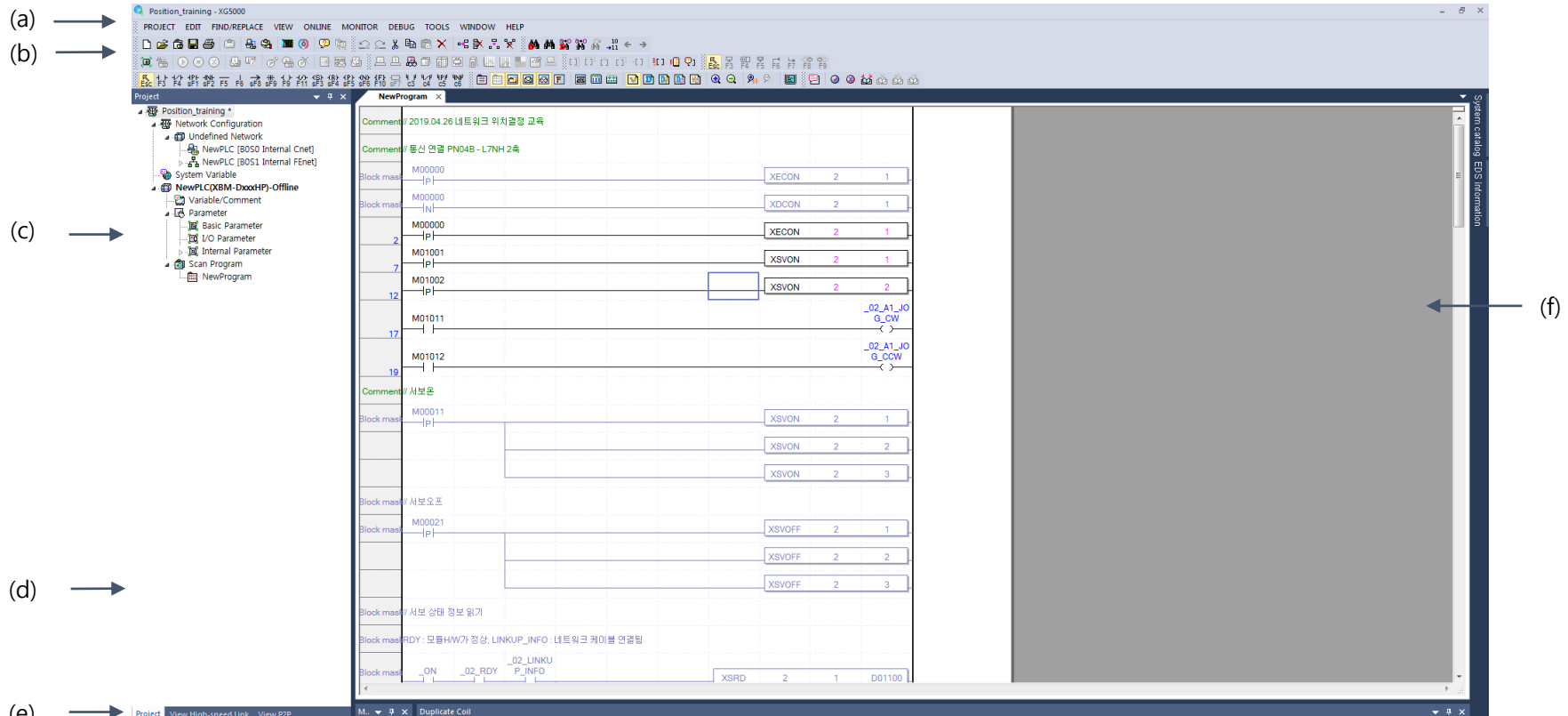
[Project Creation / Open From PLC]

Click PROJECT > New Project



1. XG5000 Basic

[Screen Configuration]

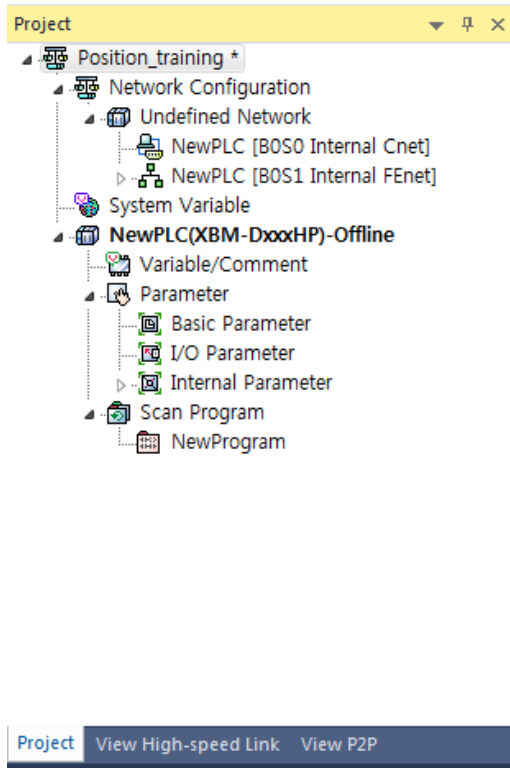


[Description of screen component]

- (a) Menu bar: It is the basic menu bar for the XG5000 software.
- (b) Tool bar: used to execute the instruction which contained in the menu conveniently.
- (c) Project window: used to display the configuration items of the presently opened project.
- (d) Message window: used to display various messages generated while XG5000 is running.
- (e) Status bar: used to display the status of XG5000, the information of connected PLC, etc.
- (f) System catalog window: used to display system catalog and EDS information

1. XG5000 Basic

[Project Window]



[Project]

- Project name is 'Test_XGB-U'. User can manage at least one PLC in one project as adding PLC.

[Network Configuration]

- Set network configuration. Basically, Cnet, FEnet are composed for Internal network. In addition, as adding extended communication module, user can add network options on XG5000.

[System Variable]

- Variables of high-Speed Link, P2P setting and devices are expressed.

[PLC]

- 1 PLC 'NewPLC(XGB-XBCU)' is configured to Variable/Comment, Parameter, Program. Program can be distinguished to Scan Program, Task Program.

[Variable/Comment]

- Edit comment of used variables or each variables on PLC program. In addition, in case of editing variables or comment, user can check it on [Variable/Comment].

[Parameter]

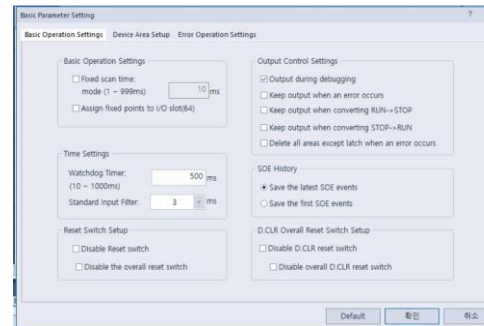
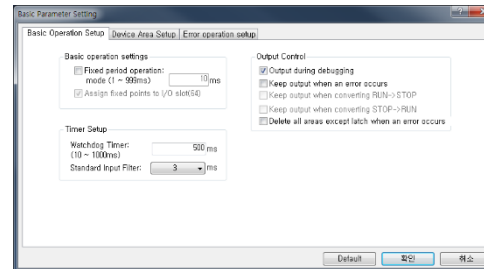
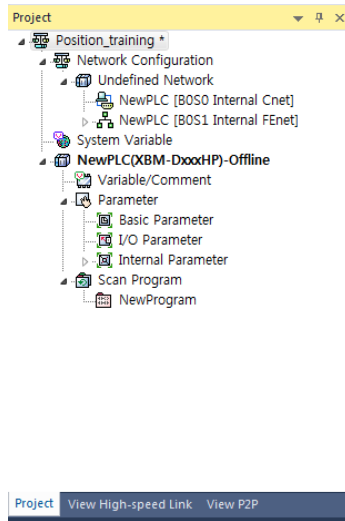
- Parameter configures 3 parameter types: Basic, I/O, Internal. And, user can set the parameter of High Speed Counter, PID and Data Log in Internal Parameter.

[Program]

- Scan Program: Performed program during PLC RUN status.
- Task Program: Performed program during PLC RUN status and satisfied special condition(Fixed-cycle, Internal contact on/OFF). To write Task Program, first of all, user must register Task

1. XG5000 Basic

[Parameter – Basic Parameter]



[Fixed period operation]

- In case that PLC must run at fixed scan time, user can set the fixed scan time. The time must be set over MAX. scan time of present programming on PLC

[Watchdog Timer]

- Watchdog Timer performs monitoring 'Operation congestion', 'Infinite loop' etc. The 1 scan of PLC must be completed within Watchdog Time user set, or PLC changes to Error status

[Standard Input Filter]

- Time to keep on or Off status about digital input module. For example, when digital input signal is OFF→ON and it returns to ON→OFF within Standard Input filter time, PLC considers it as (ON→OFF) Signal has not changed. Finally, this function performs filtering noise signal.

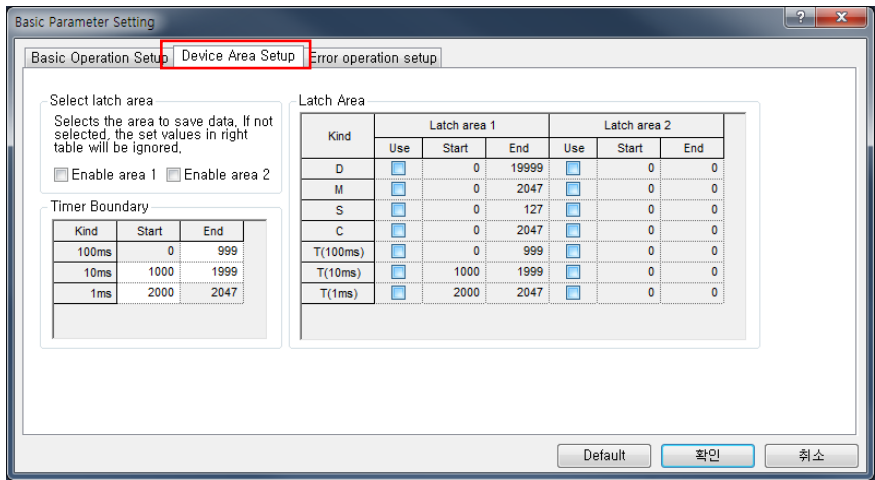
[Output Control]

- Options to whether output signal is turned OUT or not as each options.

1. XG5000 Basic

[Parameter – Basic Parameter]

In Device Area Setup, User can set Latch area. Latch area is to retain data when PLC is reset. It is divided to Latch area 1 and Latch area 2, and each area would retain or clear data as using reset function or data clear function. In addition, K, R area perform Latch function without any setting on Device Area Setup.



	K, R area	Latch 1	Latch 2
CPU STOP/RUN	Retain	Retain	Retain
Power on/OFF	Retain	Retain	Retain
Reset	Retain	Retain	Retain
Overall Reset	Clear	Clear	Retain
Latch 1 Clear	Clear	Clear	Retain
Latch 2 Clear	Clear	Clear	Clear
Program writing	Retain	Retain	Retain

[Select latch area]/[Latch Area]

- When you select 'Enable area 1' or 'Enable area 2', the checkbox is activated for each Latch area.
- When you select 'Use' on each 'Latch area', you can set device range.

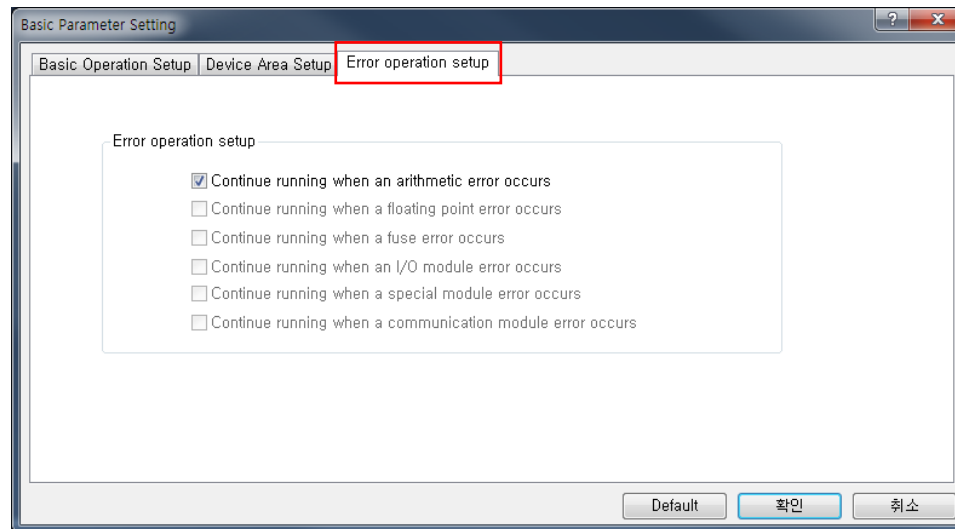
[Timer Boundary]

- In XGB-U, the area of Timer is T0000~T2047. User can set time of timer as the digit of 100ms, 10ms, 1ms.
- The range of Timer Boundary depends on the type of PLC.

1. XG5000 Basic

[Parameter – Basic Parameter]

When PLC faces on Error status, you can select whether PLC operates normally or PLC stops. In XGB-U, it is possible for user only to set 'Continue running when an arithmetic error occurs'



1. XG5000 Basic

[Parameter – IO Parameter]

In I/O Parameter, you can set parameter for Digital I/O module, Special module. But, parameter of positioning module(XG-PM), temperature control module(XG-TCon) must be set on dedicated software tool. You can set input filter time for digital input module, emergency output control for each channels during PLC→STOP mode for digital output module on I/O parameter.

① User registration

You can register module as following each slots on PLC.

I/O Parameter > Click slot (find module) > select module.

Project

Position_training

Network Configuration

Undefined Network

NewPLC [BOS0 Internal Cnet]

NewPLC [BOS1 Internal FNet]

System Variable

NewPLC(XBM-DxxxHP)-Offline

Variable/Comment

Parameter

Basic Parameter

I/O Parameter

Internal Parameter

Scan Program

NewProgram

→

Base 00 : Default

Slot 00 : XBM-DN32HP

Slot 01 : XBF-PO06E (

Slot 02 : XBF-PN04B (

Slot 03 : Default

Slot 04 : Default

Slot 05 : Default

Slot 06 : Default

Slot 07 : Default

Slot 08 : Default

All Base

Set Base

Apply

Current Consumption

Slot	Module	Comment	Input Filter	Emergency Out	Address Assignment
0(main)	XBM-DN32HP (DC 24		3 Standard[ms]	Default	P0000 ~ P003F
1	XBF-PO06E (Open Col		-	-	P0040 ~ P007F
2	XBF-PN04B (Position		-	-	P0080 ~ P011F
3	Digital Module List				
4	Special Module List				
5	Analog Input Module				
6	Analog Output Module				
7	Temp. Measuring Module				
8	Analog IO Module				
	Positioning Module				
	XBF-PD02A (Position Control, Pulse				
	XBF-PN08B (Position Control, Ether				
	XBF-PN04B (Position Control, Ether				
	High Speed Counter Module				
	Process Control Module				
	Loadcell Input Module				
	Communication Module List				

Project

View High-speed Link

View P2P

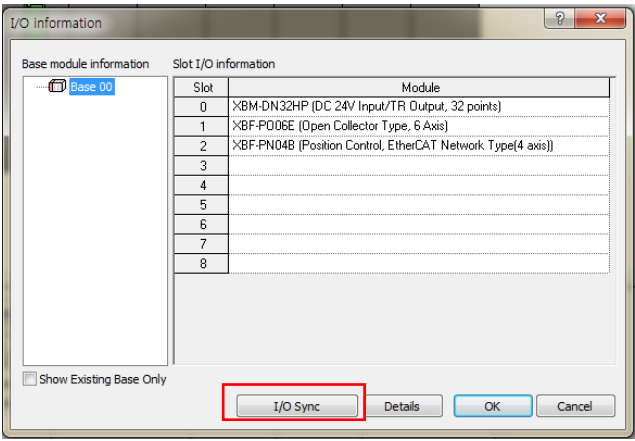
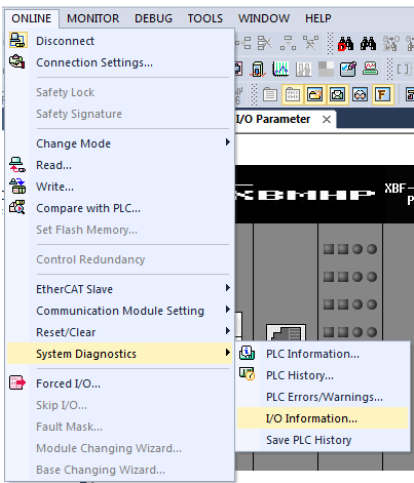
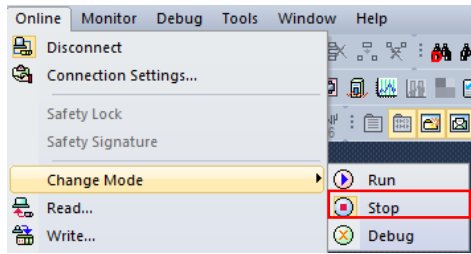
1. XG5000 Basic

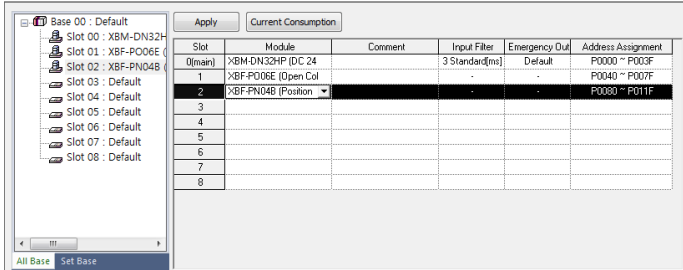
[Parameter – IO Parameter]

② Online registration

When PLC system is composed, you can read module installed PLC, and then register module to I/O parameter. It is possible only in case that PLC is connected to XG5000.

PLC STOP Mode > online > Diagnosis> I/O information > I/O Sync





Slot	Module	Comment	Input Filter	Emergency Out	Address Assignment
0 (main)	XBM-DN32HP (DC 24		3 Standard[ms]	Default	P0000 ~ P003F
1	XBF-PO06E (Open Col				P0040 ~ P007F
2	XBF-FN04B (Position				P0080 ~ P011F
3					
4					
5					
6					
7					
8					

When I/O sync is executed, module read from PLC is registered to I/O Parameter. If I/O parameter has been set in advance, all I/O parameter will be initialized by I/O sync.

1. XG5000 Basic

[Register Module Variable Comments]

To register Module Variable Comment as adding Module on I/O Parameter, Click EDIT > Register Module Variable Comments

EDIT

FIND/REPLACE

VIEW

ONLINE

MONITOR

Undo

Ctrl+Z

Redo

Ctrl+Y

Cut

Ctrl+X

Copy

Ctrl+C

Paste

Ctrl+V

Delete

Delete

Select All

Ctrl+A

Insert Line

Ctrl+L

Delete Line

Ctrl+D

Export Variables to File...

Import Variables from File...

Register Module Variable Comments

Network variable automatic registration

Add EXTERNAL Variable

Move Item Up

Move Item Down

Delete All unused variables/comments...

Reallocate All Auto-allocation Variables...

Project

Position_training *

Network Configuration

Undefined Network

NewPLC [BOS0 Internal Cnet]

NewPLC [BOS1 Internal FNet]

System Variable

NewPLC [XBM-Down] - Run

Variable/Comment

Parameter

Basic Parameter

I/O Parameter

Internal Parameter

Scan Program

NewProgram

NewProgram

Variable/Comment

View Variable

View Device

View Flag

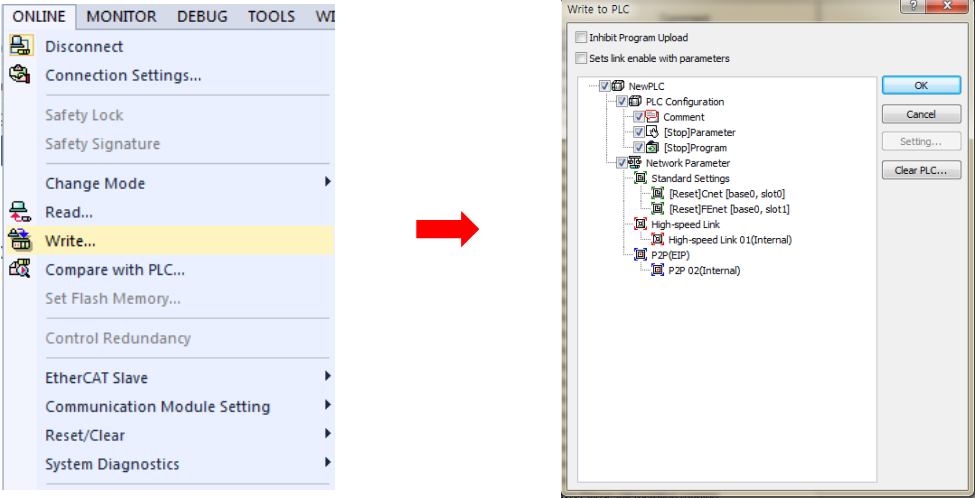
	Variable	Type	Device	Used	HMI	Comment
1	_0001_LINKUP	BIT	L17010	☐	☐	Internal FNet 01 Port Link/up down information
2	_02_A1_RDY	BIT	U02.00.0	☐	☐	Positioning Module: 1-Axis Ready
3	_02_A2_RDY	BIT	U02.00.1	☐	☐	Positioning Module: 2-Axis Ready
4	_02_A3_RDY	BIT	U02.00.2	☐	☐	Positioning Module: 3-Axis Ready
5	_02_A4_RDY	BIT	U02.00.3	☐	☐	Positioning Module: 4-Axis Ready
6	_02_LINKUP_IN	BIT	U02.00.E	☒	☐	Positioning Module: Link up/down information
7	_02_RDY	BIT	U02.00.F	☒	☐	Positioning Module: Ready Flag
8	_02_A1_JOG_C	BIT	U02.01.0	☒	☐	Positioning Module: 1-Axis JOG CW
9	_02_A1_JOG_C	BIT	U02.01.1	☒	☐	Positioning Module: 1-Axis JOG CCW
10	_02_A1_JOG_S	BIT	U02.01.2	☐	☐	Positioning Module: 1-Axis JOG SPEED
11	_02_A2_JOG_C	BIT	U02.01.4	☐	☐	Positioning Module: 2-Axis JOG CW
12	_02_A2_JOG_C	BIT	U02.01.5	☐	☐	Positioning Module: 2-Axis JOG CCW
13	_02_A2_JOG_S	BIT	U02.01.6	☐	☐	Positioning Module: 2-Axis JOG SPEED
14	_02_A3_JOG_C	BIT	U02.01.8	☐	☐	Positioning Module: 3-Axis JOG CW
15	_02_A3_JOG_C	BIT	U02.01.9	☐	☐	Positioning Module: 3-Axis JOG CCW
16	_02_A3_JOG_S	BIT	U02.01.A	☐	☐	Positioning Module: 3-Axis JOG SPEED
17	_02_A4_JOG_C	BIT	U02.01.C	☐	☐	Positioning Module: 4-Axis JOG CW
18	_02_A4_JOG_C	BIT	U02.01.D	☐	☐	Positioning Module: 4-Axis JOG CCW
19	_02_A4_JOG_S	BIT	U02.01.E	☐	☐	Positioning Module: 4-Axis JOG SPEED



1. XG5000 Basic

[Writing]

When all parameters are set, write all settings as following steps. Online > Write



1. XG5000 Basic

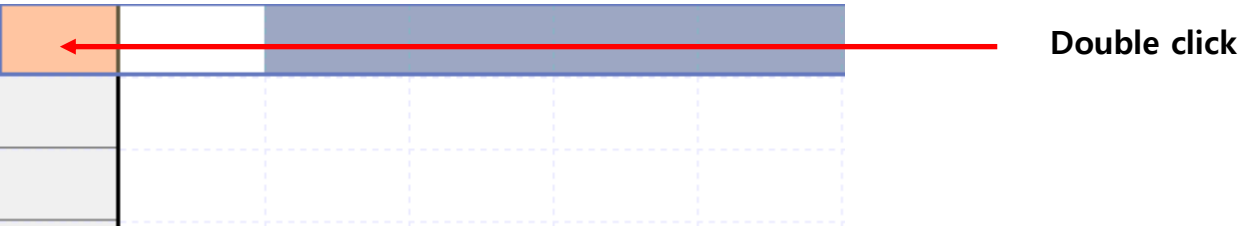
[Program Editing]

(1) Program example

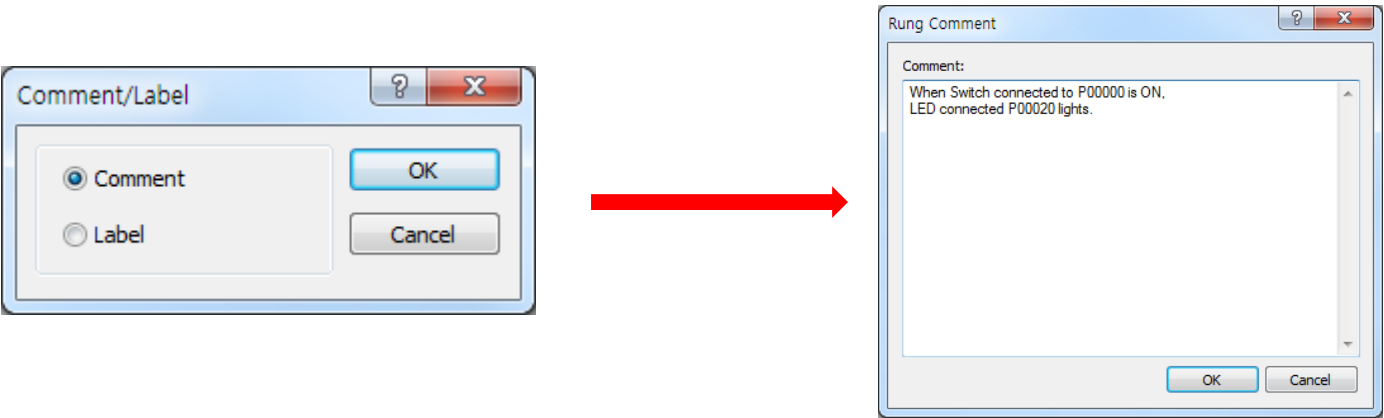
When Switch connected to P00000 is on, LED connected P00020 lights.

① Lung Comment edit

You can edit simple comment partly for program.



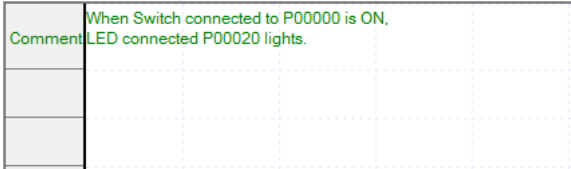
② Select [Comment] on the window, and click [OK]



1. XG5000 Basic

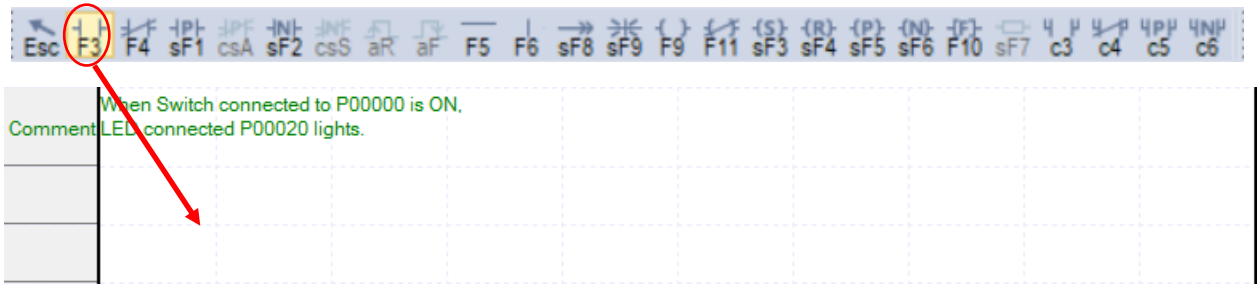
[Program Editing]

③ Lung comment has been written on the program window

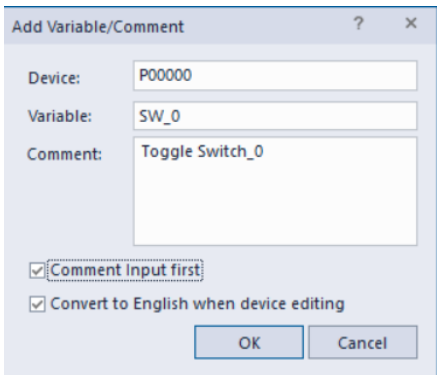
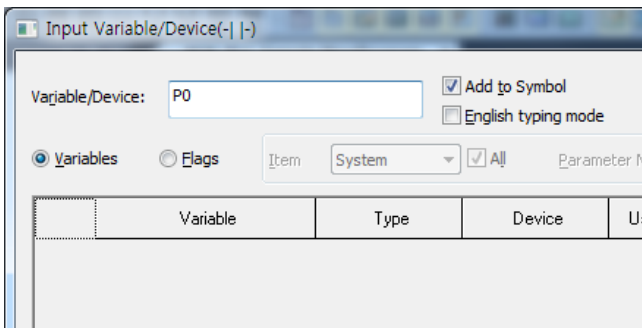


④ Contact edit

Click [Normally Open] in the Tool bar, move the mouse cursor to program window and click the position you'd like to put



⑤ When you click on the program window, 'Input Variable/Device' window is opened. Write 'P0' on Variable/Device and click [OK]. If [Add to Symbol] is checked, 'Add Variable/Comment' window is opened.



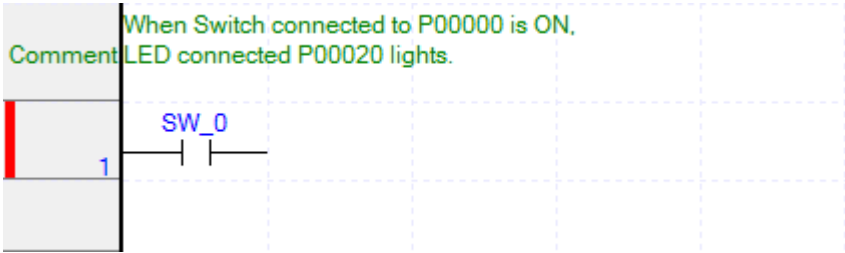
※ Reference

After registering Variable, when you want to read data, device and variable are used commonly. one variable is allocated to one device. And, for variable, only character, number, special character and () are used, and number can't be first Word for variable.

1. XG5000 Basic

[Program Editing]

⑥ Contact is on program window



- The red line on the left side of lung means that the program has errors.
- Number '1' on the left side of lung is the first step number of lung.

⑦ You can choose view option on XG5000. According to view option, the expression of variable will be different.

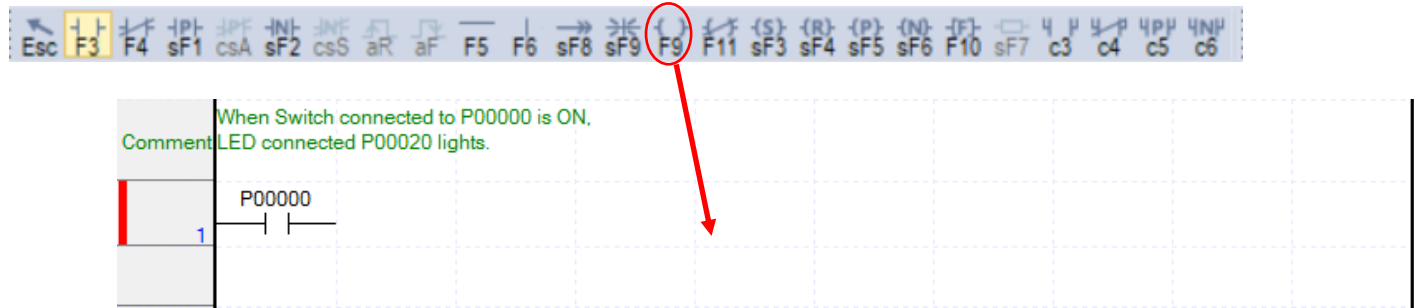
Menu bar > [View]

Tool bar

Variables	Devices	Devices/ Variables	Devices/ Comment	Variable/ Comment

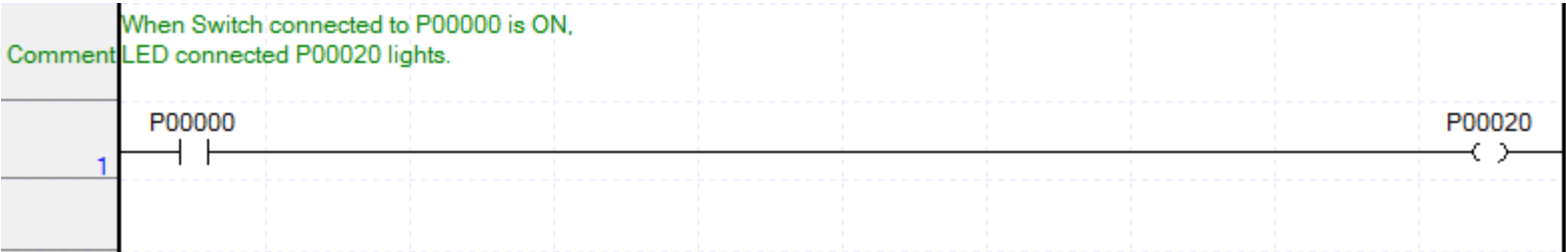
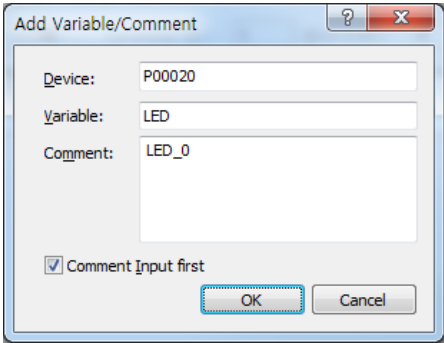
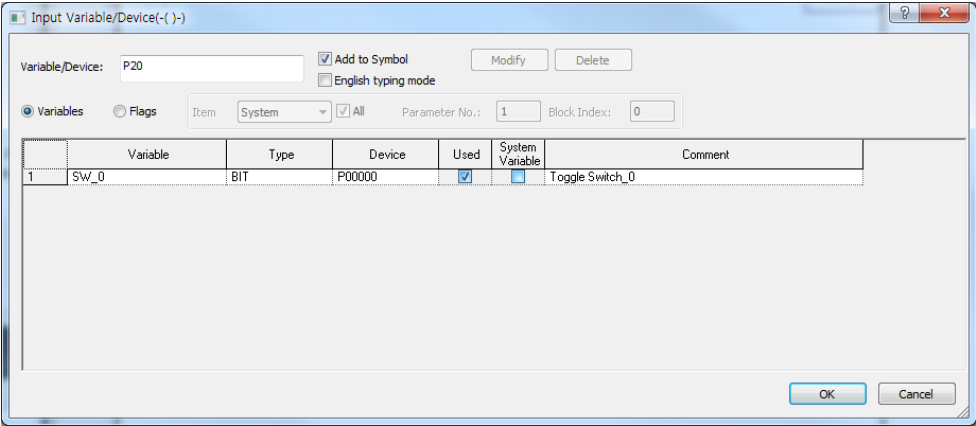
⑧ Coil edit

Select Coil Tool bar, and then click any Cell next to P00000 on Project window.



1. XG5000 Basic

⑨ Write 'P20' on Variable/Device and click [OK]. If [Add to Symbol] is checked, 'Add Variable/Comment' window is opened.

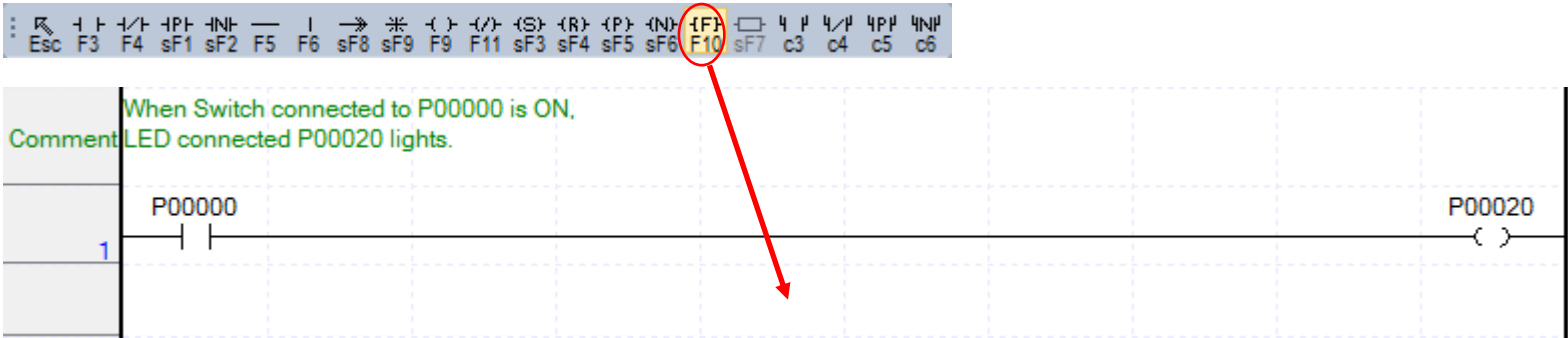


1. XG5000 Basic

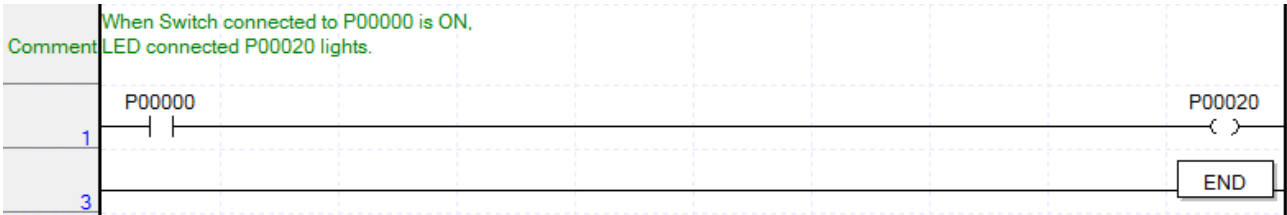
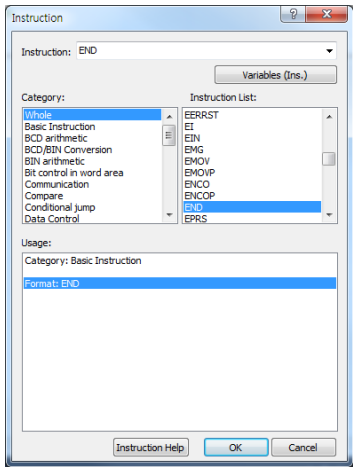
[Program Editing]

⑨ Edit 'END' instruction

After completing program edit, put END instruction at the last lung of Program



In instruction window, write 'END' and click [OK]



※ Reference
'END' instruction must be edited without condition, it doesn't have operated data

1. XG5000 Basic

[Variables/Comment]

You can edit variables and comment on when devices are used first on the program like previous pages, but variables and comment can be registered on Variable/Comment at the first step.

① Variable/Comment edit on XG5000

In Variables/Comment window, you can edit Variable/Comment.

Project

Test_XGB-U *

Network Configuration

- Unspecified Network
 - NewPLC [BOS0 Internal Cnet]
 - NewPLC [BOS1 Internal FEnet]

System Variable

- NewPLC [XGB-YBCU] Offline
 - Variable/Comment**
 - Parameter
 - Basic Parameter
 - I/O Parameter
 - Internal Parameter

Scan Program

- NewProgram

Variable/Comment x

View Variable

View Device

View Flag

	Variable	Type	Device	Used	Comment
1	SW_0	BIT	P00000	☒	Toggle Switch_0
2	LED	BIT	P00020	☐	LED_0

Project

View High-speed Link

View P2P

② Select [View Device] and write first address of devices which must be expressed. Finally, you can edit variable or comment as clicking cell of variable or comment

Variable/Comment x

View Variable

View Device

View Flag

Device: P0000 Type: BIT

	Variable	Comment
P00000	SW_0	Toggle Switch_0
P00001		
P00002		
P00003		
P00004		
P00005		

1. XG5000 Basic

[Variables/Comment]

- ③ Copy device comment
when you would like to edit similar contents of comment for continuous address, selected comment would be copied with mouse drag.

Variable/Comment x

View Variable

View Device

View Flag

Device: P0000 Type: BIT

	Variable	Comment
P00000	SW_0	Toggle Switch_0
P00001		
P00002		
P00003		
P00004		
P00005		
P00006		
P00007		

Variable/Comment x

View Variable

View Device

View Flag

Device: P0000 Type: BIT

	Variable	Comment
P00000	SW_0	Toggle Switch_0
P00001		Toggle Switch_0
P00002		Toggle Switch_0
P00003		Toggle Switch_0
P00004		Toggle Switch_0
P00005		
P00006		
P00007		

- ④ Number auto increase for device comment
Select device comment including number and drag mouse on pressing 'Ctrl' key

Variable/Comment x

View Variable

View Device

View Flag

Device: P0000 Type: BIT

	Variable	Comment
P00000	SW_0	Toggle Switch_0
P00001		
P00002		
P00003		
P00004		
P00005		
P00006		
P00007		

Variable/Comment x

View Variable

View Device

View Flag

Device: P0000 Type: BIT

	Variable	Comment
P00000	SW_0	Toggle Switch_0
P00001		Toggle Switch_1
P00002		Toggle Switch_2
P00003		Toggle Switch_3
P00004		Toggle Switch_4
P00005		
P00006		
P00007		

1. XG5000 Basic

[Variables/Comment]

- ⑤ Number auto increase for Variable
Select variable including number and drag mouse on pressing 'Ctrl' key

Variable/Comment ×

View Variable

View Device

View Flag

Device: P0000 Type: BIT

	Variable	Comment
P00000	SW_0	Toggle Switch_0
P00001		Toggle Switch_1
P00002		Toggle Switch_2
P00003		Toggle Switch_3
P00004		Toggle Switch_4
P00005		
P00006		
P00007		

→

Variable/Comment ×

View Variable

View Device

View Flag

Device: P0000 Type: BIT

	Variable	Comment
P00000	SW_0	Toggle Switch_0
P00001	SW_1	Toggle Switch_1
P00002	SW_2	Toggle Switch_2
P00003	SW_3	Toggle Switch_3
P00004	SW_4	Toggle Switch_4
P00005		
P00006		
P00007		

- ※ Reference
- If variable or comment has two or more number for each, the last number will be increased
 - With Microsoft Excel, you can edit variable/comment with copy and paste (Excel <-> XG5000)
 - You can save variable/comment to CSV file on 'Menu' bar in XG5000

1. XG5000 Basic

[Simulator (XG-SIM)]

XG-SIM is a Windows-based virtual PLC for XGT PLC series. If using XG-SIM, an operator may execute a created program without PLC and debug a PLC program by using the functions of input setup and module simulation.

(1) Features of XG-SIM

① Program simulation

An operator can simulate a program created by LD/SFC/ST languages. In addition, it supports modification function during run, with which an operator may apply modification during run of a program that is operated in XG-SIM, and it supports debugging function that trace a user-created program by unit of step.

② PLC online function

Besides program monitoring function provided by XG5000, an operator may use online diagnosis functions such as system monitor, device monitor, trend monitor, data trace and user event.

③ Module simulation

It also supports simple simulation function for modules mountable on XGK rack type PLC, such as digital I/O module, A/D conversion module, D/A conversion module, high speed counter, temperature control module and positioning module. If using the module simulation function, a program may be simulated by using input values from module.

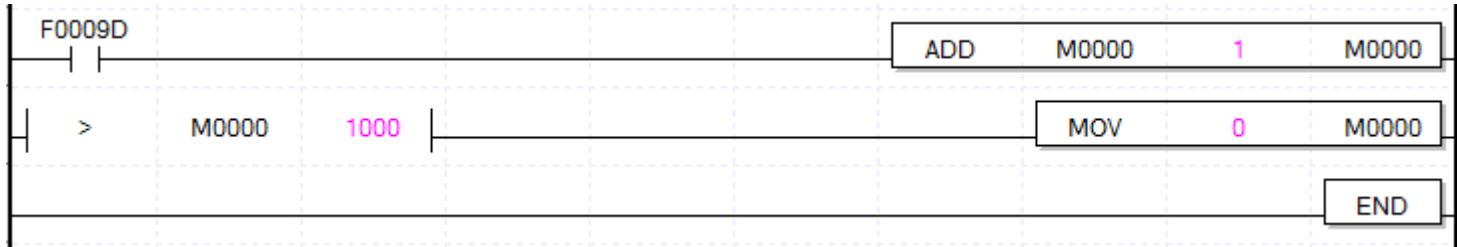
④ I/O input condition setup

The value of device may be set with value of specific device or channel value in a module set as input conditions. Using I/O input condition setup function may realize the program simulation as created with no PLC program separately created to test a created PLC program.

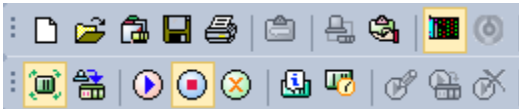
1. XG5000 Basic

[Simulator (XG-SIM)]

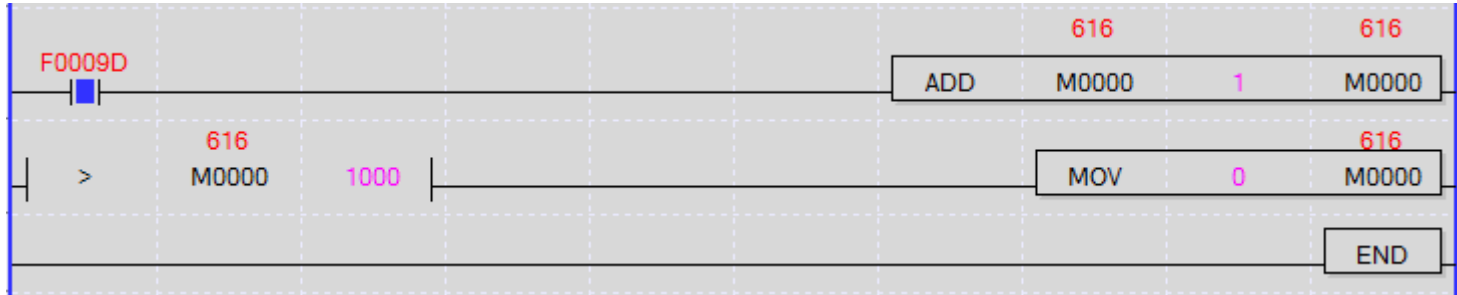
① By executing XG5000, create a program to run on XG-SIM.



② Select [Tools] – [Start Simulator] in XG5000 Menu. If XG-SIM is started, a created program is automatically downloaded onto XG-SIM. After XG-SIM is executed, it goes to online, Access and Stop.



③ By selecting [online] – [Change Mode] – [Run] in XG5000 Menu, execute the downloaded program.



※ Reference
- For available online menus of xg5000 when XG-SIM is executed, refer to the XG5000 manual.

1. XG5000 Basic

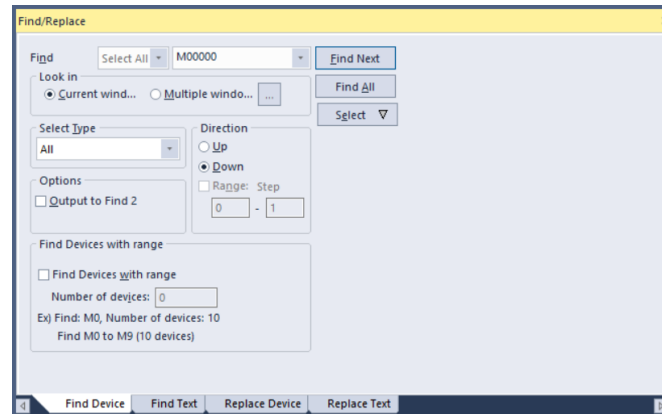
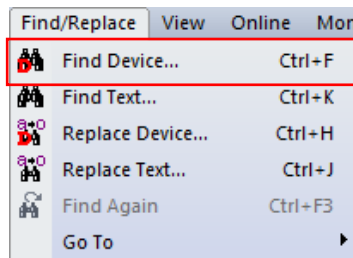
[Find/Replace]

This is used to find or replace the device and the Text in LD, IL and Variable/Comment of XG5000..

(1) Find device

This function is used to find the device previously made in LD Editor, IL Editor, or Variable/Comment Editor.

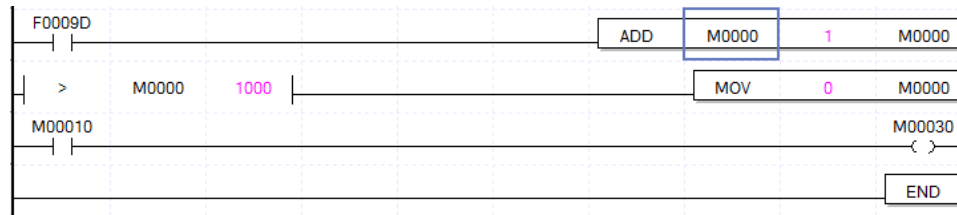
① Select the menu [Find/Replace] > [Find Device]



[Find Next Device Text]

The function is used in common for the LD editor, IL editor, SFC editor, ST editor, variable/Comment editor. The below description is based on LD.

① Select the cell that is the basis of the direction



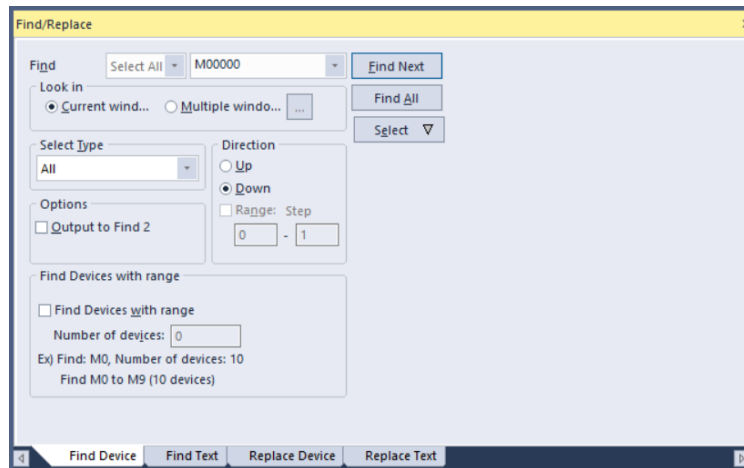
※ Reference

- The results will be displayed in the search window only when 'Find All' is applied.
- 'Find All' searches all devices in the relevant documents so selecting direction is not significant.

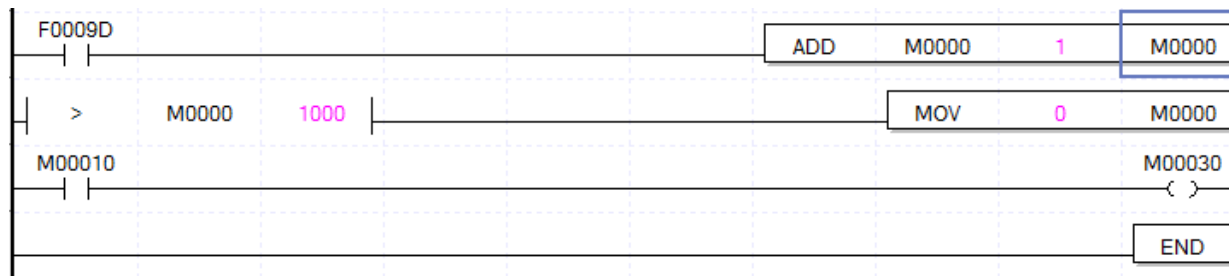
1. XG5000 Basic

[Find Next Device Text]

- ② Select the menu [Find/Replace] > [Find Device] and then, enter contents and conditions to be found.



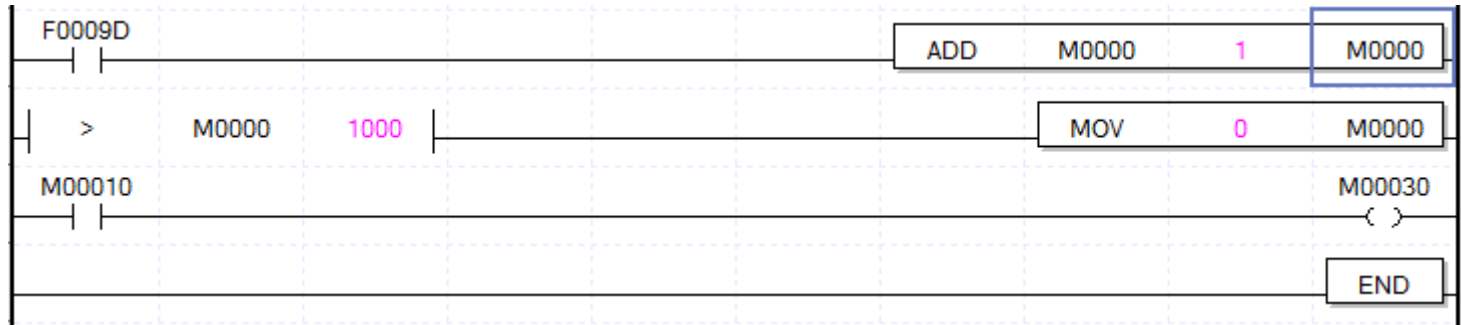
- ③ Press 'Find Next' button. If there are devices matched with what you set in the dialog box, the cell will move to the relevant device.



1. XG5000 Basic

[Find All Device]

The function is used in common for the LD editor, IL editor, SFC editor, ST editor, variable/Comment editor. The below description is based on LD.



- ① Select the menu [Find/Replace]-[Find Device].
- ② Set up the device to be found, direction, options, etc.
- ③ If you press 'Find All' button, the results will be output in the search window.
- ④ If you select the line to be found in the search window and press 'Enter' key, the cell will move to the line of the relevant LD program

Find/Replace

Find

Select All

M0000

Find Next

Look in

Current wind...

Multiple windo...

Select Type

All

Direction

Up

Down

Options

Output to Find 2

Find Devices with range

Find Devices with range

Number of devices: 0

Ex Find: M0, Number of devices: 10

Find M0 to M9 (10 devices)

Find Device

Find Text

Replace Device

Replace Text

Find 1

In NewPLC, finding address "M0000"...

Find all: NewProgram...

NewProgram(Step: 3): -[F]-

ADD M0000 1 M0000

NewProgram(Step: 7): -[F]-

> M0000 1000

NewProgram(Step: 9): -[F]-

MOV 0 M0000

4 address(es) found.

Check Program

Find 1

Find 2

Cross Reference

Used Device

Communication

Duplicate Coil

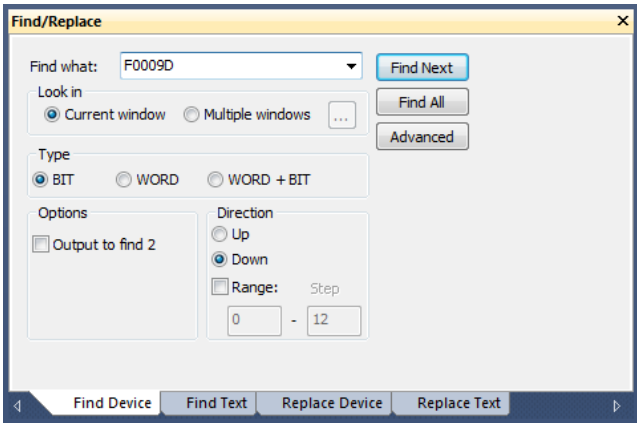
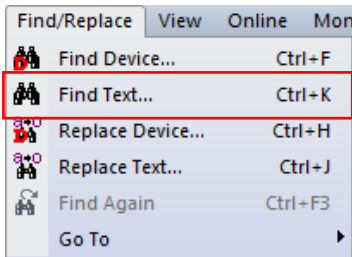
Result

1. XG5000 Basic

[Find Text]

It is the function to find the text of the rung comment, variable, variable comment, applied command, label etc. except the devices. This function is used to find the device previously made in LD Editor, IL Editor, or Variable/Comment Editor.

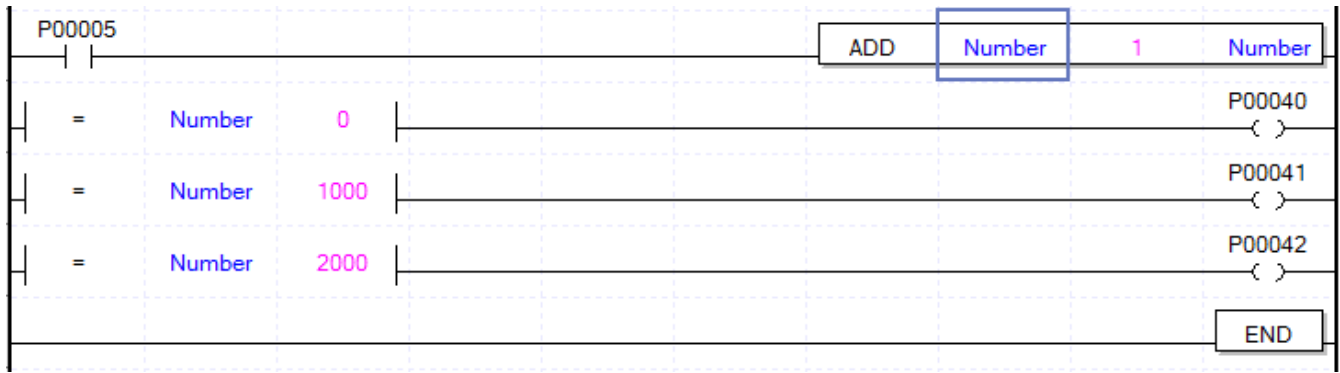
- ① Select the menu [Find/Replace] > [Find Text]



[Find Next Text]

The function is used in common for the LD editor, SFC editor, ST editor, variable/Comment editor etc. The below description is based on LD.

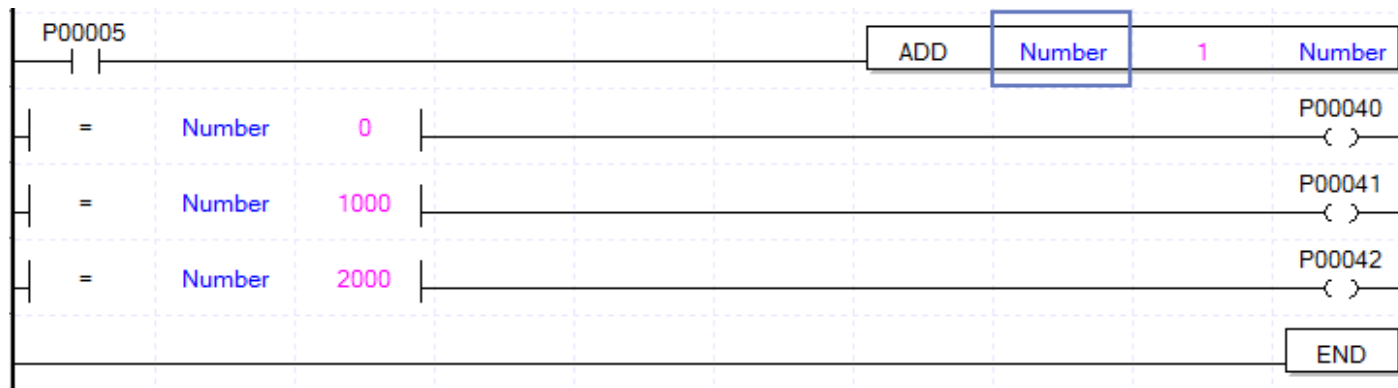
- ① Select the cell that is the basis of the direction



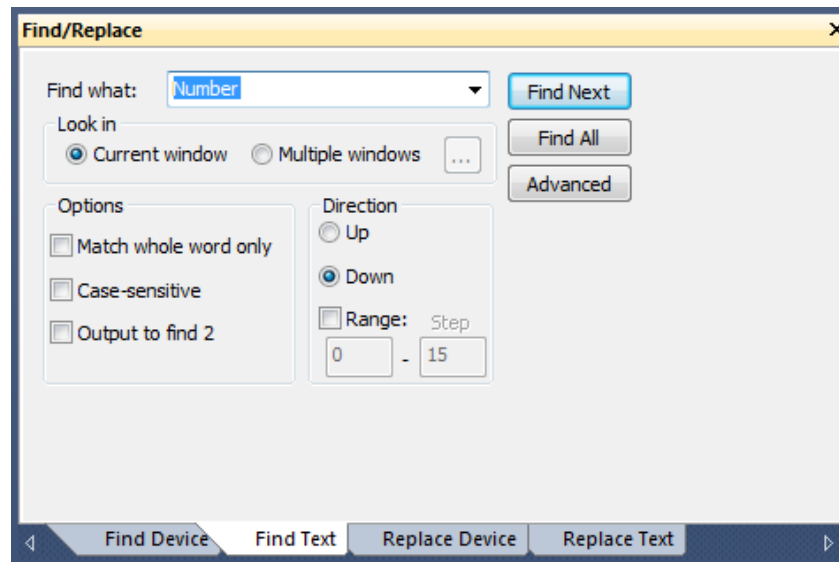
1. XG5000 Basic

[Find Next Text]

- ① Select the cell that is the basis of the direction



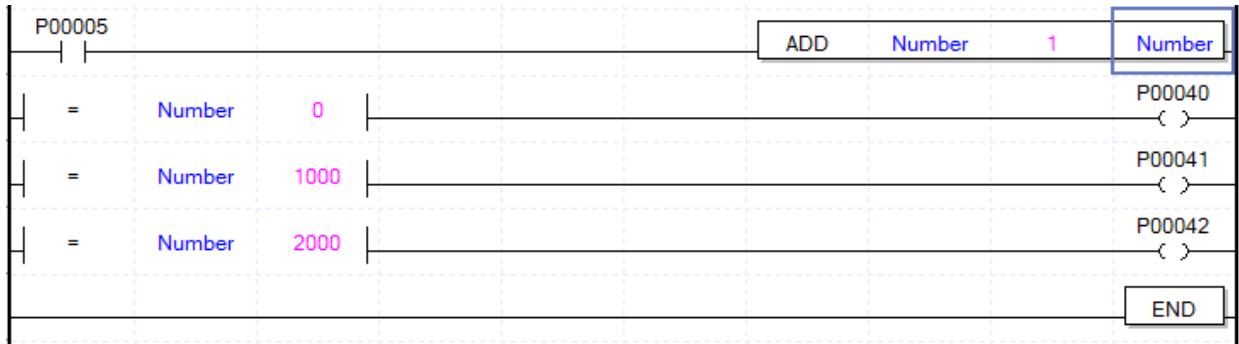
- ② Select the menu [Find/Replace] > [Find Device] and then, set up the text to be found, options, direction.



1. XG5000 Basic

[Find Next Text]

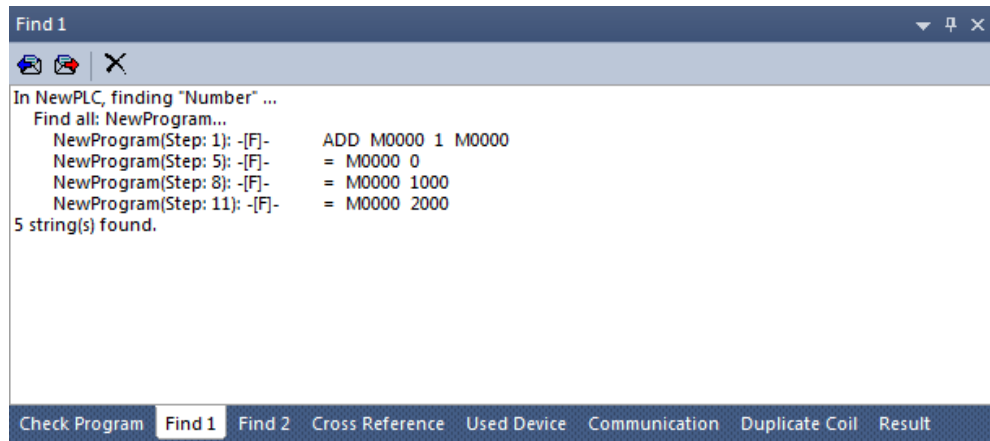
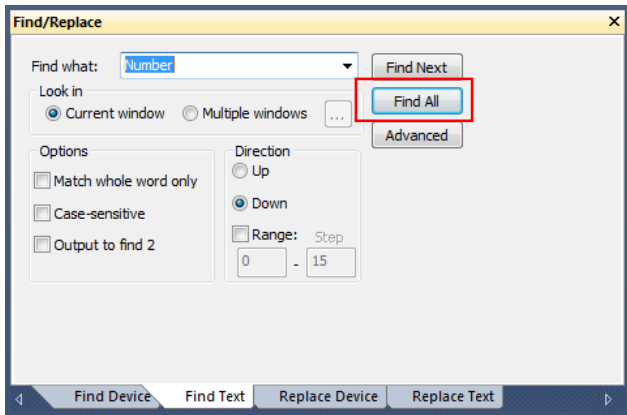
③ Press 'Find Next' button. If there are texts matched with what you set in the dialog box, the cell will move to the relevant text.



[Find All Text]

The function is used in common for the LD editor, IL editor, SFC editor, ST editor, variable/Comment editor. The below description is based on LD.

- ① Select the menu [Find/Replace] > [Find Text]
- ② Find the text to be found, options, direction, etc.



1. XG5000 Basic

[Find All Text]

③ If you select the line to be found in the search window and press 'Left Mouse Button Double Click', the cell will move to the line of the relevant LD program.

Find 1

In NewPLC, finding "Number" ...
Find all: NewProgram...
NewProgram(Step: 1): -[F]- ADD M0000 1 M0000
NewProgram(Step: 5): -[F]- = M0000 0
NewProgram(Step: 8): -[F]- = M0000 1000
NewProgram(Step: 11): -[F]- = M0000 2000
5 string(s) found.

Check Program

Find 1

Find 2

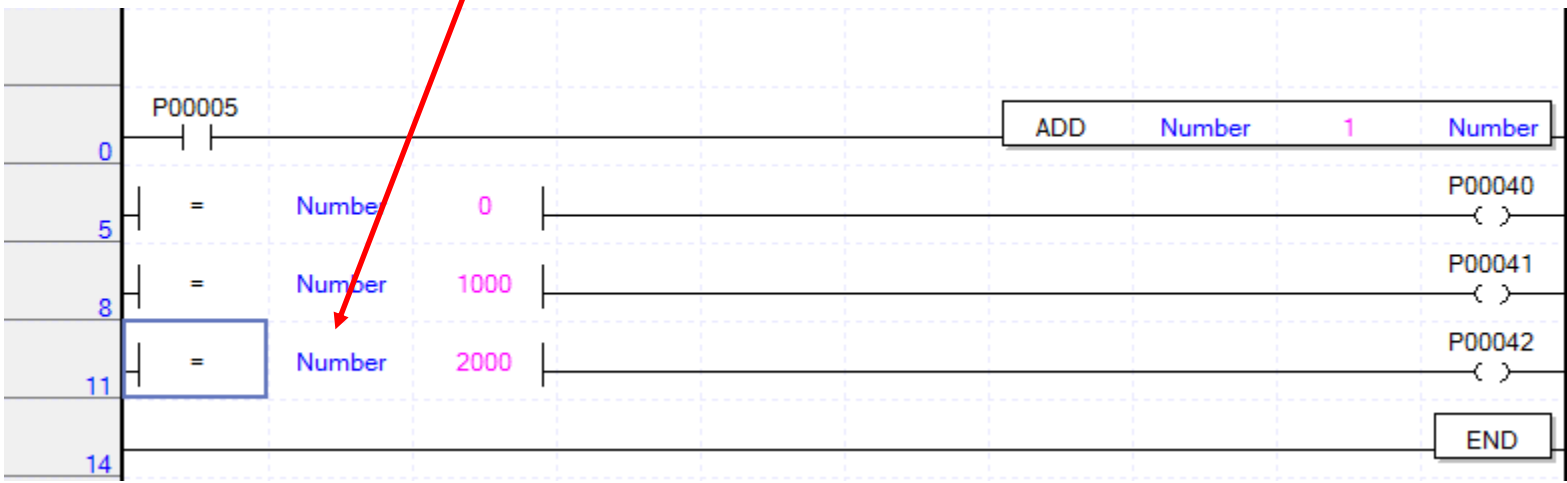
Cross Reference

Used Device

Communication

Duplicate Coil

Result

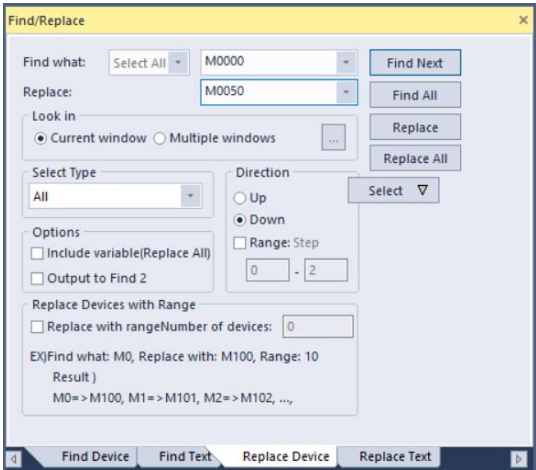
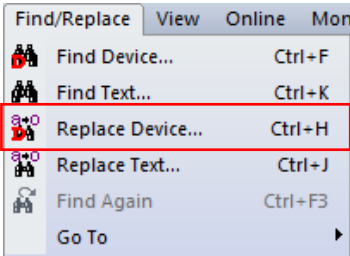


1. XG5000 Basic

[Replace Device]

It is the function to the previously made devices in the LD editor, IL editor, SFC editor, ST editor, variable/comment etc.

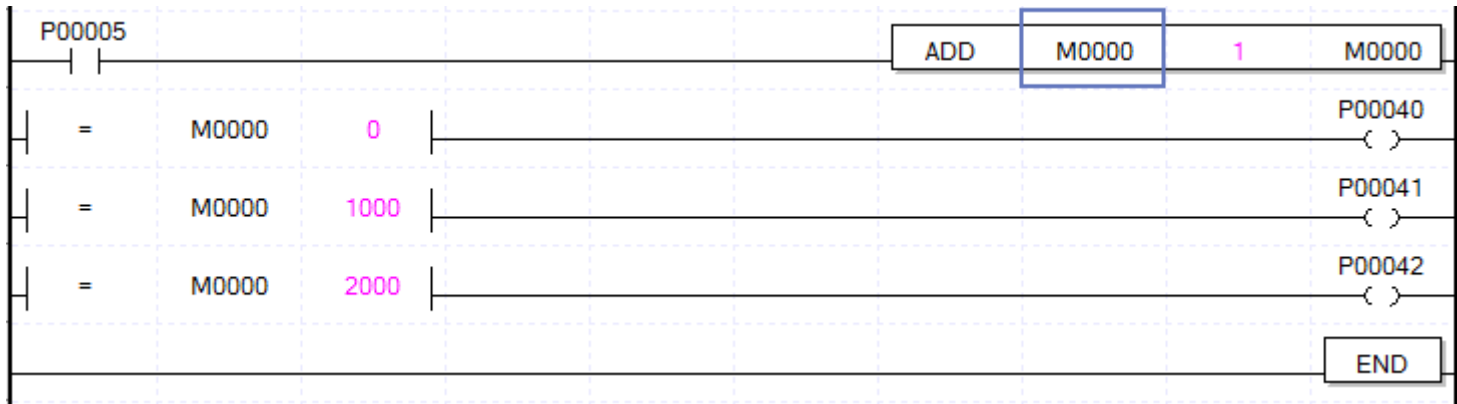
- ① Select the menu [Find/Replace] > [Replace Device]



[Replace Device]

It is the function to find the previously established device from LD editor, variable/comment editor, etc. The below description is based on LD

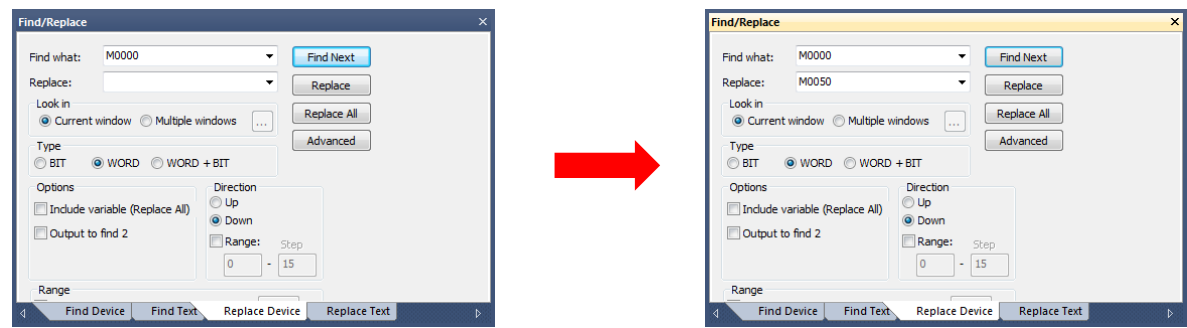
- ① Select the cell that is the basis of the direction



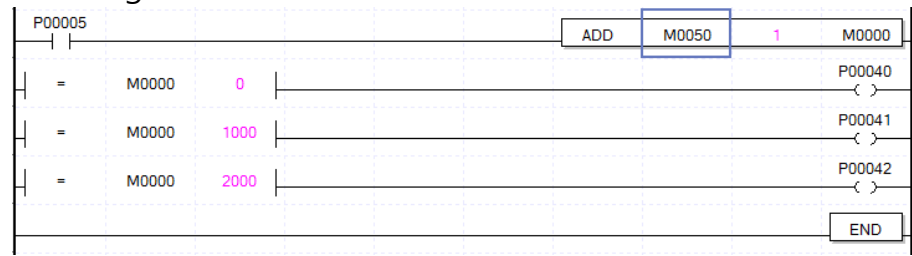
1. XG5000 Basic

[Replace Device]

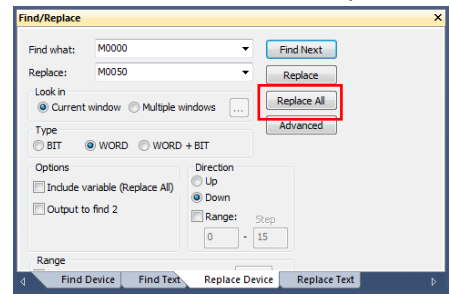
- ② Select the menu [Find/Replace] > [Replace device]
- ③ Set up the contents to be found, to be replaced with selection of type and direction, etc.



- ④ Press the 'Replace' button. If the device of the currently selected cell is the same as the find what, it replaces the current cell into what you set in the dialog box and moves to the cell that is matched with the find what.



- ⑤ As clicking [Replace All] button, all device 'M0000' will be replaced to 'M0050'

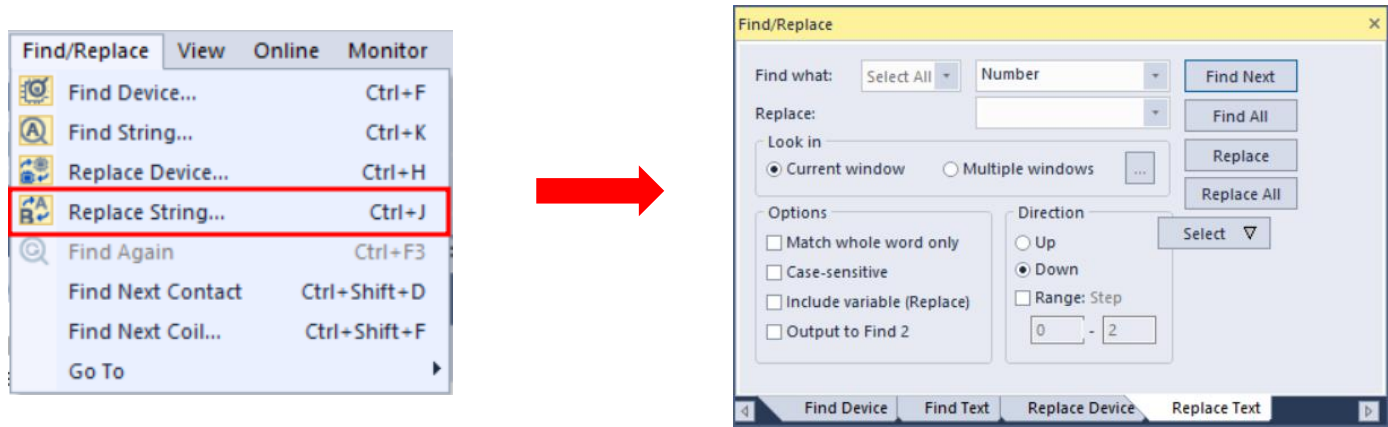


1. XG5000 Basic

[Replace String]

It is the function to find comments, variables except devices

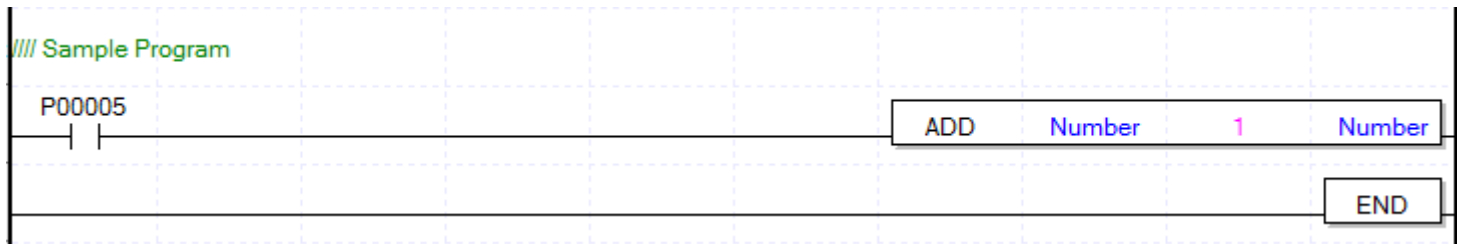
- ① Select the menu [Find/Replace] > [Replace Text]



[Replace String]

It is the function to find the previously established device from LD editor, IL editor, ST editor, SFC editor, variable/comment editor etc. The below description is based on LD

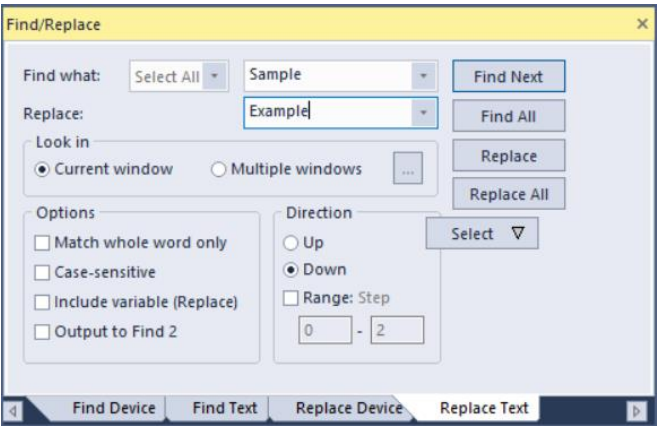
- ① Select the cell that is the basis of the direction



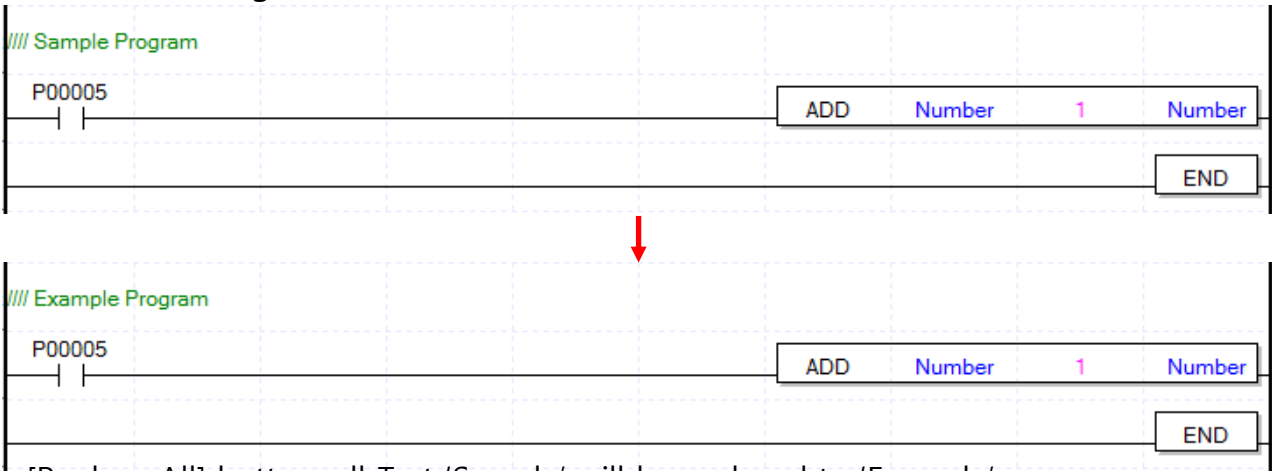
1. XG5000 Basic

[Replace String]

- ② Select the menu [Find/Replace] > [Replace Text]
- ③ Set up the text to be found, to be replaced with options and direction



- ④ Press 'Replace' button. If the text of the currently selected cell is the same as the find what, it replaces the current cell into what you set in the dialog box and moves to the cell that is matched with the find what.



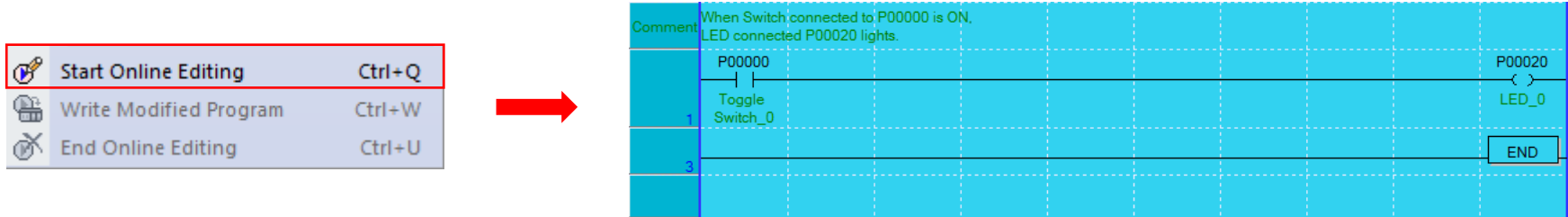
- ⑤ As clicking [Replace All] button, all Text 'Sample' will be replaced to 'Example'

1. XG5000 Basic

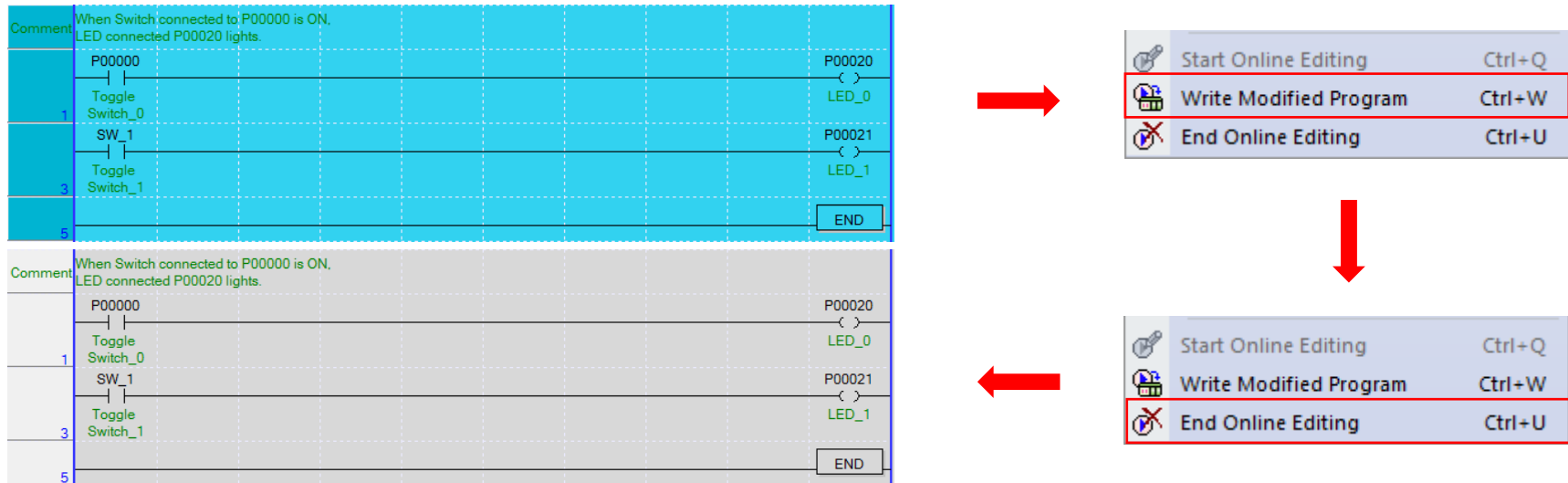
[Online Editing]

To download program to PLC, PLC must be in STOP. But, according to properties of PLC, you faces on the case that PLC doesn't change to STOP. In this case, you can revise and add program without changing PLC to STOP when online Editing is started.

- ① Select [online]-[Start online Editing]
- ② When 'online Editing' is started, the color of program window is changed.



- ③ After revising program, download modified program to PLC. Then, select [End online editing]



※ Reference

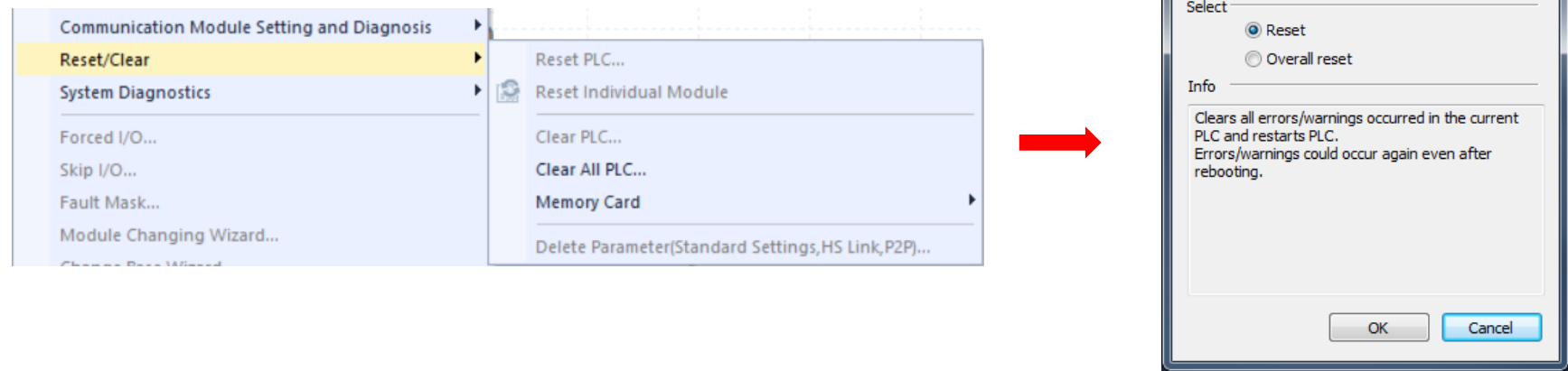
With 'online Editing', you can modify program and variable/comment only. If you completes modifying Basic Parameter and I/O Parameter, the [online] > [Writing] must be executed (the modified Basic Parameter and I/O Parameter must be written on PLC STOP).

1. XG5000 Basic

[Reset PLC]

It is used to reset PLC. Reset PLC is also available with 'Reset PLC' dip switch. Reset and Overall reset are available for its type.

- ① Select [Online]-[Reset/Clear]-[Reset PLC]
- ② After reset type selected, click [OK] button to reset PLC.



[Reset]

When PLC powered back, it will delete error/warning information to allow power on. Error/warning may occur continuously based on the situation.

[Overall Reset]

When PLC powered back, it will delete error/warning information, latch 1 area data, I/O skip, error mask and forced I/O setting area to allow power on.

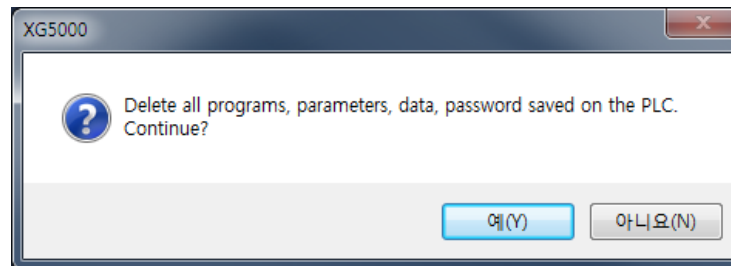
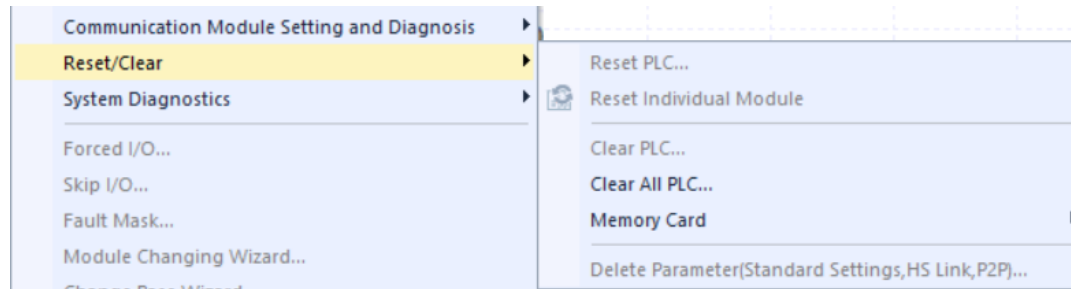
	K, R area	Latch 1	Latch 2
Reset	Retain	Retain	Retain
Overall Reset	Clear	Clear	Retain

1. XG5000 Basic

[Clear All PLC]

It is used to delete all programs, parameters, passwords, data of the PLC.

- ① Select [Online]-[Reset/Clear]-[Clear All PLC]
- ② The message will be displayed; "All programs, parameters, passwords, data of the PLC are deleted"
- ③ If you press [Yes] button, the connect setup window will be displayed.
- ④ If you press [OK] button in the connect setup window, the process will be executed.



※ Reference

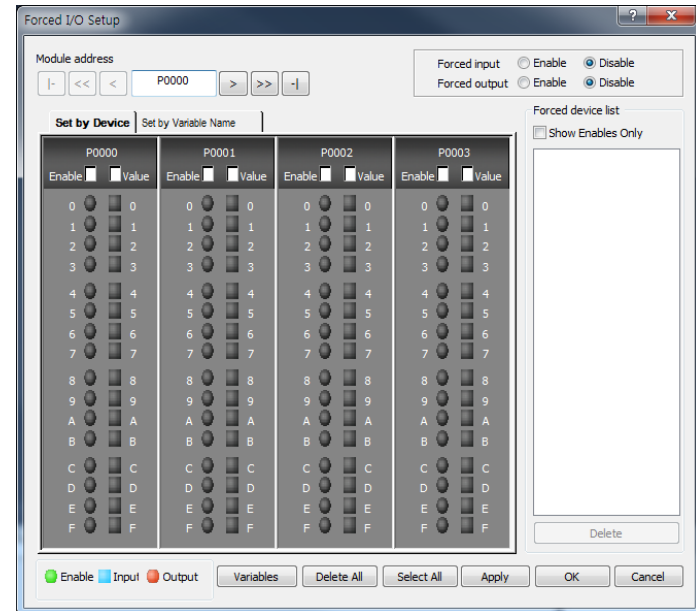
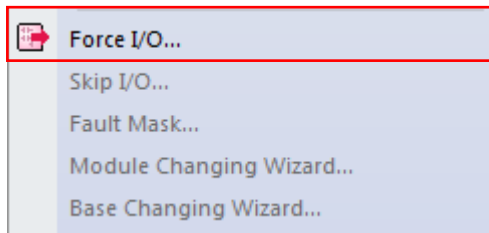
- The function 'Reset Individual Module' is available only in XGK/I/R.
- The function 'Clear All PLC' is available only in some XGB models only.
- 'Clear All PLC' is available only in PLC STOP Mode.

1. XG5000 Basic

[Force I/O]

Forced I/O of the I/O refresh area can be specified in PLC.

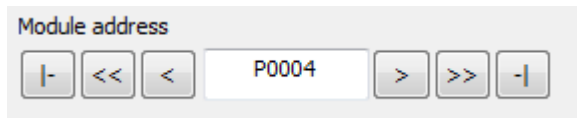
- ① Select [Online]-[Force I/O]



(1) Example of Force I/O setting

P0004 word 4th bit's forced output 1, 8th bit's forced output 0

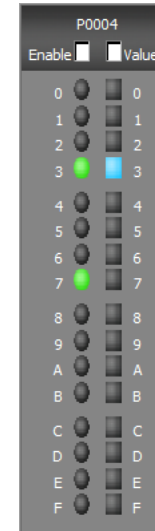
- ① Move to P0004. Use the button or directly input the area desired to move to.



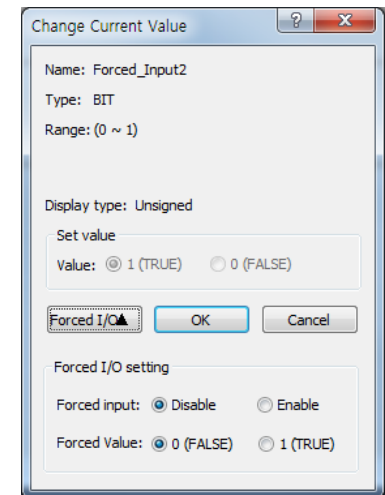
1. XG5000 Basic

[Force I/O]

- ③ Select the flag of the bit 7. as the forced output value of the bit 7 is 0, no data needs to be selected.



- ④ Selected the Forced Output Enable Flag and then click Apply button to apply the forced value.



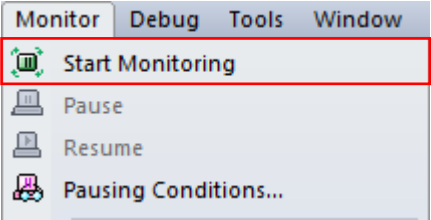
- ⑤ To cancel force I/O, you can follow the previous process, but the flags must be not selected.
- ⑥ You can set Force I/O on Ladder program. Double click the contact you want to set force I/O, then click [Forced I/O]. If you set Forced input 'Enable', you can change the value along to Forced value.

1. XG5000 Basic

[Monitoring]

Monitoring is to express status of PLC and data visually, as a result, it is useful to debug program. XG5000 supports variable monitor, system monitor, device monitor, user events, data traces etc. as well as expressing operated data. Also, you can change data for internal device (M, D etc.)

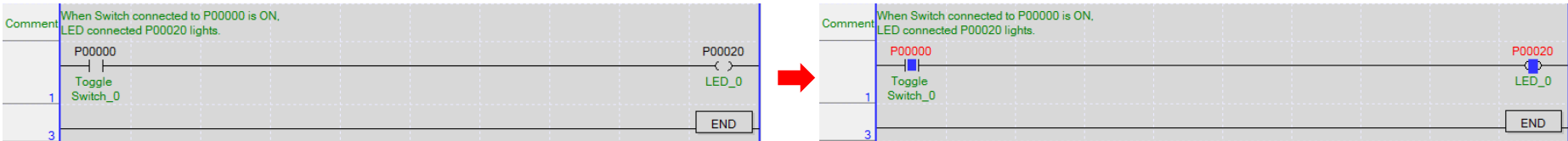
① Select [Monitor] > [Start Monitoring]



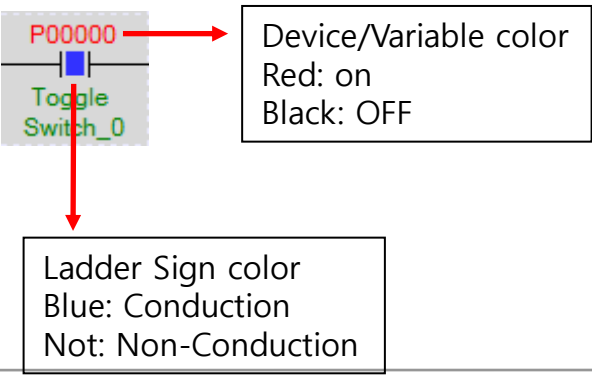
[Ladder Monitoring]

(1) Bit data expression in ladder monitoring

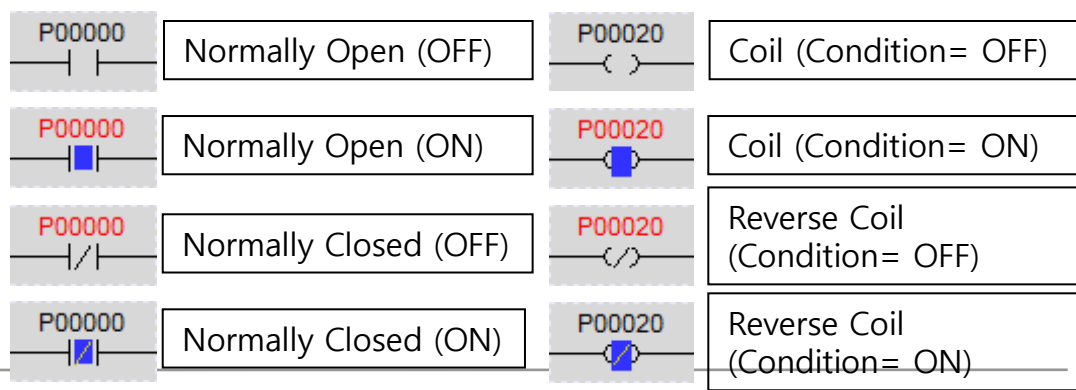
When Monitor is started on opening ladder program, the current data of ladder program is expressed. In ladder monitoring, you can check status of data, contact and coil with variable color, ladder sign color.



<When P00000 is OFF>



<When P00000 is on>

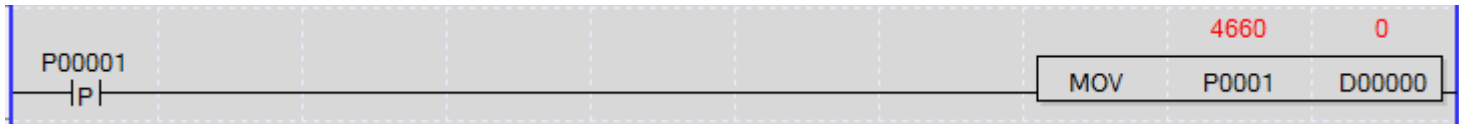


1. XG5000 Basic

[Monitoring]

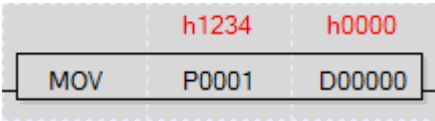
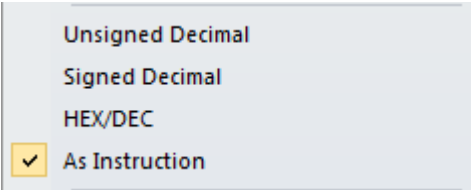
(2) Word device expression in ladder monitoring

Word data in ladder monitoring is expressed over instruction. The data type that you select is expressed



[Expressed data type]

when clicking mouse right button on program window, you can choose data type.

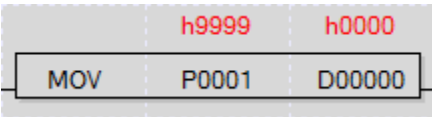


<HEX expression>

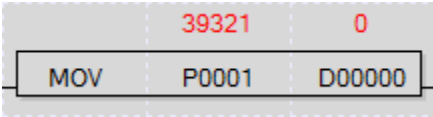


<Decimal expression>

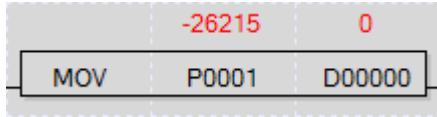
In case that expressing 1 Word data to decimal, 0 ~ 32,767 is expressed to same data regardless of sign. However, over 32,768 data in the base of unsigned decimal is expressed differently between signed decimal and unsigned decimal



<HEX expression>



<Unsigned Decimal expression>



<signed Decimal expression>

[Data range according to data type and size]

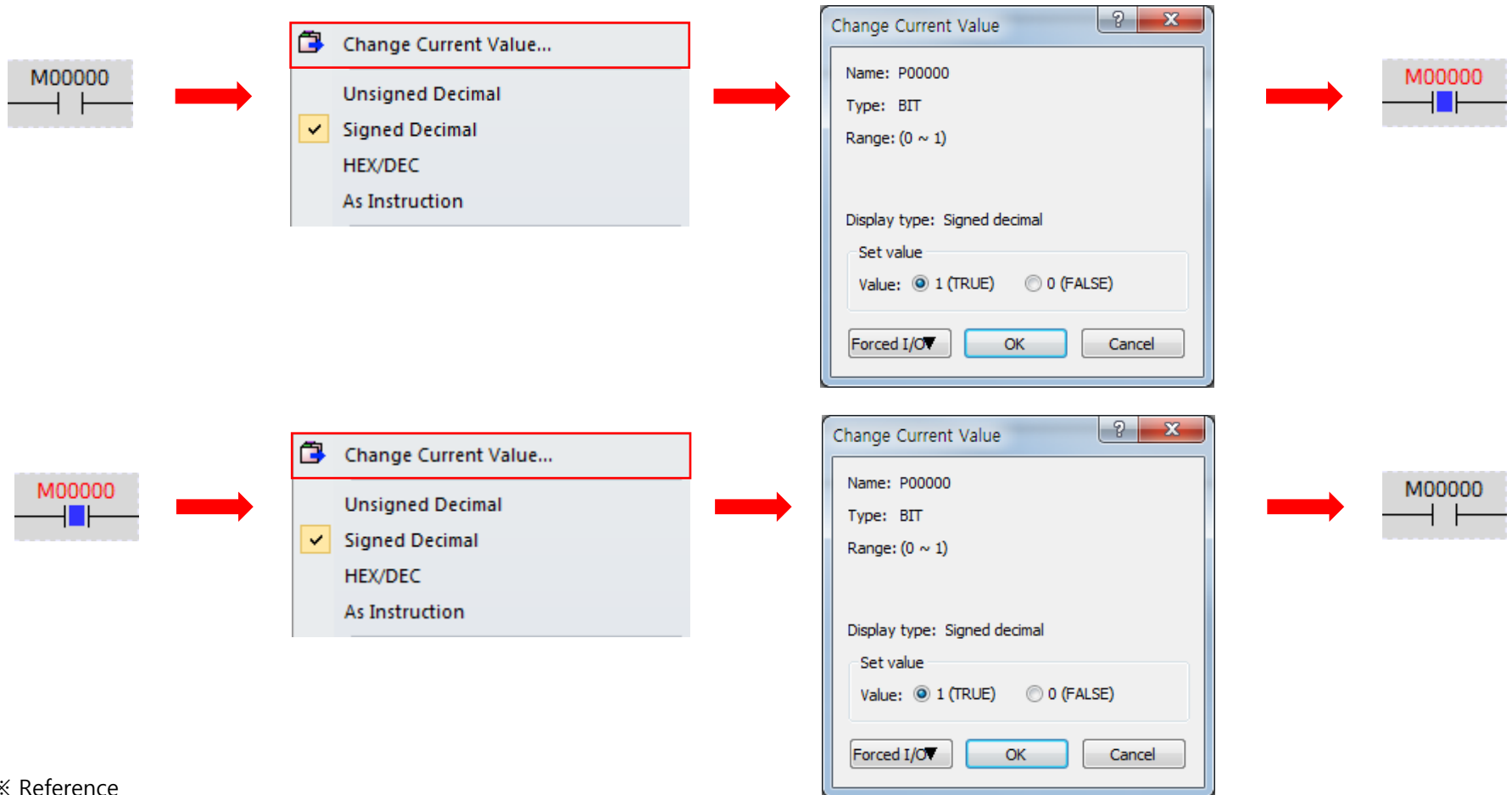
	Unsigned decimal	Signed decimal	Hex
1 Word (16 Bit)	0 ~ 65,535	-32,768 ~ 32,767	h0000 ~ hFFFF
2 Word (32 Bit)	0 ~ 4,294,967,295	-2,147,483,648 ~ 2,147,483,647	h00000000 ~ hFFFFFFFF

1. XG5000 Basic

[Changing data in ladder monitoring]

Select device and click mouse right button. Then, click [Change Current Value]

[In case of Contact (Bit)]

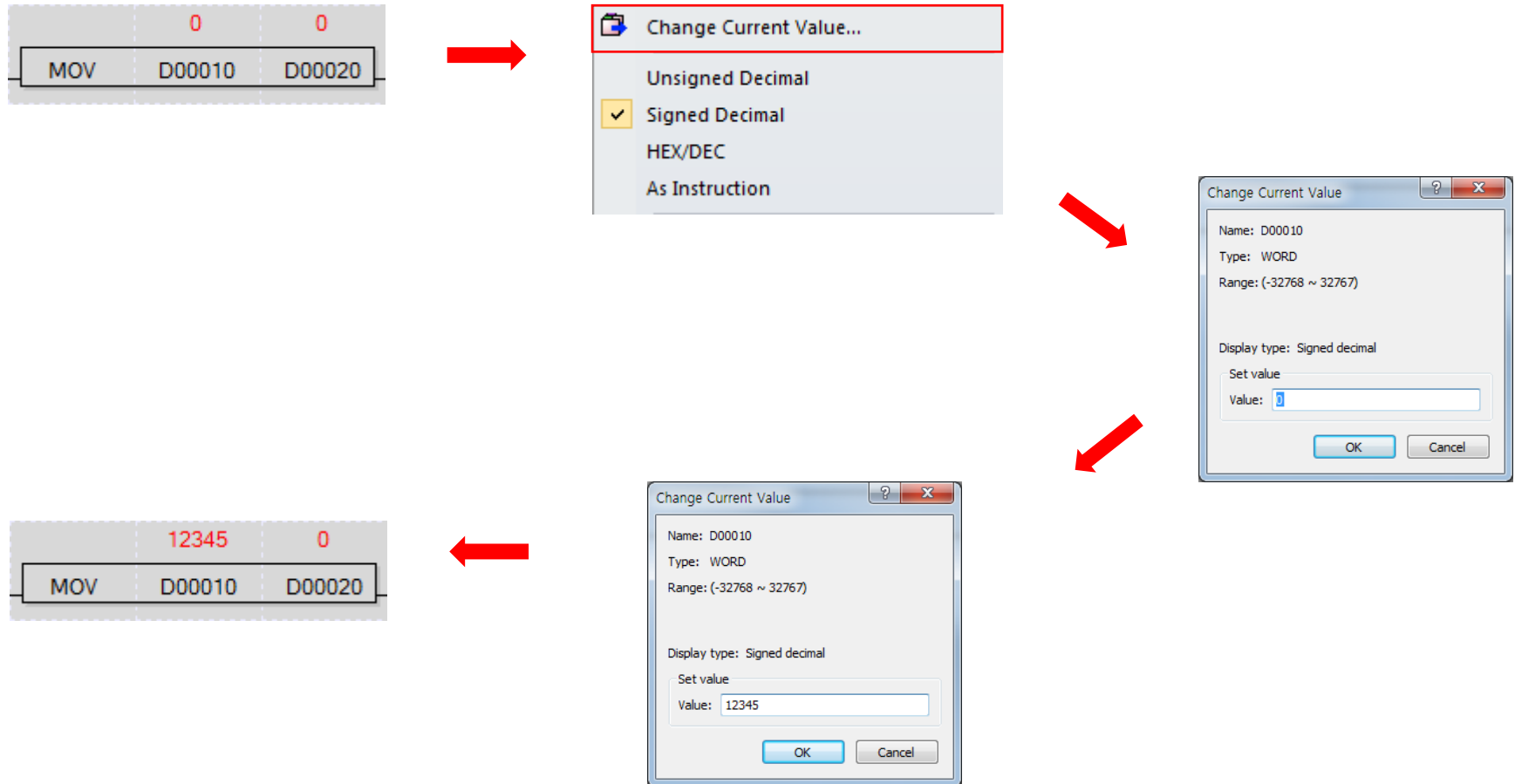


※ Reference

- The current value of Coil can't be changed.
- In case of device connected to digital input module, the current value can't be changed. In this case, data can be changed with Force I/O
- Even though internal contact in PLC RUN is selected, the current value isn't changed next using coil.

1. XG5000 Basic

[Changing data in ladder monitoring]
[In case of data (Word)]



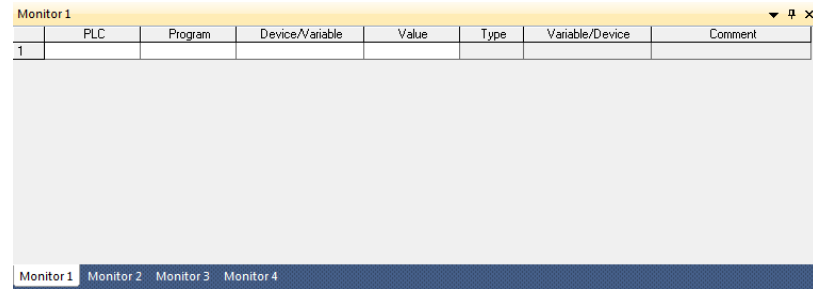
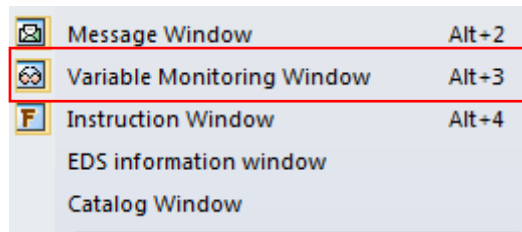
※ Reference

- It might be not changed according to operated condition
- In case of device connected to digital input module, the current value can't be changed, but Real input data will be expressed
- Data type is affected by Monitoring type of XG5000. for example, when you select Signed decimal in monitoring and put 50,000, -15,536 is expressed in ladder program

1. XG5000 Basic

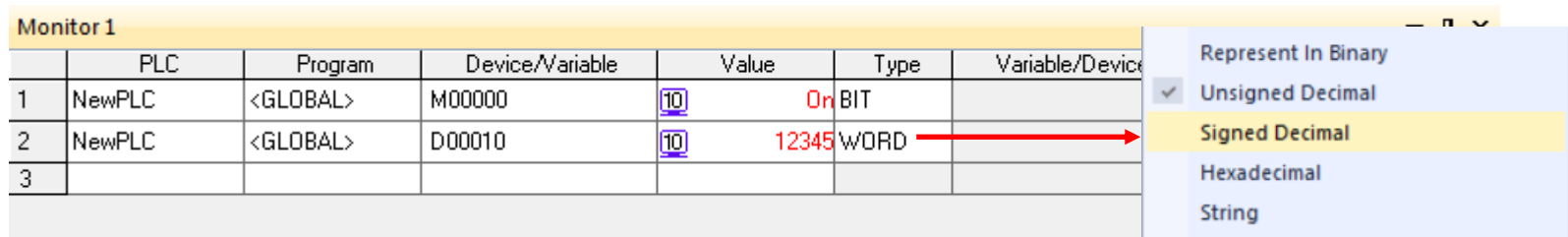
[Variable Monitoring]

you can monitor data of device which is not shown in ladder program.



[Device registration]

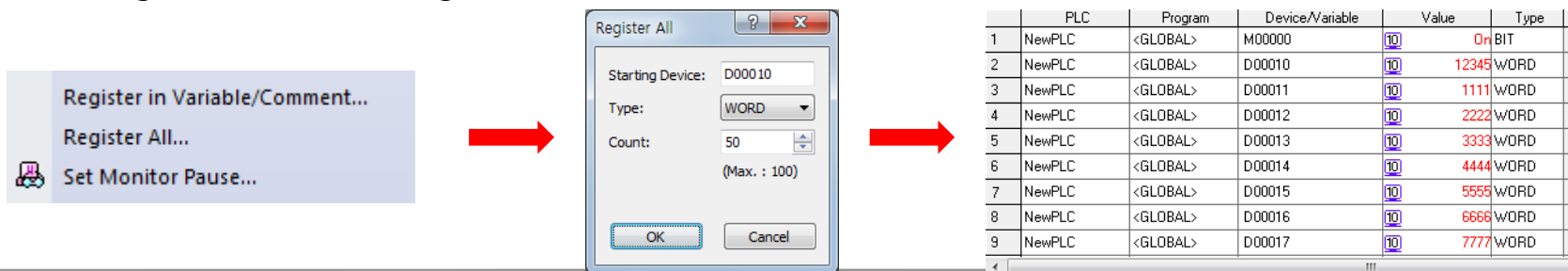
Write Device and set Type > Click mouse right button to change expressed data type



[Register All]

When it is continuous address of device, Register All can be used to monitor conveniently

Click mouse right button > click [Register All]



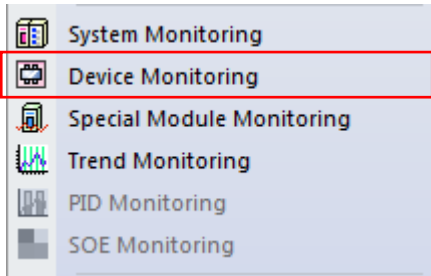
1. XG5000 Basic

[Device Monitoring]

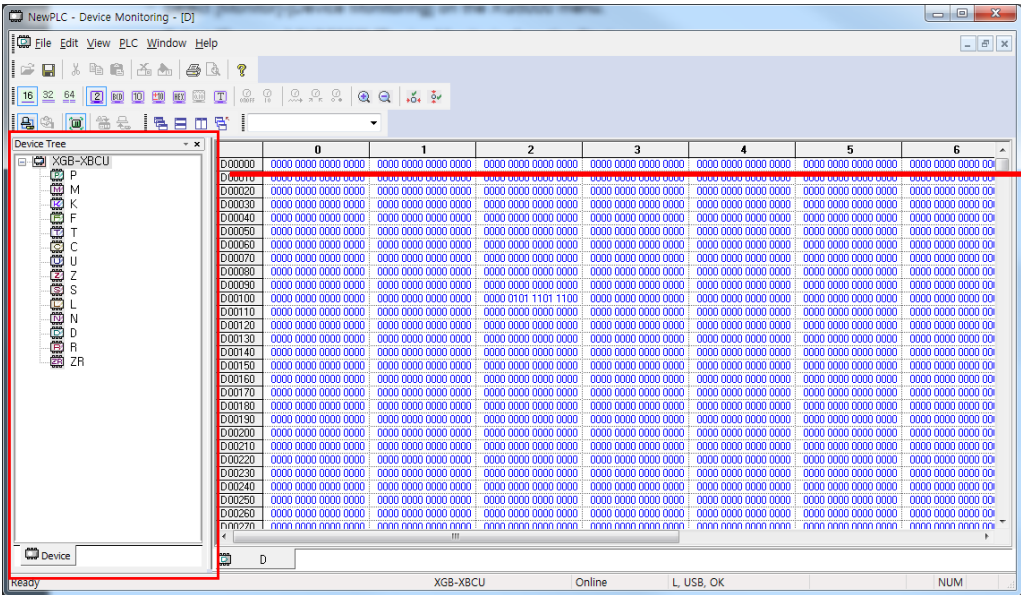
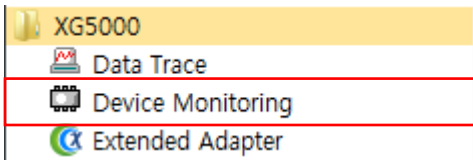
Device Monitoring can monitor all the device areas' data in PLC. It can write or read data value on the PLC's specific device. In addition, Device Monitoring can display the data value variously when displayed or input on the screen according to Bit format and display method.

(1) How to open Device monitor

① [Monitor] > [Device Monitoring]



② Windows Program > XG5000 > Device Monitoring



- The device information window displays all the device areas in PLC, based on CPU type.
- To open the device, double-click the device icon (EX. P, T, ...) or click the right button of the mouse to select [open Device] on the menu
- Device monitoring if executed on the XG5000 menu will be in Connect, Monitor status.
- If not in Monitoring mode, the device if open will display the previous data value.
- Basically the data value will be initialized to 0.

1. XG5000 Basic

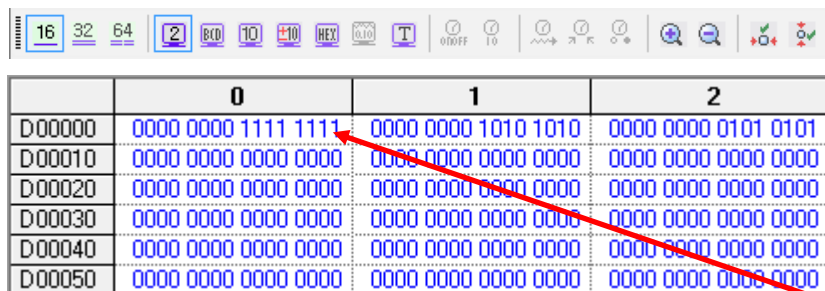
[Device Monitoring]

(2) Data Format and Display Items

There are mainly 4 methods to display data on the screen.

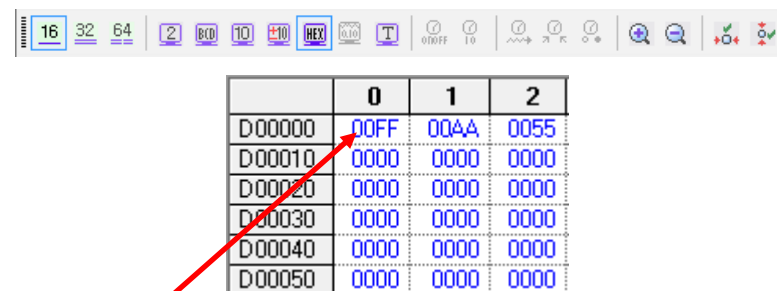
Display Setting	Description
Data Size	16 Bits, 32 Bits 64 Bits
Display Format	Binary, BCD, Unsigned decimal, Signed decimal, Hexadecimal, Real, String
View/Hide T & C device data	View Current value, View setting value, View Bit value
Display Format of T & C device Bit value	Character Bits, Figure Bits

① 16Bits / Binary



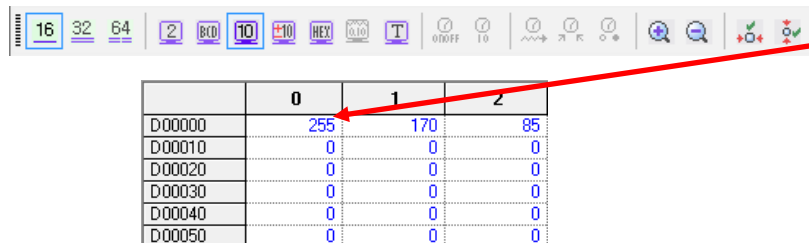
	0	1	2
D00000	0000 0000 1111 1111	0000 0000 1010 1010	0000 0000 0101 0101
D00010	0000 0000 0000 0000	0000 0000 0000 0000	0000 0000 0000 0000
D00020	0000 0000 0000 0000	0000 0000 0000 0000	0000 0000 0000 0000
D00030	0000 0000 0000 0000	0000 0000 0000 0000	0000 0000 0000 0000
D00040	0000 0000 0000 0000	0000 0000 0000 0000	0000 0000 0000 0000
D00050	0000 0000 0000 0000	0000 0000 0000 0000	0000 0000 0000 0000

② 16Bits / Hexadecimal



	0	1	2
D00000	00FF	00AA	0055
D00010	0000	0000	0000
D00020	0000	0000	0000
D00030	0000	0000	0000
D00040	0000	0000	0000
D00050	0000	0000	0000

③ 16Bits / Unsigned decimal



	0	1	2
D00000	255	170	85
D00010	0	0	0
D00020	0	0	0
D00030	0	0	0
D00040	0	0	0
D00050	0	0	0

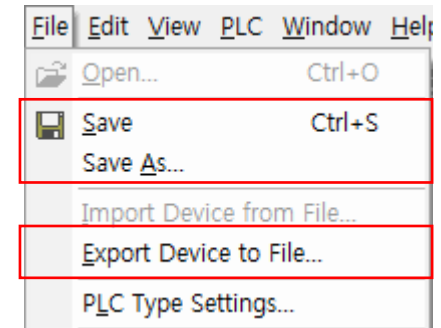
As Double clicking the cell, you can edit the data.

1. XG5000 Basic

[Device Monitoring]

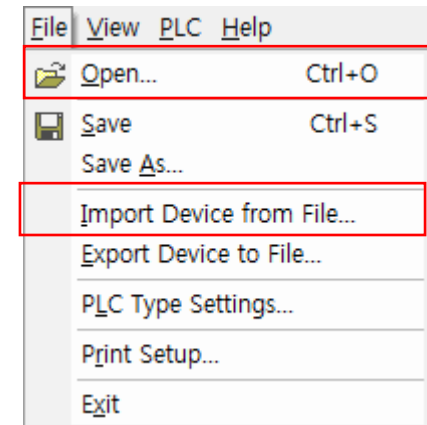
(3) Save Device

- ① Save All Device Areas
 - Select [File] > [Save] on the menu
 - Select [File] > [Save As] on the menu to save the data as a different name
 - File extension will be set as (.mem).
- ② Save Respective Device Area
 - Select [File] > [Export Device to File] on the menu
 - File extension will be set as (device+m). (Ex. Pm, mm, lm etc.)



(4) Open Device

- ① Open All Device Areas
 - Select [File] > [Open] on the menu
- ② Import Device from File
 - Select [File] > [Import Device to File] on the menu



1. XG5000 Basic

[Special Module Monitor]

Through the "Monitoring/Test" screen, the user can directly change the parameter value saved in the special module while checking the test operation and its status of the applicable module.

(1) How to open Special Module Monitor

- ① [Monitor] > [Special Module Monitoring]
- ② Example of Monitoring with XBF-RD04A

The first screenshot shows the 'Monitor' menu with 'Special Module Monitoring' highlighted. The second screenshot shows the 'Special Module List' window with 'XBF-RD04A (RTD, 4-CH)' selected. The third screenshot shows the 'Special Module Monitor' window for 'XBF-RD04A (RTD, 4-CH)' with various monitoring parameters and a 'Monitor' button.

Item	CH 0	CH 1
Temperature value	306	0
Scaling value	0	0
Min. temp value	294	0
Max. temp value	339	0

Item	CH 2	CH 3
Temperature value	0	0
Scaling value	0	0
Min. temp value	0	0
Max. temp value	0	0

Item	Setting value	Current value
Channel	CH 0	
Channel status	Disable	Enable
Sensor type	PT100	PT100
Temp. unit	Celsius	Celsius
Filter constant	0	0
Scaling	Disable	Disable

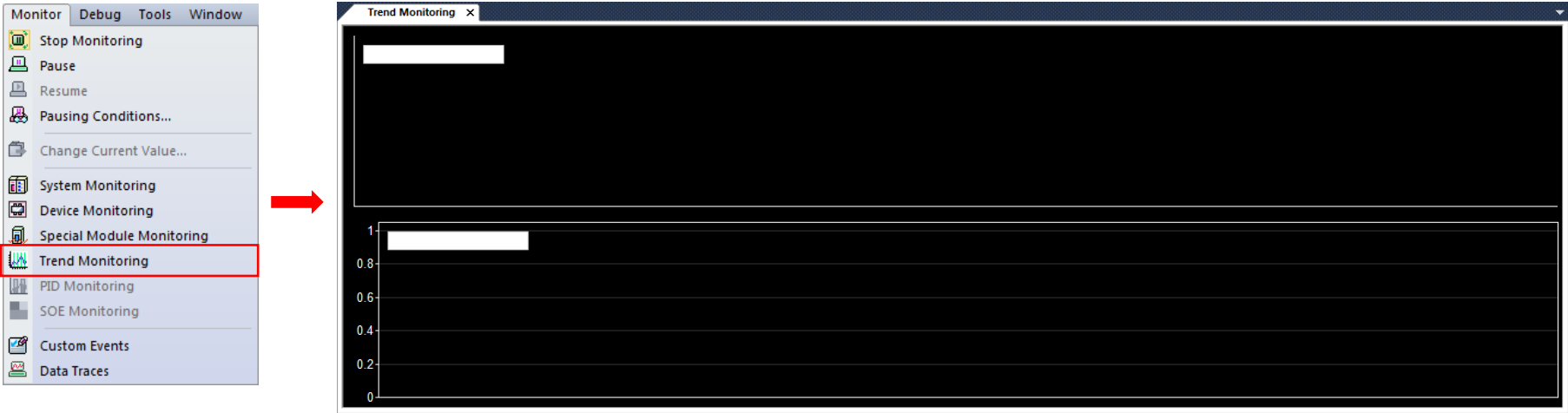
1. XG5000 Basic

[Trend Monitor]

Trend monitoring is used to read the data periodically from PLC connected so to display in a graph. The trend monitoring window is composed of Bit graph and trend graph. In the Bit graph, Bit device's on/Off status is displayed in stair-shaped graph. And in the trend graph, the variation trend of the data is displayed with the device value converted from Word to data format specified.

(1) Start Trend Monitor

① [Monitor] > [Special Module Monitoring]



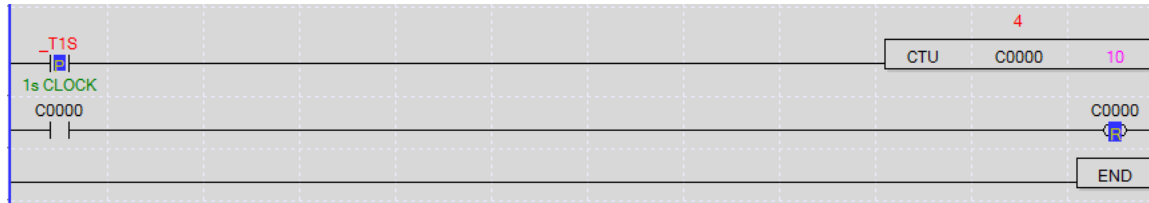
1. XG5000 Basic

[Trend Monitor]

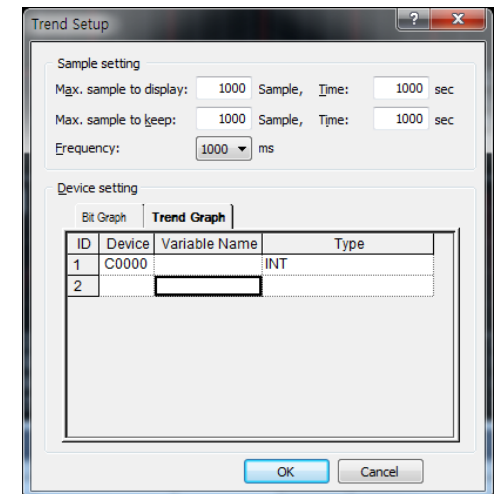
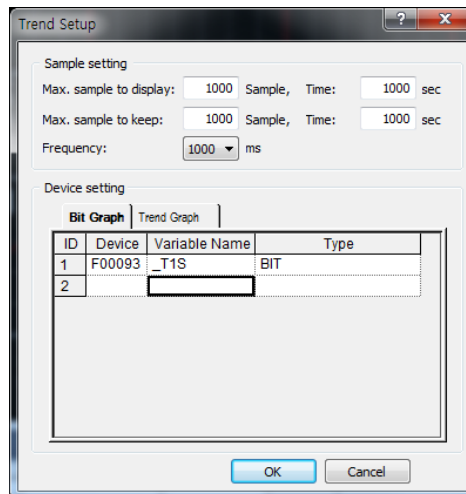
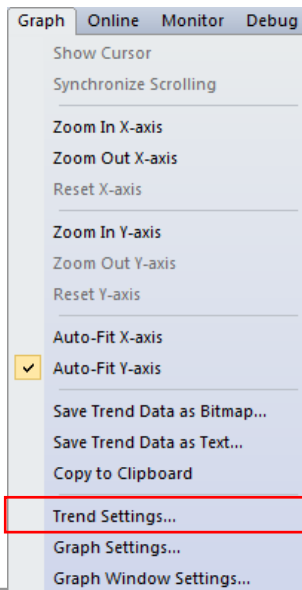
(2) Example of Trend Monitor

The current value of a counter, C0000, whose data increase by 1 every seconds and limited to 10, will be displayed with trend graph.

- ① Make a program as follows



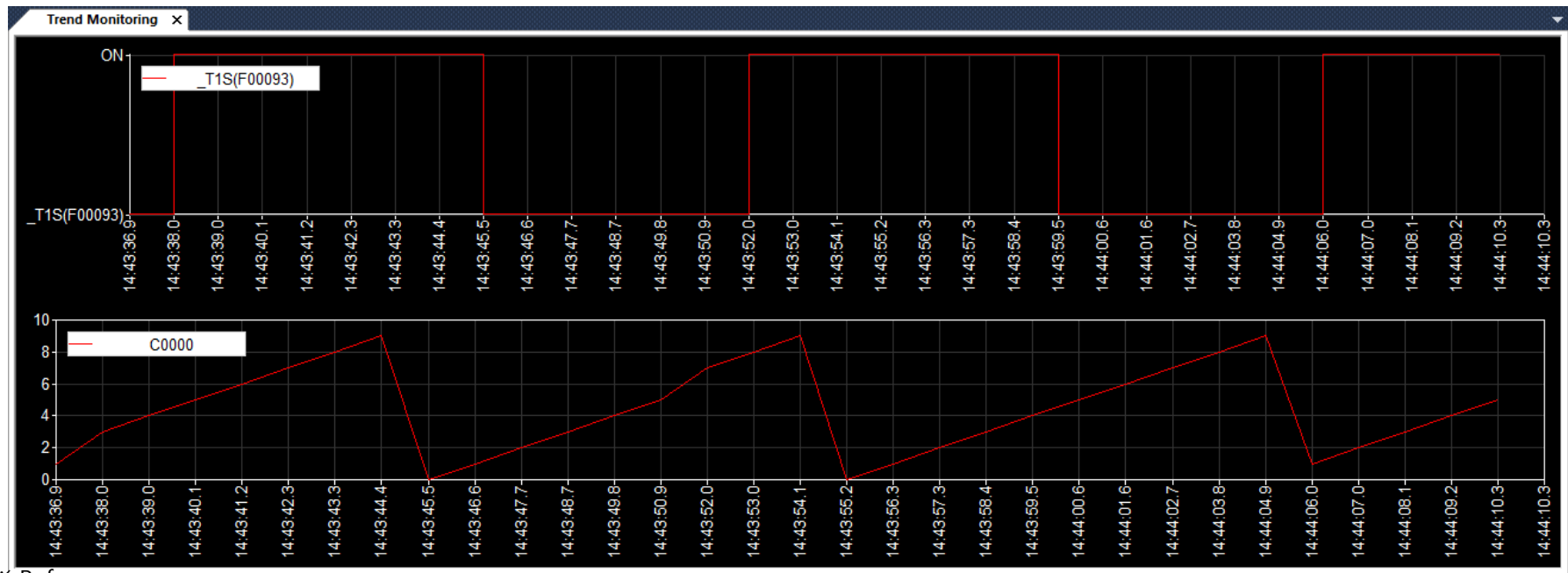
- ② Write the program to the PLC and change the mode RUN.
- ③ Select [Monitor]-[Trend Monitoring] on the menu to display the trend monitor window.
- ④ Select [Graph]-[Trend Settings].
- ⑤ Set [Bit Graph] and [Trend Graph].



1. XG5000 Basic

[Trend Monitor]

⑥ Click [Monitor]-[Start/Stop Monitoring] and the monitoring will be started. The current value of counter, C0000, will be displayed with trend graph.



- ※ Reference
- The data to display on the trend monitoring may be different from the actual data. In order to monitor correct timing, use the data trace function.
 - Please refer to XG5000 Manual for checking about further information of Trend Monitoring.

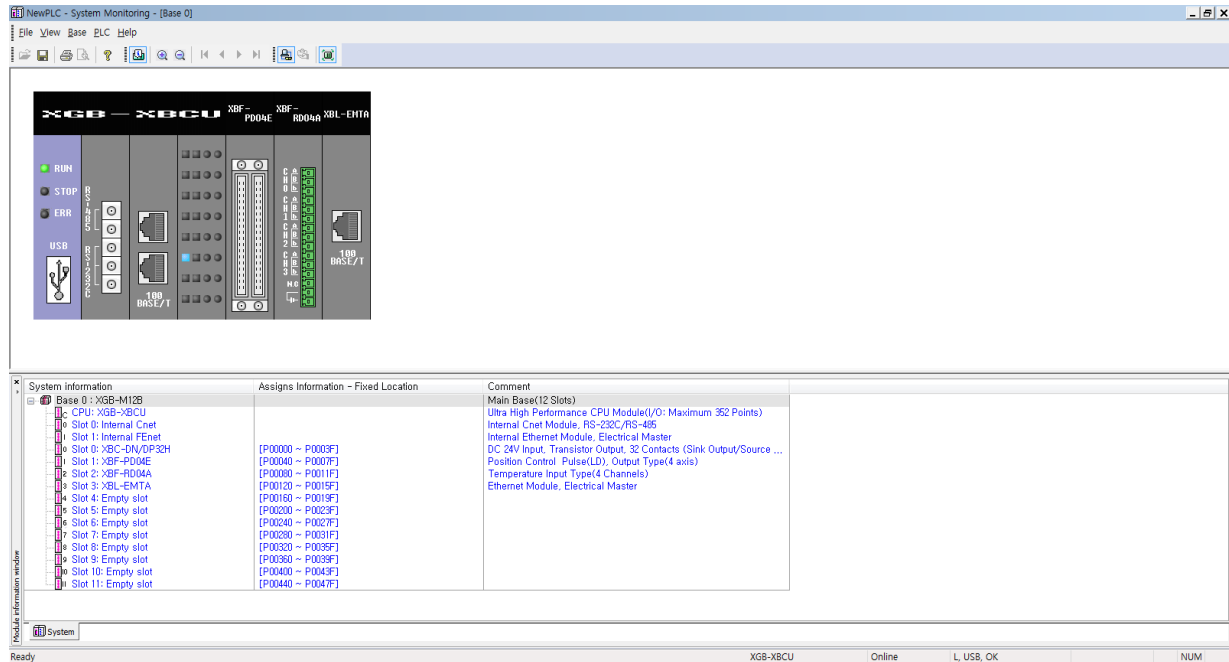
1. XG5000 Basic

[System Monitor]

System Monitoring is used to display PLC's slot information, I/O information assigned, module status and data value.

(1) How to open System Monitor

- ① [Monitor] > [System Monitor]
- ② Windows [Start] > XG5000 > System Monitoring



- Module Information Window displays the information of the slot installed on PLC. After reading the module information saved in PLC, it displays it on the data display screen of the module information window.
- ③ Select one of those specified below to view Base.
 - Select items on the module information window. (Ex. Base 0, Base 1, ...)
 - Select [Base] items on the menu. (Select First, Previous, Next, Last base)
 - Use the direction key on the keyboard to position the module's cursor on the base to select

1. XG5000 Basic

[System Monitor]

(2) Connect/Disconnect

System Monitoring can be created by a call from XG5000, or can be also executed solely. Thus, the connection is available to PLC with Connect options. If connected with PLC, base information is read from PLC to display on the module information window.

(3) System Synchronization

It reads base information, I/O assignment method and slot information specified in PLC to display on the screen. When monitored, it will read I/O skip information and forced I/O input/output information to change the current value.

(4) All I/O modules on/OFF

It is used to check output value of all the I/O modules installed on PLC.

(5) Selected I/O modules on/OFF

It is used to check output value of the selected I/O modules installed on PLC.

(6) Change Current Value

In order to change the current value, it shall be in the connection status with PLC and in monitoring mode. Click the contact point to change the data value of the selected contact point to on or OFF.

(7) Information Display of Power Module, CPU module, Communication module, Special module

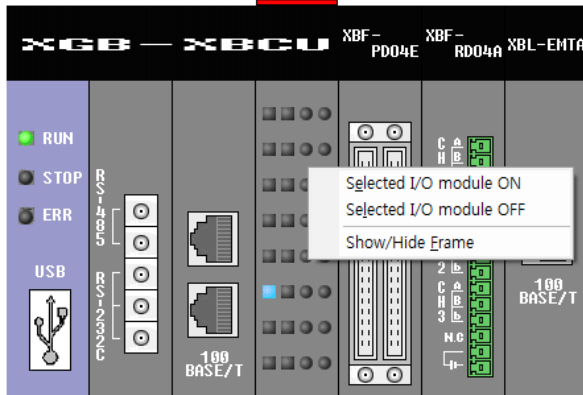
Information of Module displays base power-cut history, CPU version, type, operation mode, key status, CPU status, connection status, forced I/O setting status, I/O skip, error mask status, H/W version, O/S version and its installed date.

1. XG5000 Basic

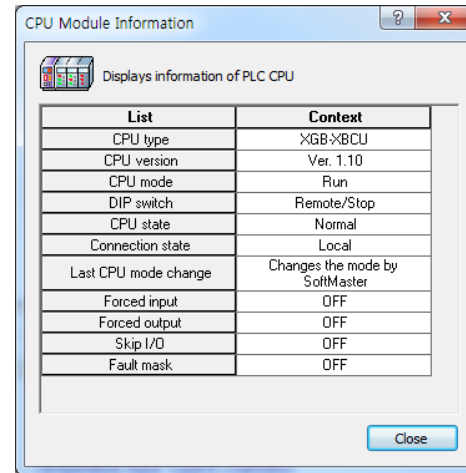
[System Monitor]

(8) Example

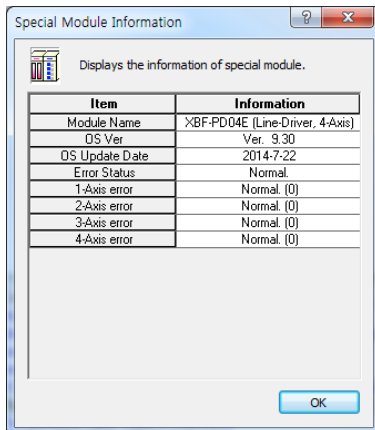
① I/O module on/OFF



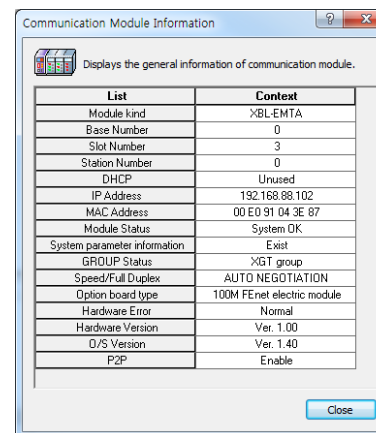
② CPU Module Information



③ Special Module Information



④ Communication Module Information



1. XG5000 Basic

[System Diagnosis]

When you can diagnose PLC status. As a result, you can check PLC Information, PLC History, PLC Errors/Warnings and I/O Information.

① [online] > [Diagnosis]

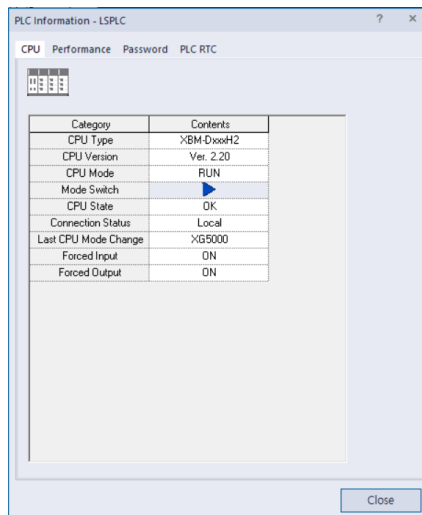


[PLC Information]

It shows the information of PLC connected, where password and PLC timer can be set.

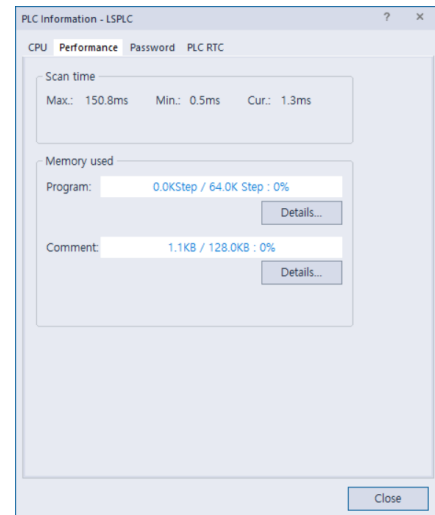
(1) CPU Information

Detailed information of PLC CPU can be checked



(2) CPU Performance

Scan time of PLC and memory application status can be checked.

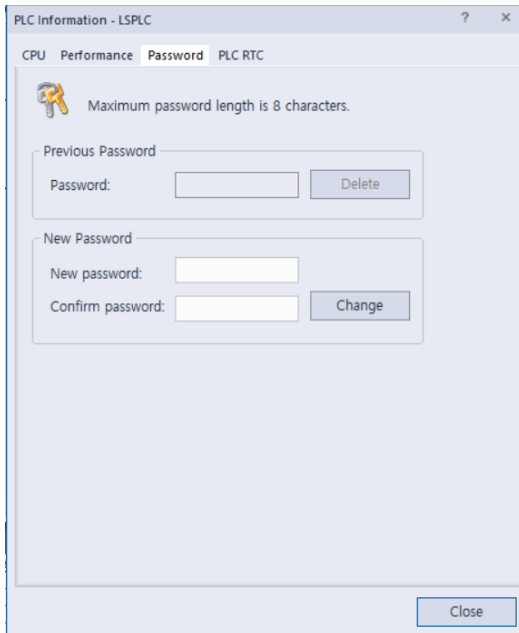


1. XG5000 Basic

[PLC Information]

(3) Password

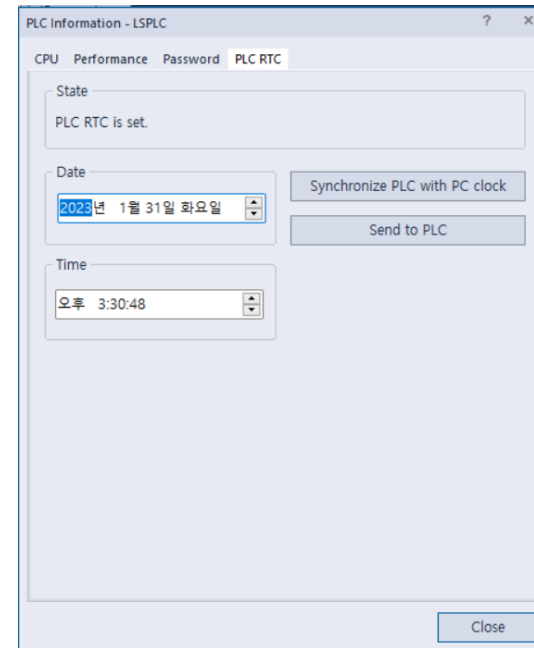
User password can be specified, changed or deleted to protect PLC information.



The screenshot shows the 'PLC Information - LSPLC' window with the 'Password' tab selected. The window has a title bar with a question mark and a close button. Below the title bar are tabs for 'CPU', 'Performance', 'Password', and 'PLC RTC'. The 'Password' tab contains a key icon and the text 'Maximum password length is 8 characters.' There are two sections: 'Previous Password' with a 'Password:' label and a text box, and a 'Delete' button; and 'New Password' with 'New password:' and 'Confirm password:' labels, text boxes, and a 'Change' button. A 'Close' button is at the bottom right.

(4) PLC RTC Setting

The RTC (Real Time Clock) of PLC can be setup as follow steps.



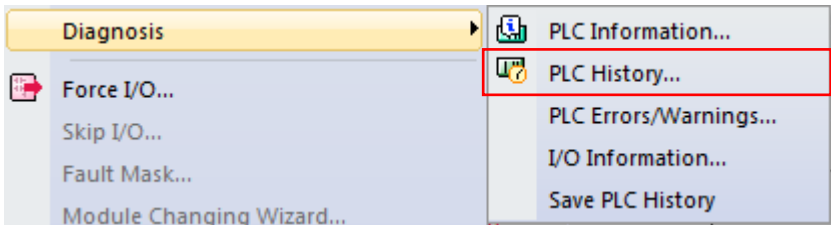
The screenshot shows the 'PLC Information - LSPLC' window with the 'PLC RTC' tab selected. The window has a title bar with a question mark and a close button. Below the title bar are tabs for 'CPU', 'Performance', 'Password', and 'PLC RTC'. The 'PLC RTC' tab contains a 'State' section with the text 'PLC RTC is set.'; a 'Date' section with a date picker showing '2023년 1월 31일 화요일' and a 'Synchronize PLC with PC clock' button; and a 'Time' section with a time picker showing '오후 3:30:48' and a 'Send to PLC' button. A 'Close' button is at the bottom right.

1. XG5000 Basic

[PLC History]

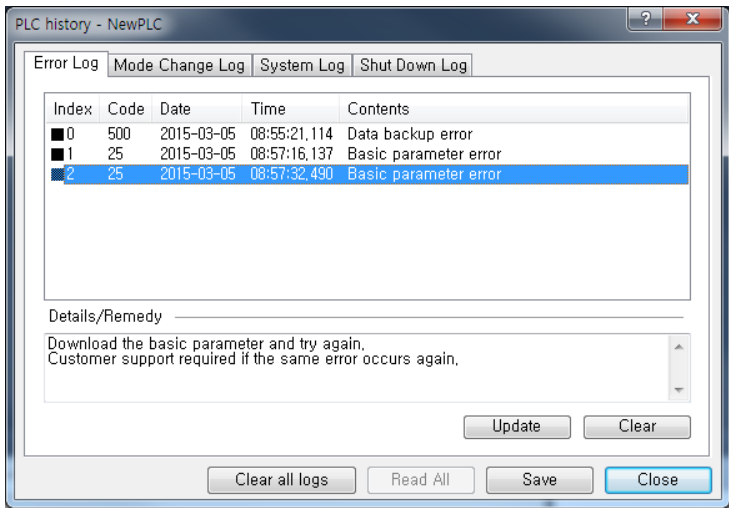
It displays the history data of error/warning, Change Mode and Shut down Log saved in PLC.

① [online] > [Diagnosis] > [PLC History]



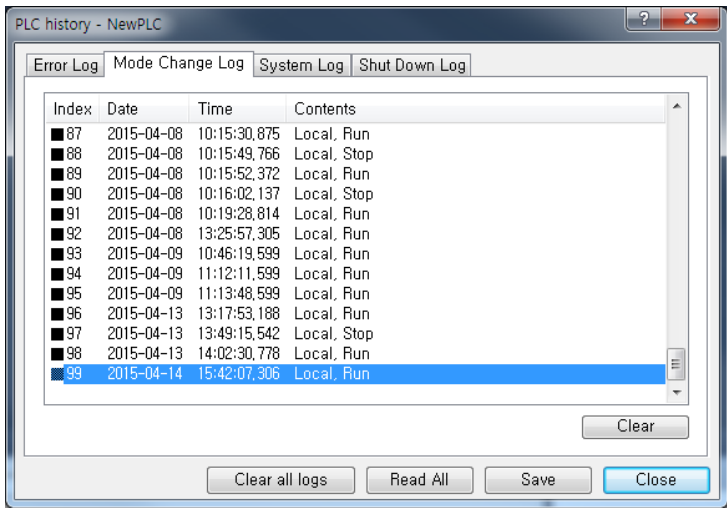
(1) Error Log

Error Log of PLC CPU can be checked. Also, it displays details of the error selected from the history along with action to take against the error



(2) Change Mode History

It shows the Mode Change Log of PLC operation modes.

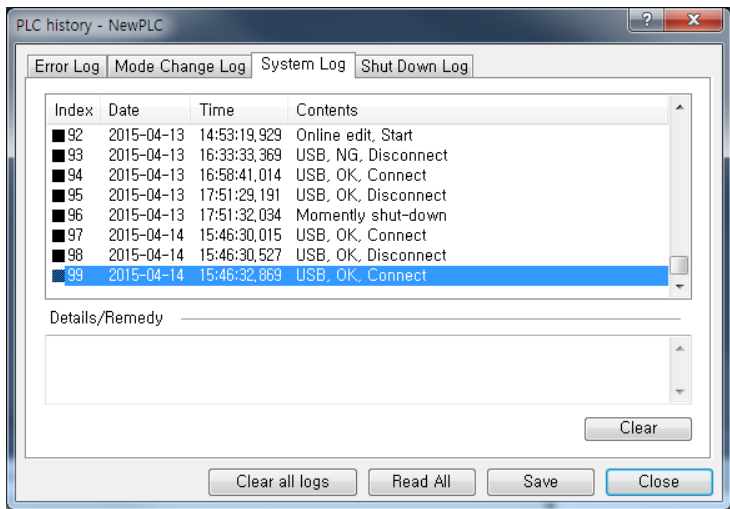


1. XG5000 Basic

[PLC History]

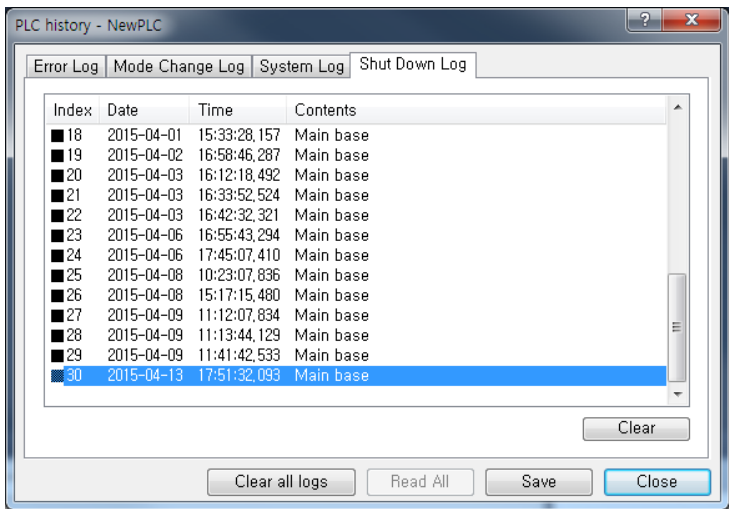
(3) System Log

It shows the history of works executed by XG5000 when PLC runs.



(4) Shut Down Log

It shows the Shut down Log history of PLC.

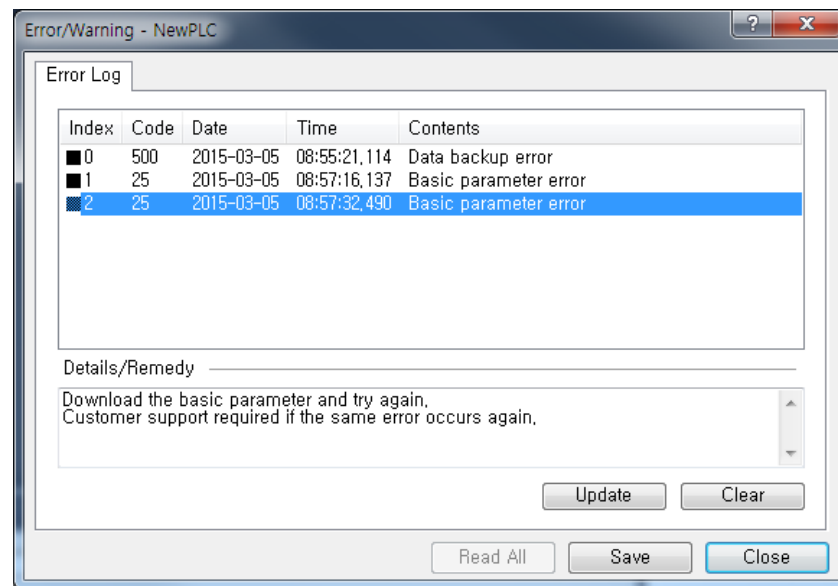
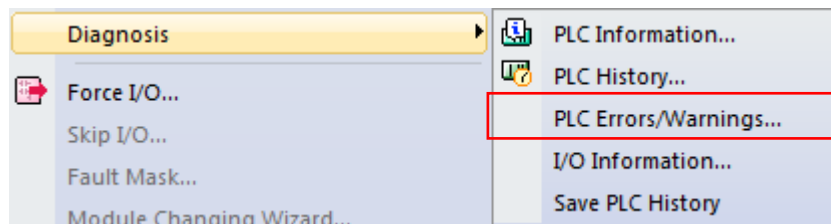


1. XG5000 Basic

[PLC Error/Warning]

Error/warning and previous Error Log presently saved in PLC can be checked.

① [online] > [Diagnosis] > [PLC Errors/Warnings]



1. XG5000 Basic

[Simple Trouble Shooting in XG5000]

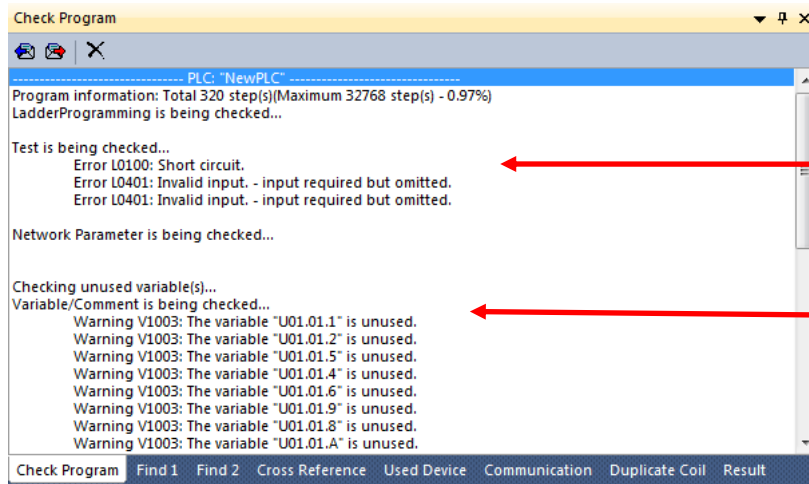
(1) Program writing Error

① Question?

I'm trying to download project to PLC. However, I failed to write the project to PLC. I don't know why the problem is caused.

② Answer

Please look for what part of program has problem. You can check this on 'Check Program' in XG5000.



There are list of Error and Warning messages in 'Check Program'.

Error: There are some errors in Program. If error message is included on 'Check Program', the Project can't be downloaded to PLC.

Warning: There are some warnings in Program. Even though warning message is existed, it doesn't matter for you to write project to PLC. However, it is required to check Ladder program.

With double clicking the messages, the window moves to step which has problem.

1. XG5000 Basic

[Simple Trouble Shooting in XG5000]

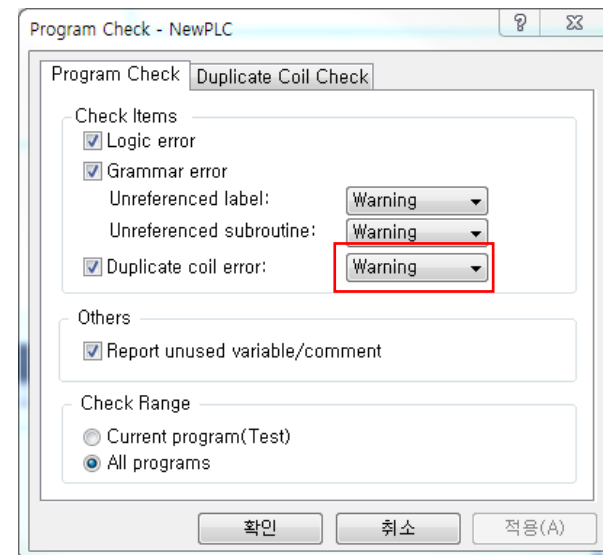
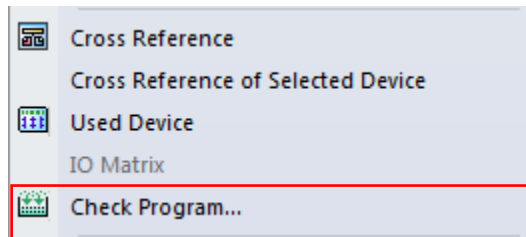
(2) Duplicated Coil Error

① Question?

I faced on the program writing error. When I look into 'Check Program', I found the message about duplicated coil Error.

② Answer

When duplicated coil error occurs, please set [Duplication coil error] to [Warning]. If it is set [Error], the duplicated Coil Error occurs.



1. XG5000 Basic

[Simple Trouble Shooting in XG5000]

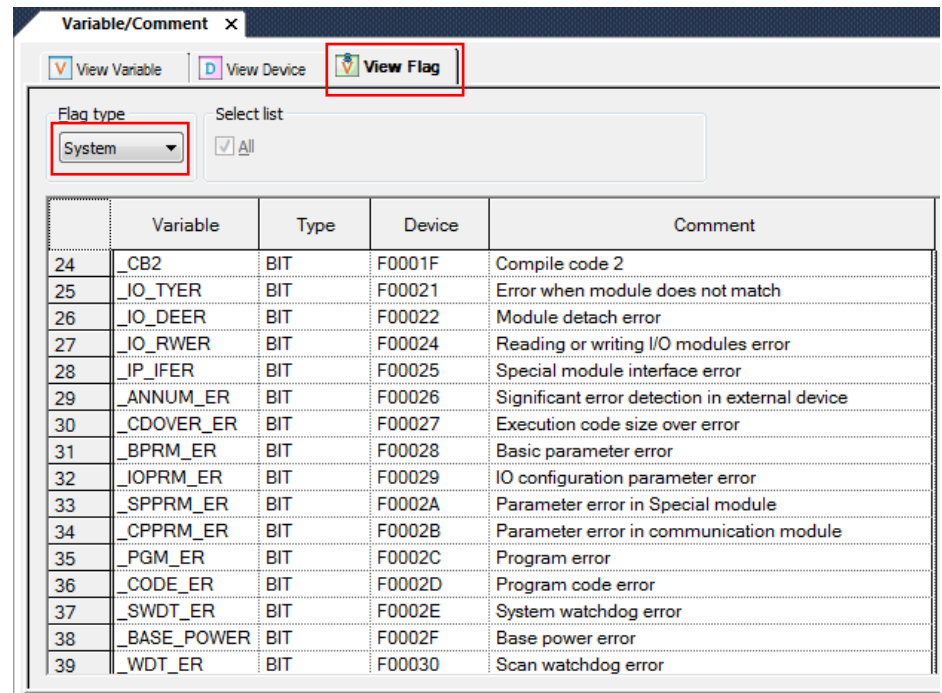
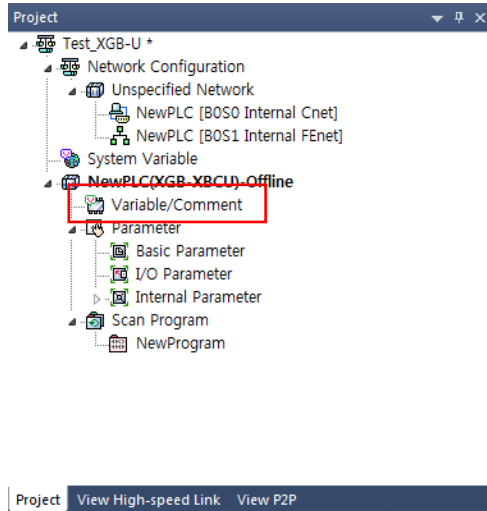
(3) Error Flag / Error Code

① Question?

I want to check PLC Error status during PLC running

② Answer

In XG5000, the flags related to PLC status exists. You can check the list of these flags on XG5000



1. PLC

2. Data Structure

2. Data Structure

[Data Type]

There are divided largely 4 data types; Bool, Integer, Real, String. Depending on how many Bits are composed, data can be divided into several types as shown below.

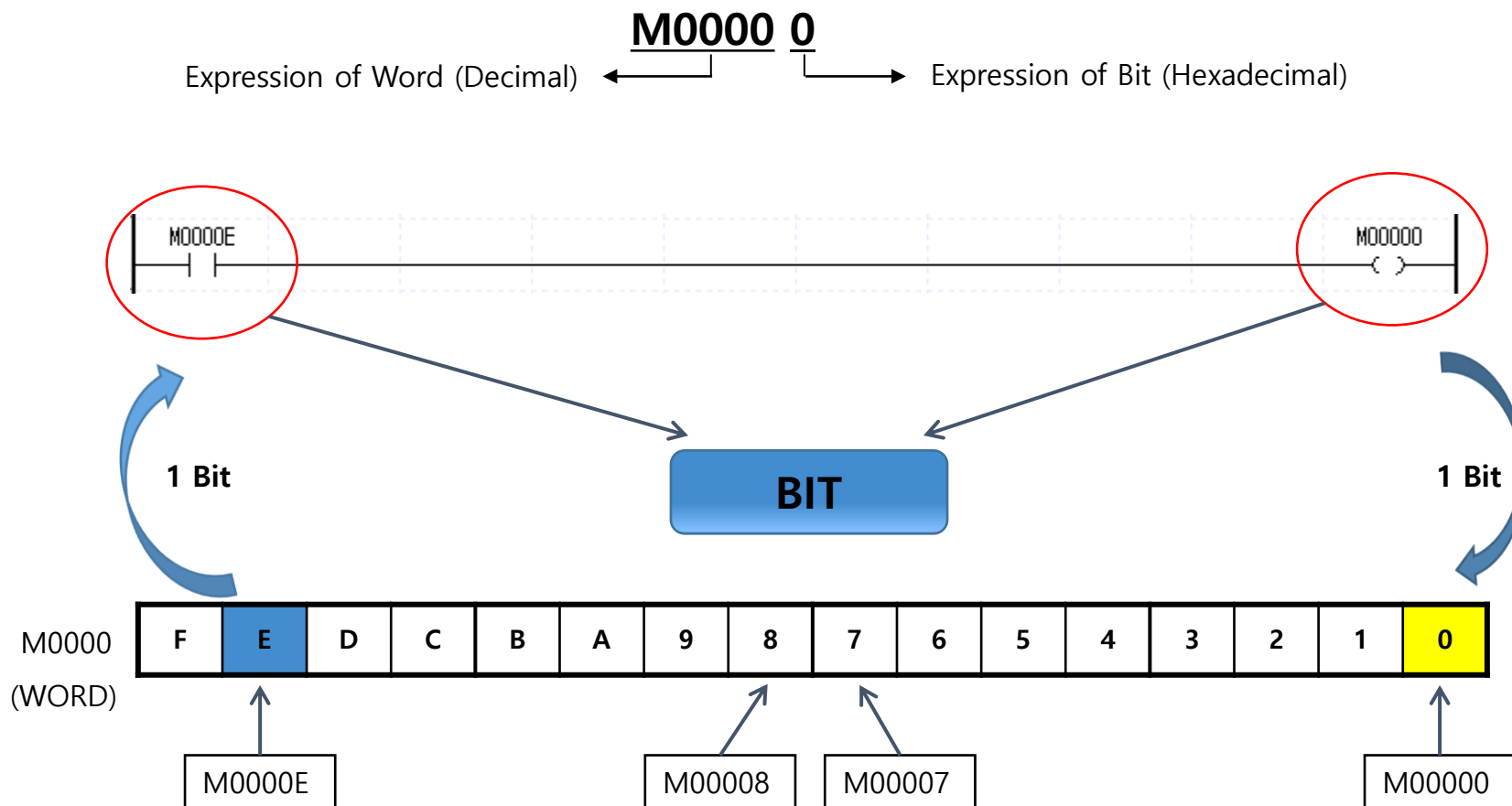
Data	Data range		
	Hexadecimal	Unsigned decimal	Signed decimal
Nibble (4 Bit)	h0 ~ hF	0 ~ 15 ($0 \sim (2^4 - 1)$)	-8 ~ 7 ($-2^3 \sim (2^3 - 1)$)
Byte (8 Bit)	h00 ~ hFF	0 ~ 127 ($0 \sim (2^8 - 1)$)	-64 ~ 63 ($-2^7 \sim (2^2 - 1)$)
Word (16 Bit)	h0000 ~ hFFFF	0 ~ 65,535 ($0 \sim (2^{16} - 1)$)	-32,768 ~ 32,767 ($-2^{15} \sim (2^{15} - 1)$)
Double Word (32 Bit)	h00000000 ~ hFFFFFFFF	0 ~ 4,294,967,295 ($0 \sim (2^{32} - 1)$)	-2,147,483,648 ~ 2,147,483,647 ($-2^{31} \sim (2^{31} - 1)$)
Float Real (32 Bit)	-3.402823466e+038 ~ -1.175494351e-038, 0, 1.175494351e-038 ~ 3.402823466e+038		
Long Real (64 Bit)	-1.7976931348623157e+308 ~ -2.2250738585072014e-308, 0, 2.2250738585072014e-308 ~ 1.7976931348623157e+308		

2. Data Structure

[Data Type (BOOL)]

1) BIT

Bit data displays on/off with 1 Bit like contact or coil, or is processed by 1 Bit unit inside the memory without I/O. In order to set the Bit of Bit device or Word device, the Bit data can be used.

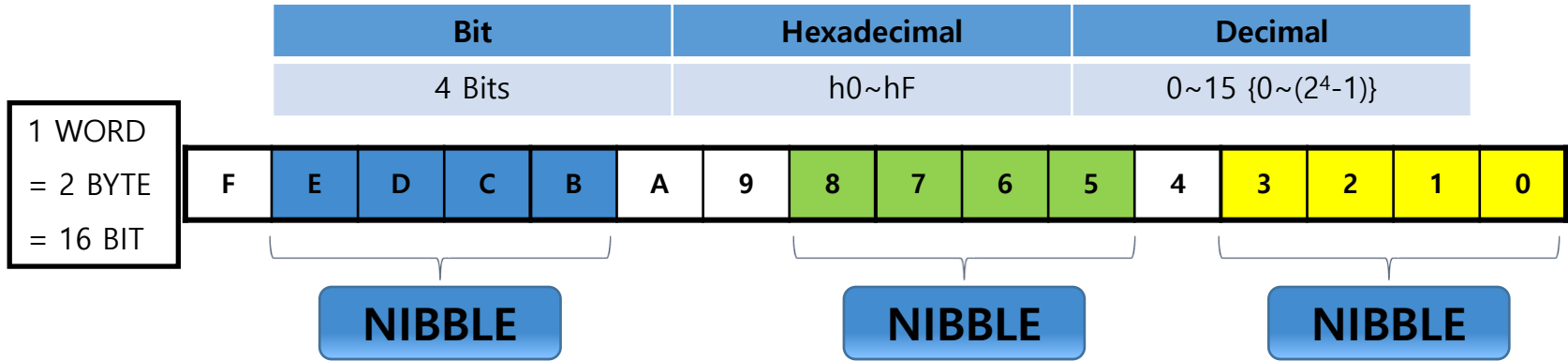


2. Data Structure

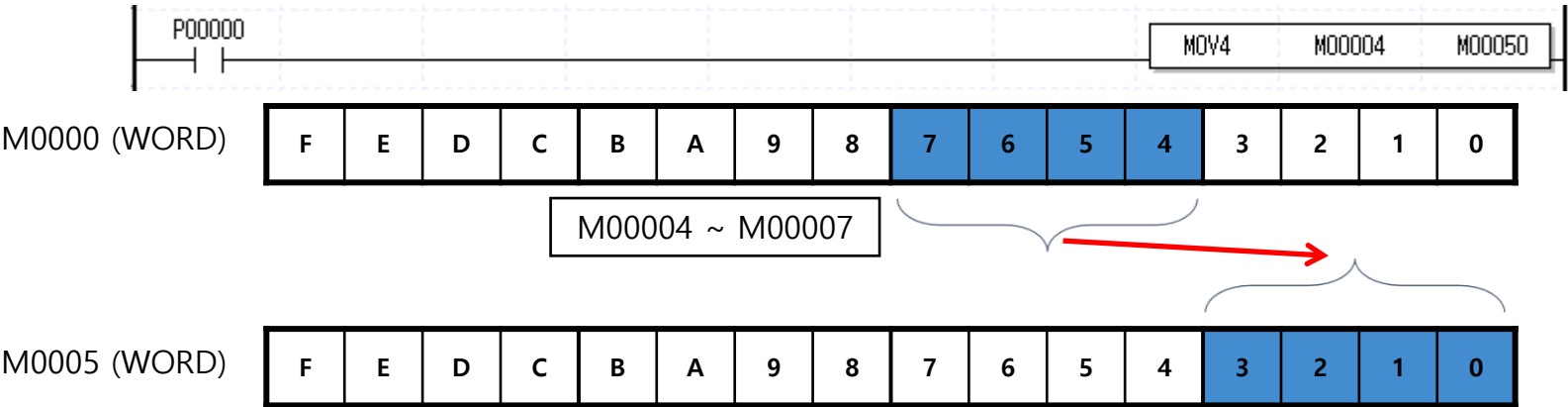
[Data Type (INTEGER)]

Nibble are used in instructions with 4 attached at the back of the name of each instruction. Nibble can be used with start Bit input. And from input contact to 4 Bits will be the data to process.

① Data Range



② Preview example: When P00000 is on, M00004's Nibble data is moved to M00000's Nibble data.



※ Reference

The operand with 4 or 8 attached at the back of instruction will be Nibble and Byte data.

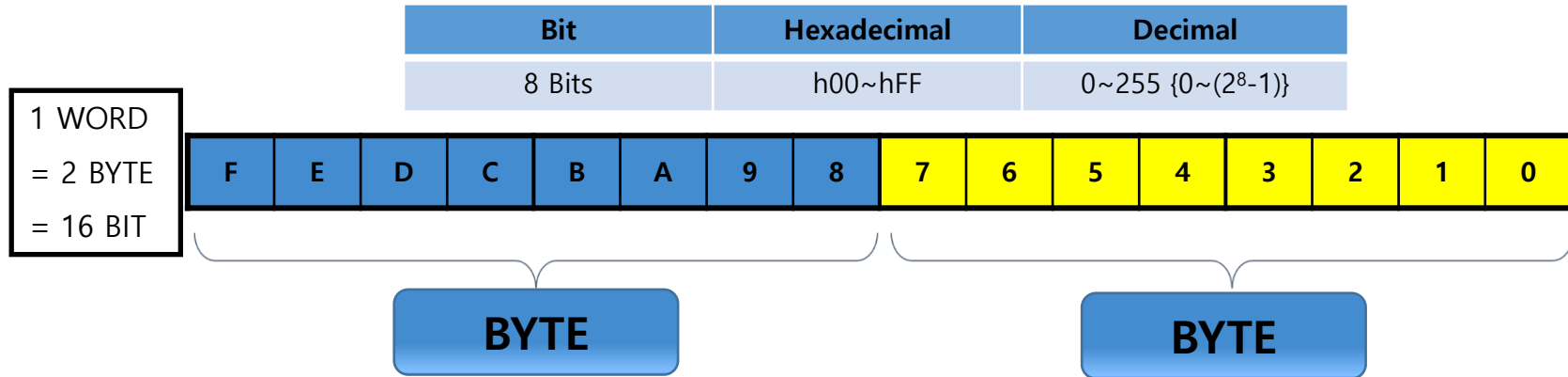
M00050 ~ M00053

2. Data Structure

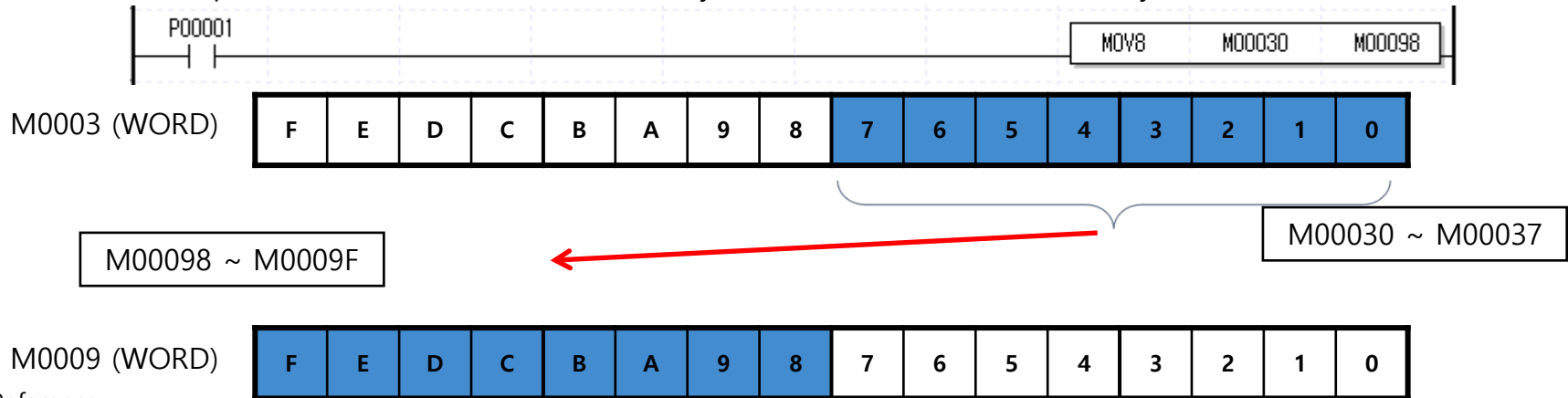
[Data Type (BYTE)]

Byte are used in instructions with 8 attached at the back of the name of each instruction. Byte can be used with start Bit input. And from input contact to 8 Bits will be the data to process.

① Data range



② Preview example: When P00001 is on, M00030's Byte data is moved to M00098's Byte data.



※ Reference

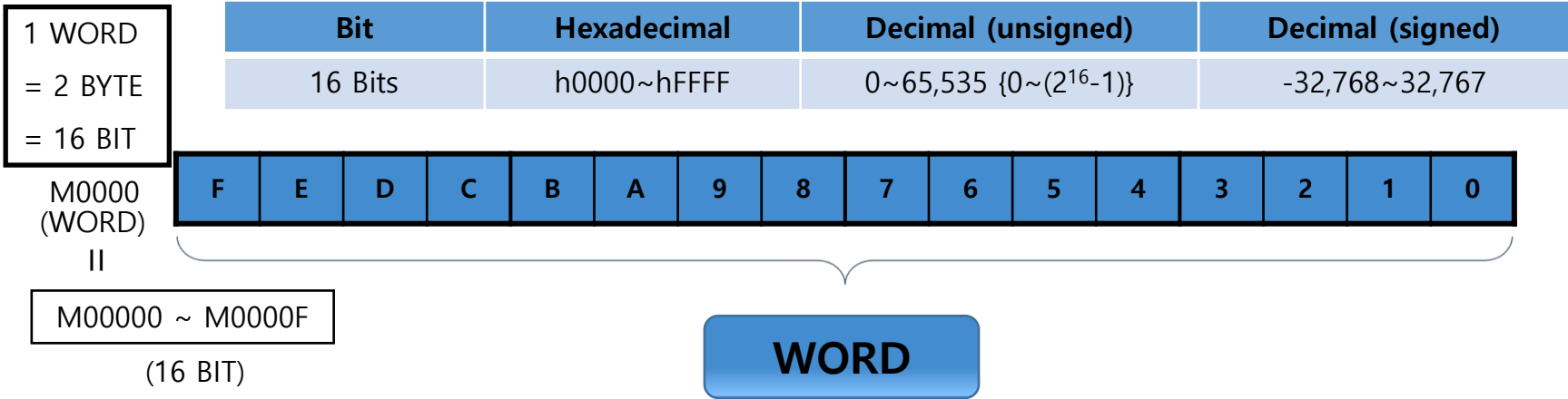
The operand with 4 or 8 attached at the back of instruction will be Nibble and Byte data.

2. Data Structure

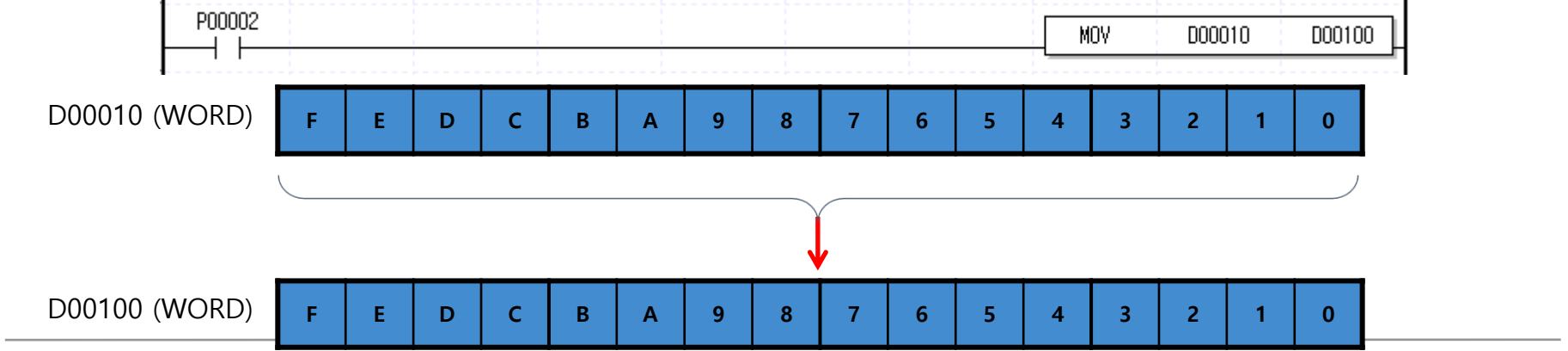
[Data Type (WORD)]

Word data is 16-Bit numeric data. It can be expressed in decimal and hexadecimal. If data is to be expressed in hexadecimal, H should be added in front of the number.

① Data range



② Preview example: When P00002 is on, D00010's Word data is moved to D00100's Word data.



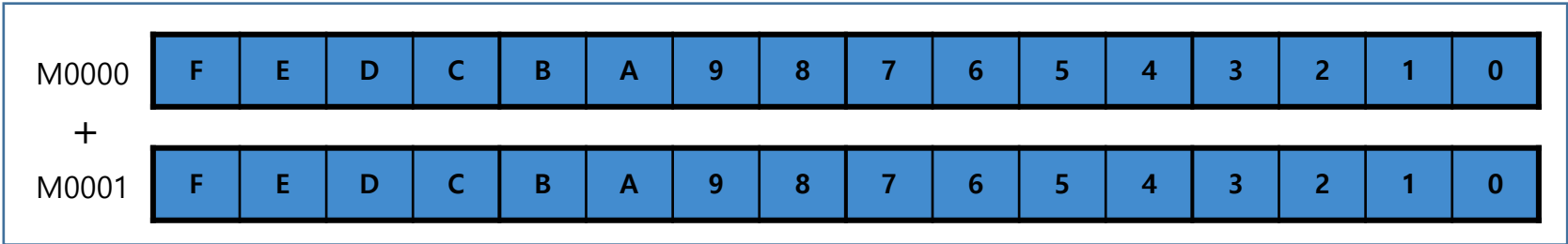
2. Data Structure

[Data Type (DOUBLE WORD)]

Double Word data is 32-Bit numeric data. It can be expressed in decimal and hexadecimal. If data is to be expressed in hexadecimal, "H" should be added in front of the number.

① Data range

Bit	Hexadecimal	Decimal (unsigned)	Decimal (signed)
32 Bits	h00000000~hFFFFFFF	0~4,294,967,295 {0~(2 ³² -1)}	-2,147,483,648~2,147,483,647

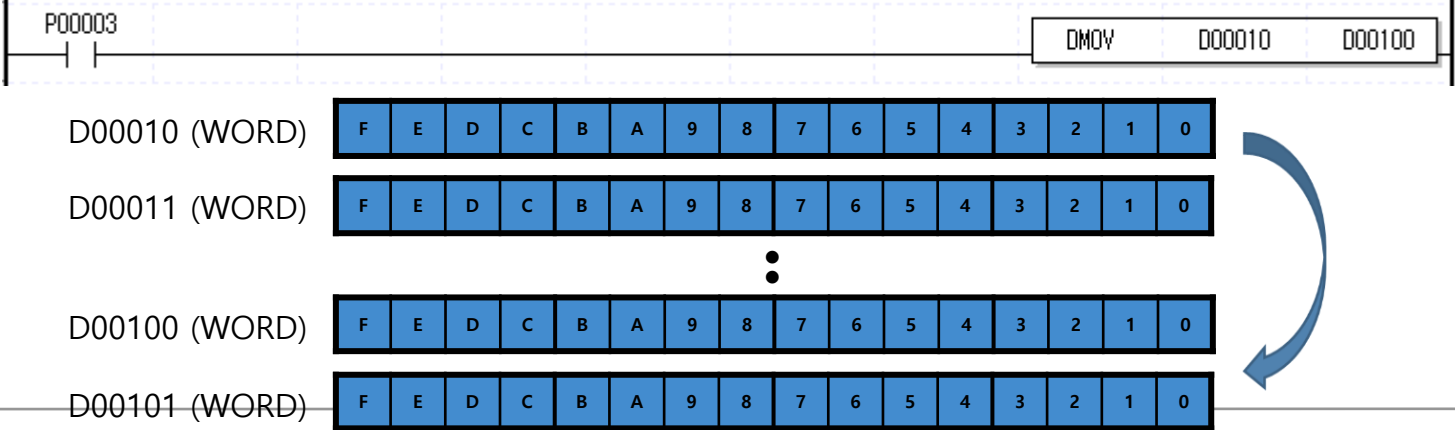


M00000 ~ M0001F

DOUBLE WORD

(32 BIT)

② Preview example: When P00003 is on, D00010's Double Word data is moved to D00100's Double Word data.



2. Data Structure

[Data Type (Real Type)]

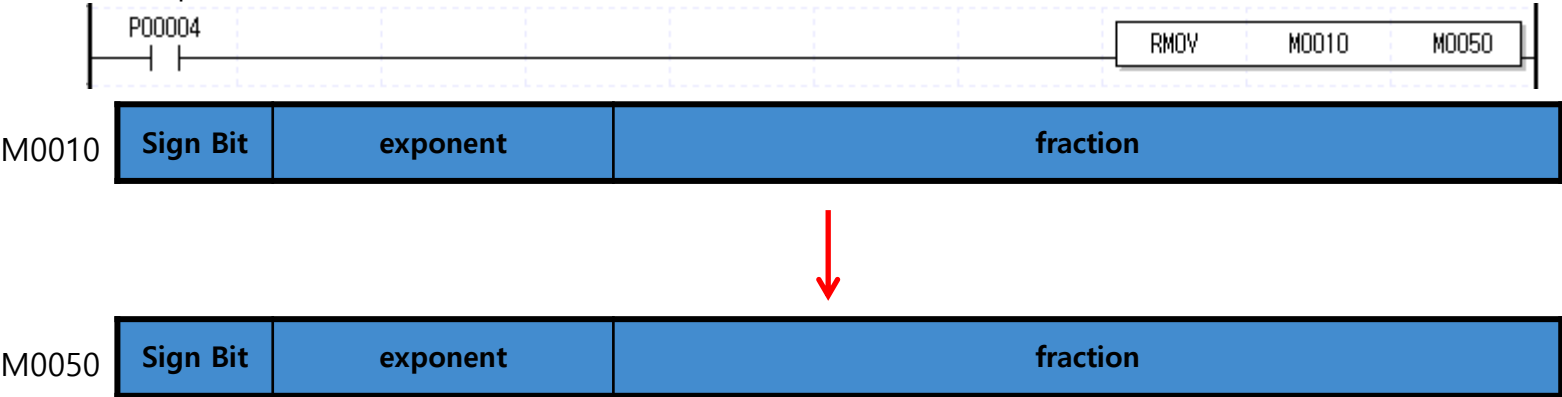
Real data is 32Bit floating decimal point data, where 32Bit floating decimal point data is called Single Real. Expression is available only in decimal format (decimal point displayed). And both Word device and Bit device are available. (M,D etc.)

① Data range

Bit	Data Range
32 Bits	-3.402823466e+038 ~ -1.175494351e-038 or 0 or 1.175494351e-038 ~ 3.402823466e+038



② Preview example: When P00004 is on, M0010's Real data is moved to M0050's Real data.



2. Data Structure

[Data Type (LReal Type)]

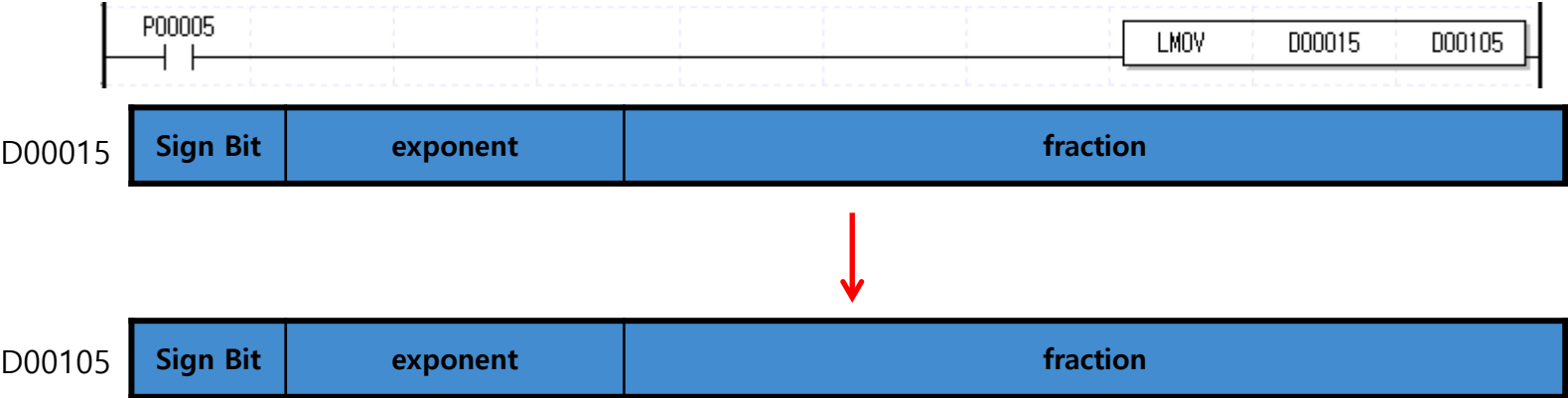
Real data is 64Bit floating decimal point data, where 64Bit floating decimal point data is called Single Real. Expression is available only in decimal format (decimal point displayed). And both Word device and Bit device are available. (M,D etc.)

① Data range

Bit	Data Range
64 Bits	-1.7976931348623157e+308 ~ -2.2250738585072014e-308 or 0 or 2.2250738585072014e-308 ~ 1.7976931348623157e+308



② Preview example: When P00005 is on, D00015's Long Real data is moved to D00105's Long Real data.



2. Data Structure

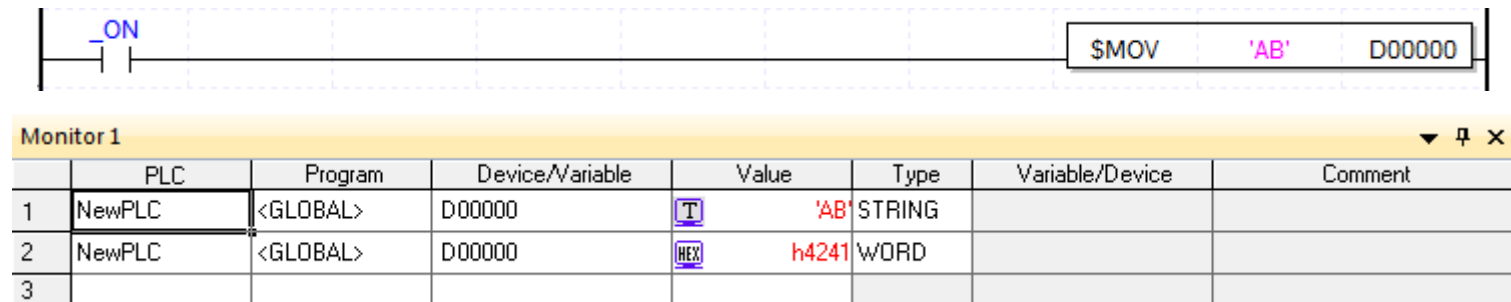
[Data Type (STRING Type)]

Among application instructions, string related instructions use the data type of number, alphabet, special sign, etc. to save in ASCII code. In addition, Korean and Chinese letters which need 16-Bit code also can be used. String data up to NULL code (h00) is regarded as one string row. And the maximum length of a string row is 32 bytes (including NULL). In other Words, up to 31 letters are available in English only. If directly input string's size exceeds the maximum limit, a warning message will be displayed in the programming tool of XG5000 to keep such string from input. Data of maximum string input is of 31 bytes + NULL (1 byte).

① How to use String

Byte	How to use String data
Up to 32 Bytes (Including NULL)	Use Single quotation marks (') on the side of characters.

② Preview example: Let's put 'AB' characters to D00000.



The screenshot shows a ladder logic program with a normally open contact labeled 'ON' connected to a coil labeled '\$MOV' with the value 'AB' and destination 'D00000'. Below the program is the 'Monitor 1' window, which displays the following data:

	PLC	Program	Device/Variable	Value	Type	Variable/Device	Comment
1	NewPLC	<GLOBAL>	D00000	T 'AB'	STRING		
2	NewPLC	<GLOBAL>	D00000	HEX h4241	WORD		
3							

ASCII Table		
Hexa	Decimal	
41	065	A
42	066	B
43	067	C

※ Reference

- \$MOV: If &MOV instruction used, string up to 16 Words will be transferred unconditionally.
- String(ASCII) configured to BYTE is saved to low byte of Word area first. So, 'AB' is expressed to 'h4241' on HEX.

1. PLC

3. Devices

3. Devices

[Bit Device]

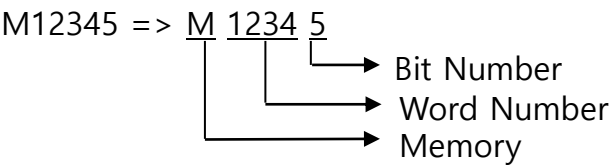
It is available to express the Bit without a '.' (dot) when used in basic instructions as LOAD or OUT. Also, it supports P, M, K, F, L for Bit devices basically. First, we will look into how to express Bit memory with Bit device and Word device. After that, the description of P, M device would be included on the next page, but the other Bit memory would be described briefly on the next page.

Device	Bit contact	Remarks
P	P00000 ~ P1023F (16,384)	Input / Output
M	M00000 ~ M1023F (16,384)	Auxiliary relay
K	K00000 ~ K4095F (65,536)	Keep relay
F	F0000 ~ F1023F (16,384)	System flag
L	L00000 ~ L2047F (32,768)	Link relay

※ XGB-CPUH

3.1.1 Bit device expression

1) The expression of Bit memory (P, M, K, L, F)



Slot	Module	Comment	Input Filter	Emergency Out	Allocation
0(main)	XBC-DN32U (DC 24V		3 Standard[ms]	Default	P0000 ~ P003F
1(Internal)	XBF-AH8E (Input 4CH		-	-	P0040 ~ P007F
2	XBE-DC16A/B (DC 24		3 Standard[ms]	-	P0080 ~ P011F
3	XBE-RY16A (RELAY		-	Default	P0120 ~ P015F
4	XBL-EMTA		-	-	P0160 ~ P019F
5	XBF-RD04A (RTD, 4-		-	-	P0200 ~ P023F
6					

2) I/O allocation

Example of I/O allocation

Allocation of I/O No. is to allocate the address to each module's I/O terminals to read the data from the input modules and output the data in the output modules when executing operation. In the XGB PLC, all modules occupy 64 points. The basic unit occupies 2 slots of No.1 so 124 points are allocated and all remaining expansion module occupies 64 points. (including special, communication modules)

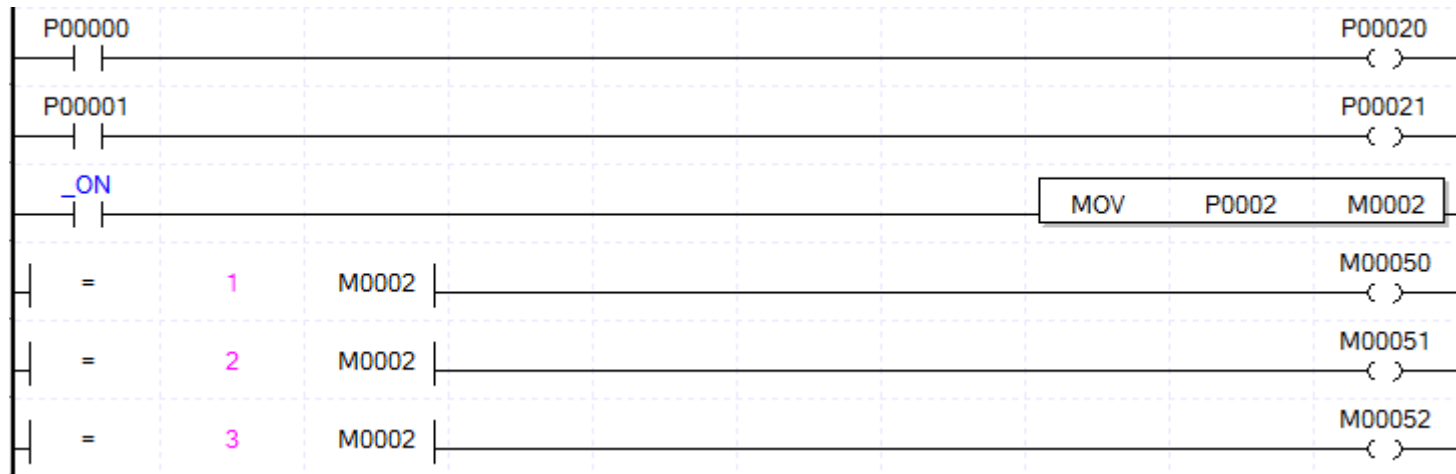
3. Devices

[Input / Output (P)]

I/O P, as the area equivalent to external equipment is composed of push button used as input device, input section to receive signals of switch or limit switch, solenoid used as output device, and output section to deliver operation result to motor and lamp. As for input section P, since input status is kept in PLC's internal memory, contact A and B are available to use. And as for output section P, contact A and B are also available. Other sections than used for I/O in P area can be used just like the auxiliary relay M. According to instructions applied, it can be used in Word unit.

1) Example of I/O Program

- When P00000 is on, M00050 is on.
- When P00001 is on, M00051 is on.
- When both P00000 and P00001 are on, M00052 is on.



※ Reference

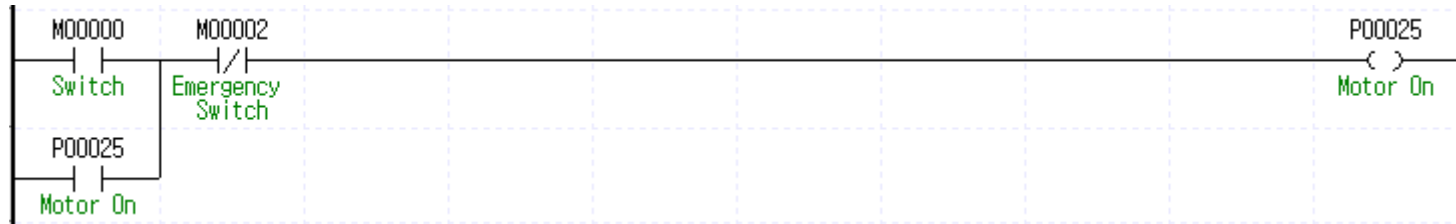
Be careful input and output's contact are assigned all to P area regardless of the status, which may cause error due to confusion between input P area and output P area when programming.

3. Devices

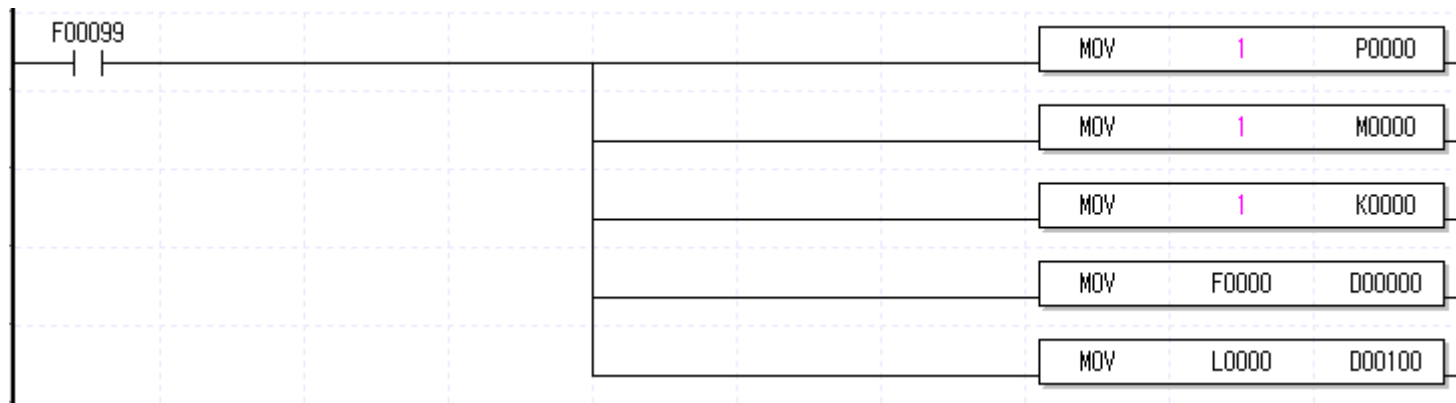
[Auxiliary relay (P)]

As an internal relay inside PLC, direct external output is impossible, but if connected with I/O P, it will be then possible. When power is on or RUN, other areas than specified as latch area by parameter setting will be all eliminated to 0. A and B contacts can be used.

1) Example of Self hold circuit with Auxiliary relay



In case that P, M, K, F, L device are used on Word position of instruction, it would be expressed up to 4 digit automatically. But, be careful about the overlap between Bit and Word area. (Ex. M0000F is the last (16th) Bit of M0000)



※ Reference

Data Register (D) is in Word device.

3. Devices

[Other Device]

1) Keep relay (K)

Its application purpose is identical to the auxiliary relay M. However, when power is on or RUN, act like a latch area 1 in basic parameter used as latch area to preserve the previous data. A and B contacts be available. The data will be eliminated by the following operation.

	How to reset keep relay K
①	Making a delete program and execute a delete program.
②	Execute a function to delete memory of the PLC delete menu in XG5000.
③	Reset key operating of CPU module or overall reset by XG5000.

2) Special relay (F)

This relay provide for information of System. It can be Read up to F0000~F2047. It provides for overall information of PLC current status, O/S information, RTC data and System clock etc. Next area of F1024 (F200 in case of XGB) Word is possible limited Write use of private instruction. This area can be use inspection of external device warning and error. Refer to CPU user's manual chapter 6.7.(in case of XGB, Chapter 6.6)

3) Link relay (L)

The area is for communication module use of flag area when communication module installed. It provides the information of communication module (O/S information, service information, flag information). It preserves the data identically to the operation characteristic of latch area 1. If communication module is not used, it can be used identically to the auxiliary relay M.

3. Devices

[Word Device]

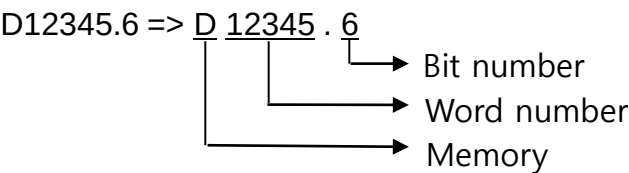
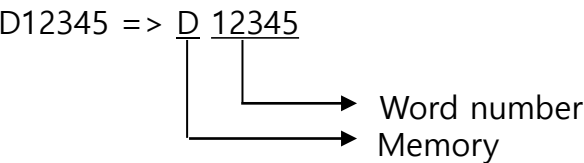
Basic expression of device is in Word unit, and a '.' (dot) is used to specify the device number's desired Bit position. Applicable device is D, R, U. First, we will look into how to express Word memory with Word Device. After that, the description of D device would be included on the next page, but the other Word memory would be described briefly on the next page.

Device	Word data	Remarks
D	D00000 ~ D10239	Data register
R	R00000 ~ R10239	File register (RAM: 2 blocks)
U	U00.00~U0A.XX	Special Module register

※ XGB-CPUH

3.2.1 Word device expression

- 1) The expression of Word memory (P, M, L, K, F, D, R(ZR), Z)
- 2) The Bit expression of Word memory (D, R, U)



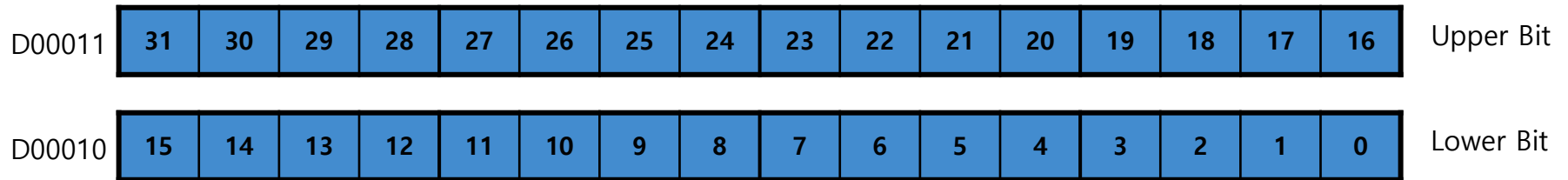
※ Reference
When P, M, K is used as Word memory, Address will be looked on 4 digit.

3. Devices

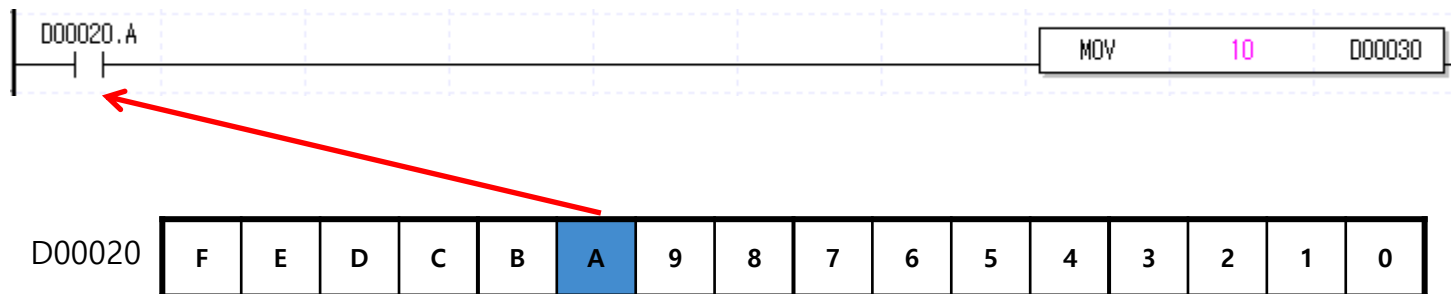
[Data Register]

It preserves internal data, where Read/Write is available in 16 Bits, 32 Bits and Bit by Bit in addition with Bit expression. As for 32 Bits, specified number is processed in the lower 16 Bits, and specified number + 1 is processed in the upper 16 Bits. Bit expression in data register uses the format of "Specified number. Specified Bit". At this moment, specified Bit is expressed in hexadecimal. When power is on or RUN, other areas than specified as latch area by parameter will be all eliminated to 0, and the latch area will be kept as before.

① If 32-Bit instruction is used with D10 specified.



② Example of expression of data register D's Bit



3. Devices

[Other Devices]

1) File Register (R)

File register is exclusively used for internal flash memory. Flash memory can not save the data when processing scan program, since it takes a little time to save the data. So scan program using the flash memory data move to the file register. If need to save the data, save to the flash memory again.

① Characteristics

Device	Word Data	Remarks
R	RAM: R00000 ~ R32,767	File Register ※ XBM-DN32H2

- As an exclusively used register for internal flash memory, it is used to read or write a block of internal flash to the file register.
- The size of one block*1) is 16K Word, identical to that of the block of internal flash memory.
- Write the data of file register in flash memory with EBWRITE instruction to keep the data permanently.
- File register operates same as latch area 1. Namely, data is eliminated by Overall reset with reset switch, reset with D.CLR and reset with XG5000.
- It will need several scans to read or write the block of file register to the block of flash memory.
- Completion state can be checked via the Bit of applicable block of F160 (_RBLOCK_RD_FLAG) and F162 (_RBLOCK_WR_FLAG).
- Both index function and indirect setting are available. At this time, indirect setting range for ZR is up to ZR0~ZR32767 Words, and Index function ([Z]) range available is -32768~32767 among device number of ZR. As for R, both indirect setting and index function are all available in the specified block range. If applicable block range is exceeded, index-exceeded error occurred

※ Reference

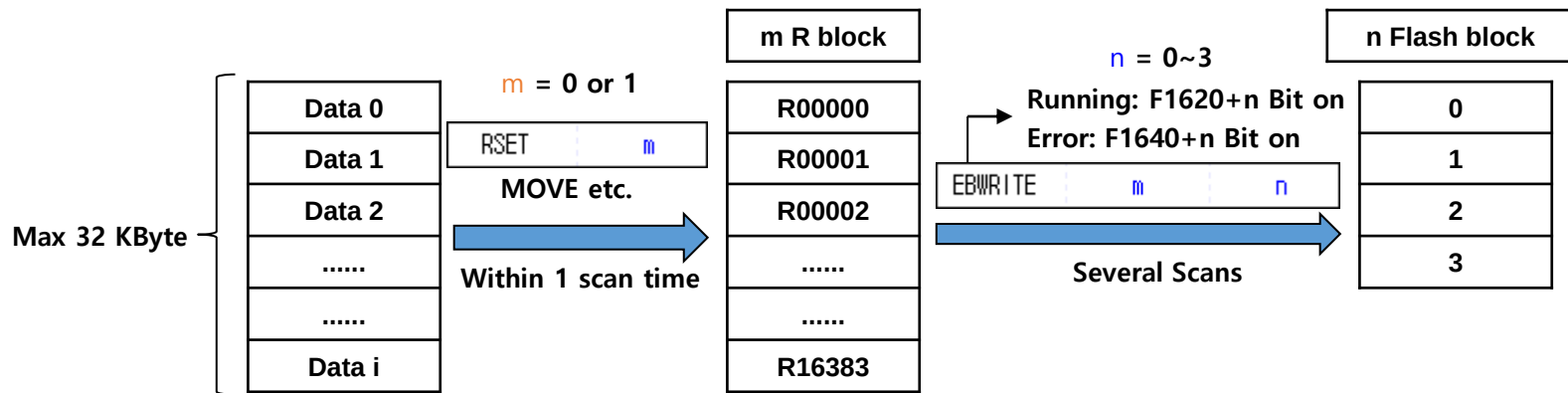
EBREAD, EBWRITE are exclusive flash instructions. Refer to manual related instructions for details.

3. Devices

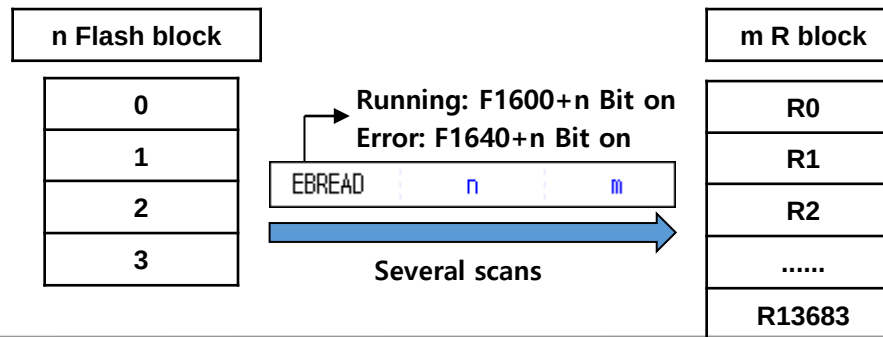
[How to use 'File register ®' for Flash Memory]

Device	Word Data	Remarks
R	FLASH: 32KByte x 4 blocks (128Kbytes)	Flash Memory

① The process for saving block data to flash memory



② The process for reading block data from flash memory,



3. Devices

[How to use 'File register ®' for Flash Memory]

③ Expression of File register

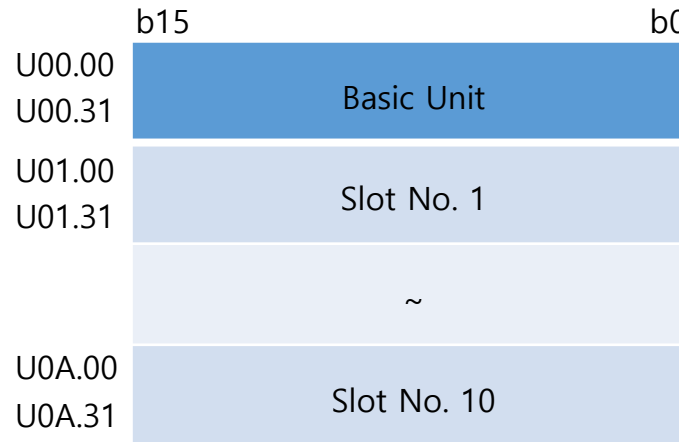
- R - File register block unit expression (16K Word fixed per block)
- ZR – Whole file register expression (range depends on unit type)
- Flash area has no device name, accessible by exclusive instruction.

Device name	Bit expression	Word expression	Write	Read	#	[Z]	Data preserved
R	O	O	O	O	O	O	Level of latch 1 area
ZR	X	O	O	O	O	O	Level of latch 1 area
Internal Flash	X	X	Exclusive instruction	Exclusive instruction	X	X	Permanently

3. Devices

[Special module register (U)]

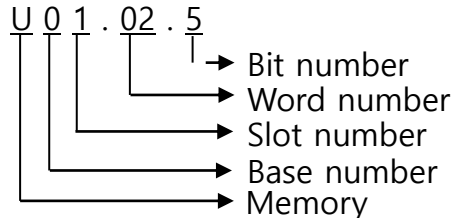
This register is used to read data from special module installed on slot. Data of special module installed by back-plane controller will be automatically updated in refresh area. 32 Words per slot are assigned to U area. Thus, in the high performance XGB case, 352 Words in total ((1 Basic Unit + 10 slots) * 32 Words = 352 Words). U area value used per slot is fixed regardless of slot which module is installed on or which is empty.



① Characteristics

Device	Word data (Bit data)	Remarks
U	U00.00.0 ~ U0A.31.F	Special module register

Basic expression of U area is in Uxy.z, where x is for base number 0~ 7, y is for slot number 0 ~ F, and z is for internal memory's Word number of special module. U01.02 =>



3. Devices

[Timer (T)]

3 types of basic cycle available are 1ms, 10ms and 100ms, whose operation method is different respectively based on 5 kinds of instructions (Ton, TOFF, TMR, TMon, TRTG). Maximum setting value is available up to hFFFF (65535) in decimal or in hexadecimal.

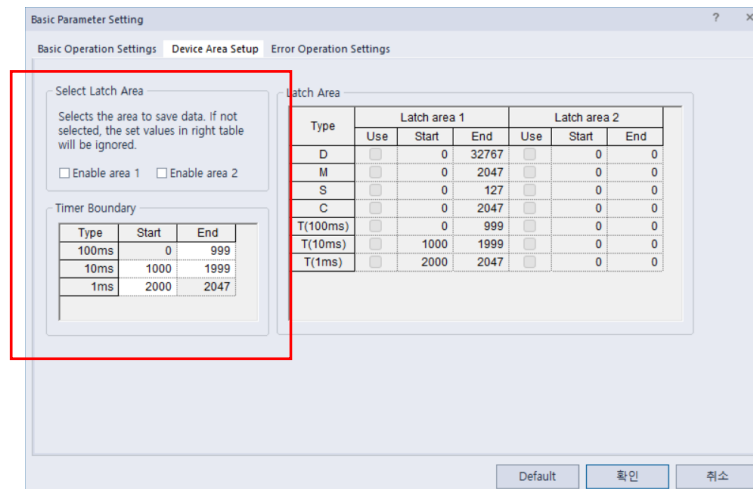
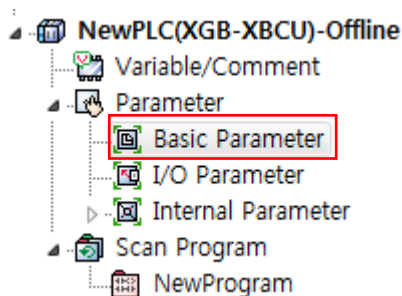
① Characteristics

Device	Word data	Remarks
T	100ms, 10ms, 1ms: T0000 ~ T2047 ※ XBM-DN32H2	Dedicated device for timer

② How to set time of (T) device.

The timer (T) device could be set to 100ms, 10ms, 1ms along to Timer Boundary. It is set in the Basic parameter.

- XG5000 > Basic parameter > Select latch area > Timer boundary



※ XBM-DN32H2

3. Devices

[Counter (C)]

The count starts at Rising Edge (Off → on) of input condition and stops if reset input then to eliminate the present value to 0 or to replace it with setting value. Operation methods are different from each other based on 4 kinds of instructions (CTU, CTD, CTUD, CTR), with maximum setting value available up to hFFFF.

① Characteristics

Device	Word data	Remarks
C	C0000 ~ C2047	Dedicated device for counter ※ XBM-DN32H2

[Step Control Relay (S)]

As a relay used to control step, it is divided into Last-input preferred and sequence control, based on instructions (OUT, SET) applied. When power is on or RUN starting, other areas than specified by parameter will be all eliminated to the first step of 0.

① Characteristics

Device	Word Data	Remarks
S	S00.00 ~ S127.99	Step control relay

1. PLC

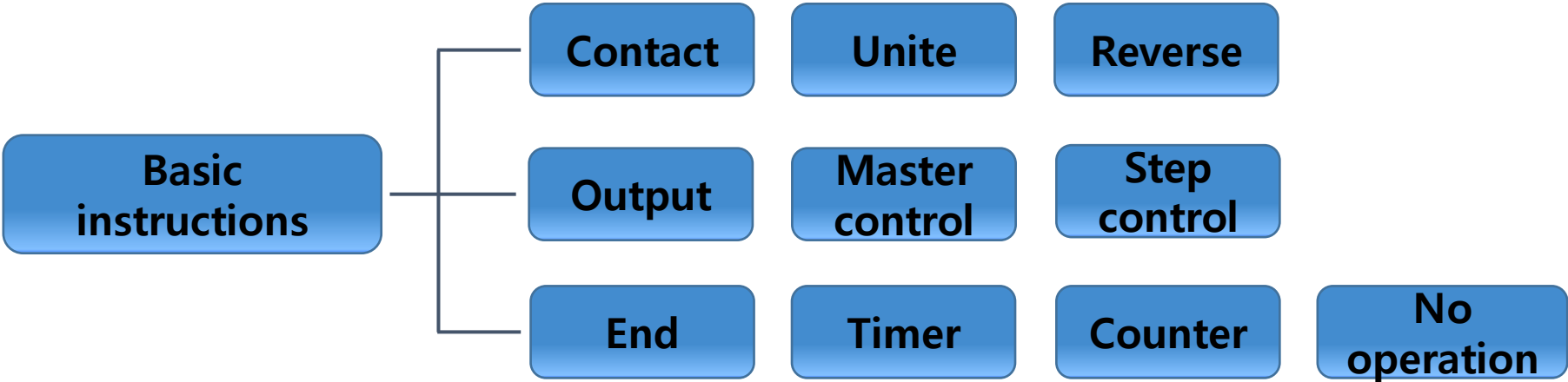
4. Instructions

4. Instructions

If you use the contacts and coils in the PLC, you can read or write to Bits data. However, you need to handle a variety of data during programming; Nibble, Byte, Word, Double Word, Long Word as well as Bit. Therefore, to use these data for calculate, you need to use instruction (instructions). This chapter describes instructions which would be used on your programming. First, you can look description for instructions structure briefly, then basic and application instructions.

4.1 Basic instructions

Basic instructions are composed of contact /coil related instruction such as LOAD/OUT, timer/counter, master control and step control instruction. In this chapter, some basic instructions will be described: contact, output(coil), timer, counter and step control.



4. Instructions

4.1.1 Contact

1) Contact instructions

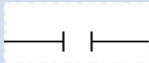
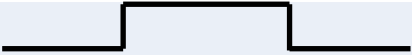
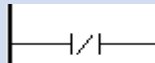
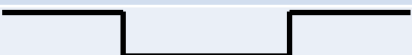
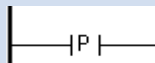
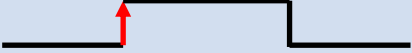
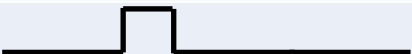
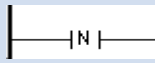
Classification	Designations	Symbol	Description	Basic steps
Contact point	LOAD		A contact point operation start	1
	LOAD NOT		B contact point operation start	1
	AND		A Contact Point Series-Connected	1
	AND NOT		B Contact Point Series-Connected	1
	OR		A Contact Point Parallel-Connected	1
	OR NOT		B Contact Point Parallel-Connected	1
	LOADP		Positive Convert Detected Contact Point	2
	LOADN		Negative Convert Detected Contact Point	2
	ANDP		Positive Convert Detected Contact Point Series-Connected	2
	ANDN		Negative Convert Detected Contact Point Series-Connected	2
	ORP		Positive Convert Detected Contact Point Parallel-Connected	2
	ORN		Negative Convert Detected Contact Point Parallel-Connected	2

※ Reference

The number of Basic Steps means the case that indirect specification, index formula and direct variable input were not used. In other words, it represents the minimum number of the steps of the applicable instruction. The number of steps depends on indirect specification, index formula and pulse application used.

4. Instructions

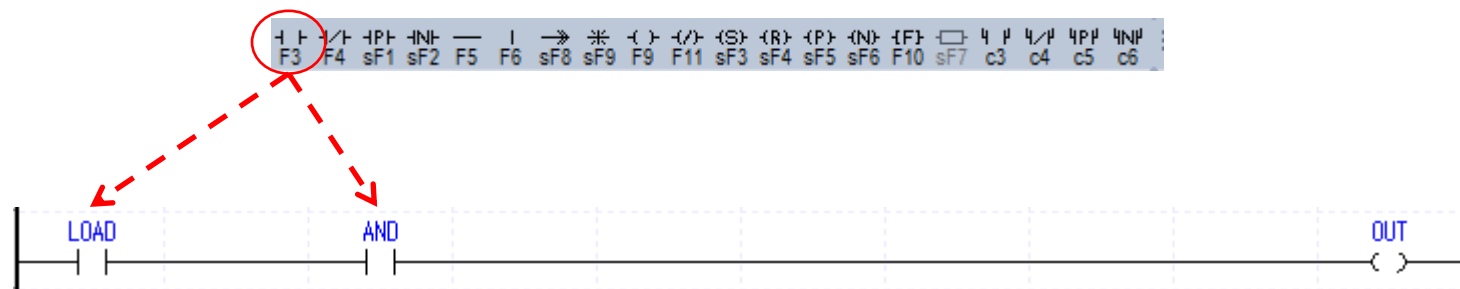
2) Contact expression

Sign	Name	Time chart	
	A contact	Input signal	
		A contact	
	B contact	Input signal	
		B contact	
	P contact	Input signal	
		P contact	
	N contact	Input signal	
		N contact	

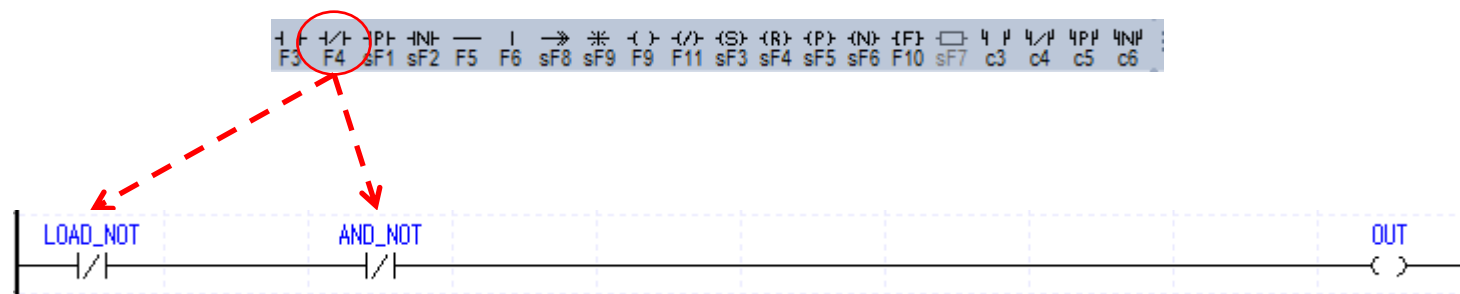
4. Instructions

3) Practice example

① LOAD, AND



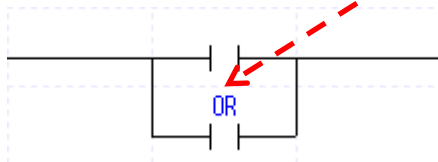
② LOAD NOT, AND NOT



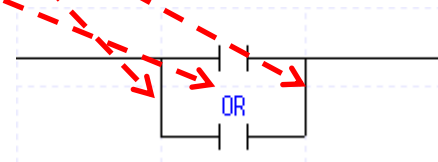
4. Instructions

③ OR

I F3 F4 sF1 sF2 F5 F6 sF8 sF9 F9 F11 sF3 sF4 sF5 sF6 F10 sF7 c3 c4 c5 c6

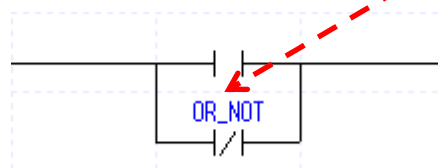


I F3 F4 sF1 sF2 F5 F6 sF8 sF9 F9 F11 sF3 sF4 sF5 sF6 F10 sF7 c3 c4 c5 c6

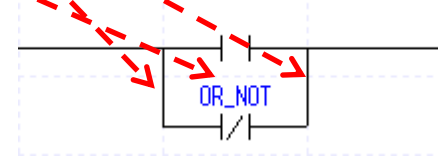


④ OR NOT

I F3 F4 sF1 sF2 F5 F6 sF8 sF9 F9 F11 sF3 sF4 sF5 sF6 F10 sF7 c3 c4 c5 c6

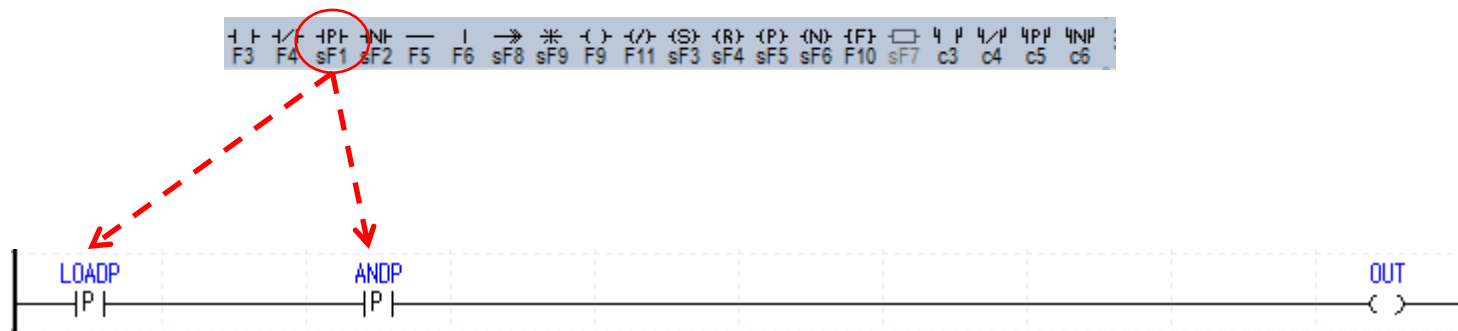


I F3 F4 sF1 sF2 F5 F6 sF8 sF9 F9 F11 sF3 sF4 sF5 sF6 F10 sF7 c3 c4 c5 c6

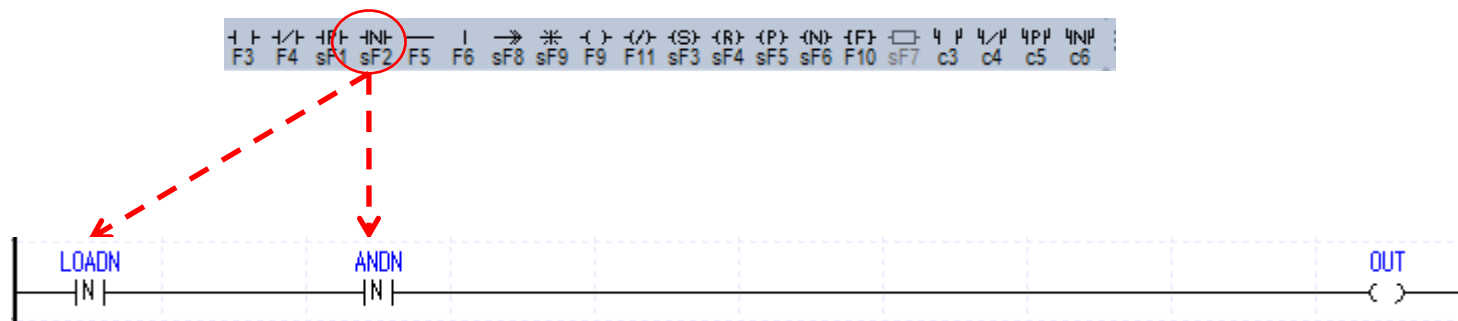


4. Instructions

⑤ LOADP, ANDP



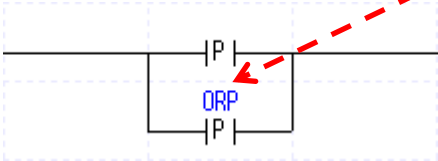
⑥ LOADN, ANDN



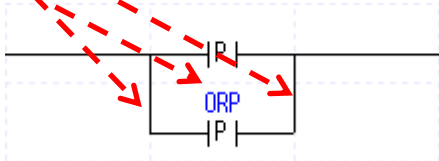
4. Instructions

⑦ ORP

LS F3 F4 sF1 sF2 F5 F6 sF8 sF9 F9 F11 sF3 sF4 sF5 sF6 F10 sF7 c3 c4 c5 c6

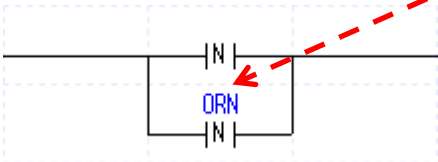


LS F3 F4 sF1 sF2 F5 F6 sF8 sF9 F9 F11 sF3 sF4 sF5 sF6 F10 sF7 c3 c4 c5 c6

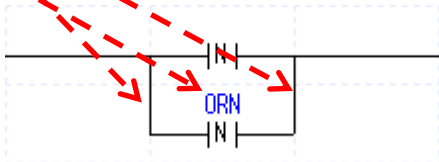


⑧ ORN

LS F3 F4 sF1 sF2 F5 F6 sF8 sF9 F9 F11 sF3 sF4 sF5 sF6 F10 sF7 c3 c4 c5 c6

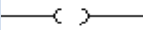
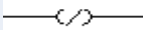
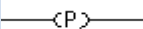
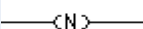
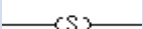
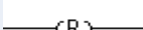


LS F3 F4 sF1 sF2 F5 F6 sF8 sF9 F9 F11 sF3 sF4 sF5 sF6 F10 sF7 c3 c4 c5 c6



4. Instructions

4.1.2 Output (Coil)

Classification	Designations	Symbol	Description	Basic Steps
Output	OUT		Operation result output	1
	OUT NOT		Operation result reverse output	1
	OUTP		1 scan output if input condition rises	2
	OUTN		1 scan output if input condition falls	2
	SET		Contact point output on kept	1
	RST		Contact point output off kept	1

※ Instruction "FF"

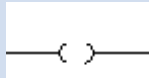
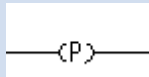
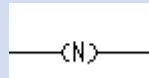
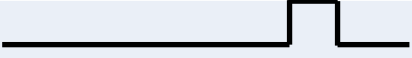
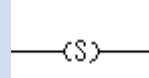
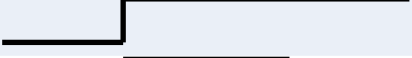
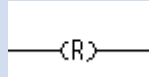
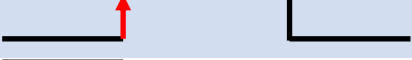
Reverses specified device's state when input contact changes Off → on by Bit output reverse instruction

Example: where P00060 state is reversed when input contact P0020 is changed from Off to on.



4. Instructions

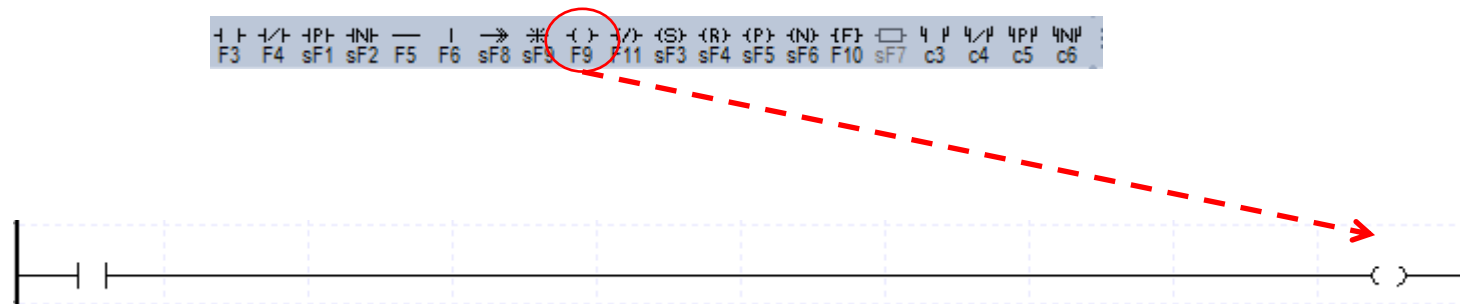
2) Coil expression

Sign	Name	Expression	
	Coil	Input signal	
		Coil	
	P Coil	Input signal	
		P Coil	
	N Coil	Input signal	
		N Coil	
	Set Coil	Input signal	
		Set Coil	
	Reset Coil	Input signal	
		Reset Coil	

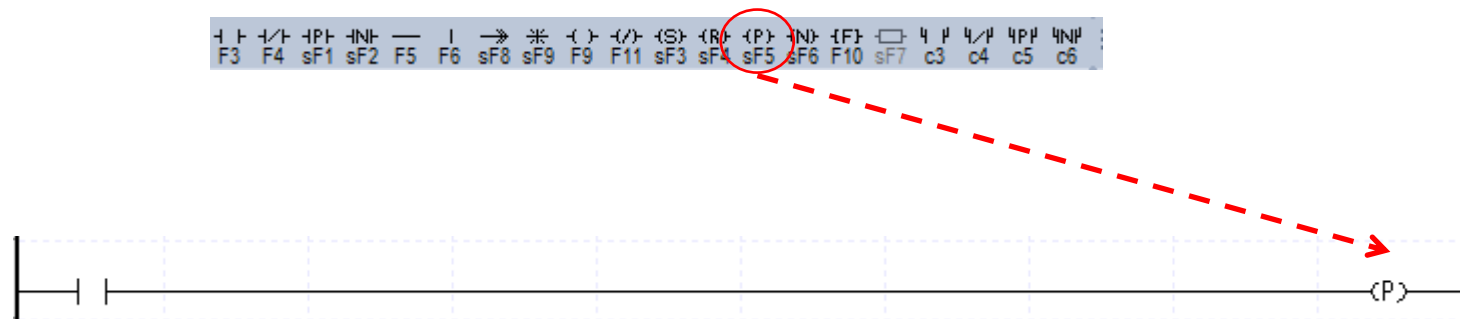
4. Instructions

3) Practice example

① Coil



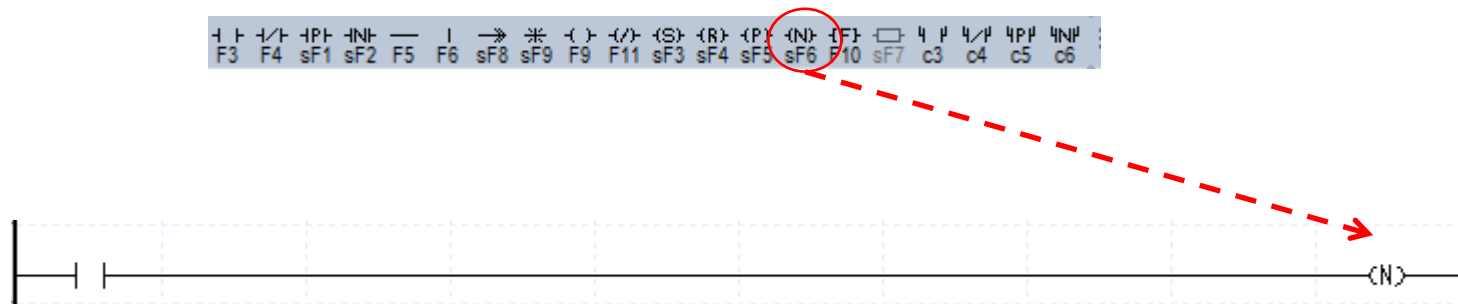
② P Coil



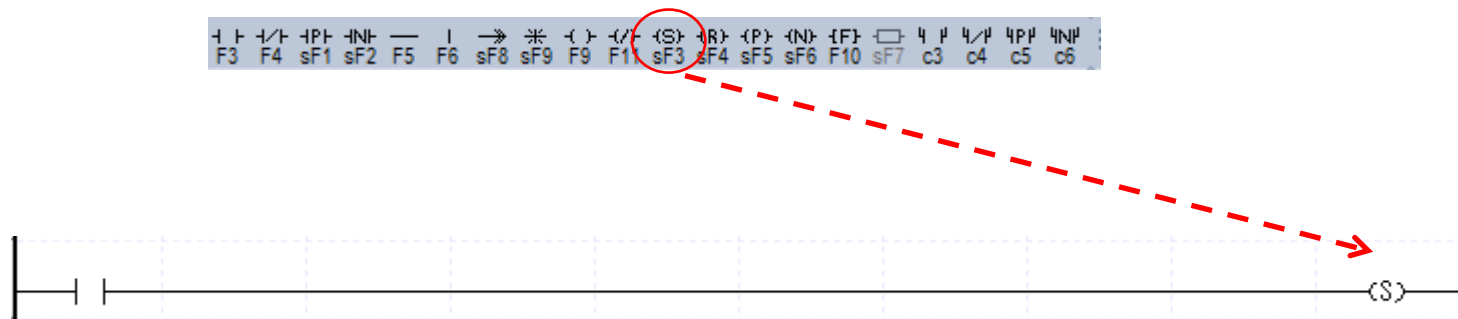
4. Instructions

3) Practice example

③ N Coil



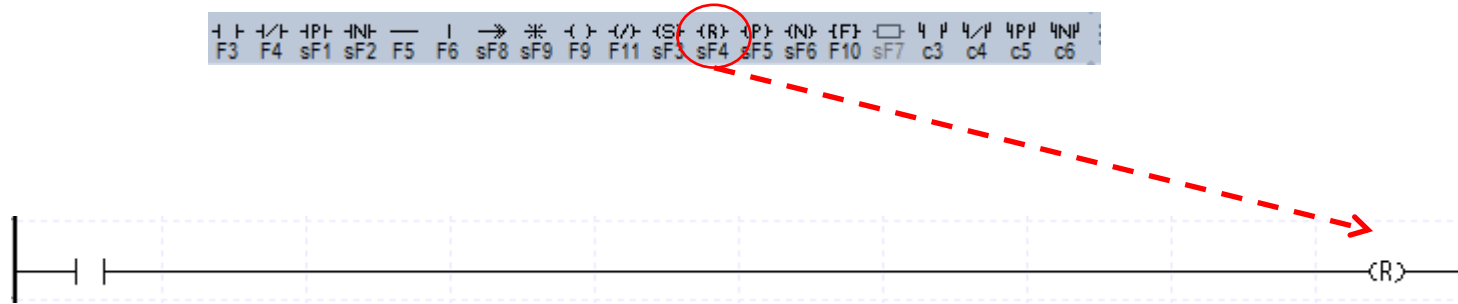
④ Set Coil



4. Instructions

3) Practice example

⑤ Reset Coil

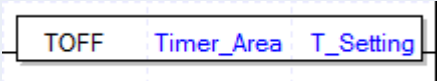
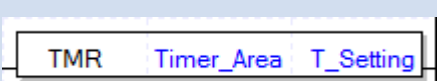
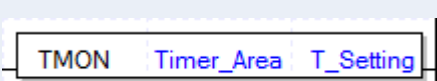
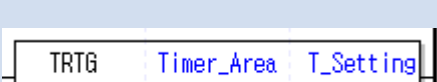


4. Instructions

4.1.3 Timer

1) Timer instructions

3 types of basic cycle available are 1ms, 10ms and 100ms, whose operation method is different respectively based on 5 kinds of instructions (TON, TOFF, TMR, TMON, TRTG). Maximum setting value is available up to hFFFF (65535) in decimal or in hexadecimal.

Classification	Designations	Symbol	Description	Basic Steps
Timer	TON		On delay timer	2
	TOFF		Off delay timer	2
	TMR		Integral timer	2
	TMON		Monostable timer	2
	TRTG		Retrigger timer	2

※ Reference

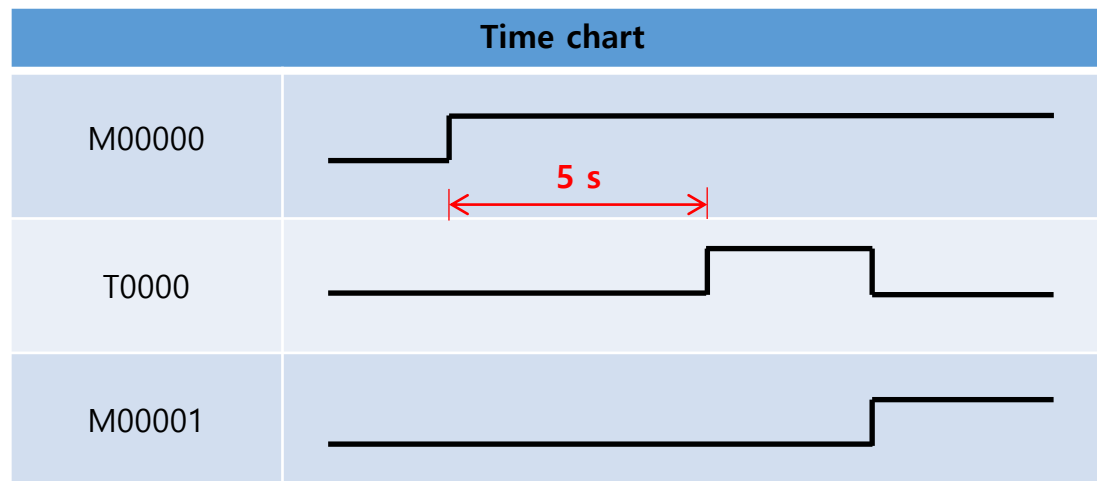
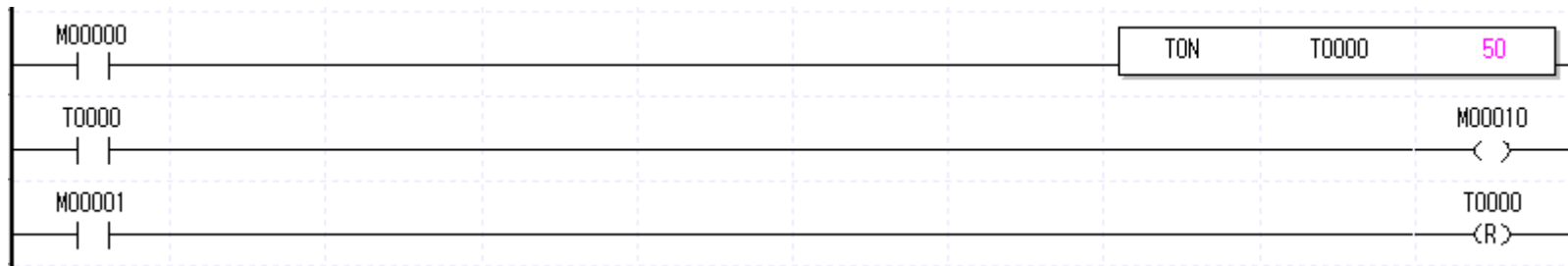
The number of Basic steps means the case that indirect specification, index formula and direct variable input were not used. In other words, it represents the minimum number of the steps of the applicable instruction. The number of steps depends on indirect specification, index formula and pulse application used.

4. Instructions

2) Practice example

① TON (On delay timer)

- When M00000 is on, the output M00010 turns on after 5 seconds.
- When M00001 is on, the T0000 is to 0 and the M00010 is off.



※ Reference

T0000's basic cycle is 100 ms as setting Timer boundary to the right.

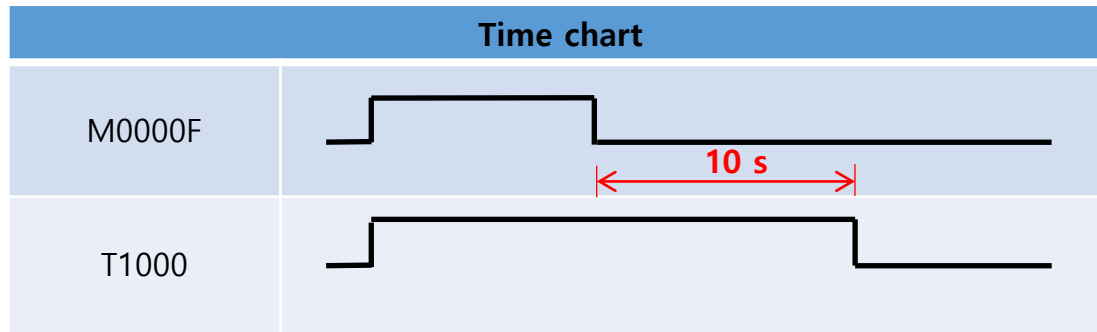
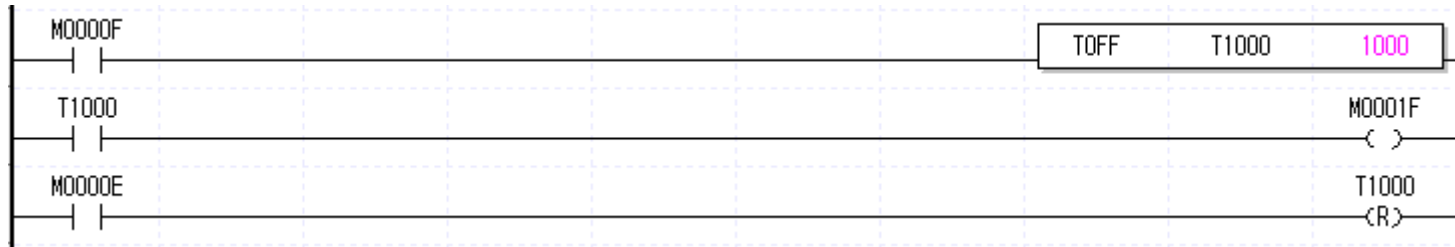
As a result, when you set it to 50, it means 5 seconds.

$$50 \times 100 \text{ (ms)} = 5000 \text{ (ms)} = 5 \text{ (s)}$$

4. Instructions

② TOFF (Off delay timer)

- When M0000F is on, the output M0001F keeps on.
- When M0000F is off, the output M0001F turns off after 10 seconds.
- When M0000E is on, the T1000 is to 0 and the M0001F is off.



※ Reference

T1000's basic cycle is 10 ms as setting Timer boundary to the right.
As a result, when you set it to 1000, it means 10 seconds.

$$1000 \times 10 \text{ (ms)} = 10000 \text{ (ms)} = 10 \text{ (s)}$$

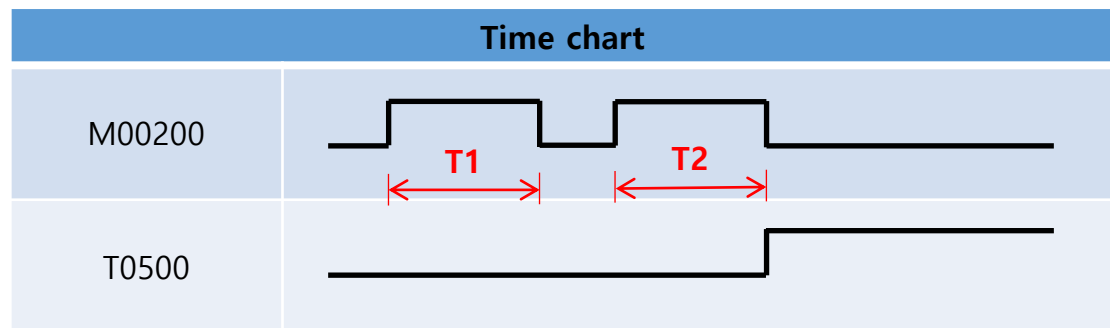
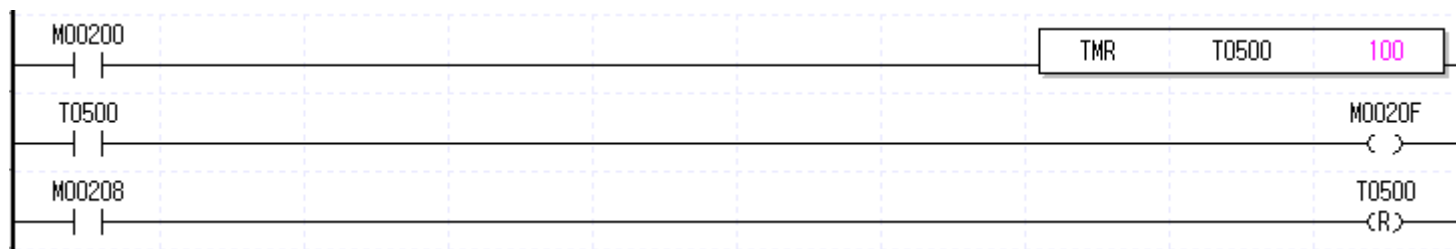
Timer Boundary

Kind	Start	End
100ms	0	999
10ms	1000	1999
1ms	2000	2047

4. Instructions

③ TMR (Integral timer)

- When the total "on" time of M00200 is 10 seconds, the output M0020F turns on.
- When M00208 is on, the T1000 is to 0 and the M0020F is off.



T1 + T2 = 10 s

※ Reference

T0500's basic cycle is 100 ms as setting Timer boundary to the right.
As a result, when you set it to 100, it means 10 seconds.

$$100 \times 100 \text{ (ms)} = 10000 \text{ (ms)} = 10 \text{ (s)}$$

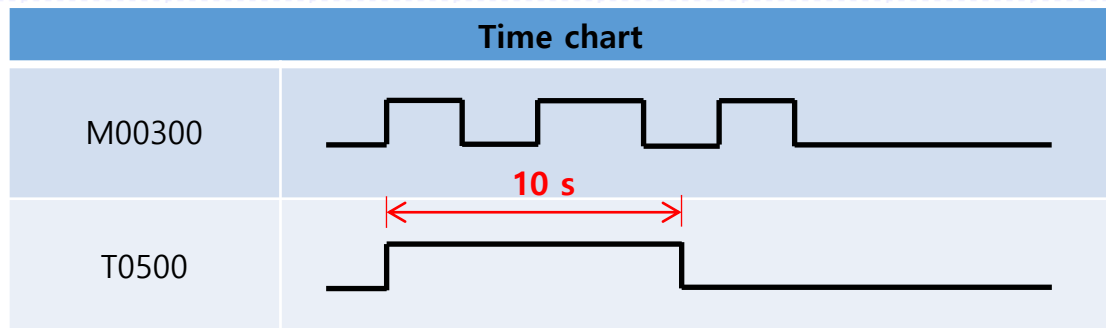
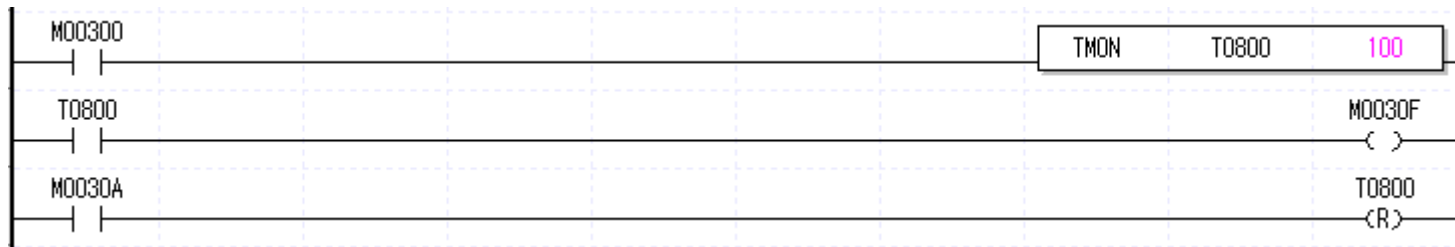
Timer Boundary

Kind	Start	End
100ms	0	999
10ms	1000	1999
1ms	2000	2047

4. Instructions

④ TMON (Monostable timer)

- When the M00300 is on, the present time is decreasing from set time (10 s).
- When T0800 is '0', M0030F is on.
- When M0030A is on, the T0800 is to 0 and the M0030F is off.



※ Reference

T0300's basic cycle is 100 ms as setting Timer boundary to the right.
As a result, when you set it to 100, it means 10 seconds.

$$100 \times 100 \text{ (ms)} = 10000 \text{ (ms)} = 10 \text{ (s)}$$

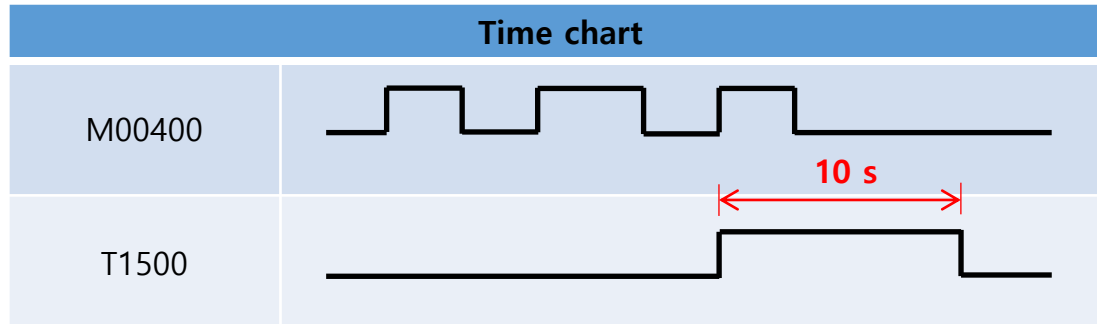
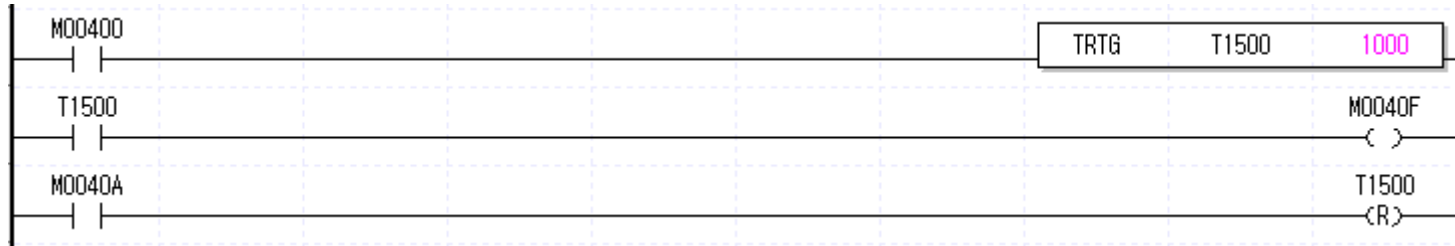
Timer Boundary

Kind	Start	End
100ms	0	999
10ms	1000	1999
1ms	2000	2047

4. Instructions

⑤ TRTG (Retrigger timer)

- When M00400 is on, the output M0040F turns on and the present time is decreasing from set time (10 s).
- When present time is "0", If M0040F turns off.
- If M00400 changes off → on again before Timer's present value is "0", Timer's present value will be reset to setting value.
- When M0040A is on, the T1500 is to 0 and the M0040F is off.



※ Reference

T1500's basic cycle is 10 ms as setting Timer boundary to the right.

As a result, when you set it to 1000, it means 10 seconds.

$$1000 \times 10 \text{ (ms)} = 10000 \text{ (ms)} = 10 \text{ (s)}$$

Timer Boundary

Kind	Start	End
100ms	0	999
10ms	1000	1999
1ms	2000	2047

4. Instructions

4.1.4 Counter

1) Counter instructions

The count starts at Rising edge (Off → on) of input condition and stops if reset input then to eliminate the present value to 0 or to replace it with setting value. Operation methods are different from each other based on 4 kinds of instructions (CTU, CTD, CTUD, CTR), with maximum setting value available up to hFFFF (65535) in decimal or in hexadecimal.

Classification	Designations	Operation	Symbol	Description	Basic Steps
Counter	CTU	Addition		Up counter	2
	CTD	Subtraction		Down counter	2
	CTUD	Addition/ Subtraction		Up/Down counter	4
	CTR	Addition		Ring counter	2

※ Reference

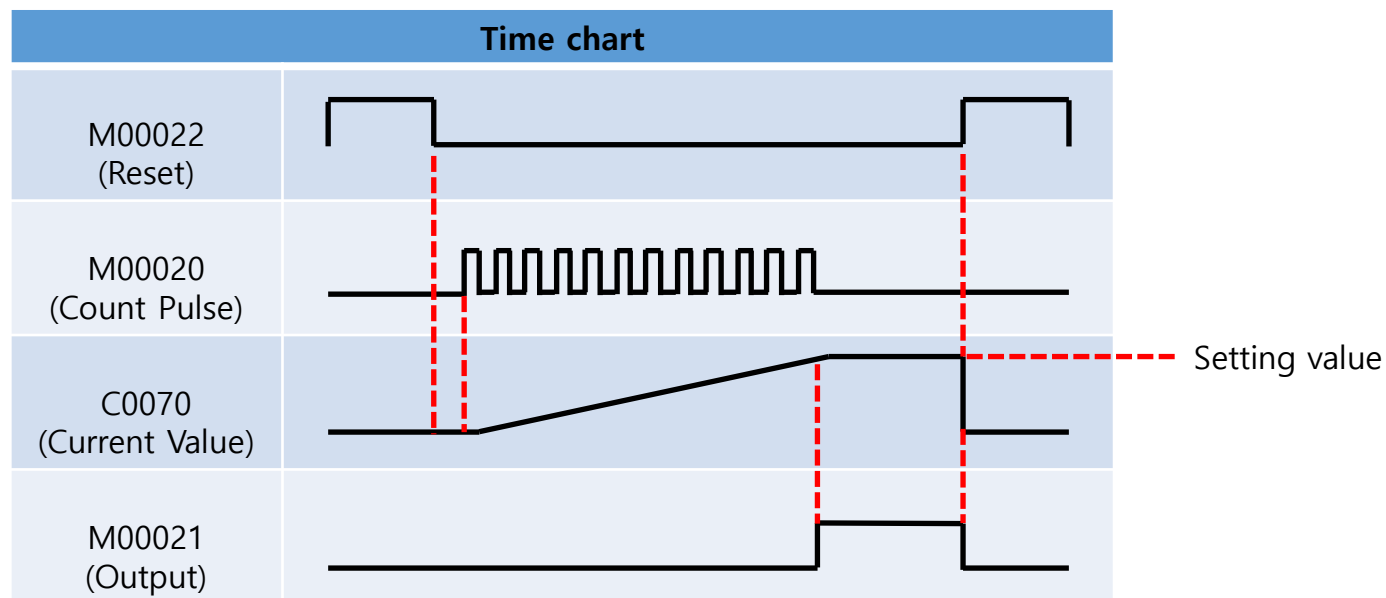
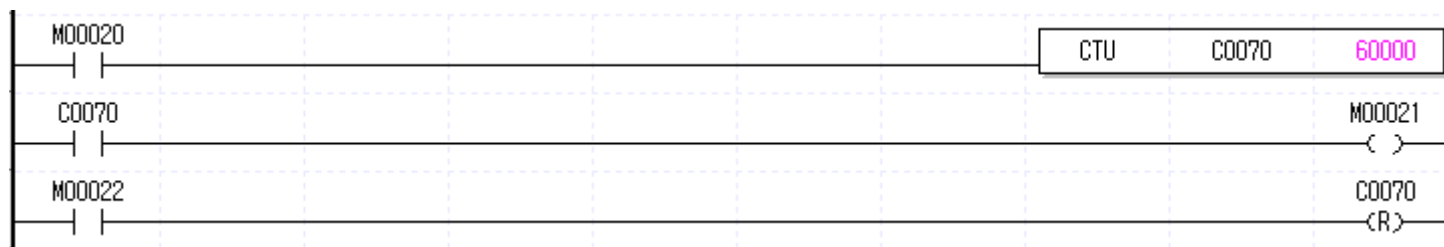
The number of Basic Steps means the case that indirect specification, index formula and direct variable input were not used. In other words, it represents the minimum number of the steps of the applicable instruction. The number of steps depends on indirect specification, index formula and pulse application used.

4. Instructions

2) Practice example

① CTU (Up counter)

- When M00020 is at rising edge, the counting number of C0070 increases by counter instructions operating.
- When C0070 reaches for 60,000, M00021 turns to on.
- When M00022 is on, C0070 is reset (to 0).

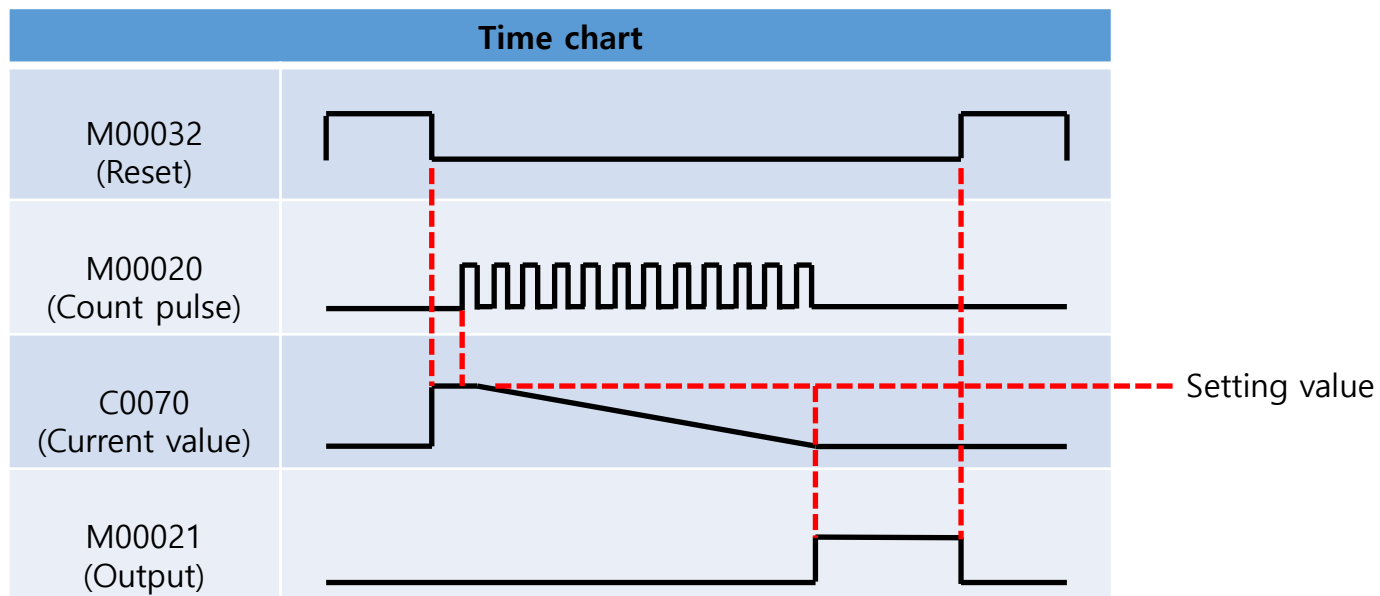
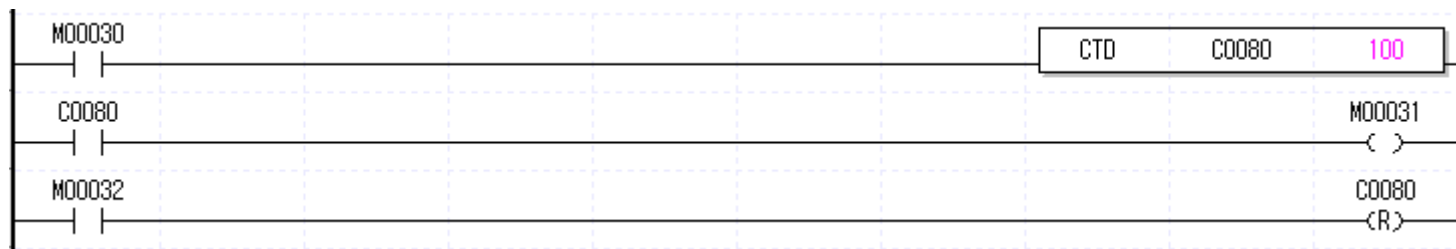


4. Instructions

2) Practice example

② CTD (Down counter)

- When M00030 is at rising edge, the counting number of C0080 decreases from 100 to 0 by counter instructions operating.
- When C0080 reaches for 0, M00031 turns to on.
- When M00032 is on, C0080 is reset (to 100).

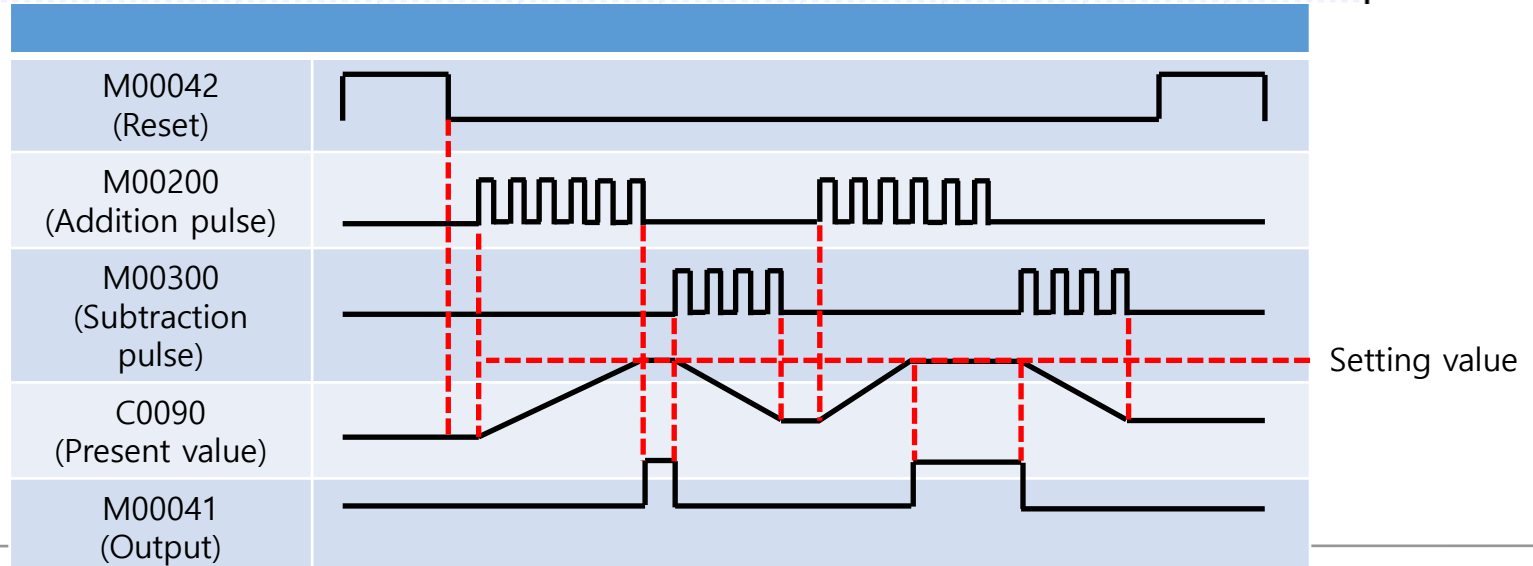
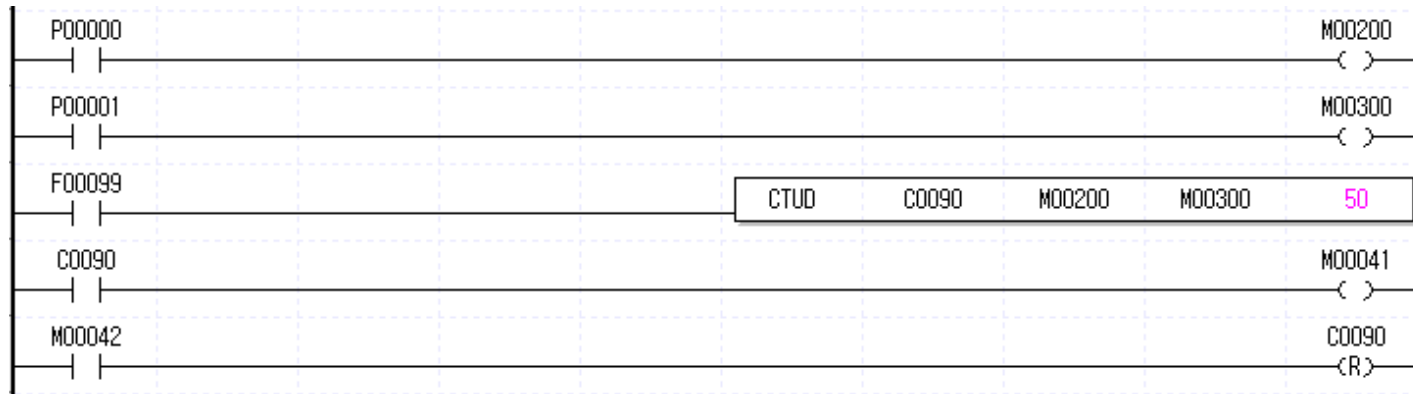


4. Instructions

2) Practice example

③ CTUD (Up/Down counter)

- When P00000 is at rising edge, the counting number of C0090 increases by counter instructions operating.
- When P00001 is at rising edge, the counting number of C0090 decreases by counter instructions operating.
- When C0090 reaches for 50, M00041 turns to on.
- When M00042 is on, C0090 is reset (to 50).

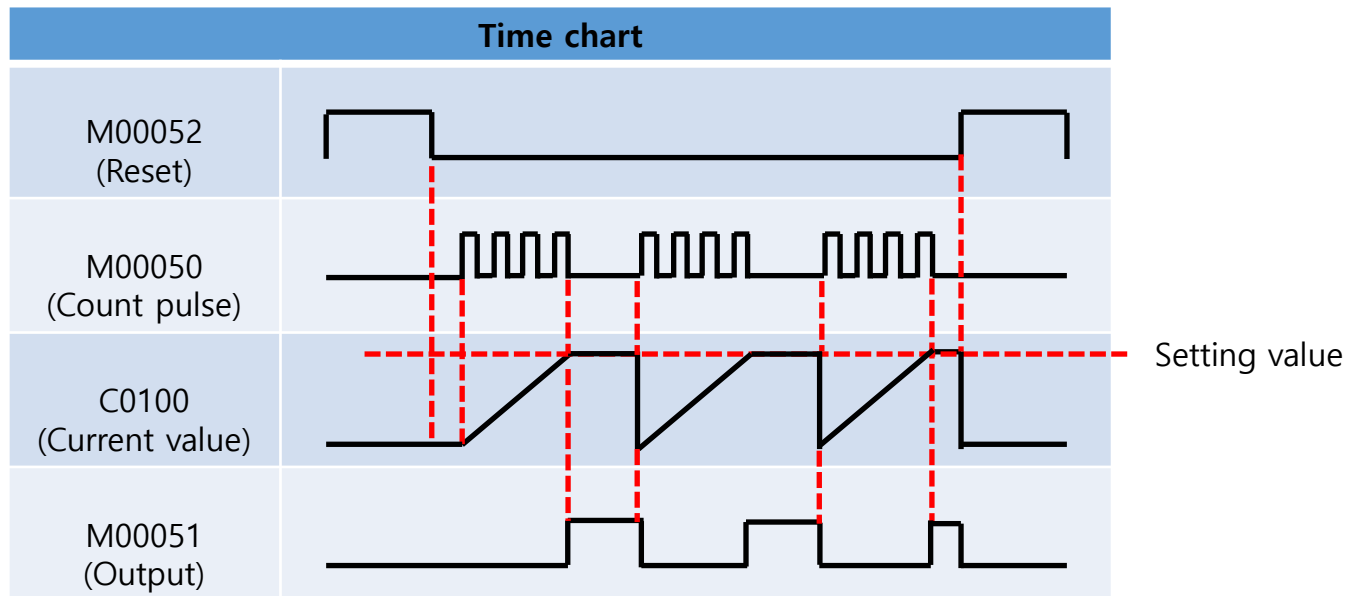
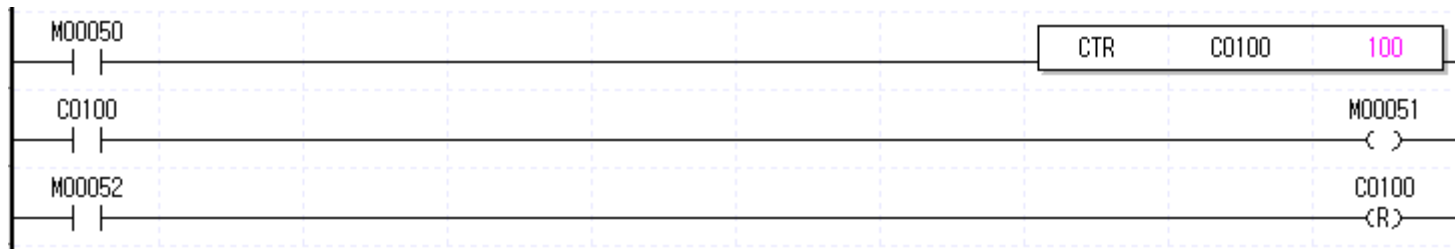


4. Instructions

2) Practice example

④ CTR (Ring counter)

- It has the counting range (0~100).
- If the counting number is over the setting value user set ("100" in this example), the C0100 is to "0".
- When M00050 is at rising edge, the counting number of C0100 increases by counter instructions operating.
- When C0100 reaches for 100, M00051 turns to on.
- When M00052 is on, C0100 is reset (to 0).



4. Instructions

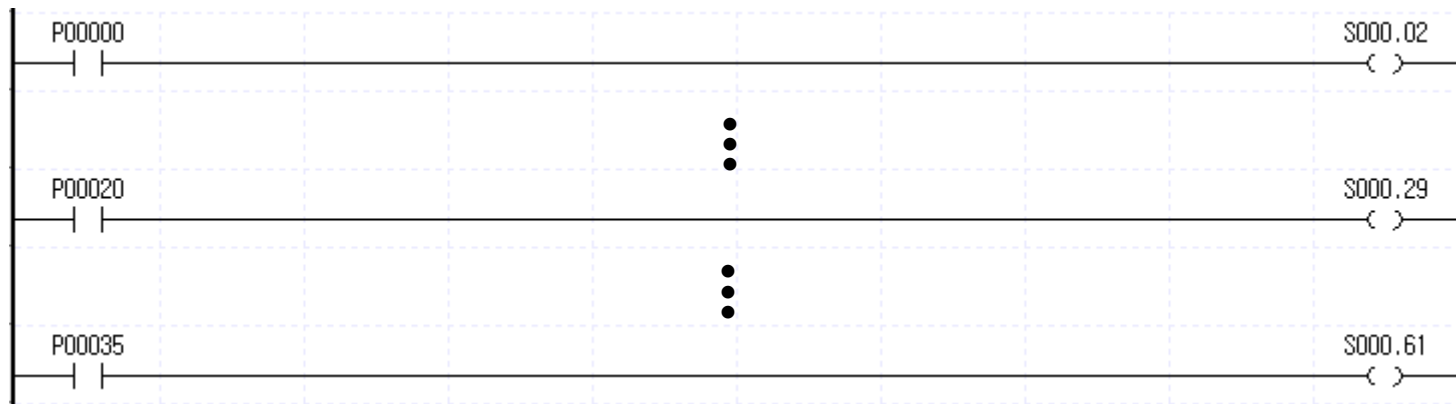
4.1.5 Step control relay

1) Step control relay (Last-input preferred/Sequence instruction)

As a relay used to control step, it is divided into Last-input preferred and sequence control, based on instructions (OUT, SET) applied. When power is on or RUN starting, other areas than specified by parameter will be all eliminated to the first step of 0.

① Last-input preferred instruction

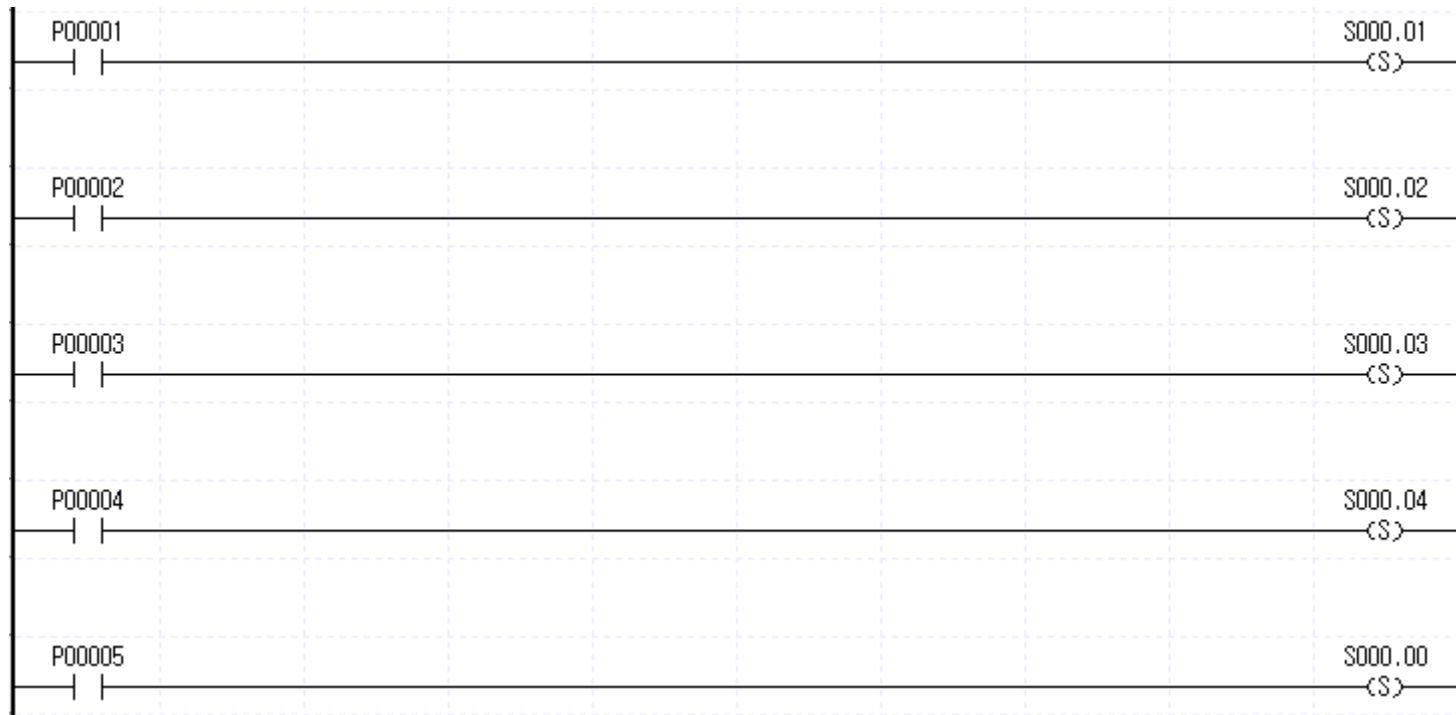
- The last programmed step is preferred under the same condition (Last-input preferred).
- Differently from SET Syy.xx, applicable step operates if input condition contact is on regardless of step sequence.
- only one step number will be on in the same group even if lots of input condition contacts are on. Finally program is on priority.
- Present step number if on will be self-held to keep on state although input contact is Off.
- OUT Syy.xx instruction will be cleared if Sxx.00's input contact is on.



4. Instructions

② Sequence instruction

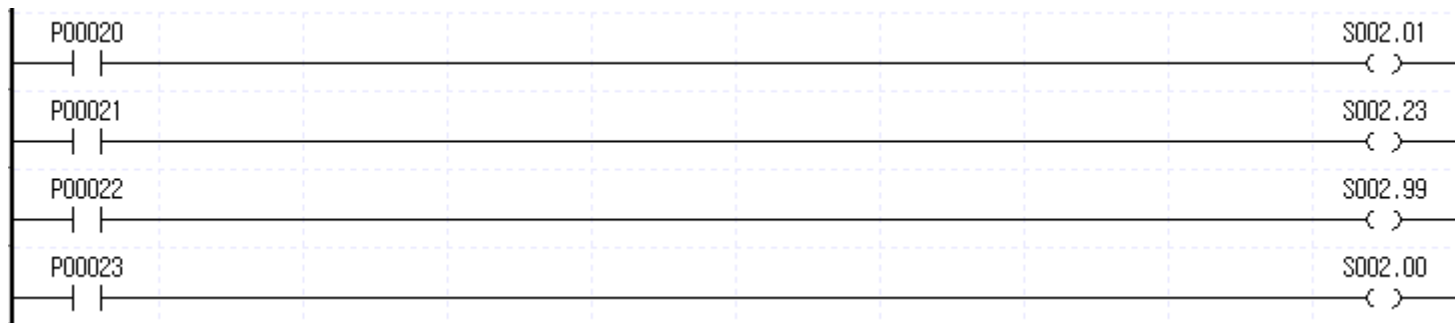
- Over after the previous step is completed, the present step will be processed by sequence control.
- In the same group with previous step number on, if present step number's input condition contact state is on, present step number will be on and the previous step number will be off.
- If present step number is on, it will be self-held to keep on state although input contact is off.
- Even if input condition contacts are on at a time, only one step number will be on in a group.
- At initial run, Syy.00 is on.
- SET Syy.xx instruction will be cleared if Syy.00's input contact is on.



4. Instructions

2) Practice example

- ① Last-input preferred instruction
- Subsequent input preferred control program with S02 group used.



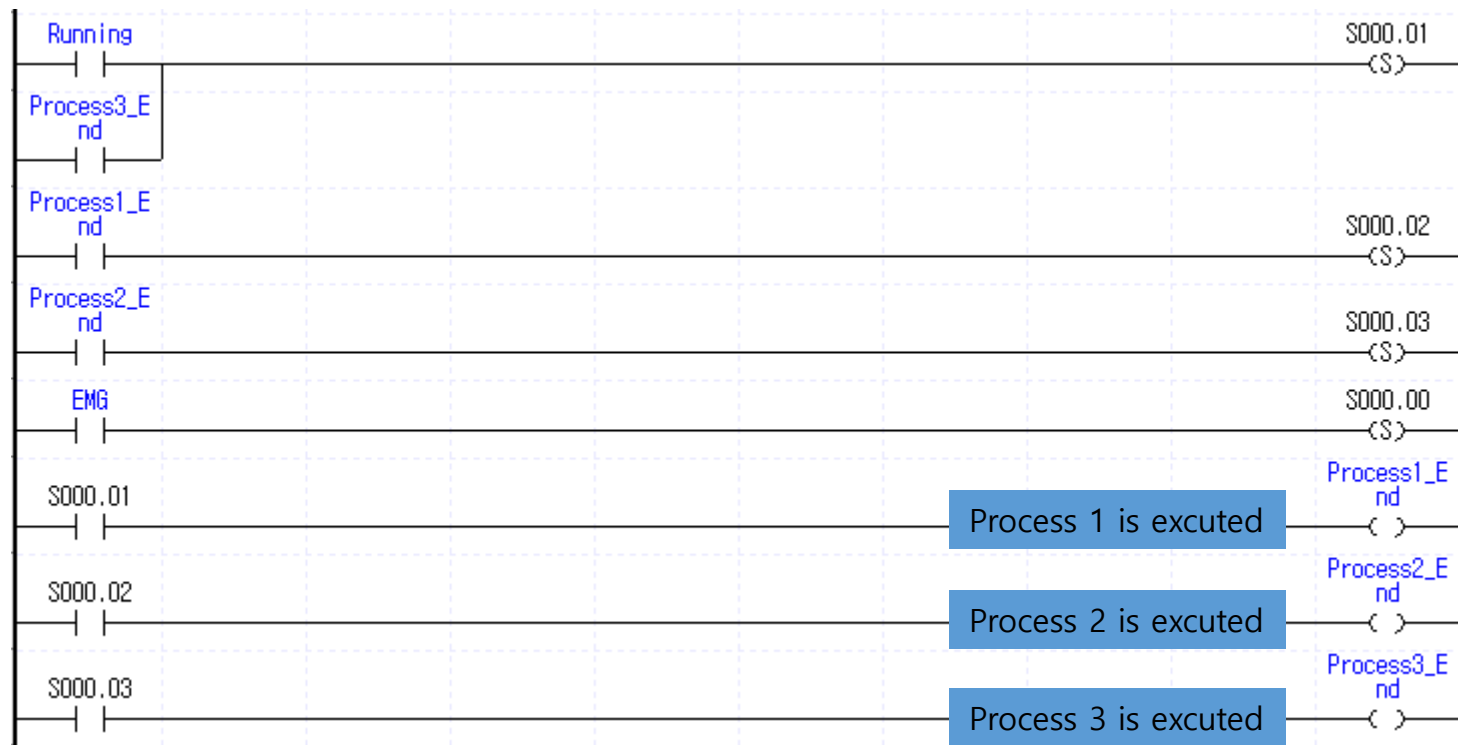
No	P00020	P00021	P00022	P00023	S02.01	S02.23	S02.98	S02.00
1	ON	OFF	OFF	OFF	ON			
2	ON	ON	OFF	OFF		ON		
3	ON	ON	ON	OFF			ON	
4	ON	ON	ON	ON				ON

4. Instructions

2) Practice example

② Sequence instruction

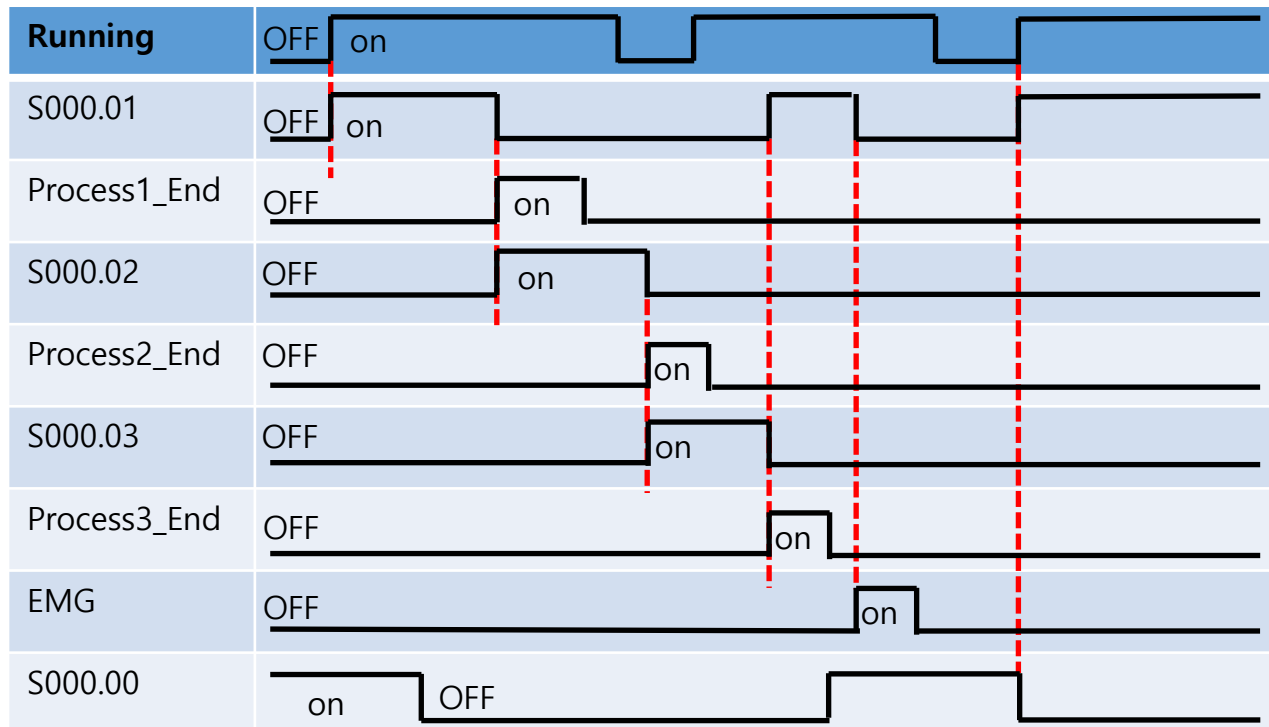
- Where process 2 is executed only after process 1 is complete, and process 1 is executed again after process 3 is complete in applicable sequence.



4. Instructions

2) Practice example

[Time chart]



4. Instructions

4.2 Application instructions

In XGB, instructions are widely classified into basic instructions, application instructions and special instructions. instructions adds the functionality of basic instructions by adding a prefix and a suffix to the basic instructions. The prefix and suffix can be used in combination to each side.



① Prefix: The prefix is added in front of the basic instructions, and support the functioning of the basic instructions. You can use up to two prefixes in one of the basic instructions.

- D: Double Word (32 Bit) Integer data operation. (ex: DMOV, D=, DADD etc.)
- R: 32 Bit real data operation. (ex: RMOV, R=, RDIV etc.)
- L: 64 Bit real data operation. (ex: LMOV, L=, LADD etc.)
- G: Group data operation. (ex: GMOV, G=, GADD etc.)
- B: Bit operation within 16 Bit. (ex: BMOV)
- \$: String data operation. (ex: \$MOV, \$ADD etc.)
- 8: 8 Bit data operation. (ex: 8=, 8<, 8<> etc.)
- 4: 4 Bit data operation. (ex: 4=, 4<> etc.)

② Suffixes: The character that support the functioning of basic instructions at the behind is called a suffix. You can use up to two suffixes to a basic instruction.

- U: Unsigned decimal integer data operation in case the basic instructions operates signed integer data. (ex: ADDU etc.)
- P: Rising edge operation of level operation. (ex: MOV, DIVP etc.)
- B: BCD data operation in case of decimal integer data operation. (ex: ADDB, MULB etc.)
- 8: 8 Bit data operation. (ex: MOV8, BCD8 etc.)
- 4: 4 Bit data. (ex: MOV4, BCD4 etc.)
- 3: Three data operation in basic instructions. (ex: =3, <>3 etc.)

4. Instructions

4.2.1 Transfer

In XGB, Transfer instruction is operating as to saving integer data to PLC's data memory and coping data to the other data memory. 'MOV', basic transfer instruction, transfers 1 Word (16 Bits) data. By adding prefix and suffixes to 'MOV' instructions, these can transfer a variety of data.

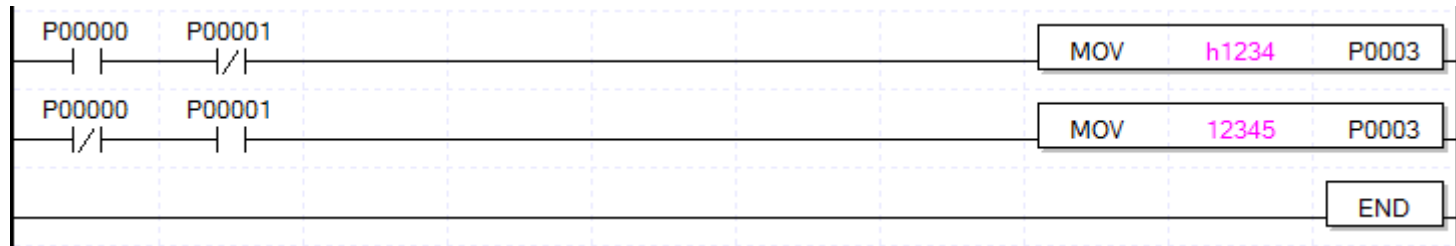
1) MOV

① [MOV S D]: Transferring 1 Word (16 Bits)

- S specified numeric data or Word device is transferred to D specified Word device.
- In case that S is specified to numeric data, it is possible to have the range h0000 ~ hFFFF.
- In case that decimal data is specified, the range is -32,768 ~ 65,535.

② Program example

- When P00000 is on, hexadecimal 1234 is displayed on BCD (P0003 Word).
- When P00001 is on, decimal 12345 is displayed on BCD (P0003 Word).



※ Reference

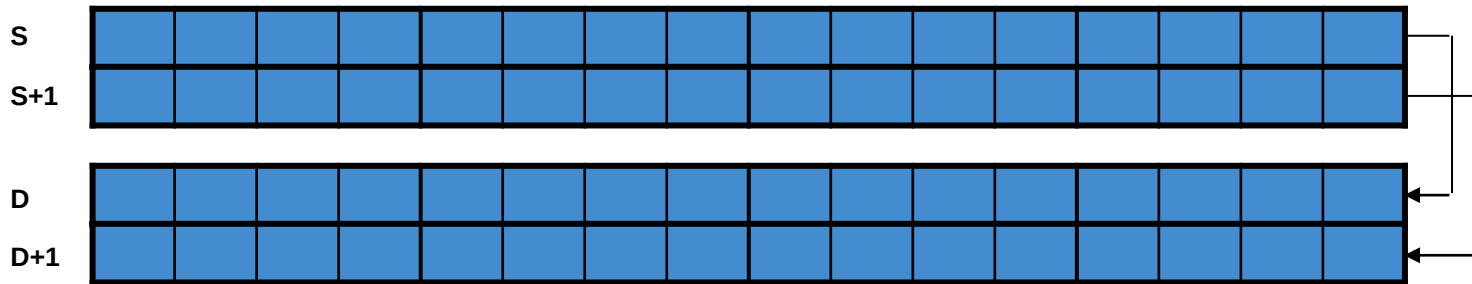
BCD(Binary Coded Decimal): It uses the h0~h9 for each digit, as the result, for the calculation in hexadecimal on BCD, h9 +h1 would be h10.
For BCD display, 1 digit of BCD needs 4 Bits.

4. Instructions

2) DMOV

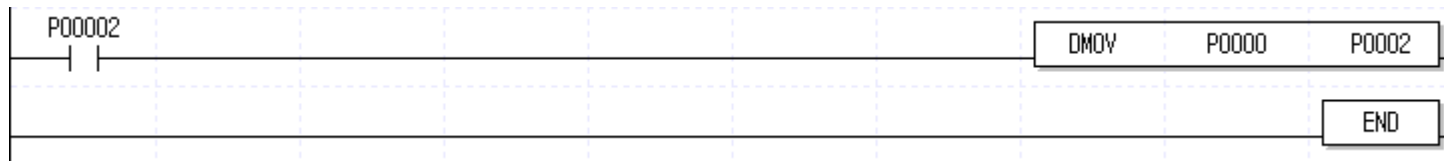
① [DMOV S D]: Transferring Double Word (32 Bits)

- S specified numeric data of double Word device is transferred to D specified Word device.
- In case that S is specified to invariable number, it is possible to have the range of h00000000 ~ hFFFFFFF.
- If S is selected as decimal number, sign does not matter whether it is plus or minus.
- If device is used for S, it needs to be Word device and it transmits selected Word data and the next Word data.
- D shall be Word device and transmitted data will be saved to selected Word device and the next Word device.



② Program example

- If P00002 switch is turned on,
- 1) display 16 points of input status connected to P00000 ~ P0000F on LED connected to P00020 ~ P0002F.
- 2) display input data of BCD switch connected to P00010 ~ P0001F on BCD display connected to P00030 ~ P0003F.

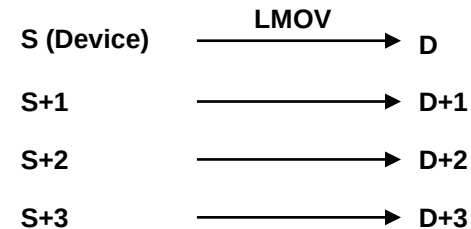
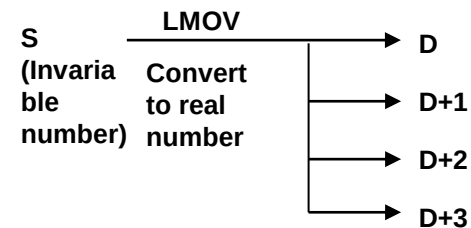
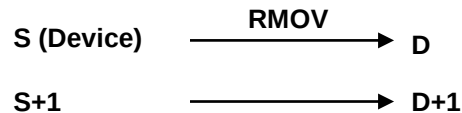
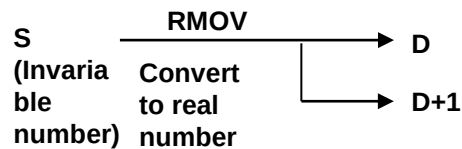


4. Instructions

3) RMOV / LMOV

① [RMOV S D] / [LMOV S D] : (Transferring Real Number) Convert S selected invariable number to real number of 32 Bit or 64 Bit and save it to D selected Word area.

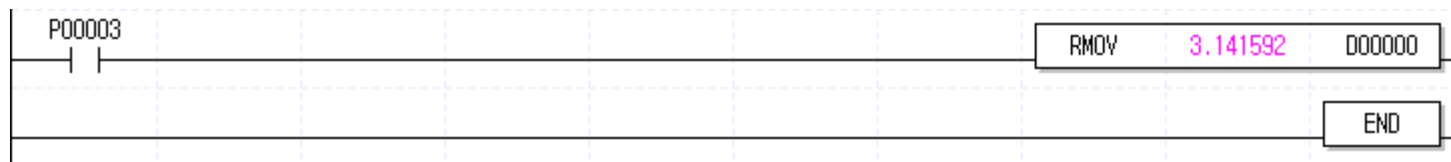
- S shall be selected as signed decimal number or real number within the limit of 32 Bit or 64 Bit real number can be handled.
- If device is used for S, data is not converted, but transfers 2 Word/4 Word from selected device to destination device.
- D shall be Word device and transmitted data will be saved to selected device for 2 Word/4 Word.



* When converting the device saved integer to real number, type conversion command (I2R, I2L, D2R, D2L, U2R, U2L, UD2R, UD2L, etc.) should be used.

② Program example

- If P00003 switch is turned on, save 3.141592 on D00000 as 32 Bit real number.

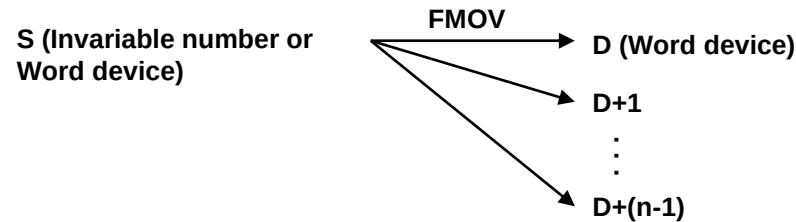


4. Instructions

4) FMOV

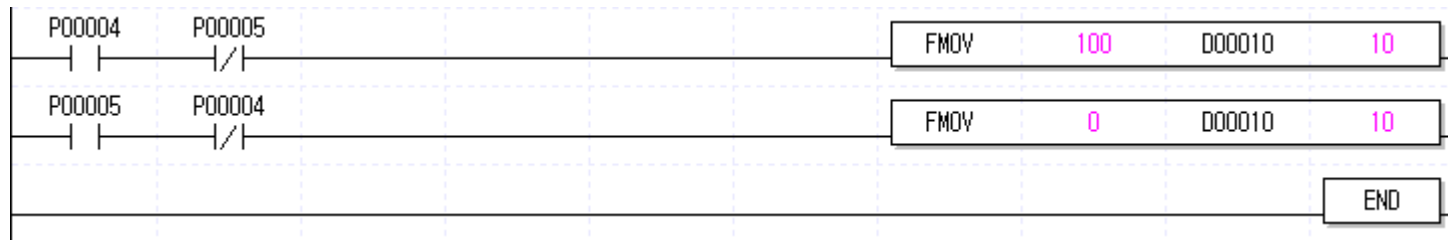
① [FMOV S D n] : (Multi-Transfer) Transfer Word data or invariable number S for N Word from D in regular order.

- In case that S is specified to invariable number, it is possible to have the range of h0000 ~ hFFFF.
- If S is selected as decimal number, it needs to use unsigned decimal number.
- N can be any between 1 ~ 65,535 and it should be used within the range of not exceeding the device area.
- If it exceeds the limit of D device, S is saved into unintended device.



② Program example

- If P00004 switch is turned on, save 100 on 10 Word device from D00010 and clear it to become 0 when P00005 switch is turned on.

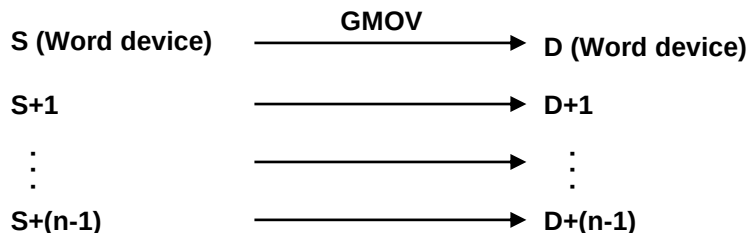


4. Instructions

5) GMOV

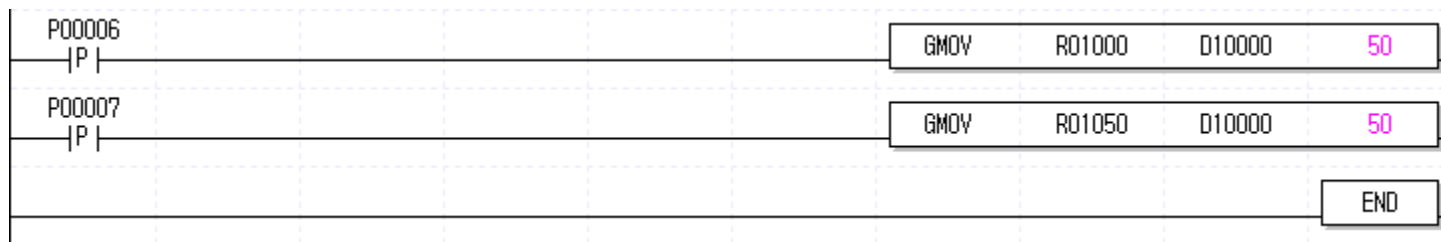
① [GMOV S D n] : (Group Transfer) Transfer Word data S for N Word from D in regular order.

- In case that S is specified to invariable number, it is possible to have the range of h0000 ~ hFFFF.
- If S is selected as decimal number, it needs to use unsigned decimal number.
- N can be any between 1 ~ 65,535 and it should be used within the range of not exceeding the device area.
- For example, in XGK-CPUH, if R30000 is used for D device, n must be less than 2,768 because last Word address for R area is R32767. (2,768 number of Words from R30000 to R32767)
- If it exceeds the limit of D device, S is saved into unintended device.



② Program example

- If P00006 switch is turned on, save 50 Word data from R01000 to 50 Word from D01000. Then, if P00007 switch is turned on, save 50 Word data from R01050 to 50 Word from D01000.

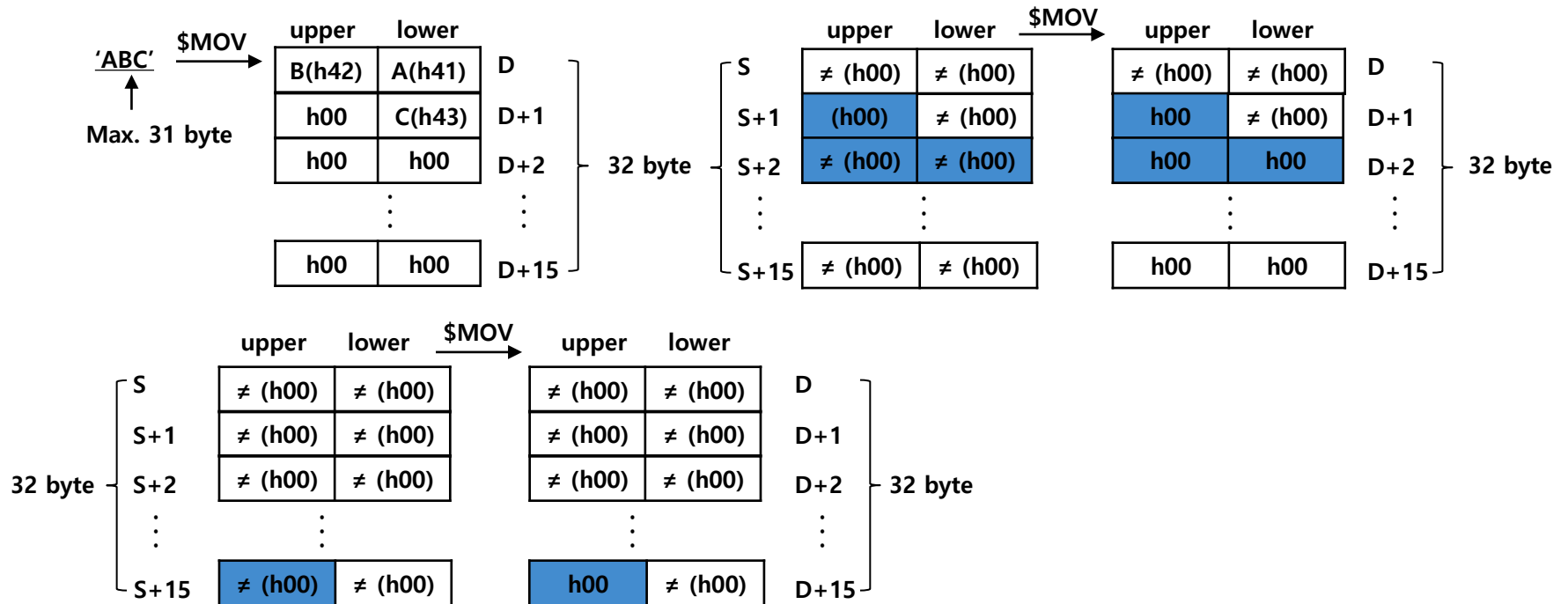


4. Instructions

6) \$MOV

① [\$MOV S D] : (Group Transfer) It transfers string starting with S to device starting with D.

- If S is specified to ASCII(string), it needs single quotation marks (' ').



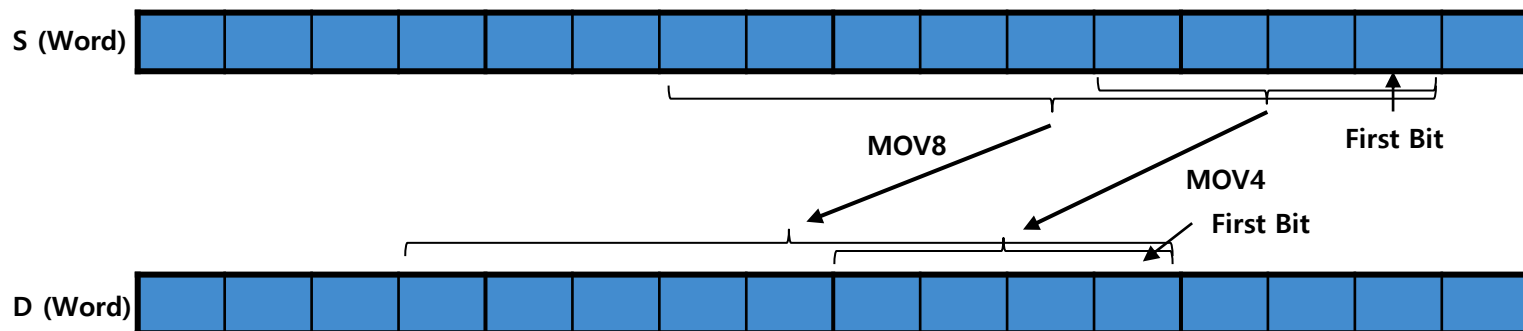
* Need to use STR instruction to change data saved in device to STRING.

4. Instructions

7) MOV4 / MOV8

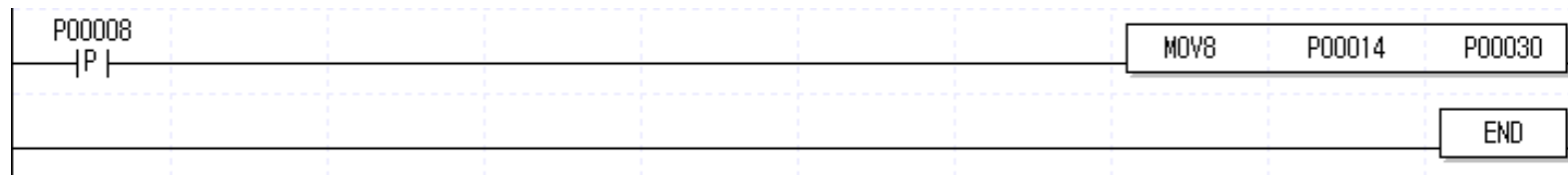
① [MOV4 S D] / [MOV8 S D] : (Transfer 4/8 Bit) It transfers 4/8 Bit starting with S to device starting with D.

- In case that S is specified to invariable number and 4 Bit, it is possible to have the range of h0 ~ hF.
- In case that S is specified to invariable number and 8 Bit, it is possible to have the range of h00 ~ hFF.
- Bit device should be selected for both S and D.



② Program example

- If P00008 switch is turned on, transfer input data of 2nd ~ 3rd digit of BCD switch connected to input contact P00010 ~ P0001F to 3rd ~ 4th digit of BCD indicator connected to P00030 ~ P0003F.

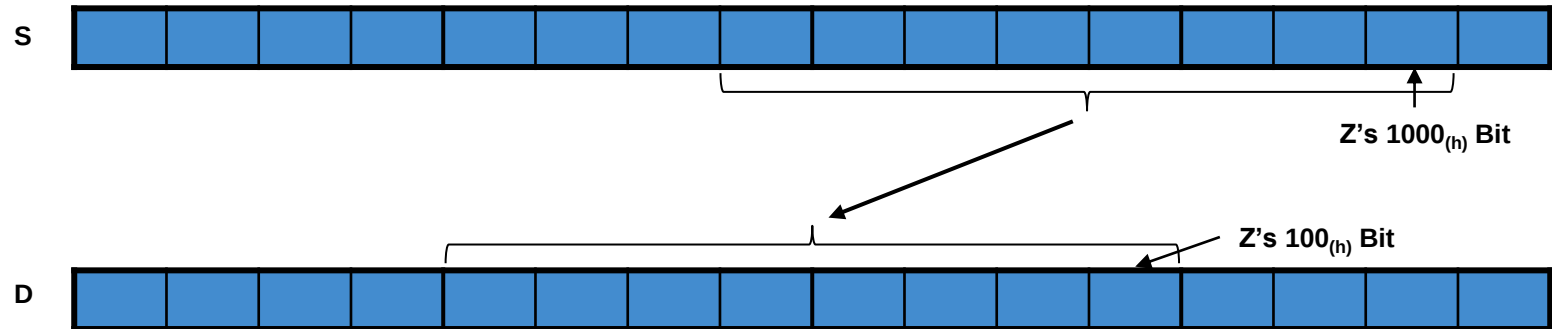


4. Instructions

8) BMOV

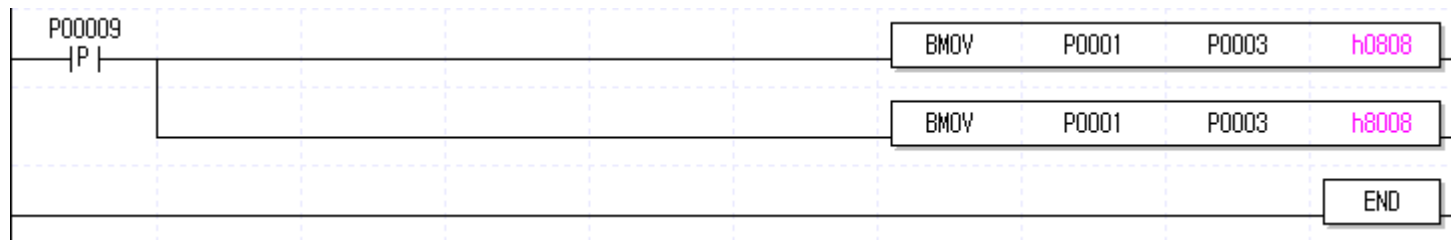
① [BMOV S D Z] : (Move Bit) By the format set in Z, specified number of Bits will be transferred to D from Word data S.

- Z's transferred Bits: up to h00 ~ h10 available.



② Program example

- If P00002 switch is turned on, switch 2 digits of input data of BCD switch connected to input contact P00010 ~ P0001F to digit of BCD indicator connected to P00030 ~ P0003F.



4. Instructions

4.2.2 Arithmetic

In XGB, basic instructions used in arithmetic are addition(ADD), subtraction(SUB), multiplication(MUL), division(DIV). These instructions need 1 Word of data and execute signed decimal calculation. However, in case of Word multiplication, it is saved to double Word. In case of division, quotient is saved to 1 Word and remainder to another 1 Word, therefore 2 Word in total.

In calculation, if the result might exceed the save area limit, resizing the data should be done prior to calculation. For example, in addition, first data's range is -8,000 ~ 8,000 and second data's range is -32,000 ~ 32,000, the result is expected to be -40,000 ~ 40,000. It exceeds the 1 Word of data that could be saved, therefore data needs to be changed to DINT and then DINT addition must be done.

If P is used as suffix, the calculation will be executed once at positive edge.

1) Basic instruction

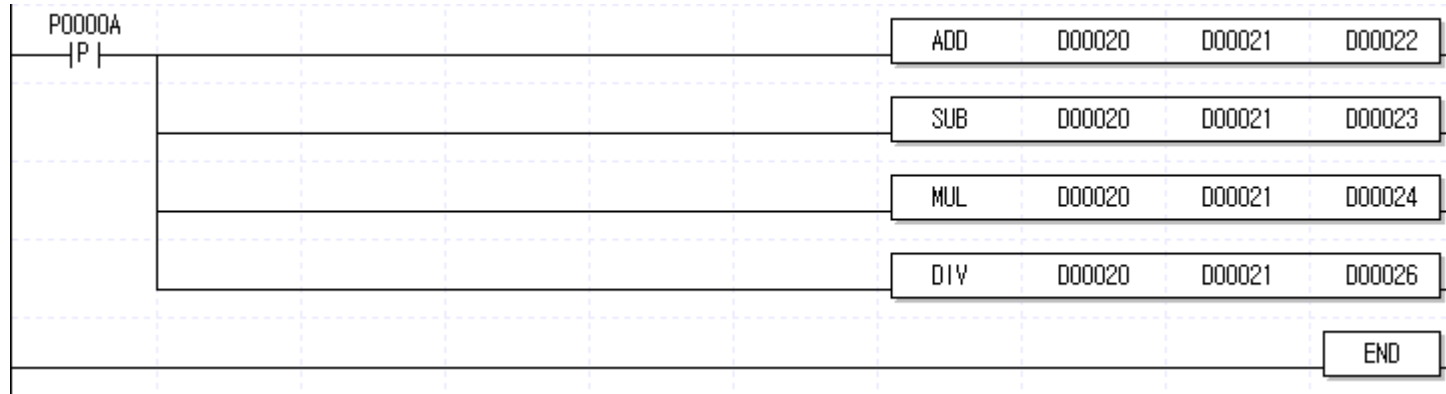
Execute 1 Word, signed decimal calculation. If data is invariable number, it can be -32,768 ~ 32,767.

- ① [ADD(P) S1 S2 D]: Convert data saved in S1, one Word, and S2, one Word, to decimal integer and execute addition. Then, the result is saved to 1 Word data memory specified as D.
- ② [SUB(P) S1 S2 D]: Convert data saved in S1, one Word, and S2, one Word, to decimal integer and execute subtraction. Then, the result is saved to 1 Word data memory specified as D.
- ③ [MUL(P) S1 S2 D]: Convert data saved in S1, one Word, and S2, one Word, to decimal integer and execute multiplication. Then, the result is saved to 2 Word data memory specified as D.
- ④ [DIV(P) S1 S2 D]: Convert data saved in S1, one Word, and S2, one Word, to decimal integer and execute division. Then, the quotient is saved to 1 Word data memory specified as D and remainder to next Word data memory.

4. Instructions

⑤ Program example

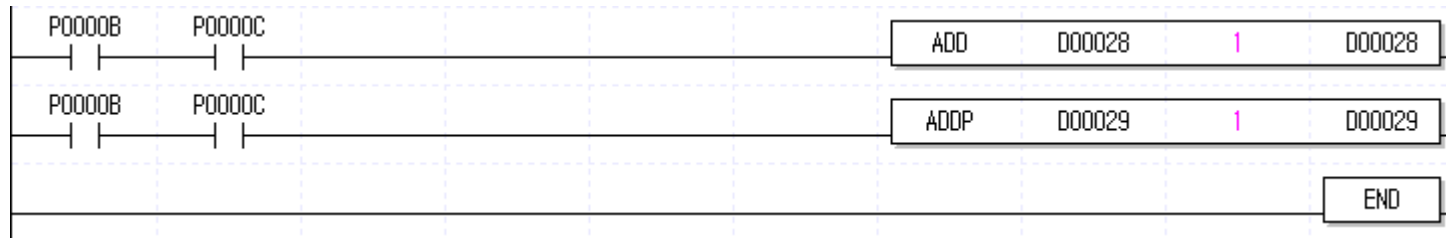
- If P0000A switch is turned on, execute four fundamental arithmetic operations with data of D00020 and D00021.



※ Area for multiplication and division result to be saved is D00024 and D00026. The result for multiplication and division is saved in 2 Word, therefore D00025 and D00027 cannot be used for other data. If any other data is saved in D00025 and D00027, it will be deleted and the result of the calculation will be saved.

⑥ Program example

- If both P0000B and P0000C switch are turned on, 1 is added to each Word data.



※ If basic instruction of addition is used as above example, it shows different result at each scan time since the operation is done every scan when the condition is satisfied. However, if P is used as the suffix, the calculation is done just once when the condition is satisfied.

4. Instructions

2) Unsigned 16Bit decimal instruction

Execute 1 Word, unsigned decimal calculation. If data is invariable number, it can be 0 ~ 65,535.

① [ADDU(P) S1 S2 D]: Convert data saved in S1, one Word, and S2, one Word, to unsigned decimal integer and execute addition. Then, the result is saved to 1 Word data memory specified as D.

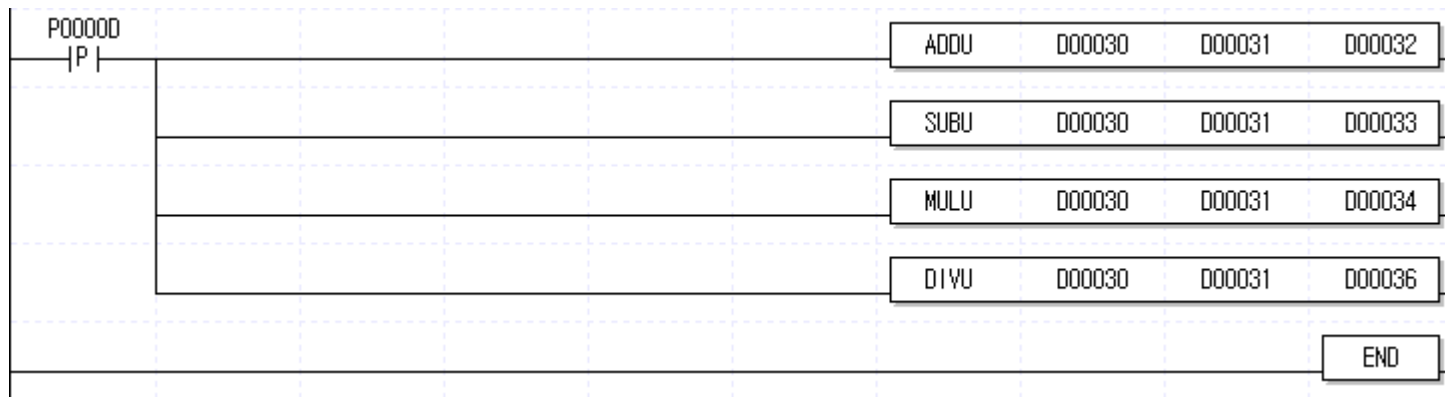
② [SUBU(P) S1 S2 D]: Convert data saved in S1, one Word, and S2, one Word, to unsigned decimal integer and execute subtraction. Then, the result is saved to 1 Word data memory specified as D.

③ [MULU(P) S1 S2 D]: Convert data saved in S1, one Word, and S2, one Word, to unsigned decimal integer and execute multiplication. Then, the result is saved to 2 Word data memory specified as D.

④ [DIVU(P) S1 S2 D]: Convert data saved in S1, one Word, and S2, one Word, to unsigned decimal integer and execute division. Then, the quotient is saved to 1 Word data memory specified as D and remainder to next Word data memory.

⑤ Program example

- If P0000D switch is turned on, execute unsigned four fundamental arithmetic operations with data of D00030 and D00031.



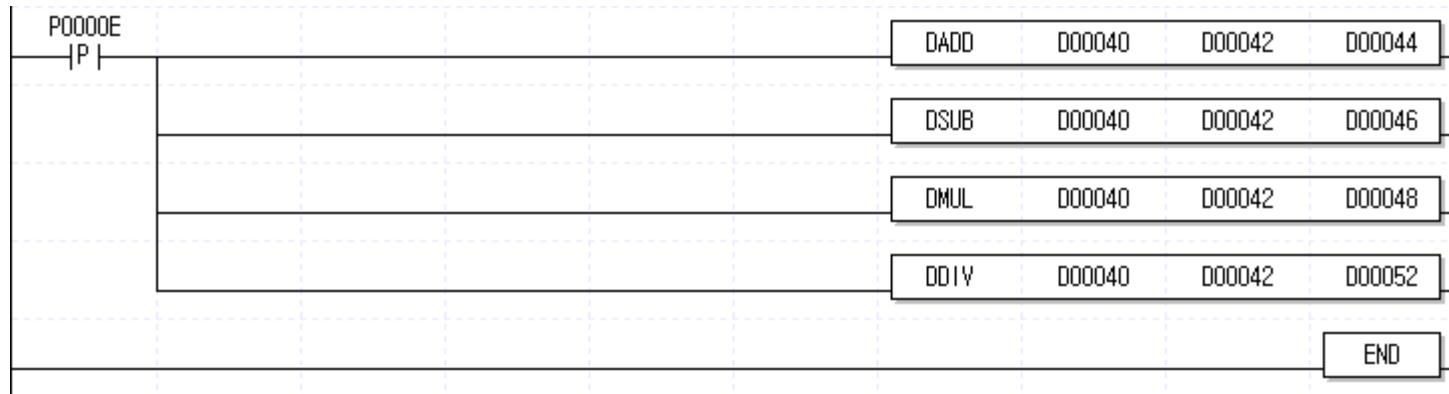
※ Area for multiplication and division result to be saved is D00034 and D00036. The result for multiplication and division is saved in 2 Word, therefore D00035 and D00037 cannot be used for other data.

4. Instructions

3) Signed 32Bit decimal instruction

Execute 2 Word, signed decimal calculation. If data is invariable number, it can be -2,147,483,648 ~ 2,147,483,647.

- ① [DADD(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to signed decimal integer and execute addition. Then, the result is saved to 2 Word data memory specified as D, D+1.
 - ② [DSUB(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to signed decimal integer and execute subtraction. Then, the result is saved to 2 Word data memory specified as D, D+1.
 - ③ [DMUL(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to signed decimal integer and execute multiplication. Then, the result is saved to 4 Word data memory specified as D, D+1, D+2, D+3.
 - ④ [DDIV(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to signed decimal integer and execute division. Then, the quotient is saved to 2 Word data memory specified as D & D+1 and remainder to next 2 Word data memory, D+2 & D+3. So, 32 Bit division needs 4 Word to be saved.
 - ⑤ Program example
- If P0000E switch is turned on, execute signed four fundamental arithmetic operations with data of D00040 and D00042.



※ Addition and subtraction result will be saved in 2 Word. However, multiplication result will be saved in D00048 and D00051. The division result is saved in 4 Word. (2 Word each for quotient and remainder)

4. Instructions

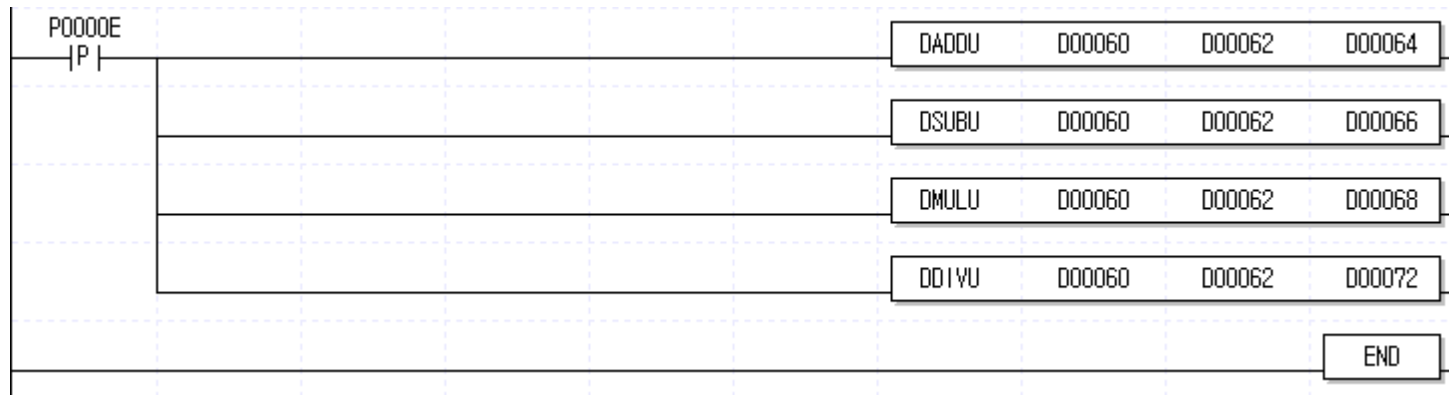
4) Unsigned 32Bit decimal instruction

Execute 2 Word, unsigned decimal calculation. If data is invariable number, it can be 0 ~ 4,294,967,295.

- ① [DADDU(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to unsigned decimal integer and execute addition. Then, the result is saved to 2 Word data memory specified as D, D+1.
- ② [DSUBU(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to unsigned decimal integer and execute subtraction. Then, the result is saved to 2 Word data memory specified as D, D+1.
- ③ [DMULU(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to unsigned decimal integer and execute multiplication. Then, the result is saved to 4 Word data memory specified as D, D+1, D+2, D+3.
- ④ [DDIVU(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to unsigned decimal integer and execute division. Then, the quotient is saved to 2 Word data memory specified as D & D+1 and remainder to next 2 Word data memory, D+2 & D+3. So, 32 Bit division needs 4 Word to be saved.

⑤ Program example

- If P0000E switch is turned on, execute signed four fundamental arithmetic operations with data of D00060 and D00062.



※ Addition and subtraction result will be saved in 2 Word. However, multiplication result will be saved in D00068 and D00071. The division result is saved in 4 Word. (2 Word each for quotient and remainder)

4. Instructions

5) Real number instruction

Execute 2 Word or 4 Word real number calculation. If data is invariable number, it can be 0 ~ 4,294,967,295.

① [RADD(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to real number and execute addition. Then, the result is saved to 2 Word data memory specified as D, D+1.

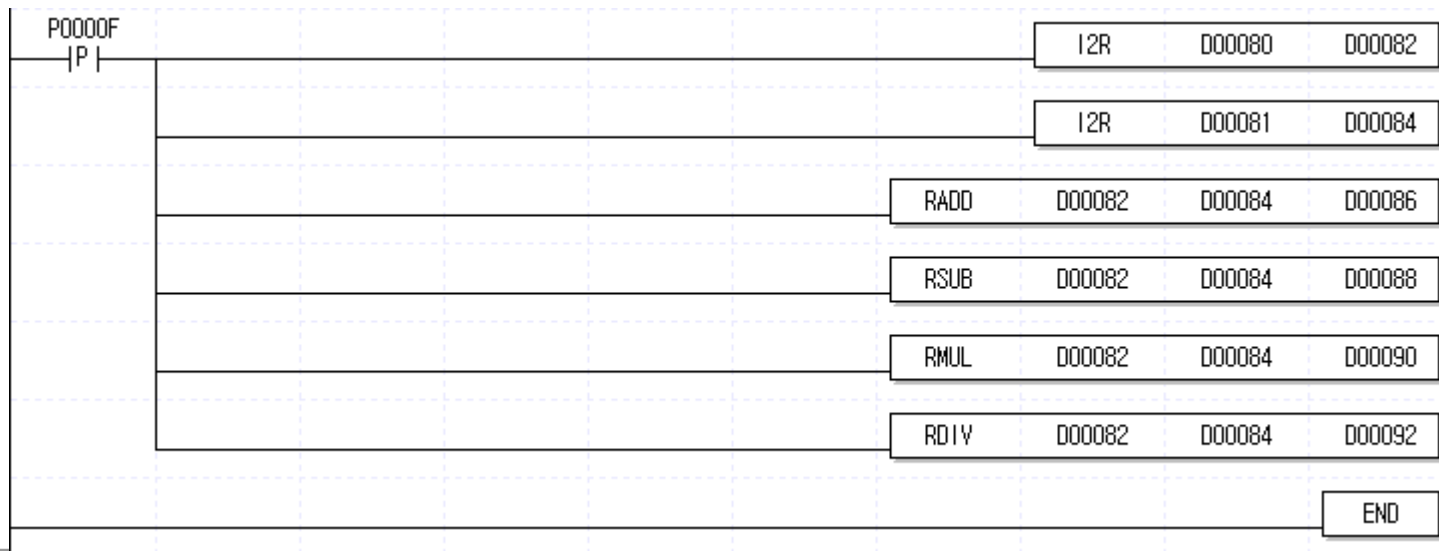
② [RSUB(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to real number and execute subtraction. Then, the result is saved to 2 Word data memory specified as D, D+1.

③ [RMUL(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to real number and execute multiplication. Then, the result is saved to 2 Word data memory specified as D, D+1.

④ [RDIV(P) S1 S2 D]: Convert data saved in S1 & S1+1, two Word, and S2 & S2+1, two Word, to real number and execute division. Then, the result is saved to 2 Word data memory specified as D, D+1.

⑤ Program example

- If P0000F switch is turned on, execute real number four fundamental arithmetic operations with data of D00080 and D00081.



4. Instructions

4.2.3 Increase/Decrease

1) Data Increase/Decrease instruction

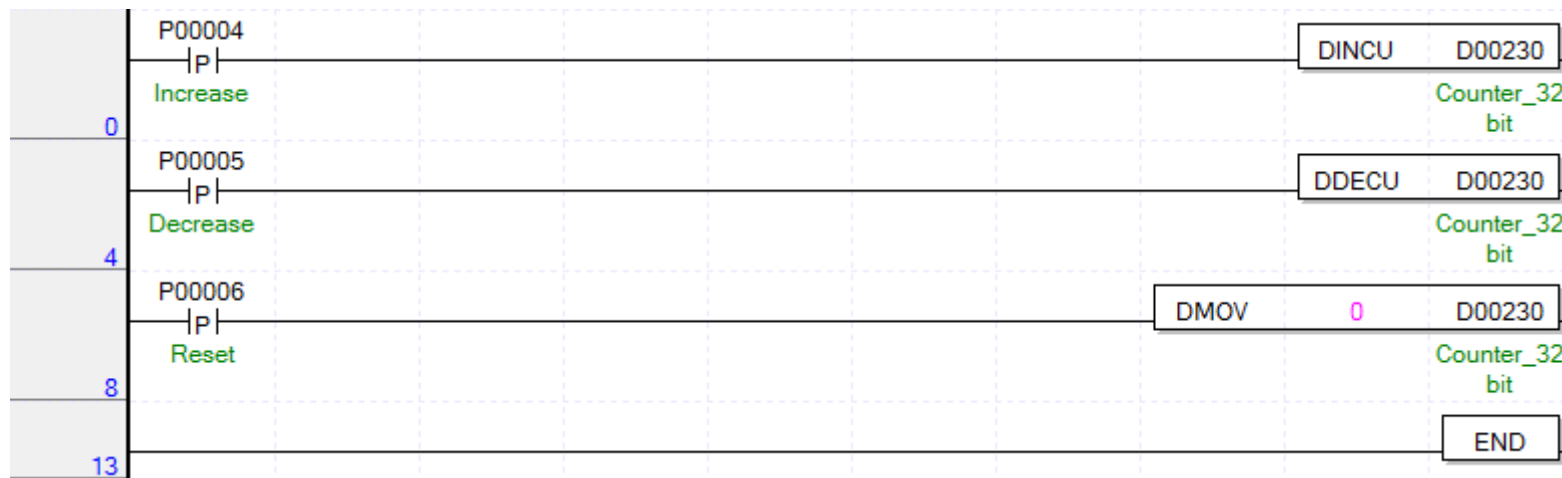
- ① [INC(P) S]: Convert 1 Word data saved in S to signed decimal integer. Then, add 1 and save it to S.
- ② [DINC(P) S]: Convert 2 Word data saved in S & S+1 to signed decimal integer. Then, add 1 and save it to S & S+1.
- ③ [DEC(P) S]: Convert 1 Word data saved in S to signed decimal integer. Then, subtract 1 and save it to S.
- ④ [DDEC(P) S]: Convert 2 Word data saved in S & S+1 to signed decimal integer. Then, subtract 1 and save it to S & S+1.

※ Suffixes for data Increase/Decrease instruction

- 4: 4 Bit operation. In this case, S must be 0 ~ C Bit of 1 Word. The range of number would be 0 ~ 15
- 8: 8 Bit operation. In this case, S must be 0 ~ 8 Bit of 1 Word. The range of number would be 0 ~ 255
- U: Unsigned decimal integer operation

⑤ Program example

- Make a UP/DOWN counter with INC/DEC instruction. It can count more than 65,535



4. Instructions

4.2.4 Comparison

1) CMP (unsigned decimal integer single data comparison)

Compare the unsigned decimal integer and save the result to flag area(F00120 ~ F00125 Bit). If the first data is smaller than second data ($S1 < S2$) F00120, if same or smaller ($S1 \leq S2$) F00121, if same ($S1 = S2$) F00122, if bigger ($S1 \geq S2$) F00124, if not same ($S1 \neq S2$) F00125 Bit turns on. User uses the contact to write program. If more than 1 CMP instruction is modified, flag area refreshes as CMP instruction is executed. Therefore, if the result is used throughout program, the result needs to be saved in internal device and then use it for calculation.

- ① [CMP(P) S1 S2]: Convert 1 Word data saved in S1 and S2 to unsigned decimal integer and compare. The result is saved to F00120 ~ F00125 Bit. If S1 or S2 is specified to invariable number, it is possible to have the range of 0 ~ 65,535.
- ② [DCMP(P) S1 S2]: Convert 2 Word data saved in S1, S1+1 and S2, S2+1 to unsigned decimal integer and compare. The result is saved to F00120 ~ F00125 Bit. If S1 or S2 is specified to invariable number, it is possible to have the range of 0 ~ 4,294,967,295.
- ③ [CMP4(P) S1 S2]: Convert each 4 Bit data saved starting from S1 and S2 to unsigned decimal integer and compare. The result is saved to F00120 ~ F00125 Bit. 0 ~ C Bit can be specified for S1 or S2. If S1 or S2 is specified to invariable number, it is possible to have the range of 0 ~ 15.
- ④ [CMP8(P) S1 S2]: Convert each 8 Bit data saved starting from S1 and S2 to unsigned decimal integer and compare. The result is saved to F00120 ~ F00125 Bit. 0 ~ 8 Bit can be specified for S1 or S2. If S1 or S2 is specified to invariable number, it is possible to have the range of 0 ~ 255.

※ If CMP instruction is used, it executes comparison after converting the data to unsigned decimal integer, therefore the result output is $-1 > 0$. It is because when numeric data is saved in memory, ordering state of Bit is $\text{hFFFF}=65,535=-1$.

4. Instructions

2) Input comparison instruction

Compare two data using sign of equality and inequality. It compares S1 with S2. And if identical to condition, result will be on, if not identical, it will be Off, and its result and present BR value will be AND operated to lead to a new operation result.

- ① [= S1 S2]: Convert 1 Word data saved in S1 and S2 to signed decimal integer and compare. Output is 1(on) if $S1 = S2$, 0(OFF) if $S1 \neq S2$.
- ② [< S1 S2]: Convert 1 Word data saved in S1 and S2 to signed decimal integer and compare. Output is 1(on) if $S1 < S2$, 0(OFF) if $S1 > S2$ or $S1 = S2$.
- ③ [<= S1 S2]: Convert 1 Word data saved in S1 and S2 to signed decimal integer and compare. Output is 1(on) if $S1 < S2$ or $S1 = S2$, 0(OFF) if $S1 > S2$.
- ④ [> S1 S2]: Convert 1 Word data saved in S1 and S2 to signed decimal integer and compare. Output is 1(on) if $S1 > S2$, 0(OFF) if $S1 < S2$ or $S1 = S2$.
- ⑤ [>= S1 S2]: Convert 1 Word data saved in S1 and S2 to signed decimal integer and compare. Output is 1(on) if $S1 \neq S2$, 0(OFF) if $S1 = S2$.
- ⑥ [<> S1 S2]: Convert each 8 Bit data saved starting from S1 and S2 to unsigned decimal integer and compare. The result is saved to F00120 ~ F00125 Bit. 0 ~ 8 Bit can be specified for S1 or S2. If S1 or S2 is specified to invariable number, it is possible to have the range of 0 ~ 255.

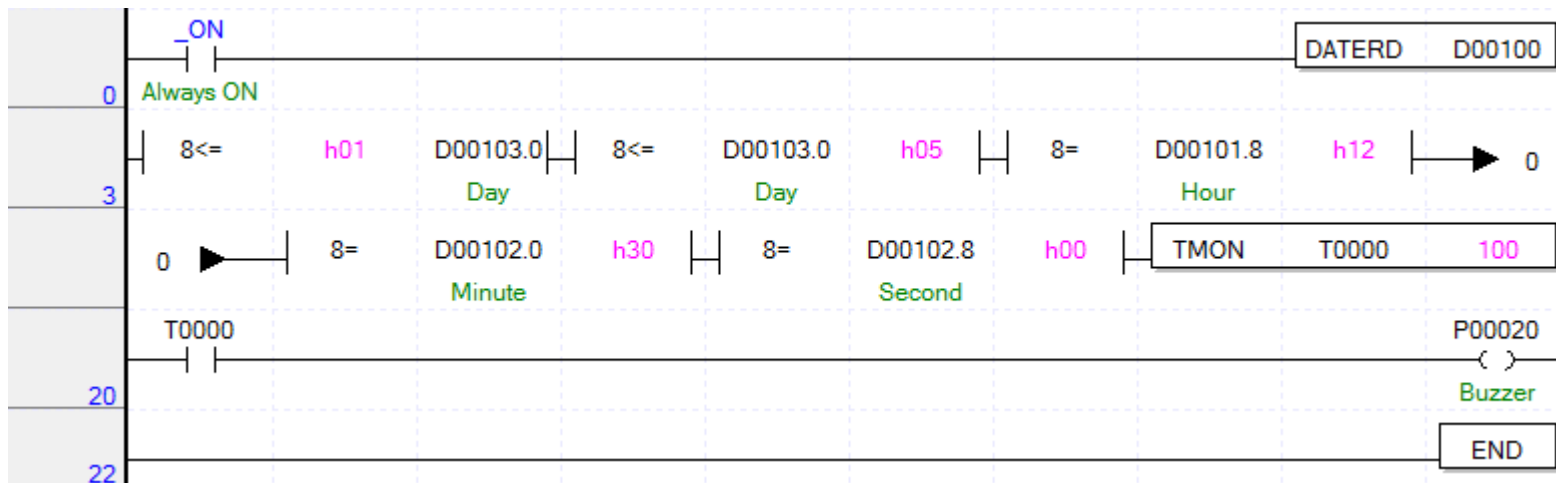
※ Usable prefix in Comparison instruction

- U : 16 Bit unsigned decimal integer (U=, U<, U<=, U>, U>=, U<>)
- D : 32 Bit signed decimal integer (D=, D<, D<=, D>, D>=, D<>)
- UD : 32 Bit unsigned decimal integer (UD=, UD<, UD<=, UD>, UD>=, UD<>)
- R : 32 Bit real number (R=, R<, R<=, R>, R>=, R<>)
- L : 64 Bit real number (L=, L<, L<=, L>, L>=, L<>)
- \$: ASCII (\$=, \$<, \$<=, \$>, \$>=, \$<>)
- 4 : 4 Bit, each operator can be 0 ~ C Bit and 0 ~ 15 if selected in invariable number.
- 8 : 8 Bit, each operator can be 0 ~ 8 Bit and 0 ~ 255 if selected in invariable number.

4. Instructions

⑦ Program example

- Using PLC clock data, buzzer sounds for 10 seconds at 12:30 from Monday to Friday.



※ XGK PLC's clock data (RTC) : Provides 8 data of year, month, date, hour, minute, second, day, 100 year in 1 byte BCD type. RTC data is saved in F0053 ~ F0056 Word. Since, F area cannot calculate 8 Bit, RTC data saved in F area should be moved to other area (in above example, D00100 ~ D00103) using DATERD instruction, then calculate afterwards.

Area	Upper byte(F ~ 8 Bit)	Lower byte(0 ~ 7 Bit)
F0053	month (h00 ~ h12)	year (h00~h99)
F0054	hour (h00 ~ h23)	date (h00 ~ h31)
F0055	second (h00 ~ h59)	minute (h00 ~ h59)
F0056	100 year (h19 ~ h21)	day (h00(Sun) ~ h06(Sat))

4. Instructions

3) 3(three) Input comparison instruction:

Compare 3(three) data using sign of equality and inequality. It compares S1 with S2 with S3. And if identical to condition, result will be on, if not identical, it will be Off, and its result and present BR value will be AND operated to lead to a new operation result.

- ① [$=3$ S1 S2 S3]: Convert 1 Word data saved in S1, S2 and S3 to signed decimal integer and compare. Output is 1(on) if $S1 = S2 = S3$, 0(OFF) if others.
- ② [<3 S1 S2 S3]: Convert 1 Word data saved in S1, S2 and S3 to signed decimal integer and compare. Output is 1(on) if $S1 < S2$ and $S2 < S3$, 0(OFF) if others.
- ③ [≤ 3 S1 S2 S3]: Convert 1 Word data saved in S1, S2 and S3 to signed decimal integer and compare. Output is 1(on) if $S1 \leq S2$ and $S2 \leq S3$, 0(OFF) if others.
- ④ [>3 S1 S2 S3]: Convert 1 Word data saved in S1, S2 and S3 to signed decimal integer and compare. Output is 1(on) if $S1 > S2$ and $S2 > S3$, 0(OFF) if others.
- ⑤ [≥ 3 S1 S2 S3]: Convert 1 Word data saved in S1, S2 and S3 to signed decimal integer and compare. Output is 1(on) if $S1 \geq S2$ and $S2 \geq S3$, 0(OFF) if others.
- ⑥ [$<>3$ S1 S2 S3]: Convert 1 Word data saved in S1, S2 and S3 to signed decimal integer and compare. Output is 1(on) if $S1 \neq S2$ and $S2 \neq S3$ and $S1 \neq S3$, 0(OFF) if others.

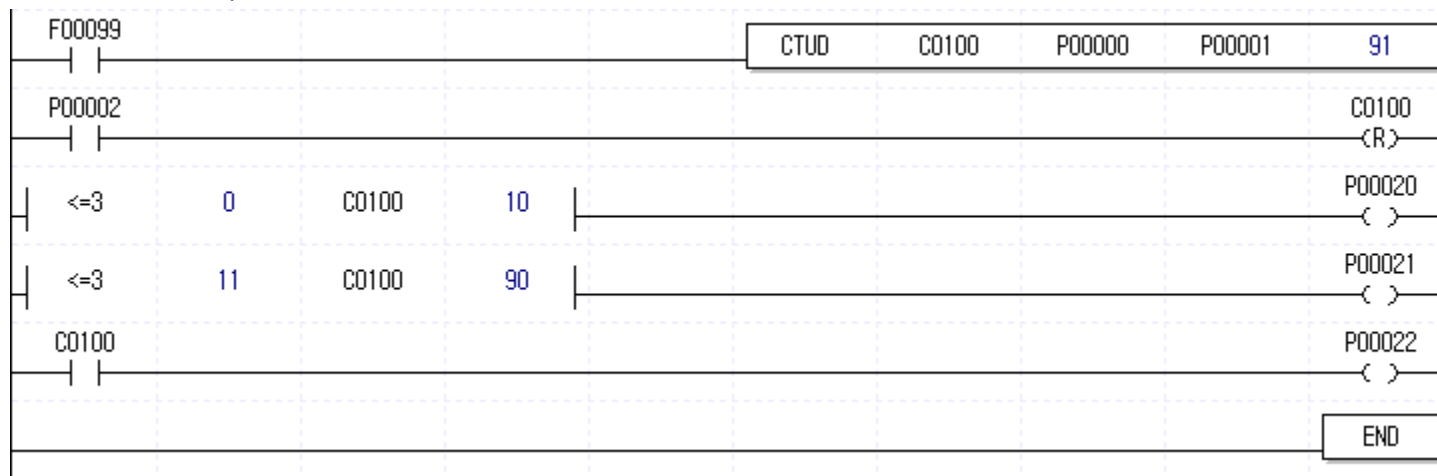
※ Reference

Prefix that can be used with 3 input comparison instruction is D, which means 32 Bit instruction. In this case, if the data is specified in invariable number, it is possible to have the range of -2,147,483,648 ~ 2,147,483,647. (D=3, D<3, D<=3, D>3, D>=3, D<>3)

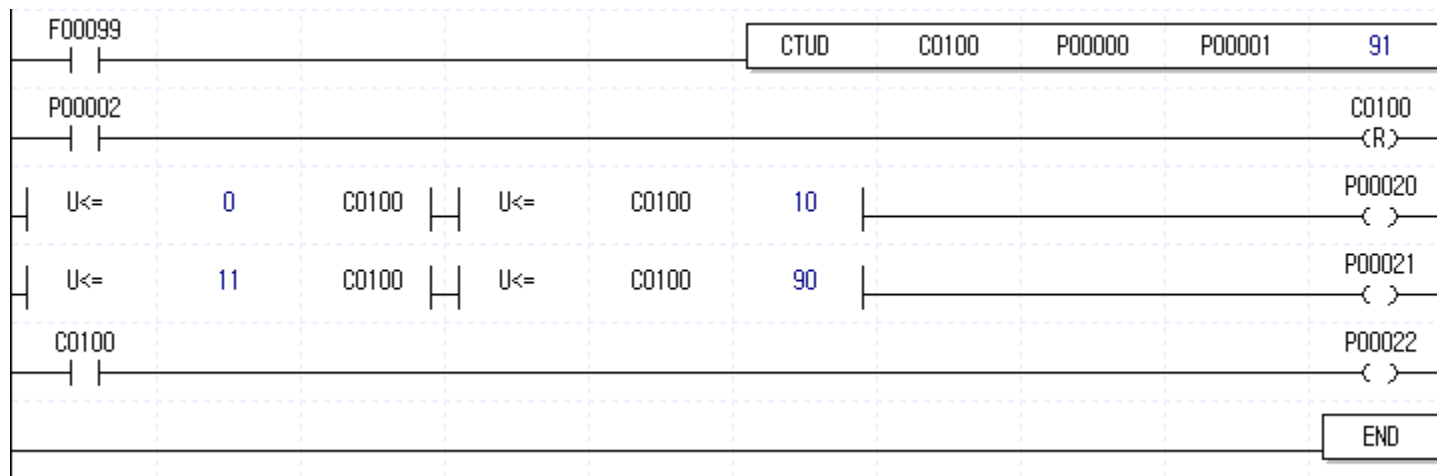
4. Instructions

⑦ Program example

- Turn on the LED of P00020 if present value of counter is between 0 and 10
- Turn on the LED of P00021 if present value of counter is between 11 and 90
- Turn on the LED of P00022 if present value of counter is more than 91



※ If the present value of C0100 exceeds 32,767, the program shall be below,



4. Instructions

4) Group comparison instruction

It compares specified compare data S1 with N Word data starting S2 1:1 to save in specified D number's lower Bit up to Nth Bit. If compare condition is met, 1 will be saved in D. If compare condition is not met, 0 will be saved in D. It can input constant value from 0 to 65,535 in S1. Operation of instruction is as shown below.

- ① [GEQ(P) S1 S2 D N]: Convert N Word data of S1 and N Word data of S2 to unsigned decimal integer. It compares 1:1 to save in specified D number's lower Bit up to Nth Bit. It will save 1 if two data is same, 0 if it is not same.
- ② [GLT(P) S1 S2 D N]: Convert N Word data of S1 and N Word data of S2 to unsigned decimal integer. It compares 1:1 to save in specified D Nth Bit. Output is 1 if $S1+n < S2+n$ ($1 \leq n \leq N-1$), 0 if others.
- ③ [GLE(P) S1 S2 D N]: Convert N Word data of S1 and N Word data of S2 to unsigned decimal integer. It compares 1:1 to save in specified D Nth Bit. Output is 1 if $S1+n \leq S2+n$ ($1 \leq n \leq N-1$), 0 if others.
- ④ [GGT(P) S1 S2 D N]: Convert N Word data of S1 and N Word data of S2 to unsigned decimal integer. It compares 1:1 to save in specified D Nth Bit. Output is 1 if $S1+n > S2+n$ ($1 \leq n \leq N-1$), 0 if others.
- ⑤ [GGE(P) S1 S2 D N]: Convert N Word data of S1 and N Word data of S2 to unsigned decimal integer. It compares 1:1 to save in specified D Nth Bit. Output is 1 if $S1+n \geq S2+n$ ($1 \leq n \leq N-1$), 0 if others.
- ⑥ [GNE(P) S1 S2 D N]: Convert N Word data of S1 and N Word data of S2 to unsigned decimal integer. It compares 1:1 to save in specified D Nth Bit. Output is 1 if $S1+n \neq S2+n$ ($1 \leq n \leq N-1$), 0 if others.

※ Reference

Prefix that can be used with 3 input comparison instruction is D, which means 32 Bit instruction. In this case, if the data is specified in invariable number, it is possible to have the range of 0 ~ 4,294,967,295. (GDEQ(P), GDLT(P), GDLE(P), GDGT(P), GDGE(P), GDNE(P))

5) Table data comparison instruction

It compares specified compare data S1 with 16-Word data starting S2 to output to specified D area's 16 Bits ('1' if identical, '0' if different). S1 sets Word data or data address, and S2 sets table head area address.

- ① [TCMP(P) S1 S2 D]: It compares specified compare data S1 with 16-Word data starting S2 to output to specified D area. If S1 is specified with invariable number, the range is 0~65,535.
- ② [DTCMP(P) S1 S2 D]: It compares specified compare data S1,S1+1 with 16-double-Word data starting S2,S2+1 to output to specified D area. If S1 is specified with invariable number, the range is 0~4,294,967,295

4. Instructions

6) Input Group comparison instruction

Compare data in sequential order and result will be 1(on) if the condition is met, it will be 2(Off) if not met. N can be constant value from 0 to 65,535.

- ① [G= S1 S2 N]: Convert N Word data of S1 and N Word data of S2 to signed decimal integer. It compares with each data. It will save 1(on) if $S1=S2$, $S1+1=S2+1$, ..., $S1+(N-1)=S2+(N-1)$, 0(OFF) if others.
- ② [G< S1 S2 N]: Convert N Word data of S1 and N Word data of S2 to signed decimal integer. It compares with each data. It will save 1(on) if $S1<S2$, $S1+1<S2+1$, ..., $S1+(N-1)<S2+(N-1)$, 0(OFF) if others.
- ③ [G<= S1 S2 N]: Convert N Word data of S1 and N Word data of S2 to signed decimal integer. It compares with each data. It will save 1(on) if $S1\leq S2$, $S1+1\leq S2+1$, ..., $S1+(N-1)\leq S2+(N-1)$, 0(OFF) if others.
- ④ [G> S1 S2 N]: Convert N Word data of S1 and N Word data of S2 to signed decimal integer. It compares with each data. It will save 1(on) if $S1>S2$, $S1+1>S2+1$, ..., $S1+(N-1)>S2+(N-1)$, 0(OFF) if others.
- ⑤ [G>= S1 S2 N]: Convert N Word data of S1 and N Word data of S2 to signed decimal integer. It compares with each data. It will save 1(on) if $S1\geq S2$, $S1+1\geq S2+1$, ..., $S1+(N-1)\geq S2+(N-1)$, 0(OFF) if others.
- ⑥ [G<>3 S1 S2 N]: Convert N Word data of S1 and N Word data of S2 to signed decimal integer. It compares with each data. It will save 1(on) if $S1\neq S2$, $S1+1\neq S2+1$, ..., $S1+(N-1)\neq S2+(N-1)$, 0(OFF) if others.

※ Reference

Prefix that can be used with 3 input comparison instruction is D, which means 32 Bit instruction. In this case, if the data is specified in invariable number, it is possible to have the range of -2,147,483,648 ~ 2,147,483,647. (D=3, D<3, D<=3, D>3, D>=3, D<>3)

4. Instructions

4.2.5 Data type conversion

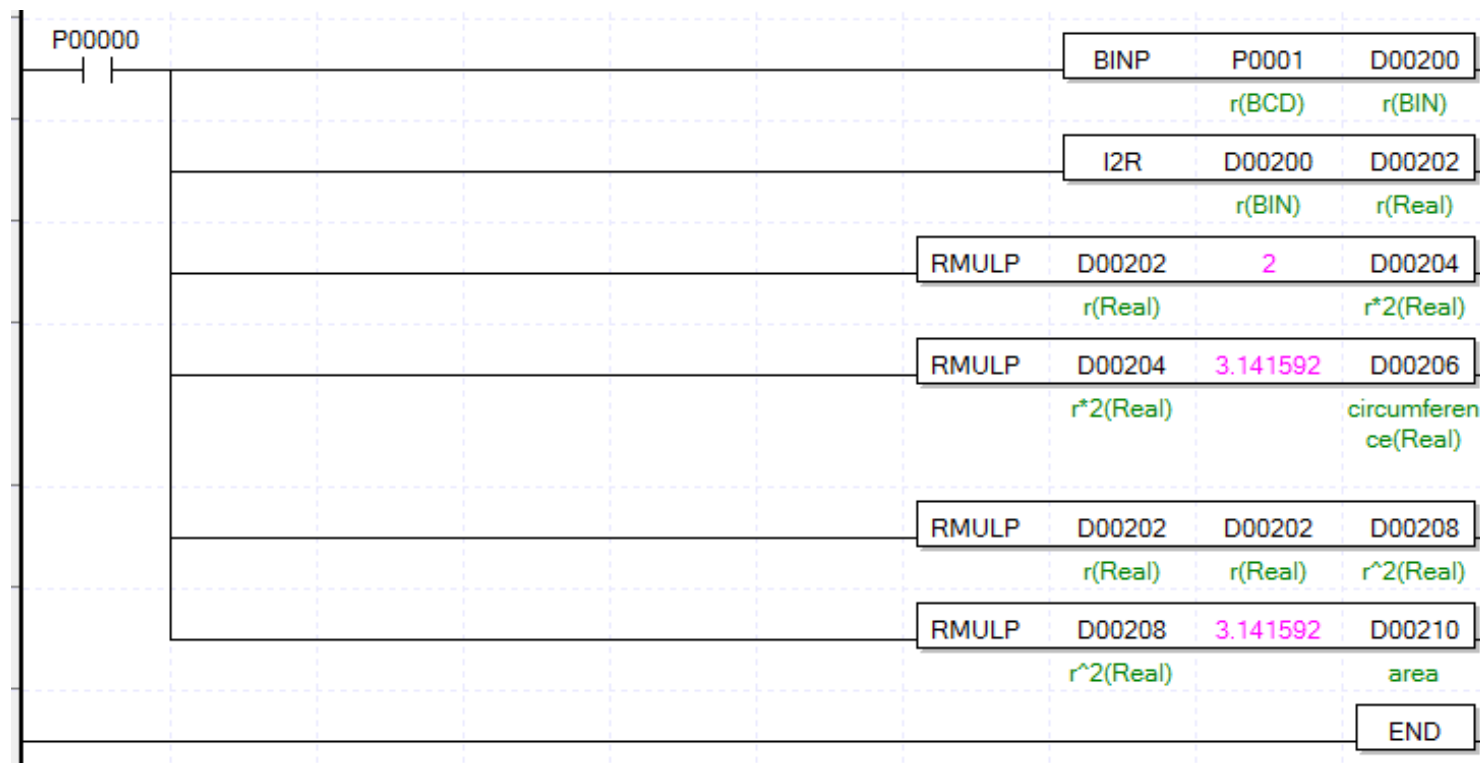
1) Integer → Real number data type conversion instruction

- ① [I2R(P) S D]: It converts specified S 16-Bit Integer data to Single Real (32-Bit) and saves in D+1, D.
- ② [U2R(P) S D]: Converts 16 Bit unsigned integer data set by S to short real data (32 Bit) and save it in D+1, D.
- ③ [D2R(P) S D]: D2R(P) converts S+1,S specified 32-Bit Double Integer data to Real Number(32-Bit) to save in D+1, D. If 32-Bit Integer data value exceeds valid range (24-Bit) of Floating point Real data, accuracy become lower and then inaccuracy error flag (F0057A) is set. But PLC operation status does not change.
- ④ [UD2R(P) S D]: Converts 32 Bit double unsigned integer data set by S+1, S to short real data (32 Bit) and save it D+1, D. In case 23 Bit unsigned integer data exceeds effective range (24 Bit) of short floating point type real data, accuracy is removed and inaccuracy error flag (F0057A) is set. Tough inaccuracy error flag is set, there is no changed in PLC operation.
- ⑤ [I2L(P) S D]: Converts specified S 16-Bit Integer data to long real (64-Bit) and saves in D+3,D+2,D+1,D.
- ⑥ [U2L(P) S D]: Converts 16 Bit unsigned integer data set by S to long real data (64 Bit) and save it in D+3, D+2, D+1, D.
- ⑦ [D2L(P) S D]: D2L(P) converts S+1,S specified 32-Bit Double Integer data to long real number (64-Bit) to save in D+3,D+2,D+1,D.
- ⑧ [UD2L(P) S D]: Converts 32 Bit double unsigned integer data set by S+1, S to long real data (64 Bit) and save it D+3, D+2, D+1, D.

4. Instructions

⑨ Program example

- If P00000 switch turns on, use BCD switch and get circumference length and area of circle with given radius.



4. Instructions

2) Real number → Integer data type conversion instruction :

- ① [R2I(P) S D]: R2I converts S+1,S specified Real Number(32-Bit) to 16-Bit Integer data to save in D.
- ② [R2U(P) S D]: Converts short real data (32Bit) set by S+1, S to unsigned integer data and save it in D.
- ③ [R2D(P) S D]: Converts S+1,S specified 32-Bit Integer data to long real number(32-Bit) and saves in D+1,D.
- ④ [R2UD(P) S D]: Converts short real data (32Bit) set by S+1, S to double unsigned integer data (32 Bit) and save it in D+1, D.
- ⑤ [L2I(P) S D]: Converts S+3,S+2,S+1,S specified long real number to integer (16-Bit) to save in D.
- ⑥ [L2U(P) S D]: Converts long real data set by S+3, S+2, S+1, S to unsigned integer (16 Bit) and save it in D.
- ⑦ [L2D(P) S D]: Converts S+3,S+2,S+1,S specified long real data to double integer data (32-Bit) to save in D+1,D.
- ⑧ [L2UD(P) S D]: Converts long real data set by S+3, S+2, S+1, S to double unsigned integer data (32 Bit) and save it in D+1, D.

※ If S+1,S specified Real Number value exceeds -32,768~32,767 range, operation error occurs. At this moment, the result of 32,767 will be saved if input value is bigger than 32,767, and -32,768 will be saved if input value is smaller than -32,768.

※ Value of below decimals is will be omitted after rounding off the nearest integer.

4. Instructions

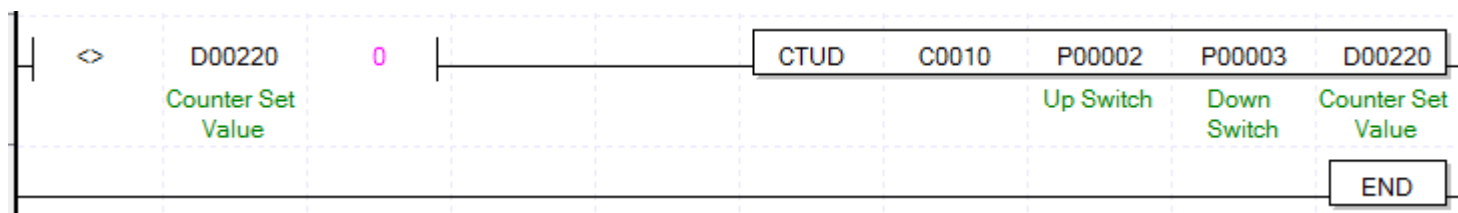
3) BCD → BIN data type conversion instruction :

① [BIN(P) S D]: Converts specified S device's BCD data (0~9999) to BIN data(16 Bit unsigned integer) and it is saved in D.

② [DBIN(P) S D]: Converts specified S, S+1 device's BCD data to BIN data(32 Bit unsigned integer) and it is saved in D, D+1.

③ Program example

- Set counter to 10 using UpDown Counter. UpSwitch : P0002, DownSwitch : P0003



4. Instructions

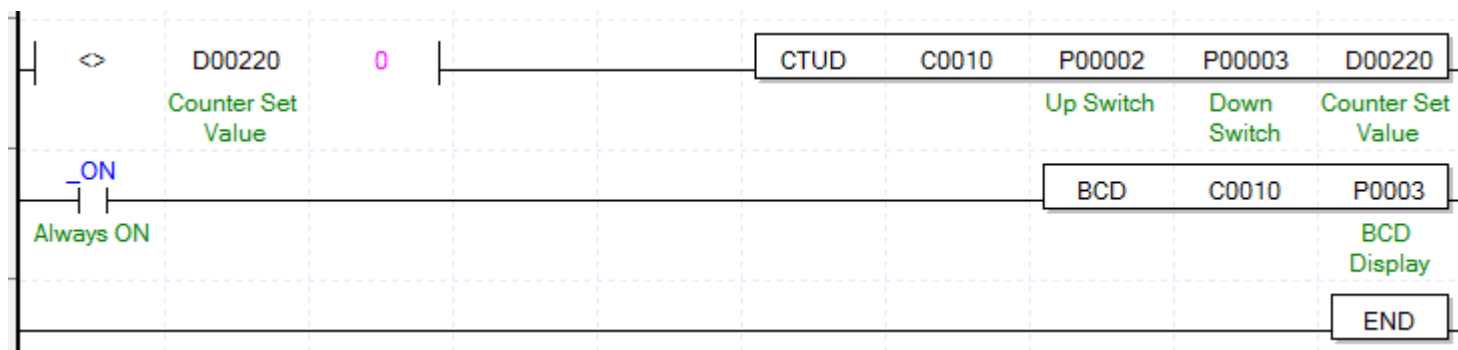
4) BIN → BCD data type conversion instruction :

① [BCD(P) S D]: Converts specified S device's BIN data (0~h270F) to BCD so to save in D.

② [DBCD(P) S D]: Converts specified (S+1,S) device's BIN Data (0~h05F5E0FF) to BCD so to save in D+1 and D respectively.

③ Program example

- Show the previous counter value(program example 4.2.3) to BCD display.



4. Instructions

4.3 Index (Z)

The conception of Index is considering device as variable. Z is used for Index device, and the area is Z000 ~ Z127. Index is used as type of standard device [Index device], and the allocated Index device will be Standard Device address + Index Device. For example, when D00000[Z000] of Index is used, it considers D00000 as standard device, and Z000 is to Index device. When Z000 has data '100', the device of D00000[Z000] is D00100. It means that if Index device is changed, standard device[Index device] is changed. By using this conception, table data of more than 1 Word can be operated.



The data of Index device can be considered differently as following standard device. When it is used as Word device[Index], Index is to Word device index. on the other hand, when it is used as Bit device[Index], Index is to Bit device Index. For example, in case that 100 is saved to Z000, D00000[Z000] is D00100, M00000[Z000] is M00063 and D00000.5[Z000] is D00100.5.

D00000[Z000] → D00100

M00000[Z000] → M00063

D00000.5[Z000] → D00100.5

※ Reference

When the standard device area is exceed by Index device, it might make problem to PLC. Therefore, you must be careful to set Index data. For example, in case that you use D10000[Z000], the data set to Z00 is only possible between -10,000 and 6387, because XGB-U has D device area D00000 ~ D32767.

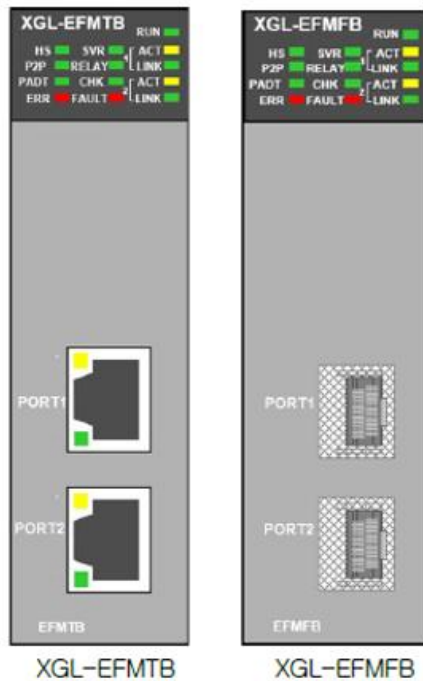
1. PLC

5. Communication (Ethernet)

5. Communication (Ethernet)

[XGL-EFMTB (Fast Ethernet)]

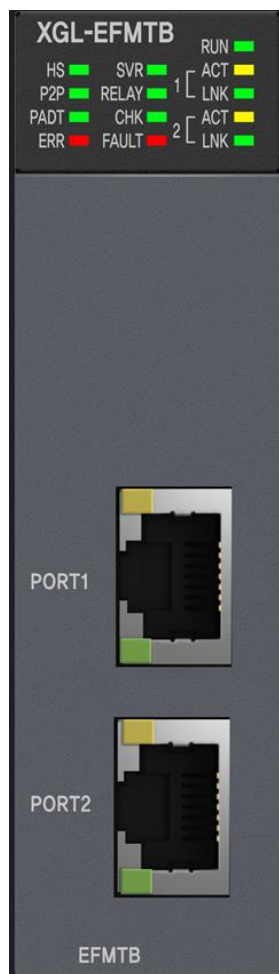
LS ELECTRIC dedicated Industrial network based on Ethernet / Ring topology
(XGT, Modbus, RAPIEnet, EtherNet/IP Protocol support)



Item		XGL-EFMTB/EFMFB/EFMHB
transmission speed		UTP: 10/100/1000Mbps Optical: 100/1000Mbps
Max. extended length between nodes.		UTP: 100m Optical: 2km(@100Mbps, MM)
Transmission media		UDP: CAT5e~ Optical: Multi mode(MMF) / Single mode(SMF) cable
Topology		Line, Star, Tree, Ring(RAPIEnet Enable)
Diagnosis function		Station number/IP address collision detection User diagnosis in XG5000
Communication Speed	P2P	XGT Protocol, Modbus TCP/IP RAPIEnet(ver6.0~)
	High Speed	Fenet, RAPIEet(ver6.0~)
	Smart extend service	RAPIEnet(ver8.0~) Ethernet/IP(ver8.0~)

5. Communication (Ethernet)

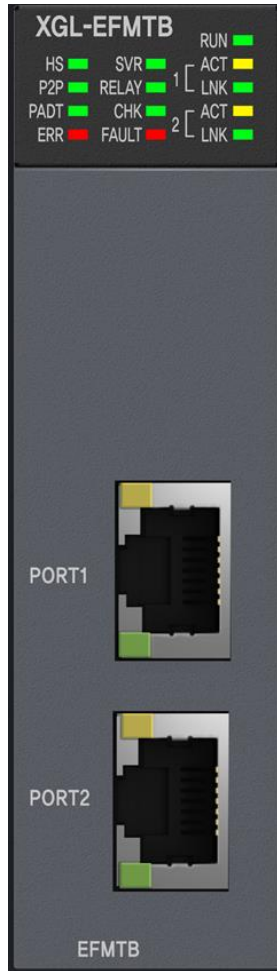
[XGL-EFMTB (Fast Ethernet)]



Silk	LED contents	LED status	contents
RUN	Indicates whether initialization is done successfully and it works properly	ON	In case initialization is done successfully
		OFF	In case initialization is not normal or done abnormally
HS	Indicates whether High speed link service is normal or not	ON	Service is enabled and service is executed normally
		OFF	Service is not executed
P2P	Indicates whether P2P service is normal or not	ON	Service is enabled and service is executed normally
		OFF	Service is not executed
PADT	Indicates whether PADT is connected or not through FEnet I/F module	ON	If PADT is connected
		OFF	If PADT is not connected
ERR	Indicates hardware error	ON	It means heavy error
		OFF	It is executed normally
SVR	Indicate whether connection with remote Ethernet interface communication device(include PC) and dedicated communication service.	ON	In case dedicated communication is used
		OFF	In case dedicated communication is not used

5. Communication (Ethernet)

[XGL-EFMTB (Fast Ethernet)]



Silk	LED contents	LED status	contents
RELAY	Indicates whether Relay service works properly or not	ON	when the media speed of Port 1 and Port 2 is the same and the frame can be relayed.
		BLINKING	if the media speed is different between the two ports even if the Relay option is checked
		OFF	Relay function is not supported at 10Mbps, so if Relay option is checked at 10Mbps
CHK	Indicates whether CHECK LED works properly or not	ON	-If there is no basic parameter -If there is an error in the basic parameter 1) when the module type is different 2) when the media type and the type of the installed media are different) 3) if there is an error in the IP setting -If there is an error in the RAPIenet slave parameter(V6.0 or above) -When there is a station number duplicated on the RAPIenet network. (Not a collision of the local station number)(V6.0 or above)
		BLINKING	- During Flash Write (V6.0 or above) - When the RAPIenet network is changed to Ring -> Line (250ms On, 250ms Off) (V6.0 or above) - When discarding received messages due to heavy load (V6.0 or above)
		OFF	normal operation

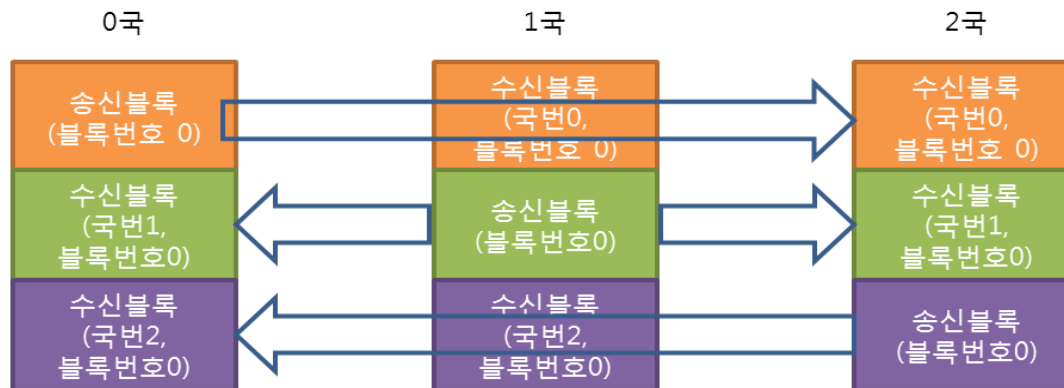
5. Communication (Ethernet)

[High Speed Link]

High-speed link communication is basically a method that allows communication to be opened only with easy parameter setting between LS PLC. Although it is a dedicated communication method between LS PLC and has limitations that it can be used only in LAN environments, it has the advantage of being able to open communication easily with multiple server and multiple client structures using station numbers and block numbers.

Type	Max. Size	Max. Block	Max. Transmission Size	Max. Comm. Size
XGL-EFMT(B)	200WORD (400BYTE)	128블록 (Total Tr/Rcv)	6,400WORD (32 Block)	25,600WORD (Total Tr/Rcv 128 Block)

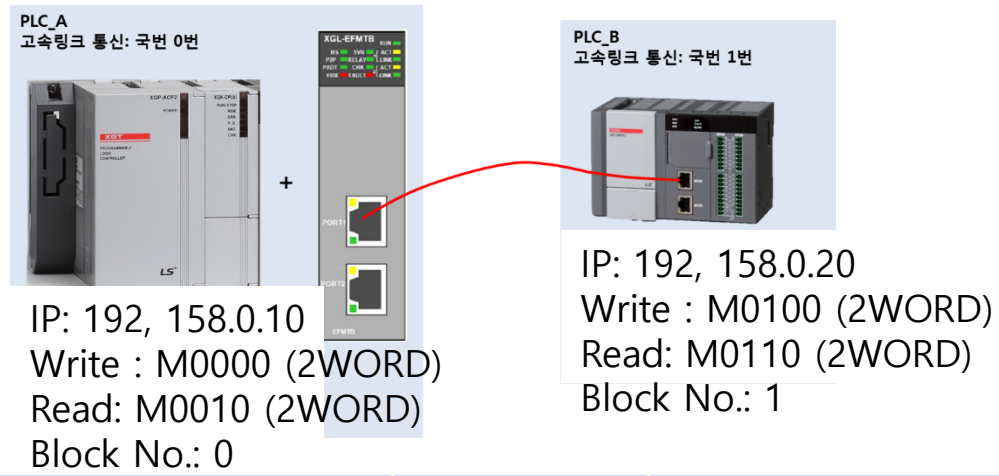
High-speed link data processing on Ethernet is performed in the following structure: The high-speed link parameter has 32 block numbers for transmitting and receiving data and 128 block numbers for transmitting and receiving data can be specified from 0 to 31 or 0 to 127 for receiving. When transmitting data, the sender can decide which data to read and which block to send without specifying the other country number.



5. Communication (Ethernet)

[High Speed Link]

High-speed link communication is periodic communication. The transmission period for each module can be set by the user, and the transmission period for each high-speed link module is set in batches from 20 ms to 10s. For optimal high-speed link communication systems, it is effective to reduce the number of blocks by setting data close to the maximum configurable transmission/reception size (400 bytes) per high-speed link block. To test high-speed link communication, configure the system as follows. Ethernet high-speed link communication between XGT series PLC with station number 0 and XGB PLC with station number 1.



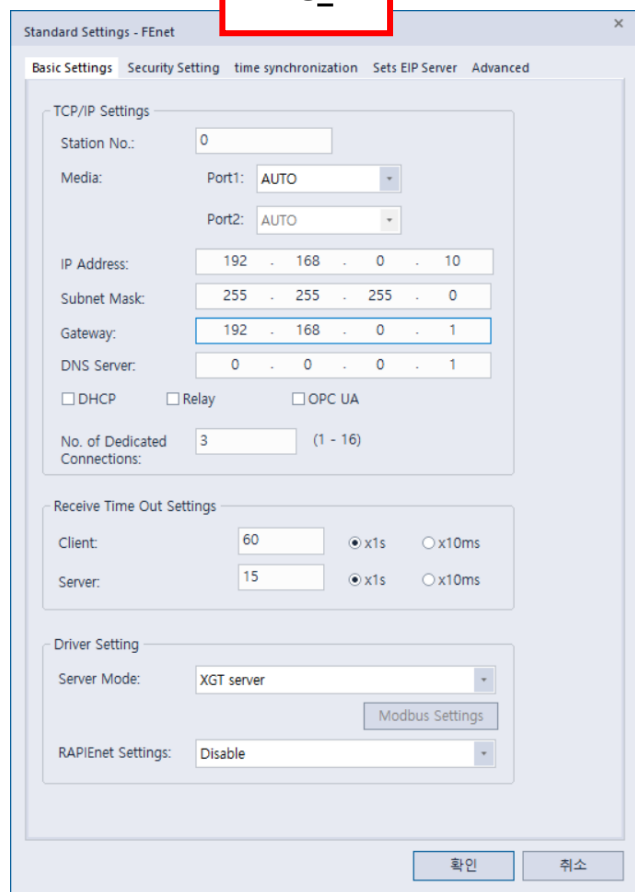
Station 0			Station 1	
Transmission Block (0)	M0000 M0001	→	Receive Block (0)	M0110 M0111
Receive Block (0)	M0010 M0011	←	Transmission Block (0)	M0100 M0101

5. Communication (Ethernet)

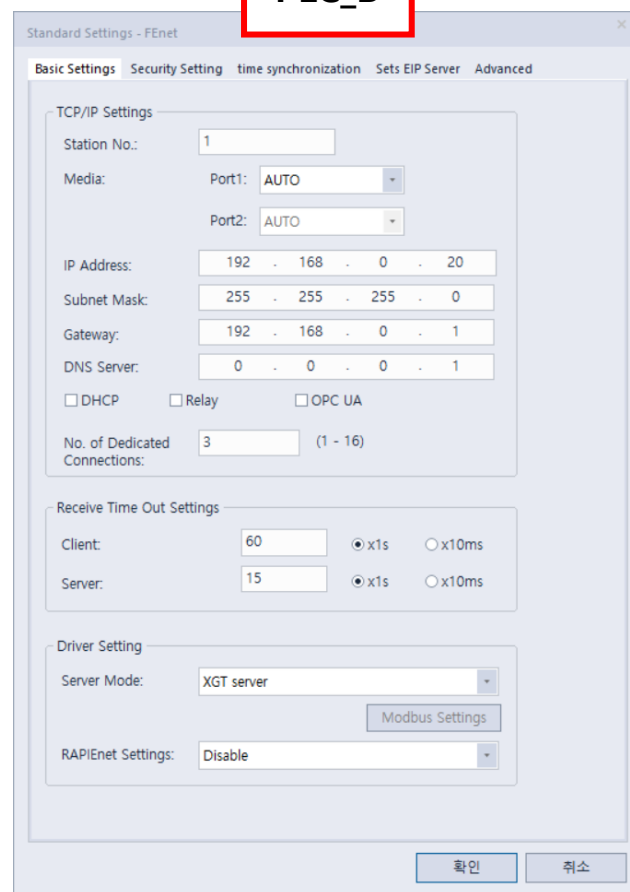
[High Speed Link]

Set the default settings for the FENET module for high-speed link communication. Set PLC_A (Country number: 0) and PLC_B (Country number: 1).

PLC_A



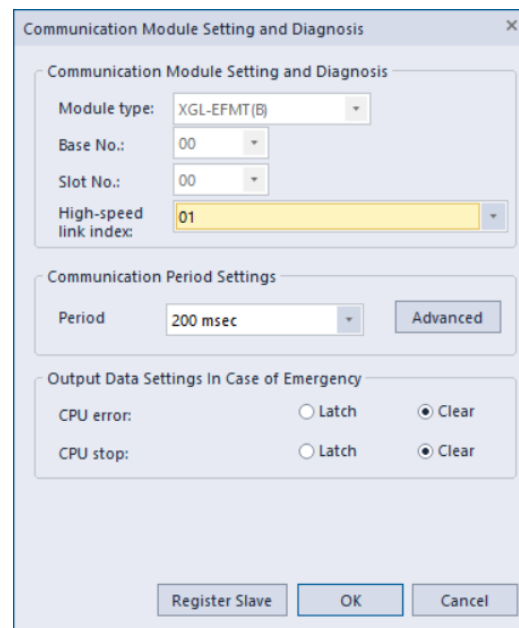
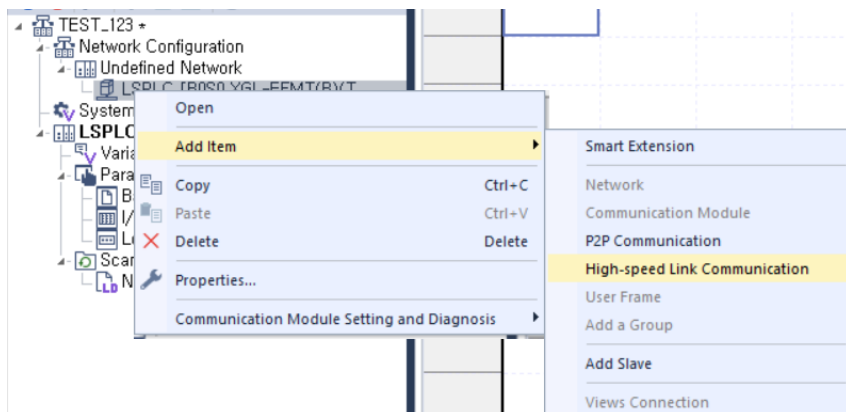
PLC_B



5. Communication (Ethernet)

[High Speed Link]

Add a high speed link from the registered FENET communication module. Based on the XGT series PLC, you can choose from 1 to 12.



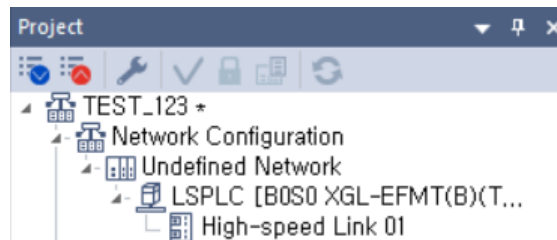
5. Communication (Ethernet)

[High Speed Link]

The high-speed link communication module settings are as follows.

Classification		Content
Communication module setting	Module type	Register Ethernet module.
	Base Number	Base number where Ethernet module is mounted. Set to 00.
	Slot No.	Slot number where Ethernet module is mounted.
	High speed link number	Ethernet module can be set from 1 to 12 in CPU system and maximum 12 devices can be installed.
Communication period settings	Period type	It can be set from 5ms to 10sec in communication cycle. - The send cycle is only for blocks set to send. - The transmission period is equally applied to the entire transmission block within one HS link parameter.
Output data settings in case of emergency	CPU error	Latch: It keeps and transmits the latest data received from the CPU. Even if it is set to latch, it may appear to be cleared by giving data to the CPU before the error. Clear: Clear ignores the data received from the CPU and sets HS link send data to 0 for transmission. Even if the emergency output data setting of the device area of the high speed link sending side is set to latch in the CPU, the data set to 0 is transmitted when set to clear in the emergency output data setting of XG5000 high speed link setting.
	CPU stop	
Slave Registration*1 (V6.0 or higher)		Select Smart I/O block to be registered in HS link.

You can see that registered high-speed link communication is added as a child of the FENET module. Here, double-click the high-speed link 01 to select the actual transmission area and size.



5. Communication (Ethernet)

[High Speed Link]

A window for creating indexes for communication with multiple stations.

LSPLC - HS Link 01														
Index	Station type	Mode	Station number	Block number	Module Type	Read (Transmit) Area	Variable name	Variable name comment	Read area Word size	Save (Receive) Area	Variable name	Variable name comment	Save area Word size	Diagnostic information device
0														
1														
2														

Classification		Content
Station type	FNet	Select for Ethernet high speed link communication.
	RAPiEnet	Select for RAPiEnet high speed link communication. The driver setting must be set to RAPiEnet in the Standard settings.
	RAPiEnet Remote *1	RAPiEnet Select to control Smart I/O block with high speed link. RAPiEnet Smart I/O block can be set by loading slave in HS link parameter after setting in slave configuration screen. However, RAPiEnet v1 must be enabled in Standard settings.
Mode	Send	Sends data.
	Receive	Receives data.
	Send/Receive*1	It is automatically input when connected to I / O mixed module slave of RAPiEnet Smart I/O block. (Ethernet V6.0 or higher)
Station No		Send means the module station number of own station, and when receiving, it means the module number of other station. It has the range of '0 ~ 63' as the station number. 'Do not use duplicated station number because it is 'own station' or unique number that distinguishes communication module in the same network system.
Block number		Set the send / receive block. * RAPiEnet Smart I/O block is automatically set to the same station number and block number.
Read area		Memory area of own module. Available memory area. 1) XGK: P,M,K,F,T,C,U,Z,L,N,D,R,ZR 2) XGI/XGR: M,I,Q,R,W For the size and range of each memory area, refer to the XGT CPU memory device list in the Appendix.
Variable		The variable name of the device set in the read area is displayed.
Variable Comment		The variable comment of the device set in the read area is displayed.
Read area Word size		Set the data size to be sent in words.
Save area		This area is to receive and store data from the destination station. 1) XGK: P,M,K,F,T,C,U,Z,L,N,D,R,ZR 2) XGI/XGR: M,I,Q,R,W For the size and range of each memory area, refer to the Synchronous start XGT CPU memory device list in the Appendix.
Variable		The variable name of the device set in the save area is displayed.
Variable Comment		The variable comment of the device set in the save area is displayed..
Save area Word size		Set the data size to be receive in words
Diagnostic information device		Displays the diagnostic information device when the use diagnostic area option of the registered slave is enabled.

5. Communication (Ethernet)

[High Speed Link]

From now on, enter each item.

Index 0: Send country code 0 data, receive country code 1 data

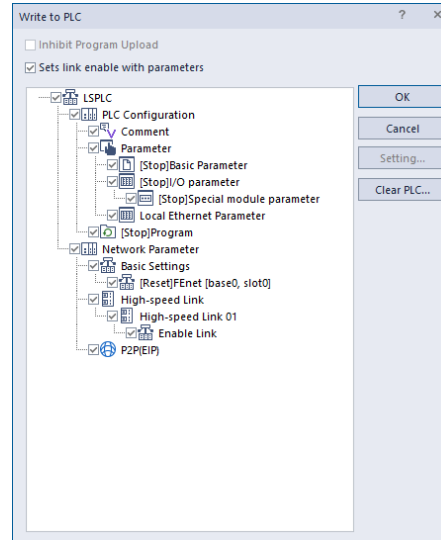
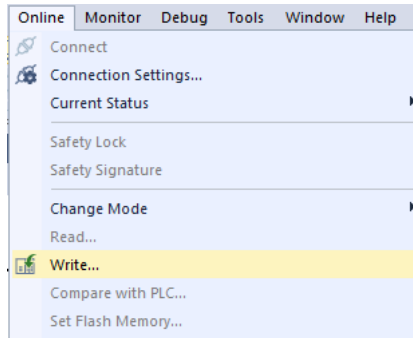
Index 1: Send National Number 1 data. Number 0 country code data received

Type	Settings														
Station 0	LSPLC - HS Link 01 ✕														
	Index	Station type	Mode	Station number	Block number	Module Type	Read (Transmit) Area	Variable name	Variable name comment	Read area Word size	Save (Receive) Area	Variable name	Variable name comment	Save area Word size	Diagnostic information device
	0	FEnet	1. Send	0	0		M0000			2					
	1	FEnet	2. Receive	1	0						M0010			2	
Station 1	LSPLC - HS Link 01 ✕														
	Index	Station type	Mode	Station number	Block number	Module Type	Read (Transmit) Area	Variable name	Variable name comment	Read area Word size	Save (Receive) Area	Variable name	Variable name comment	Save area Word size	Diagnostic information device
	0	FEnet	1. Send	1	0		M0100			2					
	1	FEnet	2. Receive	0	0						M0110			2	

5. Communication (Ethernet)

[High Speed Link]

This completes the creation of the high-speed link parameters. Next, perform a fast link parameter write operation. Under 'Online – Write', check 'High Speed Link 01', then click the OK button.

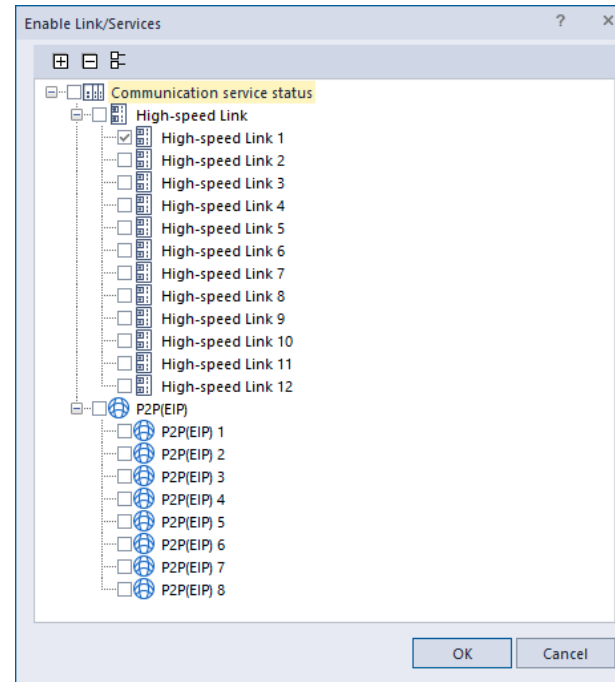
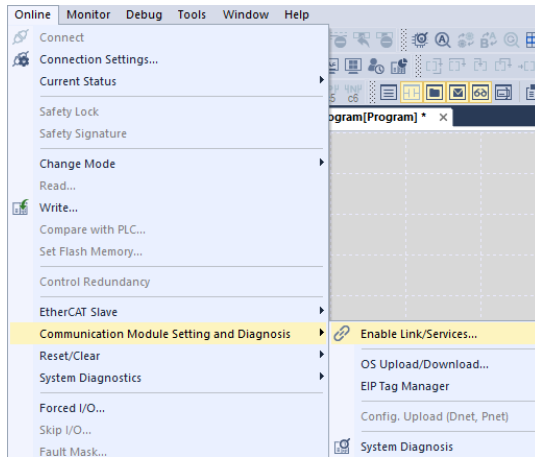


Next changed FENET preference (country number, IP address..)etc.) the PLC communication module must be reset to apply. XGK, XGI maintains the RUN state of the PLC CPU by resetting the individual communication modules and can only change the state of the communication module. Resetting individual modules only needs to be done when the default settings are changed. If the high-speed link parameter is changed, the modified parameter is applied only by writing the parameter.

5. Communication (Ethernet)

[High Speed Link]

The communication is now initiated via 'Online – Communication Module Settings – Link Enable'. In the Reset window below, select the high-speed link number you added and 'write'.



Once the link is enabled, the communication proceeds at the communication cycle you set.

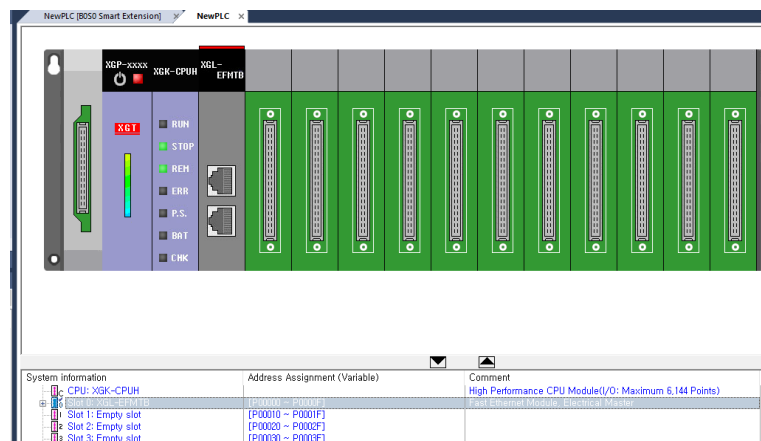
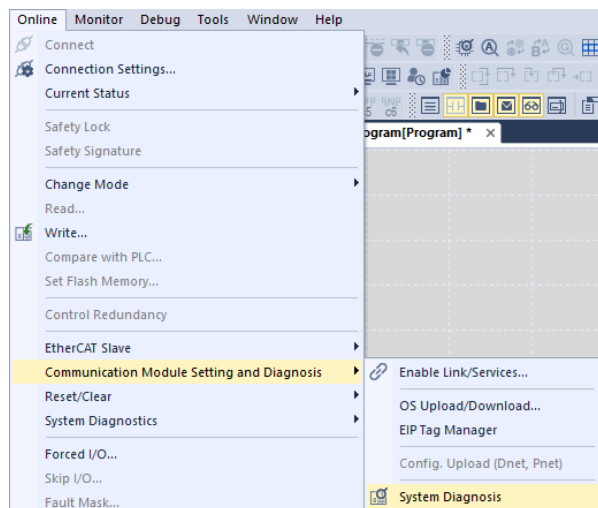
5. Communication (Ethernet)

[High Speed Link]

Check the LEDs on the FENET module to verify the actual communication status.

LED	Contents
ACT	If the Ethernet I/F module transmits and receives data, it is turned on, and if not, it is turned off. The LED flashes when sending and stopping the transmission repeatedly.
LINK	The Ethernet I/F module is ON when a link is formed in the network, otherwise OFF.
HS	LED ON when the high-speed link communication function starts to operate.

It is also possible to verify the communication status through system diagnostics. Click 'Online – Communication Module Setup – System Diagnostics', right-click the appropriate communication module (XGL_EFMTB) and select 'Status by Service'.



5. Communication (Ethernet)

[High Speed Link]

Under Service-Specific Status, you can view the status of index communication created by clicking the High-Link Services tab. The index status for country numbers 0 and 1 is as follows.

Standard Information

Base No.: 0

Slot No.: 0

HS Link Service Information

Send Packets: 5 In normal communication.

Receive Packets: 5 (Run link: 1, Link trouble: 0)

Communication Diagnostics:

Index	Station number	Block number	Block type	Data size	Read area	Save area	High-speed link state	Mode	HS
00	02	02	Slave reception	08		M0010	Normal	Run	No
01	04	04	Slave transmission	04	M0000		Normal	Run	No

Read Refresh

Close

5. Communication (Ethernet)

[High Speed Link]

It also provides a high-speed link flag so that the PLC can check the corresponding communication status. This can be used in the program to check the communication status.

Variables (x: HighSpeedLink No. , yy: Block Index No.)	Type	Descriptions
_HSx_RLINK	Bit	Full information indicating that the high-speed link is running normally
_HSx_LTRBL	Bit	During high-speed link communication, whether trouble occurs
_HSx_STATUSyyy	Bit	Comprehensive service status of the corresponding parameter and block (MOD+TRX+ERR=STATE)
_HSx_MODyyy	Bit	Corresponding parameter, service mode status of block
_HSx_TRXyyy	Bit	Corresponding parameter, service transmission/reception status of the block
_HSx_ERRyyy	Bit	Corresponding parameter, service error status of block
_HSx_SETBLOCKyyy	Bit	Corresponding parameter, setting status of block

1. PLC

6. Communication (Cnet)

6. Communication (Cnet)

[Modbus RTU]

<Standard Setting>

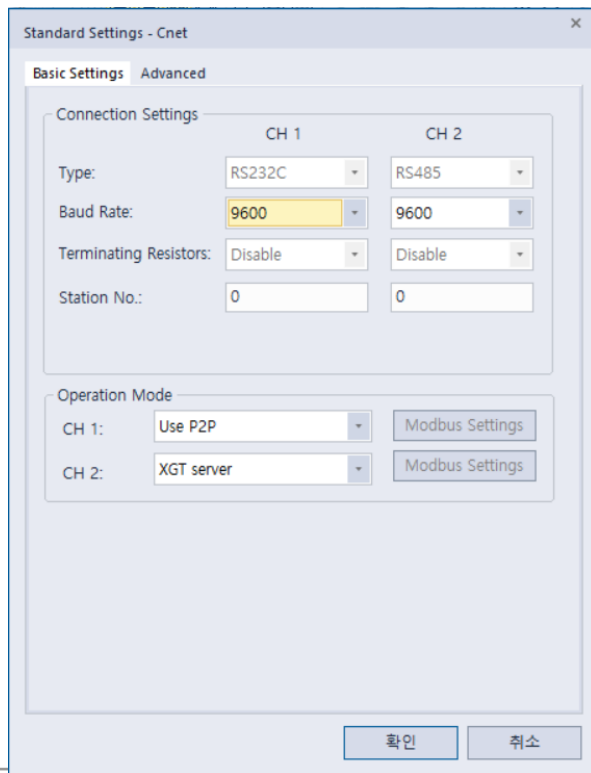
Speed: 19200 bps -> It must same with Server.

Station No.: 0 -> It must be different from the other device (PLC's Station Number).

Operation mode: P2P -> Client must set P2P setting.

<Advanced Setting>

Speed, Parity bit, Stop bit: Must be same with Server.



Standard Settings - Cnet

Basic Settings Advanced

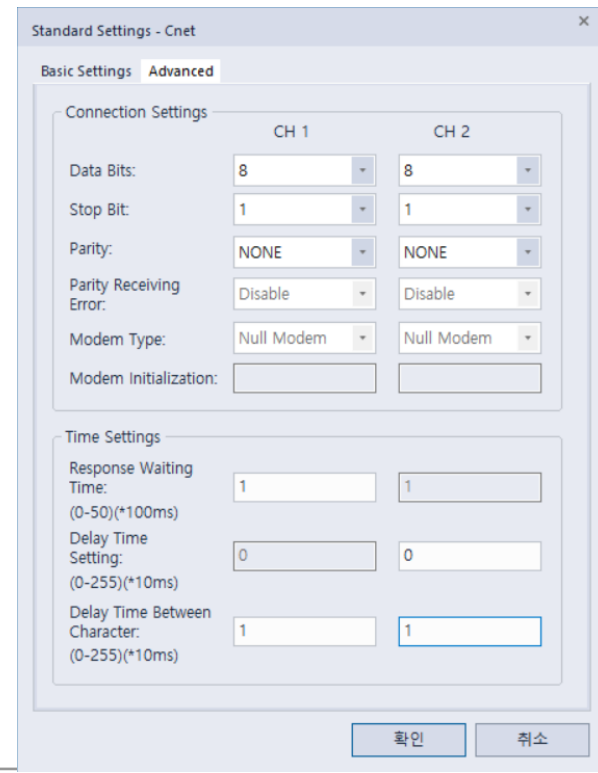
Connection Settings

	CH 1	CH 2
Type:	RS232C	RS485
Baud Rate:	9600	9600
Terminating Resistors:	Disable	Disable
Station No.:	0	0

Operation Mode

CH 1:	Use P2P	Modbus Settings
CH 2:	XGT server	Modbus Settings

확인 취소



Standard Settings - Cnet

Basic Settings Advanced

Connection Settings

	CH 1	CH 2
Data Bits:	8	8
Stop Bit:	1	1
Parity:	NONE	NONE
Parity Receiving Error:	Disable	Disable
Modem Type:	Null Modem	Null Modem
Modem Initialization:		

Time Settings

Response Waiting Time: (0-50)(*100ms)	1	1
Delay Time Setting: (0-255)(*10ms)	0	0
Delay Time Between Character: (0-255)(*10ms)	1	1

확인 취소

6. Communication (Cnet)

[Modbus RTU]

Modbus Address	PLC Address
0x40000	M0000
0x40001	M0001
0x40009	M0009
0x4000A	M0010
0x4000F	M0015
0x40010	M0016

Modbus Address	PLC Address
0x30000	M0010
0x30001	M0011
0x30009	M0019
0x3000A	M0020
0x3000F	M0025
0x30010	M0026

(example 0x40000)

0x : HEX address

4 : Function code is selected using this device. It can read/write data in word units. For write message, function code 16(h10) use.

For read message, function code 03 use.

(For read message, device 3 or 4 available. 3 means function code 04.)

0000 : Modbus address. 0000 is first Modbus address.

Purpose	Function	Modbus Address	Function code	
			Read	Write
Bit Read	Read	0x 1xxxx	01	-
Bit Write	R/W	0x 0xxxx	02	05 (single bit) 0F (multi bit)
Word Read	Read	0x 3xxxx	04	-
Word Write	R/W	0x 4xxxx	03	06 (single word) 16(h10) (multi word)

<Function code>

6. Communication (Cnet)

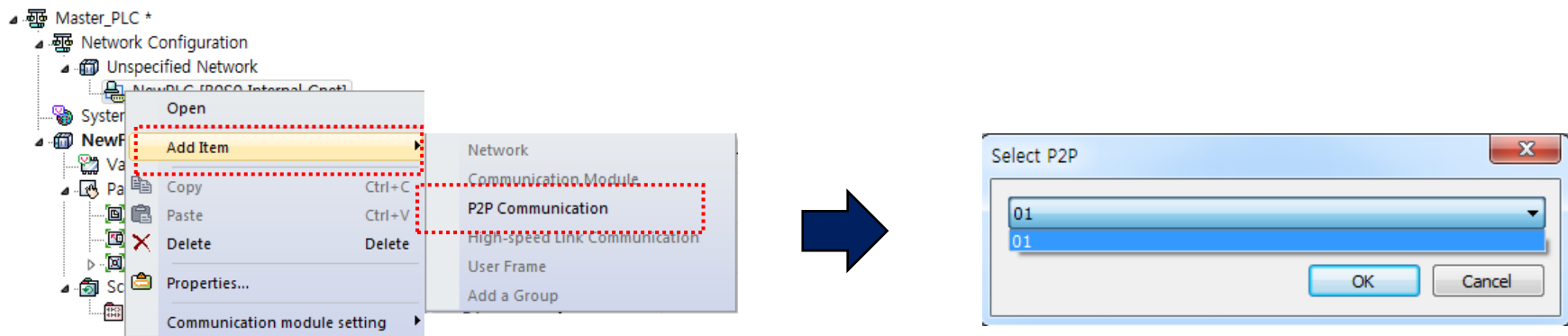
[Modbus RTU]

<P2P Parameter>

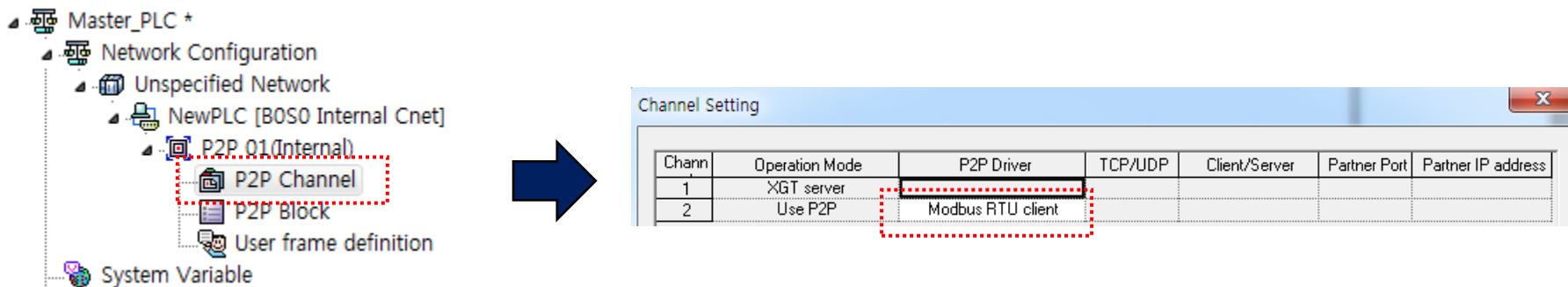
Server PLC don't need P2P parameter setting.

When PLC act as Server, just set standard setting and then write to PLC. Server don't need extra work.

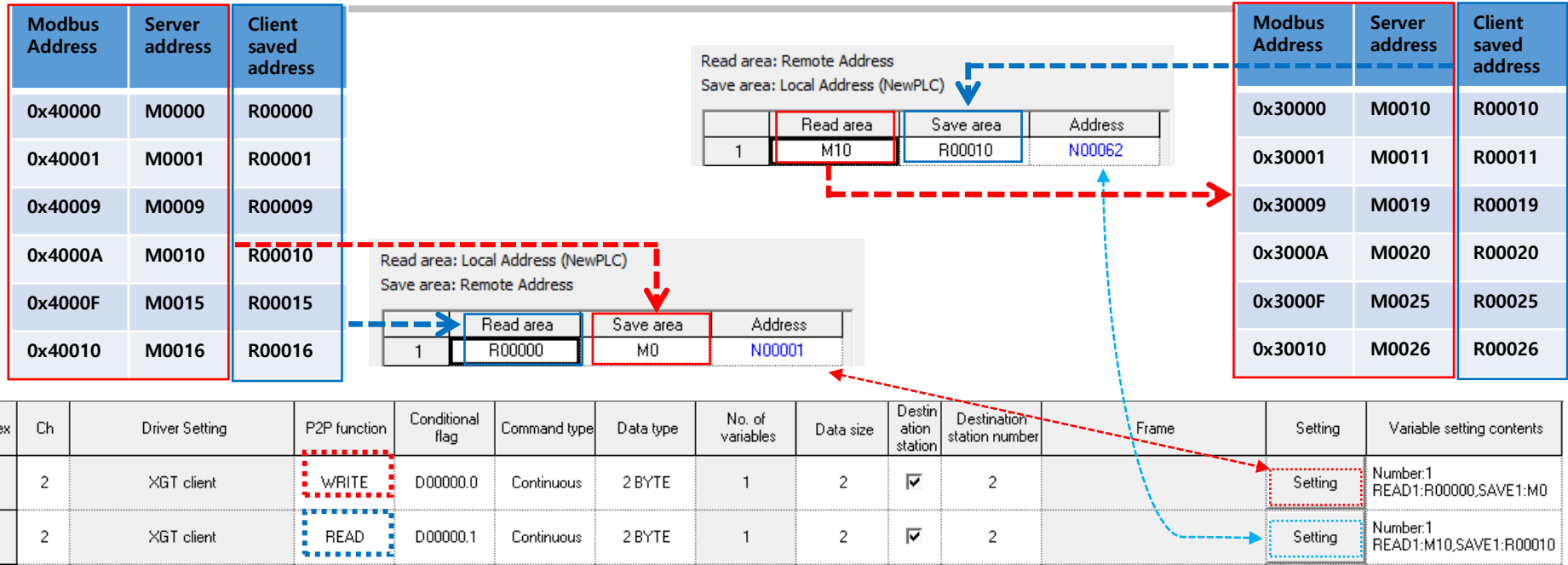
Client PLC's P2P setting step is shown as below.



Built-in Cnet right click -> add item -> P2P communication -> P2P block 01 (Built-in Cnet is fixed always P2P 01)



6. Communication (Cnet)



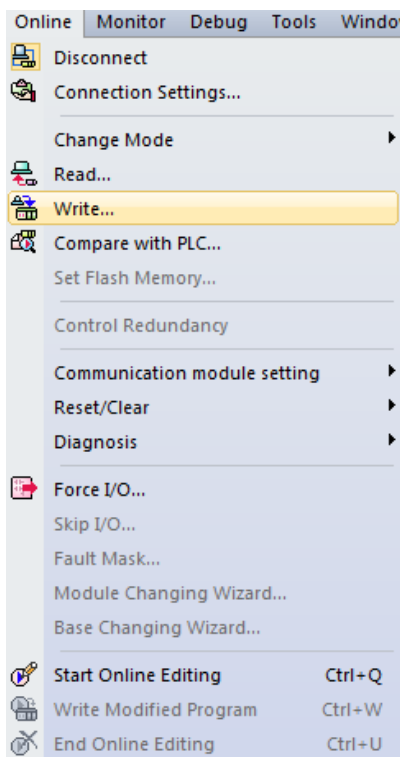
- Index:** Each P2P block has 32 index. XGK, XGI, XGR Cnet has 64 index.
- Ch:** Cnet has two channels. If you choose Modbus RTU client in P2P channel, driver setting set Modbus RTU client automatically.
- P2P function:** You can choose 'Read' or 'Write'.
'Read' means receive and save Server's data. 'write' means send data to Server. In other words control Server.
- Conditional flag:** Enter the time of data transmission by using special flag(F device) or bit contact. When the bit on(rising edge), message send.
- Command type:** (1) Single: Read or write max. four data of memory area to destination station. Ex) M01, M10, M20, M30
(2) Continuous: Read or write continuous data to destination station
Ex) M01~M10

- Data type:** In Modbus RTU/ASCII protocol, choose bit or word.
- Data size:** It is activate when command type is continuous.
- Destination station number:** It means destination station number and setting range is 0~31.
- Setting:**
 - When Read command
 - Read area: Device area of data saved in destination station(Server).
 - Save area: Device area of self station to save the data
 - when write command
 - Read area: Device area of self station to send the data
 - Save area: Device area of data saved in destination station(Server).

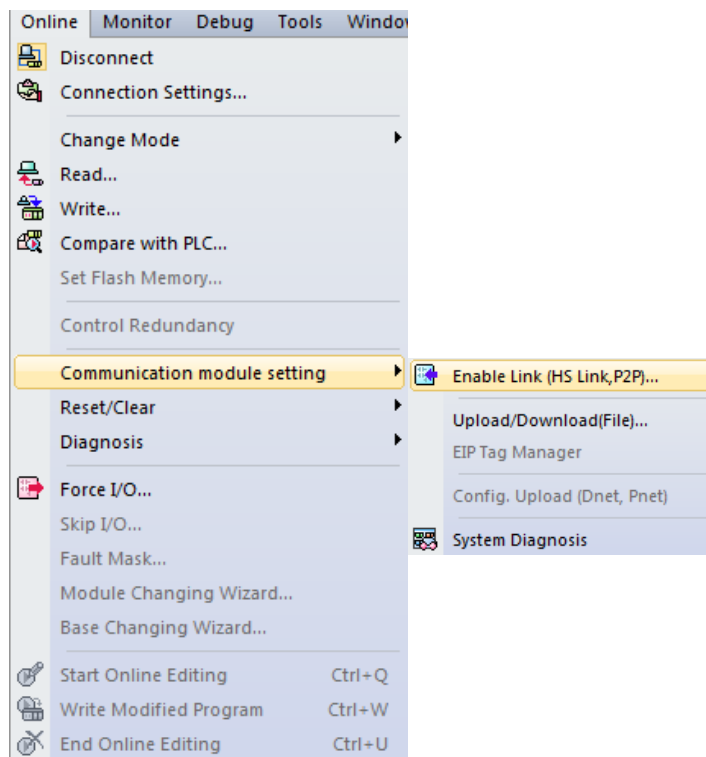
6. Communication (Cnet)

[Modbus RTU]

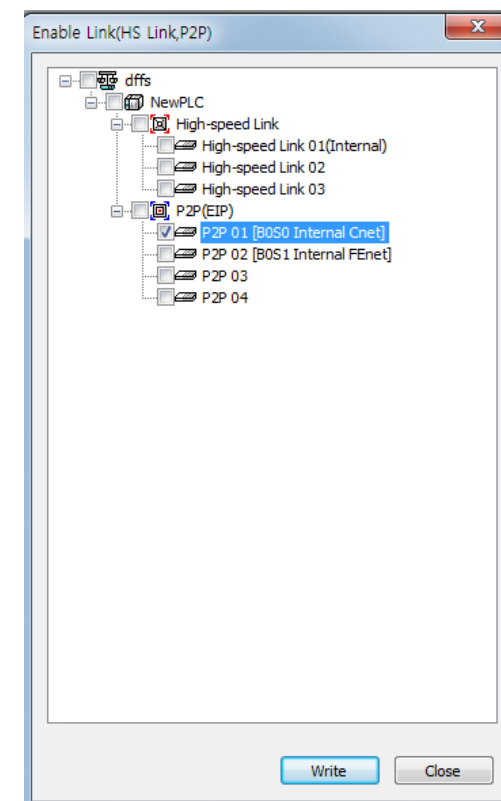
Write to PLC after set standard setting and P2P setting.
It means all changed information apply to PLC. Go to online → write



It needs enable link.
It means unlock the P2P block to use communication as Client.



Check the P2P block that you want to use. After write, communication start.

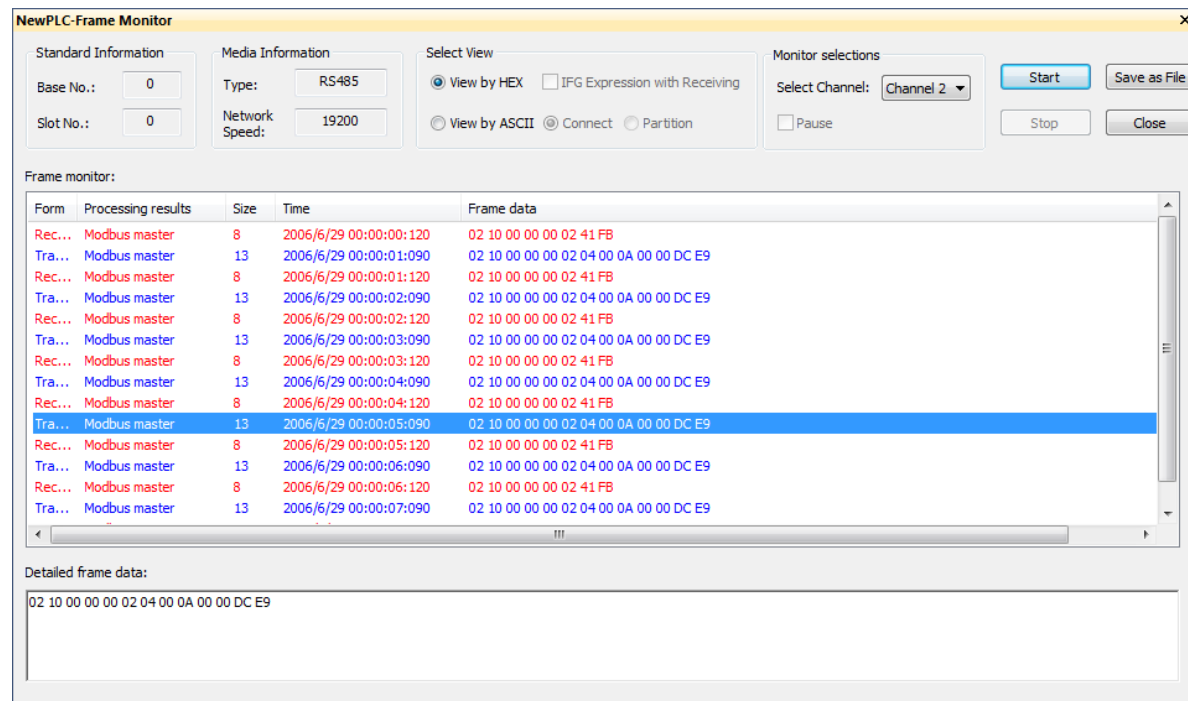
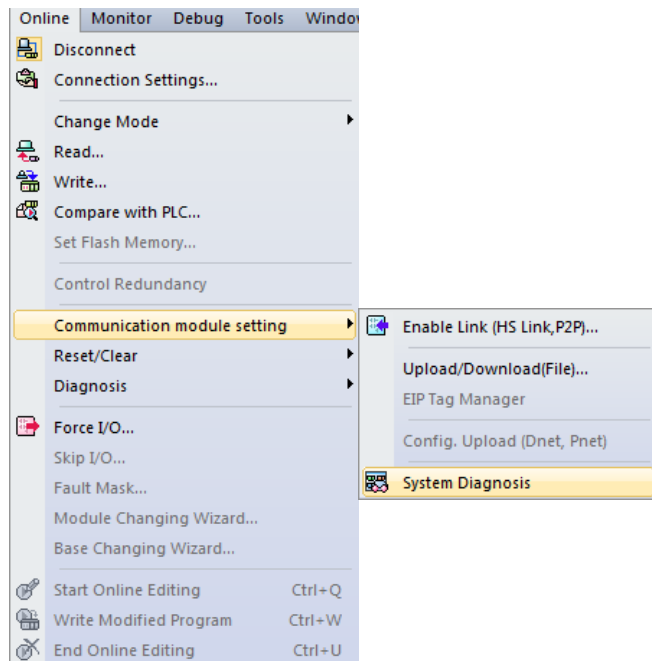


6. Communication (Cnet)

[Modbus RTU]

<Frame Monitor>

After enable link, Client PLC send message to Server. Server response to the message. These step is called data communication. In XG5000 you can diagnosis using frame monitor. Frame monitor shows all message. Go to Online -> Communication module setting -> System Diagnosis -> right click on module -> frame monitor. In frame monitor window select channel and start. And then message shown up. Blue message is transmission(TX). Red message is receive(RX). Also change view type HEX or ASCII. You can save all message as a csv file.

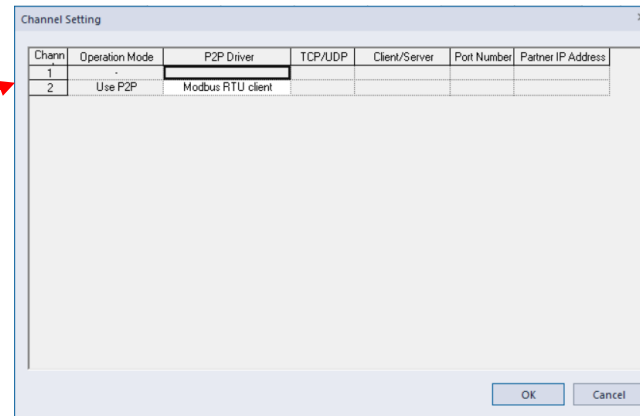
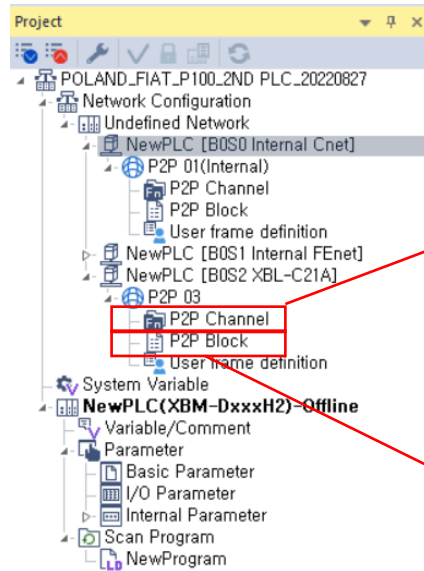


Write request frame monitor

6. Communication (Cnet)

[Modbus RTU]

“POLAND_FIAT_P100_2ND PLC_20220827” PLC Setting (Cnet)



NewPLC - P2P 03												
Index	Channel	Driver Setting	P2P function	Conditional flag	Command type	Data Type	No. of variables	Data size	Destination station	Destination station number	Frame	Variable Setting Content
0	2	Modbus RTU client	READ	P00004	1. Single	WORD	1		☒	1		Setting Number:1 READ1:0x40C83,SAVE1:D00400
1	2	Modbus RTU client	READ	P00006	1. Single	WORD	1		☒	1		Setting Number:1 READ1:0x40C83,SAVE1:D00400
2												Setting
3												Setting
4	2	Modbus RTU client	READ	M00110	1. Single	WORD	1		☒	1		Setting Number:1 READ1:0x40002,SAVE1:D00200
5												Setting
6												Setting
7	2	Modbus RTU client	READ	M00210	1. Single	WORD	1		☒	1		Setting Number:1 READ1:0x40008,SAVE1:D00210
8												Setting
9												Setting
10	2	Modbus RTU client	READ	M00310	1. Single	WORD	1		☒	1		Setting Number:1 READ1:0x30C84,SAVE1:D00310

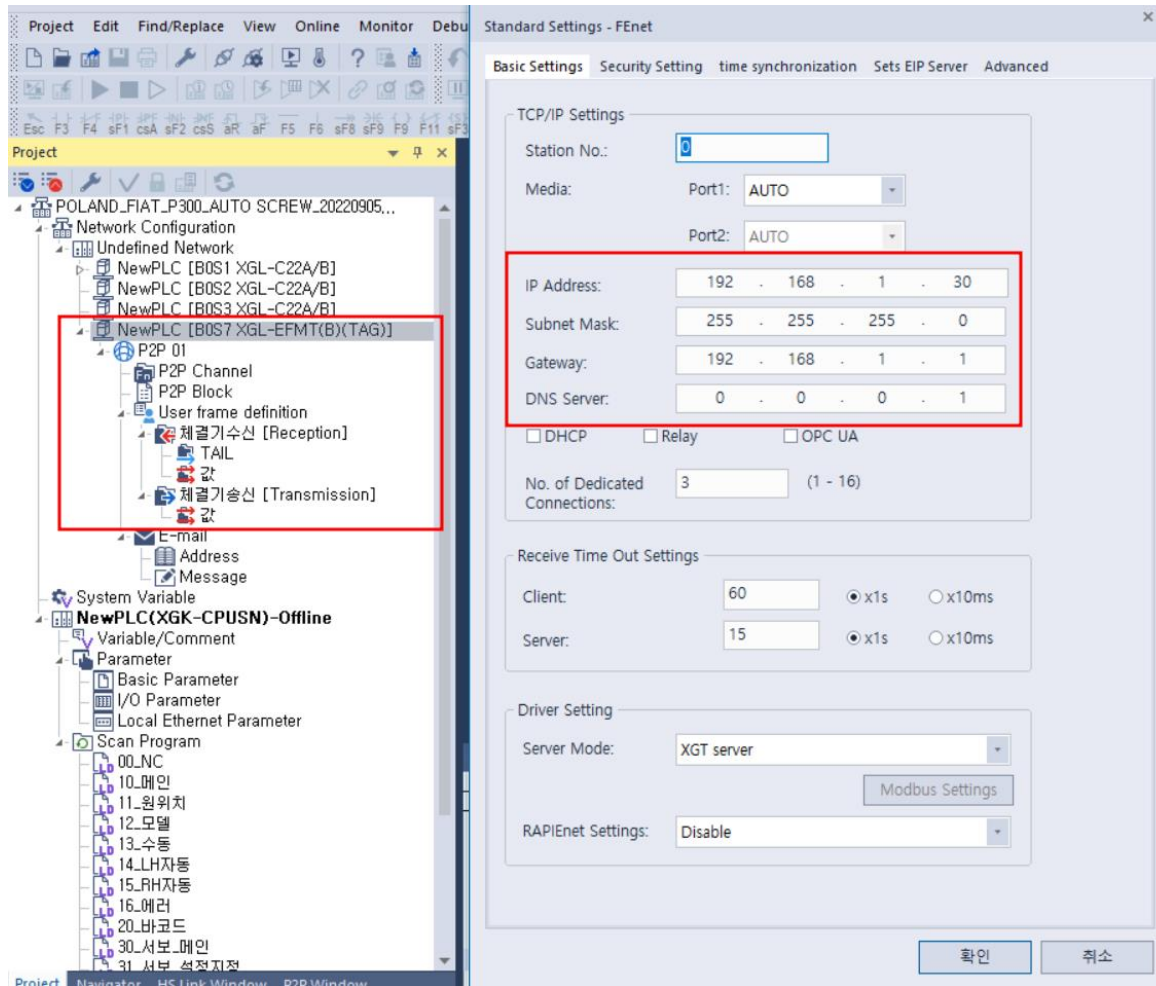
1. PLC

7. Communication (User Defined Protocol)

7. Communication (User Defined Protocol)

[User Defined Protocol]

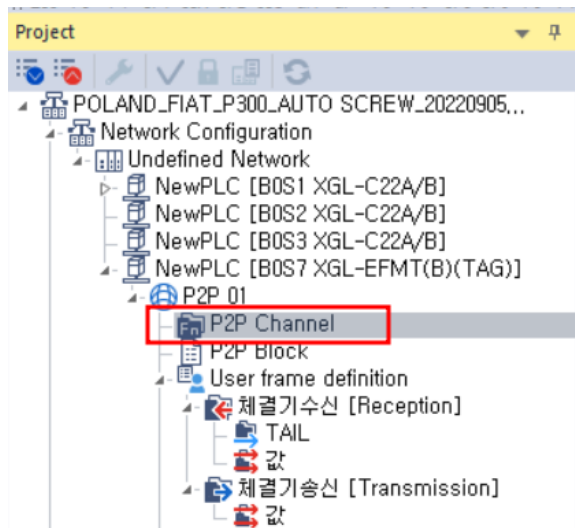
POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트



7. Communication (User Defined Protocol)

[User Defined Protocol]

POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트

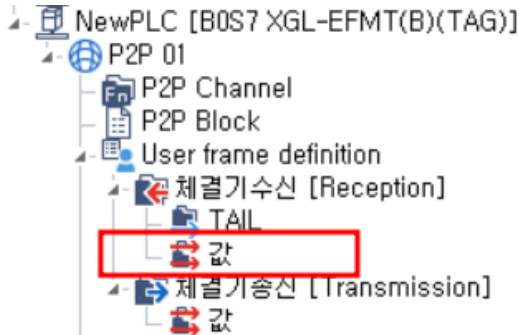


Channel Setting						
Chann	Operation Mode	P2P Driver	TCP/UDP	Client/Server	Port Number	Partner IP Address
0		Modbus TCP client	TCP	Client	502	192.168.1.32
1		User frame definition	TCP	Client	4545	192.168.1.33
2						0.0.0.1
3						0.0.0.1
4						0.0.0.1

7. Communication (User Defined Protocol)

[User Defined Protocol]

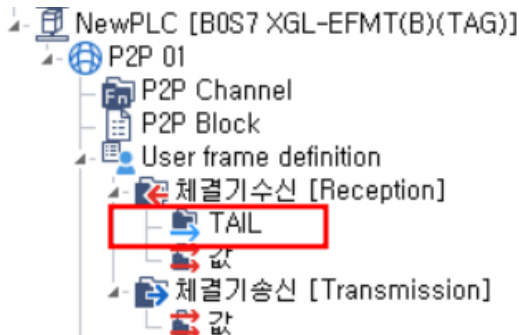
POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트



NewPLC P2P 01 Frame - 체결기수신.값

Up Down Add BCC Add Line Delete Line

	Segment	Frame	Size	Data	Assign memory	Conversion	Swap
	00	Fix sized variable	4		☐		
	01	Variable sized variable			☒	NONE	NONE

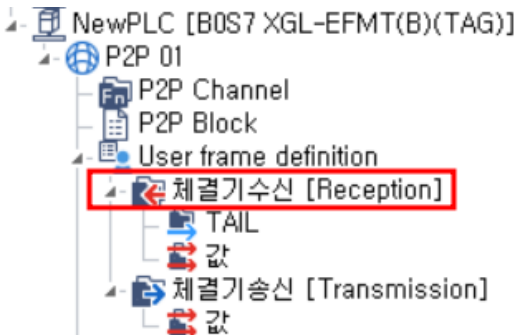


NewPLC P2P 01 Frame - 체결기수신.값

NewPLC P2P 01 Frame - 체결기수신.TAIL

Up Down Add BCC Add Line Delete Line

	Segment	Frame	Size	Data	Assign memory
	00	Numerical constant	1	00	☐



NewPLC P2P 01 Frame - 체결기수신.값

NewPLC P2P 01 Frame - 체결기수신.TAIL

NewPLC P2P - 체결기수신 [Reception]

	Segment	Field	Frame	Size	Data	Assign memory	Conversion	Swap
1	00	TAIL	Numerical constant	1	00	☐		
2	00	값	Fix sized variable	4		☐		
3	01	값	Variable sized variable			☒	NONE	NONE

7. Communication (User Defined Protocol)

[User Defined Protocol]
 POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트

NewPLC [B0S7 XGL-EFMT(B)(TAG)]

P2P 01

P2P Channel

P2P Block

User frame definition

채결기수신 [Reception]

TAIL

값

채결기송신 [Transmission]

값

NewPLC P2P 01 Frame - 채결기수신.값

NewPLC P2P 01 Frame - 채결기수신.TAIL

NewPLC P2P - 채결기수신 [Reception]

NewPLC P2P 01 Frame - 채결기송

Up

Down

Add BCC

Add Line

Delete Line

	Segment	Frame	Size	Data	Assign memory	Conversion	Swap
	00	String Constant	4	0020	☐		
	01	Variable sized variable	8		☒	NONE	NONE
	02	Numerical constant	8	2020202020202020	☐		
	03	Numerical constant	1	00	☐		

NewPLC [B0S7 XGL-EFMT(B)(TAG)]

P2P 01

P2P Channel

P2P Block

User frame definition

채결기수신 [Reception]

TAIL

값

채결기송신 [Transmission]

값

NewPLC P2P - 채결기송신 [Transmission]

NewPLC P2P 01 Frame - 채결기수신.값

NewPLC P2P 01 Frame - 채결기수신.TAIL

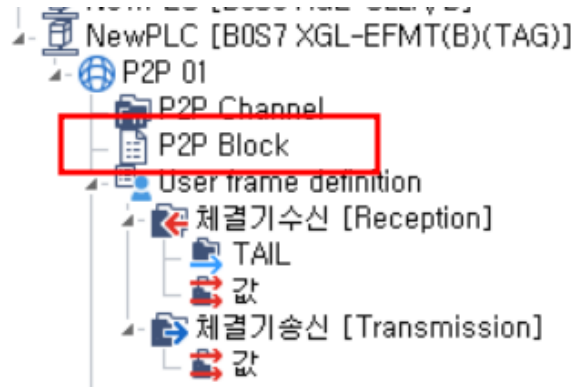
NewPLC P2P - 채결기수신 [Reception]

	Segment	Field	Frame	Size	Data	Assign memory	Conversion	Swap
1	00	값	String Constant	4	0020	☐		
2	01	값	Variable sized variable	8		☒	NONE	NONE
3	02	값	Numerical constant	8	2020202020202020	☐		
4	03	값	Numerical constant	1	00	☐		

7. Communication (User Defined Protocol)

[User Defined Protocol]

POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트



NewPLC - P2P 01															
Index	E-mail	Channel	Driver Setting	Header	P2P function	Conditional flag	Command type	Data Type	No. of variables	Data size	Destination station	Destination station number	Frame	Setting	Variable Setting Content
0	☐	0	Modbus TCP client		READ	_T200MS	2. Continuous	WORD	1		☒	1		Setting	Number:1 READ1:0x30100,SAVE1:D000500,SIZE1:2
1	☐	1	User frame definition		RECEIVE								채결기수신 값	Setting	Number:1 SAVE1:D007100
2	☐	1	User frame definition		SEND	M01900							채결기송신 값	Setting	Number:1 READ1:D007000,SIZE1:8

7. Communication (User Defined Protocol)

[User Defined Protocol]

POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트

The screenshot displays a project configuration window with a tree view on the left and a 'Standard Settings - Cnet' dialog box on the right. The tree view shows a project named 'POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트' under 'Network Configuration'. A red box highlights the 'Undefined Network' section, which contains a 'NewPLC [B0S1 XGL-C22A/B]' node. This node has a 'P2P 02' sub-node, which further contains 'P2P Channel', 'P2P Block', 'User frame definition', 'L_BAR [Reception]', 'TAIL', 'LH', 'R_BAR [Reception]', 'TAIL', and 'RH'. Below this, there are three more 'NewPLC' nodes: 'NewPLC [B0S2 XGL-C22A/B]', 'NewPLC [B0S3 XGL-C22A/B]', and 'NewPLC [B0S7 XGL-EFMT(B)(TAG)]'. The 'Standard Settings - Cnet' dialog box is open, showing 'Basic Settings' and 'Advanced' tabs. The 'Connection Settings' section is active, showing 'Type' as 'RS232C' for both 'CH 1' and 'CH 2'. The 'CH 1' 'Port' is set to '9600' and 'CH 2' 'Port' is set to 'Disable'. The 'Modbus Settings' section is also visible, with 'Modbus Settings' buttons for both channels. At the bottom of the dialog are '확인' (OK) and '취소' (Cancel) buttons.

The screenshot shows a 'Channel Setting' dialog box with a table containing two channels. The table has columns for 'Chann', 'Operation Mode', 'P2P Driver', 'TCP/UDP', 'Client/Server', 'Port Number', and 'Partner IP Address'. Channel 1 is configured with 'Use P2P' operation mode, 'User frame definition' P2P driver, and an empty 'Partner IP Address' field. Channel 2 is also configured with 'Use P2P' operation mode, 'User frame definition' P2P driver, and an empty 'Partner IP Address' field.

Chann	Operation Mode	P2P Driver	TCP/UDP	Client/Server	Port Number	Partner IP Address
1	Use P2P	User frame definition				
2	Use P2P	User frame definition				

7. Communication (User Defined Protocol)

[User Defined Protocol]
POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트

NewPLC [B0S1 XGL-C22A/B]

P2P 02

P2P Channel

P2P Block

User frame definition

L_BAR [Reception]

TAIL

LH

R_BAR [Reception]

TAIL

RH

NewPLC [B0S2 XGL-C22A/B]

NewPLC [B0S3 XGL-C22A/B]

NewPLC [B0S7 XGL-EFMT(B)(TAG)]

NewPLC [B0S1 XGL-C22A/B]

P2P 02

P2P Channel

P2P Block

User frame definition

L_BAR [Reception]

TAIL

LH

R_BAR [Reception]

TAIL

RH

NewPLC [B0S2 XGL-C22A/B]

NewPLC [B0S3 XGL-C22A/B]

NewPLC [B0S7 XGL-EFMT(B)(TAG)]

NewPLC [B0S1 XGL-C22A/B]

P2P 02

P2P Channel

P2P Block

User frame definition

L_BAR [Reception]

TAIL

LH

R_BAR [Reception]

TAIL

RH

NewPLC [B0S2 XGL-C22A/B]

NewPLC [B0S3 XGL-C22A/B]

NewPLC [B0S7 XGL-EFMT(B)(TAG)]

NewPLC P2P 02 Frame - L_BAR.LH

NewPLC P2P 02 Frame - R_BAR.RH

NewPLC P2P 01 Frame - 체결기수신.값

NewPLC P2P 01 Frame - 체결기수신

Up

Down

Add BCC

Add Line

Delete Line

	Segment	Frame	Size	Data	Assign memory	Conversion	Swap
	00	Variable sized variable			☒	NONE	NONE

NewPLC P2P 02 Frame - L_BAR.TAIL

NewPLC P2P 02 Frame - L_BAR.LH

Up

Down

Add BCC

Add Line

Delete Line

	Segment	Frame	Size	Data	Assign memory
	00	Numerical constant	2	0D0A	☐

NewPLC P2P - L_BAR [Reception]

NewPLC P2P 01 Frame - 체결기수신.값

NewPLC P2P 01 Frame - 체결기수신.TAIL

NewPLC P2P - 체결기수신 [Reception]

	Segment	Field	Frame	Size	Data	Assign memory	Conversion	Swap
1	00	TAIL	Numerical constant	2	0D0A	☐		
2	00	LH	Variable sized variable			☒	NONE	NONE

LS ELECTRIC

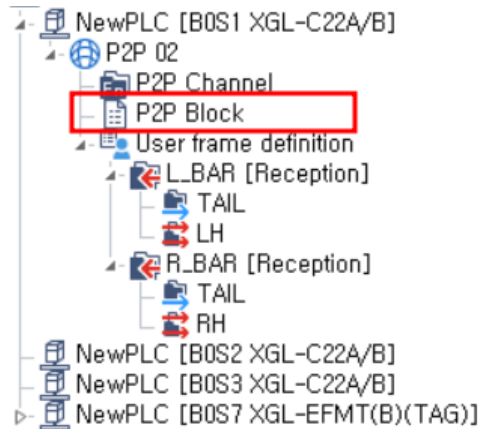
185

FUTURING SMART ENERGY

7. Communication (User Defined Protocol)

[User Defined Protocol]

POLAND_FIAT_P300_AUTO SCREW_20220905_통신테스트



NewPLC - P2P 02													
Index	Channel	Driver Setting	P2P function	Conditional flag	Command type	Data Type	No. of variables	Data size	Destination station	Destination station number	Frame	Setting	Variable Setting Content
0	1	User frame definition	RECEIVE								L_BAR.LH	Setting	Number:1 SAVE1:D008000
1	2	User frame definition	RECEIVE								R_BAR.RH	Setting	Number:1 SAVE1:D008200

3. XGI-CPUZ

1. Overview

1. Overview

- 1 All-in-One Controller Provides Sequence and Motion Controllers at the same time
- 2 Simulation-based high reliability power modules: 67% efficiency -> 90% or higher
- 3 IIoT Technical Support: Web Server, Cloud, FTP, Wireless Solutions, etc
- 4 Development of three strategic market response lineups, including automotive battery pack manufacturing and motor manufacturing processes:
XGI-CPUZ3, XGI-CPUZ5, XGI-CPUZ7



1. Overview

- 2** Operational capabilities are 2.4x better, and on-site responsiveness is improved by providing a variety of commands
- For example, providing socket commands makes it possible to easily respond to third-party communication protocols.



Category		XGI-CPUZ3	XGI-CPUZ5	XGI-CPUZ7	Remarks
Operation Method		Main task/Cycle task: Fixed period and cyclic operation Initialization task: Only once at the time of entering the RUN			
Control Cycle		Main task time: 1ms ~ 1,000ms (1ms unit setting) Cycle task time: Multiple setting of main task (2~4,000ms)			
Calculation processing speed (Basic Instructions)	LD	6.5 ns/command			Maximum performance condition
	MOVE	5.2 ns/command			Maximum performance condition
	Real calculation	±: 31 ns(S), 121 ns(D) x : 31 ns(S), 119 ns(D) ÷ : 34 ns(S), 142 ns(D)			S : Short real D : Long real Maximum performance condition
Program memory	Capacity	2MB: seq+motion 5MB: NC control;	4MB: seq+motion 5MB: NC control	10MB: seq+motion 10MB: NC control	
Motion control	Real/Virtual Axis	Axis 8	Axis 16	Axis 32	
	Dedicated virtual Axis	Axis 1	Axis 2	Axis 4	
	Slave (Real Axis)	16 Slave	32 Slave	64 Slave	
External memory	Memory Type	Micro SD/SDHC			
	Service	Program back-up/restoration, booting operation			
Built-in communication port	USB Device	Loader service (XG5000)			
	USB Host	Loader service (XG5000), Web server			Supported devices: WiFi Dongle
	Ethernet	Loader service (XG5000) (Multi-connection support, up to 4 people including USB) (XGT server) – Dedicated communication 3rd party protocol support (MODBUS TCP server) FTP server / NTP client / Web Server Socket Service (LS electric and third-party client service correspondence)			
Security		Project password TLS support: Loader service, Web server, FTP server			

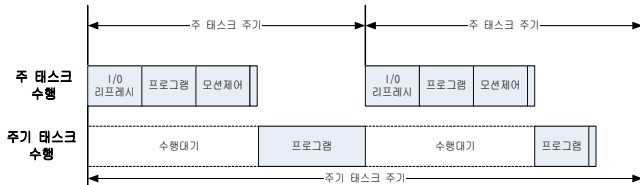
3. XGI-CPUZ

2. Specification

2. Specification

Performance

1. Precision control period provided: minimum period 1 ms, precision: $\leq 10\mu s$



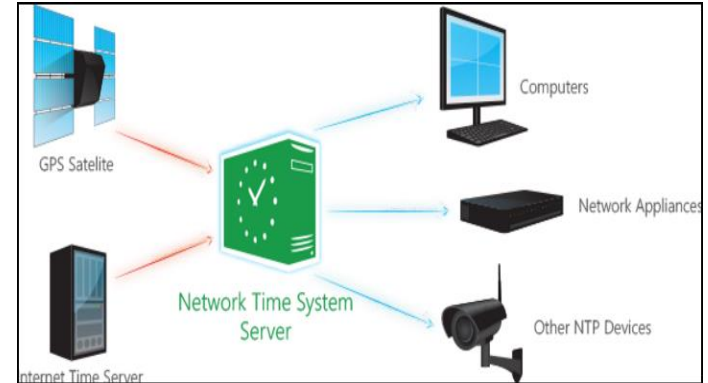
2. Faster processing

Test Method	Max. SCAN Time		Rate of Improvement
	XGI-CPUUN *	XGI-CPUZ7	
ADD 10,000 (Word)	8.9ms	3.7ms	x 2.4
LD/ST 10,000 (Bit)	1.9ms	1.1ms	x 1.7

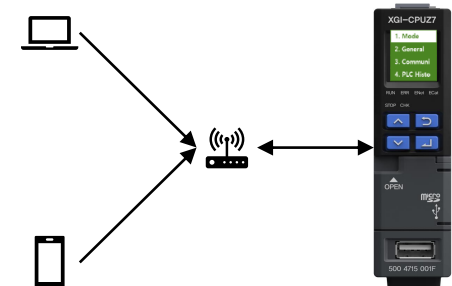


Connectivity

3. NTP Time Synchronization Settings:
Synchronize system time using NTP server



4. Wireless Solutions



2. Specification

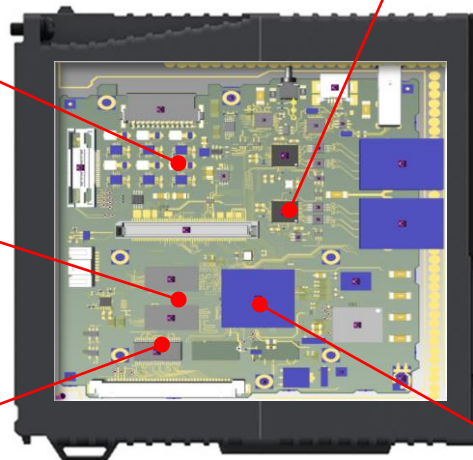
Rapid response and prevention by providing a variety of diagnostic functions

Diagnostics

1. I/O module check function

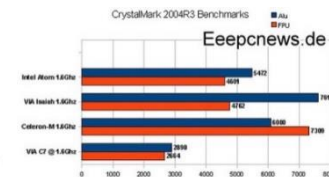
2. PBM(Power Backup Module) check function

3. Error history saving function

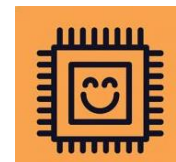


Diagnostics

4. Error checking function in system configuration



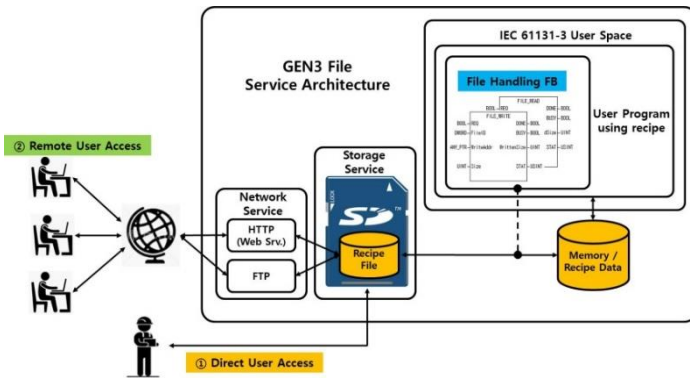
5. Health monitoring: CPU module temperature/humidity check function, MCU temperature check function



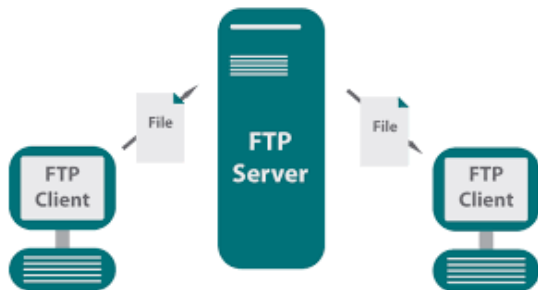
2. Specification

Convenience

1. Provides a variety of features, including multiple backups using SD Card



2. File-by-file data sharing with FTP server delivery

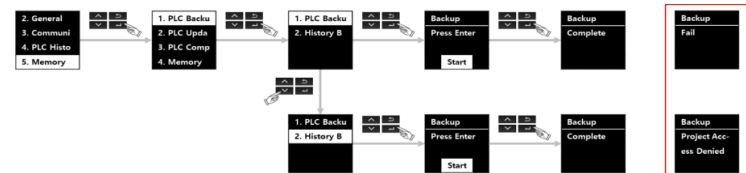


Convenience

3. Web server



4. Built-In UI Display: Remote Control without XG5000 (Diagnosis, Control)



2. Specification

Security

1. 1. Security Enhancements

1) XG5000 Secure Access: Applying TLS Encryption Protocol

세부 사항

Ethernet

IP 주소 설정

☒ IP 주소(I): 192 . 168 . 250 . 100

☐ 호스트 이름(H):

☐ 사용자 지정 포트

포트 설정: 2002

경고: 공유기를 이용한 포트포워딩시에만, 포트 번호를 바꿔주시기 바랍니다.

☒ TLS 사용

IP 찾기(S) OK Cancel

2) FTP Server security Access

기본 설정 - FNet

기본 설정 보안 설정 시간 동기화 FTP 웹 서버

☒ FTP 서버 기능 활성화

☒ 보안 프로토콜 사용(HTTPS)

사용자 ID: LS

비밀번호: *****

☐ 비밀번호 보이기

타일아웃 시간: 1 분(1 - 5)

OK Cancel



Security

3. Web server Security

기본 설정 - FNet

기본 설정 보안 설정 시간 동기화 FTP 웹 서버

☒ 웹 서버 사용

☒ 보안 프로토콜 사용(HTTPS)

관리자 비밀번호: *****

사용자1 비밀번호: *****

사용자2 비밀번호: *****

사용자3 비밀번호: *****

사용자4 비밀번호: *****

☐ 비밀번호 보이기

OK Cancel

4. Permission list setting function

기본 설정 - FNet

기본 설정 보안 설정 시간 동기화 FTP 웹 서버

☒ 할당 허용 + 설정 목록

타입	동작 모드	시작(초)	종(초)
1	허용	192.168.1.1	192.168.1.254
2	허용	192.168.2.3	
3			
4	허용	3000	5000
5	허용	7777	
6			
7	허용	00:02:29 FF FF 00	00:02:29 FF FF FF
8	허용	00:02:29 FF 13 33	

☐ 사용자 지정 포트

포트 번호	추가 포트	계수
205000 포트(TCP)		
205000 보안 포트(TCP)		
2051 보안 포트(TCP)		
2051 보안 포트(TCP)		
2051 보안 포트(TCP)		
2051 보안 포트(TCP)		
2051 보안 포트(TCP)		
2051 보안 포트(TCP)		
2051 보안 포트(TCP)		

OK Cancel

2. Specification



Convenience

1. FoE(File Access over EtherCAT)
Servo F/W : Each Firmware Update → simultaneous Firmware Update



2. Firmware Update with SD memory without the XG5000 (for internal users)

