

EasyBuilder Pro

Ver. 5.02.01

Copyright © 2013 Weintek Labs., Inc. All rights reserved.

1.	Easy	Build	er Pro Installation and Startup Guide	1-1
	1.1.	Insta	allation Requirements	1-2
	1.2.	Step	s to Install EasyBuilder Pro	1-3
2.	Utili	ty Ma	anager	2-1
	2.1.	Ove	rview	2-2
	2.2.	HMI	IP, Password	2-4
	2.3.	Editi	ing Tools	2-5
	2.3	.1.	Build Download Data for SD/USB Disk	2-5
	2.3	.2.	Steps to Download Project to HMI via USB Disk or SD Card	2-5
	2.4.	Tran	sfer	2-6
	2.4	.1.	Download	2-6
	2.4	.2.	Upload	2-7
	2.5.	Simu	ulation	2-8
	2.5	.1.	Off-line Simulation / On-line Simulation	2-8
	2.6.	Pass	-Through	2-9
3.	Crea	ite ar	a EasyBuilder Pro Project	3-1
	3.1.	Ove	rview	3-2
	3.2.	Crea	ite a New Project File	3-2
	3.3.	Save	and Compile the Project File	3-4
	3.4.	Run	On-Line or Off-Line Simulation	3-5
	3.5.	cMT	Viewer	3-5
	3.6.	Dow	nload the Project File to HMI	3-6
	3.6	.1.	Configure in EasyBuilder Pro	3-6
	3.6	.2.	Use HMI Name	3-8
	3.6	.3.	Use USB cable	3-9
	3.6	.4.	Use USB Drive or SD Card	3-9
4.	Hard	dware	e Settings	4-1
	4.1.	Ove	rview	4-2
	4.2.	I/O I	Ports	4-2
	4.3.	LED	Indicators	4-2
	4.4.	Syst	em Reset	4-3
	4.5.	Syst	em Toolbar	4-4
	4.5	.1.	System Setting	4-4
	4.5	.2.	System Information	4-8
	4.6.	Easy	SystemSetting	4-8
5.	Syst	em Pa	arameter Settings	5-1
	5.1.	Ove	rview	5-2
	5.2.	Devi	ce	5-2



5.2.1. 5.2.2.		.1.	How to Control a Local PLC	5-3
		.2.	How to Control a Remote PLC	5-7
	5.2.3.		How to Control a Remote HMI	5-9
	5.3.	Mod	lel	5-11
	5.4.	Gene	eral	5-13
	5.5.	Syste	em Setting	5-16
	5.6.	Secu	ırity	5-18
	5.6	.1.	General Mode	5-18
	5.6	.2.	Enhanced Security Mode	5-19
	5.7.	Font		5-20
	5.7	.1.	eMT, iE, XE, mTV Series	5-20
	5.7	.2.	cMT Series	5-21
	5.8.	Exte	nded Memory	5-22
	5.9.	Print	ter / Backup Server	5-24
	5.10.	Time	e Sync./DST	5-25
	5.11.	e-Ma	ail	5-26
	5.12.	Reci	pes	5-28
6.	Wine	dow (Operations	6-1
	6.1.	Over	rview	6-2
	6.2.	Wind	dow Types	6-2
	6.2.1.		Base Window	6-2
	6.2.2.		Fast Selection Window	6-2
	6.2	.3.	Common Window	6-3
	6.2	.4.	System Message Window	6-3
	6.3.	Crea	te, Set, and Delete a Window	6-4
	6.3	.1.	Creating and Setting a Window	6-5
	6.3	.2.	Open, Close, and Delete a Window	6-7
7.	Even	it Log		7-1
	7.1.	Over	rview	7-2
	7.2.	Even	it Log Management	7-2
	7.2	.1.	eMT, iE, XE, mTV Series	7-2
	7.2.2.		cMT Series	7-4
	7.2	.3.	Excel Editing	7-5
	7.3.	Crea	ting a New Event Log	7-6
8.	Data	Sam	pling	8-1
	8.1.	Over	rview	8-2
	8.2.	2. Data Sampling Management		8-2
	8.3.	Crea	ting a New Data Sampling	8-2



8.3	3.1.	Demonstration of Auto. stop	8-6
8.3	3.2.	Customized File Handling	8-6
8.4.	Syn	chronizing cMT Viewer data and Saving to External Device	
8.5.	Che	cking History Data of a Specific Date on cMT Viewer	8-9
9. Obj	ect G	eneral Properties	9-1
9.1.	Ove	rview	
9.2.	Sele	cting PLC and Setting Read/Write Address	
9.3.	Usir	ng Shape Library and Picture Library	
9.3	3.1.	Shape manager	9-5
9.3	3.2.	Pictures manager	9-5
9.4.	Sett	ing Label Text	
9.5.	Adju	usting Profile Size	
10. Use	r Pas	sword and Object Security	10-1
10.1.	Ove	rview	10-2
10.2.	Use	r Password and Operable Object Classes	10-2
10	.2.1.	General Mode	10-2
10	.2.2.	Enhanced Security Mode	10-3
10.3.	Enh	anced Security Mode and Control Address	10-4
10	.3.1.	Control Address Settings	10-4
10	.3.2.	Commands	10-5
10	.3.3.	Command Execution Results	10-6
10.4.	Enh	anced Security Mode Usage	10-7
10	.4.1.	Importing User Accounts	10-7
10	.4.2.	Login with USB Security Key	10-8
10	.4.3.	Login / Logout Automatically with USB Security Key	10-10
10	.4.4.	Enhanced Security Mode with Option List Object	10-11
10.5.	Obj	ect Security Settings	10-12
10.6.	Exai	nple of Object Security Settings	10-13
11. Inde	ex Re	gister	11-1
11.1.	Ove	rview	11-2
11.2.	Exai	nples of Index Register	11-2
12. Key	board	Design and Usage	12-1
12.1.	Ove	rview	12-2
12.2.	Step	os to Design a Popup Keyboard	12-2
12.3.	Step	os to Design a Keyboard with Direct Window	12-5
12.4.	Step	os to Design a Fixed Keyboard on Screen	12-6
12.5.	Step	os to Design a UNICODE Keyboard	12-7
13. Obj	ects.		



13.1. Bit Lamp	
13.1.1. Overview	13-3
13.1.2. Configuration	13-3
13.2. Word Lamp	
13.2.1. Overview	13-5
13.2.2. Configuration	13-5
13.3. Set Bit	
13.3.1. Overview	13-9
13.3.2. Configuration	13-9
13.4. Set Word	
13.4.1. Overview	13-11
13.4.2. Configuration	13-11
13.5. Function Key	
13.5.1. Overview	
13.5.2. Configuration	
13.6. Toggle Switch	
13.6.1. Overview	13-22
13.6.2. Configuration	13-22
13.7. Multi-State Switch	
13.7.1. Overview	13-24
13.7.2. Configuration	13-24
13.8. Slider	
13.8.1. Overview	13-27
13.8.2. Configuration	13-27
13.9. Numeric	
13.9.1. Overview	13-31
13.9.2. Configuration	13-31
13.10. ASCII	
13.10.1. Overview	
13.10.2. Configuration	
13.11. Indirect Window	
13.11.1. Overview	13-47
13.11.2. Configuration	13-47
13.12. Direct Window	
13.12.1. Overview	13-50
13.12.2. Configuration	13-50
13.13. Moving Shape	
13.13.1. Overview	



13.13.2. Configuration	13-53
13.14. Animation	
13.14.1. Overview	13-57
13.14.2. Configuration	13-57
13.15. Bar Graph	
13.15.1. Overview	13-61
13.15.2. Configuration	13-61
13.16. Meter Display	
13.16.1. Overview	13-65
13.16.2. Configuration	13-65
13.17. Trend Display	
13.17.1. Overview	13-73
13.17.2. Configuration	13-73
13.18. History Data Display	
13.18.1. Overview	13-86
13.18.2. Configuration	13-86
13.19. Data Block Display	
13.19.1. Overview	13-92
13.19.2. Configuration	13-92
13.20. XY Plot	
13.20.1. Overview	13-100
13.20.2. Configuration	13-100
13.21. Alarm Bar and Alarm Display	
13.21.1. Overview	13-106
13.21.2. Configuration	13-106
13.22. Event Display	
13.22.1. Overview	
13.22.2. Configuration	13-110
13.23. Data Transfer (Trigger-based)	
13.23.1. Overview	13-117
13.23.2. Configuration	13-117
13.24. Backup	
13.24.1. Overview	13-119
13.24.2. Configuration	13-119
13.25. Media Player	
13.25.1. Overview	13-125
13.25.2. Configuration	13-125
13.26. Data Transfer	



13.26.1. Overview	13-131
13.26.2. Configuration	13-131
13.27. PLC Control	13-134
13.27.1. Overview	13-134
13.27.2. Configuration	13-134
13.28. Scheduler	13-141
13.28.1. Overview	13-141
13.28.2. Configuration	13-141
13.29. Option List	
13.29.1. Overview	13-150
13.29.2. Configuration	13-150
13.30. Timer	
13.30.1. Overview	13-158
13.30.2. Configuration	13-158
13.31. Video In and Video In (USB Camera)	13-162
13.31.1. Overview	13-162
13.31.2. Configuration	13-162
13.32. System Message	13-166
13.32.1. Overview	13-166
13.32.2. Configuration	13-166
13.33. Recipe View	13-168
13.33.1. Overview	13-168
13.33.2. Configuration	13-168
13.34. Flow Block	13-174
13.34.1. Overview	13-174
13.34.2. Configuration	13-174
13.35. Operation Log	13-178
13.35.1. Operation Log Settings	13-178
13.35.2. Operation Log View	13-181
13.35.3. Operation Log Printing	13-183
13.36. Combo Button	
13.36.1. Overview	13-190
13.36.2. Configuration	13-190
13.37. Circular Trend Display	
13.37.1. Overview	13-193
13.37.2. Configureation	13-193
13.38. Picture View	
13.38.1. Overview	13-202



13.38.2. Configuration	13-202
13.39. File Browser	
13.39.1. Overview	
13.39.2. Configuration	
13.40. Recipe Import/Export	
13.40.1. Overview	
13.40.2. Configuration	
13.41. Pie Chart	
13.41.1. Overview	
13.41.2. Configuration	
13.42. QR Code	
13.42.1. Overview	13-213
13.42.2. Configuration	13-213
14. Shape Library and Picture Library	14-1
14.1. Overview	
14.2. Building Shape Library	
14.2.1. Shape manager	14-2
14.2.2. Steps to Build Shape Library	14-5
14.3. Building Picture Library	
14.3.1. Picture manager	14-9
14.3.2. Steps to Build Picture Library	14-11
14.3.3. Steps to Import Picture by Pasting	14-13
15. Label Tag Library and Multi-Language	15-1
15.1. Overview	15-2
15.2. Label Tag Library Manager	15-2
15.3. Steps to create Label Tag Library	15-3
15.4. Using Label Tag Library	15-4
15.5. Settings of Multi-Language	15-5
16. Address Tag Library	
16.1. Overview	
16.2. Building Address Tag Library	
16.3. Using Address Tag Library	
17. Transferring Recipe Data	17-1
17.1. Overview	17-2
17.2. Steps to Update Recipe Data with Ethernet or USB Cable	
17.3. Steps to Update Recipe Data with SD Card or USB Disk	17-3
17.4. Transferring Recipe Data	
17.5. Saving Recipe Data Automatically	



18. Macr	ro Re	eference	l8-1		
18.1.	18.1. Overview				
18.2.	Instr	ructions to use the Macro Editor	18-2		
18.3.	Conf	figuration	18-7		
18.4.	Synt	tax	18-8		
18.4	4.1.	Constants and Variables	18-8		
18.4	4.2.	Operators18	3-10		
18.5.	State	ement	8-13		
18.5	5.1.	Definition Statement18	3-13		
18.5	5.2.	Assignment Statement18	3-13		
18.5	5.3.	Logical Statements18	3-13		
18.5	5.4.	Selective Statements18	3-15		
18.5	5.5.	Iterative Statements18	3-16		
18.6.	Fund	ction Blocks1	8-18		
18.7.	Built	t-In Function Block1	8-21		
18.7	7.1.	Mathematical Functions18	3-21		
18.7	7.2.	Data Transformation18	3-27		
18.7	7.3.	Data Manipulation18	3-31		
18.7	7.4.	Bit Transformation18	3-33		
18.7	7.5.	Communication18	3-35		
18.7	7.6.	String Operation Functions18	3-49		
18.7	7.7.	Recipe Query Function18	3-76		
18.7	7.8.	Miscellaneous18	3-78		
18.8.	How	v to Create and Execute a Macro1	8-85		
18.8	8.1.	How to Create a Macro18	3-85		
18.8	8.2.	Execute a Macro18	3-89		
18.9.	User	r Defined Macro Function1	8-89		
18.9	9.1.	Import Function Library File18	3-90		
18.9	9.2.	How to Use Macro Function Library18	3-91		
18.9	9.3.	Function Library Management Interface18	3-93		
18.10.	Som	ne Notes about Using the Macro18-	-101		
18.11.	Use	the Free Protocol to Control a Device	-101		
18.12.	Com	npiler Error Message	-106		
18.13.	Sam	nple Macro Code18-	-112		
18.14.	Mac	cro TRACE Function	-117		
18.15.	Exan	mple of String Operation Functions	-121		
18.16.	Mac	cro Password Protection18-	-129		
19. Conf	igure	e HMI as a MODBUS Server	19-1		



19.1.	Overview	19-2
19.2.	Steps to Create a MODBUS Server	19-2
19.3.	Steps to Access a MODBUS Server	19-4
19.4.	Changing MODBUS Server Station Number Online	19-6
19.5.	MODBUS Address Type	19-6
20. How	v to Connect a Barcode Reader	20-1
20.1.	Overview	20-2
20.2.	Steps to Connect a Barcode Reader	20-2
21. Ethe	ernet Communication and Multi-HMI Connection	21-1
21.1.	Overview	21-2
21.2.	HMI to HMI Communication	21-2
21.3.	PC to HMI Communication	21-3
21.4.	Operating the PLC Connected with Other HMI	21-4
21.	.4.1. Settings of eMT / mTV Series	21-5
21.	.4.2. Settings of cMT-SVR Series	21-6
22. Syst	em Registers	22-1
22.1.	Overview	22-3
22.2.	The Address Ranges of Local HMI	22-4
22.	.2.1. Bits	22-4
22.	.2.2. Words	22-4
22.3.	System Registers	22-5
22.	.3.1. HMI Time	22-5
22.	.3.2. HMI Operation	22-6
22.	.3.3. Touch Position	22-7
22.	.3.4. Local HMI Network Information	22-7
22.	.3.5. Project File Information	22-9
22.	.3.6. Storage Space Management	22-10
22.	.3.7. Recipe and Extended Memory	22-10
22.	.3.8. Data Sampling	22-12
22.	.3.9. Event Log	22-12
22.	.3.10. Station Number Variables	22-13
22.	.3.11. Index Registers	22-14
22.	.3.12. MODBUS Server Communication	22-15
22.	.3.13. Communication Parameter Settings	22-16
22.	.3.14. Communication Status and Control with PLC (COM)	22-20
22.	.3.15. Communication Status and Control with PLC (Ethernet)	22-21
22.	.3.16. Communication Status and Control with PLC (USB)	22-23
22.	.3.17. Communication Status and Control with PLC (CAN Bus)	22-24



EasyBuilder Pro V5.02.01

22	3.18. Communication Status and Control with Remote HMI	22-24
22	22-28	
22	3.20. Local/Remote Operation Restrictions	22-30
22	3.21. Communication Error Codes	22-30
22	3.22. Driver ID	22-32
22	3.23. DLT645 Controller	22-32
22	3.24. [PLC No Response] Window Control	22-32
22	3.25. [Fast Selection] Window Control	22-33
22	3.26. EasyAccess	22-33
22	3.27. EasyAccess 2.0	22-33
22	3.28. Remote Print/Backup Server	22-34
22	3.29. Pass-Through Settings	22-34
22	3.30. VNC Control	22-35
22	3.31. Project Key and HMI Key	22-36
22	3.32. USB Security Key	22-36
22	3.33. User Name and Password	22-37
22	3.34. Macro	22-38
22	3.35. Input Object Function	22-39
22	3.36. Time Sync./Daylight Saving Time	22-39
22	3.37. Miscellaneous	22-41
23. HM	Supported Printers	23-1
23.1.	The Supported Printer Types	
23.2.	Steps to Add a New Printer and Start Printing	
24. Rec	pe Editor	24-1
24.1.	Overview	
24.2.	Recipe / Extended Memory Editor Setting	
24.3.	Recipe Records Setting	
25. Easy	Converter	25-1
25.1.	Overview	
25.2.	Converting Data Sampling File to Excel File	
25.3.	Converting Event Log File to Excel File	
25.4.	Converting Operation Log File to Excel File	
25.5.		
	Converting Multiple Files	
25.6.	Converting Multiple Files Scaling Function	
25.6.		
25.6.	Scaling Function	25-7 26-1
25.6. 26. Easy	Scaling Function	25-7 26-1 26-2



26.2.2.	Setup Procedure in EasyBuilder Pro	26-4
26.3. Usir	ng EasyPrinter as a Backup Sever	
26.3.1.	Setup Procedure in EasyPrinter	26-6
26.3.2.	Setup Procedure in EasyBuilder Pro	26-7
26.4. Easy	Printer Operation Guide	
26.4.1.	EasyPrinter Managing Window	26-10
26.4.2.	Operation Guide	26-11
26.5. Con	vert Batch File	
26.5.1.	The Default Value of Convert Batch File	26-15
26.5.2.	Specialized Criteria	
26.5.3.	The Format of a Convert Batch File	26-17
26.5.4.	The Order of Examining Criterion	26-17
27. EasySimu	ılator	27-1
27.1. Ove	rview	
27.2. Step	os to setup EasySimulator	
28. Multi-HN	II Communication (Master Slave Mode)	
28.1. Ove	rview	
28.2. Step	os to Create a Project of Master HMI	
28.3. Step	os to Create a Project of Slave HMI	
28.4. Step	os to Connect with MT500 Slave HMI	28-5
28.4.1.	Settings in EasyBuilder Pro	28-5
28.4.2.	Settings in EasyBuilder500	28-6
29. Pass-thro	ough	29-1
29.1. Ove	rview	
29.2. Ethe	ernet Mode	
29.2.1.	Steps to install virtual serial port driver	29-2
29.2.2.	Steps to Change the Virtual Serial Port	29-3
29.2.3.	Settings of Ethernet mode	29-3
29.3. CON	/ Port Mode	29-5
29.3.1.	Settings of COM Port Mode	29-5
29.3.2.	Using Utility Manager	29-5
29.3.3.	Using System Registers	29-7
29.4. Pass	s-through Control	29-7
29.5. SIEN	/IENS S7-200 PPI and S7-300 MPI Pass-through Settings	
29.5.1.	EasyBuilder Pro Settings	29-8
29.5.2.	S7-200 PPI Connection	29-8
29.5.3.	S7-300 MPI Connection	29-9
29.5.4.	Registers of SIEMENS Pass-through	29-11



30. Pro	ject P	rotection			
30.1.	30.1. Overview				
30.2.	EXO	B Password	30-2		
30.3.	Dec	ompilation is Prohibited	30-3		
30.4.	Disa	ble EXOB Upload Function	30-3		
30.5.	Proj	ect Key	30-3		
30.6.	EMT	P Password	30-4		
31. Me	mory	Мар	31-1		
31.1.	Ove	rview			
31.2.	PIN	Settings			
31.3.	Com	nmunication Flowchart			
31.4.	Add	ress Types			
31	.4.1.	Communication Examples	31-5		
31.5.	Sett	ings			
31	.5.1.	Steps to Add a Memory Map Device	31-7		
31	.5.2.	Object Settings	31-8		
31	.5.3.	Executing the Settings	31-9		
32. FTP	Serve	er Application	32-1		
32.1.	Ove	rview			
32.2.	Step	s to Log in FTP Server			
32.3.	Bacl	up History Data and Update Recipe Data	32-3		
33. Eas	yDiag	noser			
33.1.	Ove	rview			
33.2.	Con	figuration			
33.3.	Easy	Diagnoser Settings	33-3		
33	.3.1.	Main Menu			
33	.3.2.	Activity Area			
33	.3.3.	Polling Packages			
33	.3.4.	Devices			
33	.3.5.	Output (Macro debug)			
33.4.	Erro	r Code	33-9		
33.5.	Win	dow Adjustment			
34. Roc	kwell	EtherNet/IP Free Tag Names			
34.1.	Ove	rview			
34.2.	Step	s to Import User-Defined AB Tag CSV File to EasyBuilder Pro			
34.3.	Step	s to Add a New Data Type			
34.4.	Step	s to Paste			
34.5.	Mis	cellaneous Functions			



34.6.	Мос	dule-Defined	
35. Easy	/Wato	ch	35-1
35.1.	Ove	rview	
35.2.	Con	figuration	
35	.2.1.	Basic Functions	35-2
35	.2.2.	Quick Selection Tools	35-3
35.3.	Mor	nitor Settings	
35	.3.1.	Adding a Monitor	35-4
35	.3.2.	Monitor Settings	35-4
35	.3.3.	Steps to Add a New Monitor	35-5
35.4.	Mac	ro Settings	
35	.4.1.	Adding a Macro	35-8
35	.4.2.	Macro Settings	35-8
35	.4.3.	Steps to Add a New Macro	35-8
35.5.	HMI	Manager	
35	.5.1.	Opening HMI Setting	35-9
35	.5.2.	HMI Manager	35-10
35.6.	Obje	ect List	
35	.6.1.	Object List Columns	35-11
35	.6.2.	Editing Object List	35-12
36. Adn	ninist	rator Tools	
36.1.	Ove	rview	
36.2.	Usei	r Accounts	
36	.2.1.	User Accounts Settings	
36	.2.2.	Steps to Set User Accounts	
36	.2.3.	Steps to Import Accounts Using EasyBuilder Pro	
36.3.	USB	Security Key	
36	.3.1.	USB Security Key Settings	
36	.3.2.	Steps to Set USB Security Key	
36	.3.3.	Steps to Set USB Security Key Using EasyBuilder Pro	
36.4.	e-M	ail SMTP Server Settings	
36	.4.1.	Steps to set e-Mail SMTP Server Settings	
36.5.	e-M	ail Contacts	
36	.5.1.	e-Mail Contacts Settings	
36	.5.2.	Steps to set e-Mail Contacts	
36	.5.3.	Steps to Import e-Mail Settings and Contacts Using EasyBuilder Pro	
37. MO	DBUS	TCP/IP Gateway	
37.1.	Ove	rview	



37.2.	Steps to Create an Address Mapping Table	37-2
37.3.	Notes about Configuring Address Mapping	37-5
38. Easy	/Download	38-1
38.1.	Overview	38-2
38.2.	Configuration	38-2
39. Data	a Security	39-1
39.1.	Overview	39-2
39.2.	Configuration	39-2
39.	2.1. Word address settings	39-3
39.	2.2. Bit address settings	39-5
Appendix	A. Comparison of HMI Software Features	A-1



1. EasyBuilder Pro Installation and Startup Guide

This chapter explains how to install EasyBuilder Pro.

1.1.	Installation Requirements	1-2
1.2.	Steps to Install EasyBuilder Pro	1-3



1.1. Installation Requirements

Software:

Download EasyBuilder Pro from CD-ROM or visit Weintek Labs, Inc.'s website at http://www.weintek.com. The language versions include Simplified Chinese, Traditional Chinese, English, Italian, Korean, Spanish, Russian, French, and Turkish. The latest upgraded files can be downloaded too.

Hardware (Recommended):

CPU: INTEL Pentium II or higher Memory: 256MB or higher Hard Disk: 2.5GB or higher (Disc space available at least 500MB) CD-ROM: 4X or higher Display: 1024 x 768 resolution or greater Keyboard and Mouse Ethernet: for project downloading/uploading USB Port 2.0: for project downloading/uploading RS-232 COM: for on-line simulation Printer

Operating System:

Windows XP / SP3 Windows Vista Windows 7 (32bit / 64bit) Windows 8 (32bit / 64bit) Windows 8.1 (32bit / 64bit)

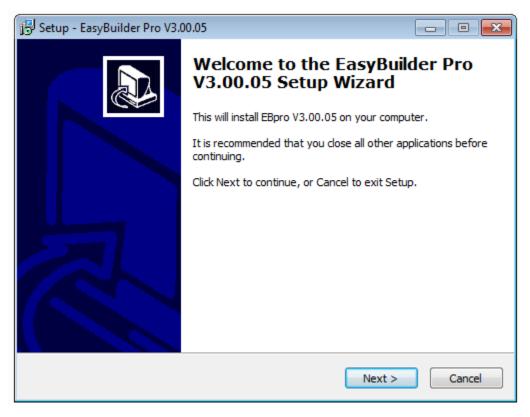


1.2. Steps to Install EasyBuilder Pro

 Insert the CD-ROM into your CD-ROM drive. The computer will automatically install EasyBuilder Pro driver. Or, you can manually execute [Autorun.exe] file under the root directory. The installation screen is shown as the following figure.



2. Click [Install] and select the language for the installation process, and then click [Next].





EasyBuilder Pro V5.02.01

- *3.* If there is a previous version of EasyBuilder Pro on the PC, please remove it before installation.
- 4. Select a folder for EasyBuilder Pro installation, or use the default folder. Click [Next].

🔂 Setup - EasyBuilder Pro V3.00.05	
Select Destination Location Where should EasyBuilder Pro V3.00.05 be installed?	
Setup will install EasyBuilder Pro V3.00.05 into the following fo	older.
To continue, click Next. If you would like to select a different folder, cli	ick Browse.
C:\EBpro	Browse
At least 448.5 MB of free disk space is required.	
< Back Next >	Cancel

5. Select a start menu folder, or use the defulat folder. Click [Next].

Setup - EasyBuilder Pro V3.00.05	
Select Start Menu Folder Where should Setup place the program's shortcuts?	
Setup will create the program's shortcuts in the following Start I	Menu folder.
To continue, click Next. If you would like to select a different folder, click	Browse.
EasyBuilder Pro	Browse
< Back Next >	Cancel



6. Select additional tasks, for example: [Create a desktop icon]. Click [Next].

🔁 Setup - EasyBuilder Pro V3.00.05	- • ×
Select Additional Tasks Which additional tasks should be performed?	
Select the additional tasks you would like Setup to perform while installing Pro V3.00.05, then dick Next.	EasyBuilder
Additional icons:	
Create a desktop icon	
< Back Next >	Cancel

7. Confirm all setting. To change the setting, click [Back]. To start installation, click [Install].

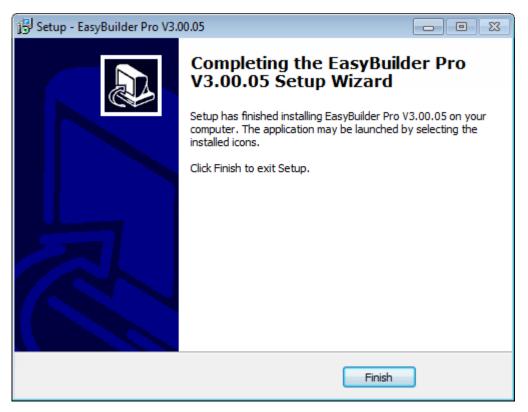
😰 Setup - EasyBuilder Pro V3.00.05	- • •
Ready to Install Setup is now ready to begin installing EasyBuilder Pro V3.00.05 on your computer.	
Click Install to continue with the installation, or click Back if you want to re change any settings.	eview or
Destination location: C:\EBpro	*
Start Menu folder: EasyBuilder Pro	
Additional tasks: Additional icons: Create a desktop icon	
<	*
< Back Install	Cancel



8. Installation progress.

Setup - EasyBuilder Pro V3.00.05	- • 💌
Installing Please wait while Setup installs EasyBuilder Pro V3.00.05 on your computer.	
Extracting files C:\EBpro\project\mTV_demo.emtp	
	Cancel

9. Click [Finish] to complete the installation.





10. The EasyBuilder shortcut can be found in [Start] » [All Programs] » [EasyBuilder Pro].

Installed file	Description
Administrator Tool	Saves the data of User Accounts, USB Security Key,
	e-Mail SMTP Server Setting, e-Mail Contacts to USB
	disk and import to HMI.
Easy Access	Supports access to any HMI connected to the
	internet. The HMI can be operated on PC.
EasyBuilder Pro	EasyBuilder Pro project editor.
EasyConverter	Conversion tool for Data Sampling and Event Log.
EasyDiagnoser	Monitoring and debugging tool operated on HMI.
EasyPrinter	Remote screen hardcopy and backup server.
EasySimulator	Executes simulation.
EasySystemSetting	Allows updating hardware system settings by using
	SD card or USB drive.
EasyWatch	On PC to monitor or set HMI and PLC address value.
Recipe Editor	Tool for setting the format of Recipe data. Users can
	open Recipe data or data in the External Memory.
Release Note	Software release notes.
Structure Editor	Supports AB TAG and improve the flexibility to read /
	write an object.
Utility Manager	EasyBuilder Pro management tool.

The description of each item in EasyBuilder Pro menu:



HMI supports downloading/uploading projects via USB cable. After installing EasyBuilder Pro, please go to [Computer Management] » [Device Manager] to check if the USB driver is installed, if not, please install it manually.



2. Utility Manager

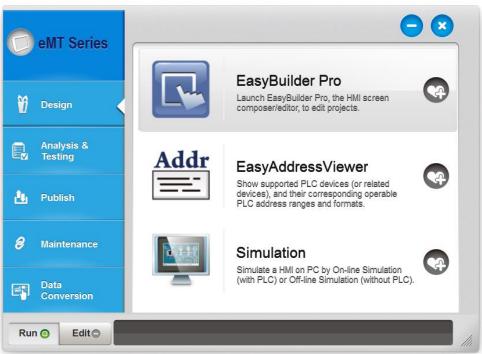
This chapter explains how to use Utility Manager.

2.1.	Overview	. 2-2
2.2.	HMI IP, Password	. 2-4
2.3.	Editing Tools	. 2-5
2.4.	Transfer	. 2-6
2.5.	Simulation	. 2-8
2.6.	Pass-Through	. 2-9



2.1. Overview

After installing EasyBuilder Pro, double click [UtilityManagerEx] shortcut on the desktop to start. Utility Manager is for launching several utilities and it is a stand-alone program.



Utilities	Description
Select Model	Select your HMI model. Please note that if the model is incorrect,
	certain features may not work correctly.
Design	EasyBuilder Pro: Launch EasyBuilder Pro to edit projects.
	EasyAddressViewer: Review the address ranges and formats of
	supported PLCs.
	Simulation: Simulate a HMI on PC by On-line Simulation (with PLC)
	or Off-line Simulation (without PLC).
Analysis & Testing	EasyDiagnoser: On-line monitoring and debugging tool. Diagnose
	the connection status between PC/HMI and PLC.
	🖙 See "33 EasyDiagnoser".
	EasyWatch: Allows users to monitor HMI or PLC address values via
	Ethernet on PC.
	🖙 See "35 EasyWatch".
	Reboot HMI: Restart a HMI to its initial condition by Ethernet or
	USB connection.
	Pass-Through: Allows PC applications to control PLC via HMI. In this
	case the HMI is an adaptor.

	See "29 Pass-through".
Publish	Download: Download project file to HMI via Ethernet.
	Upload: Upload files on HMI to PC via Ethernet.
	Build Download Data for SD/USB Disk:
	Build the data to be saved in SD card / USB drive and then insert
	the device to HMI to download the data. This feature is not
	supported by cMT Series.
Maintenance	EasyPrinter, Backup/Printer Server: A backup/printer server on PC,
	which receives backups from HMI and run a defined batch to
	convert, or HMI screenshots to print out on PC.
	Administrator Tools: Allows storing the data of [User Accounts],
	[USB Security Key], [e-Mail SMTP Server Settings], and [e-Mail
	Contacts] to USB. This feature is not supported by cMT Series.
	🖙 See "36 Administrator Tools".
	cMT Viewer: Connect to a cMT-SVR machine. PC acts as a display
	terminal for cMT-SVR.
	Data/Event log Information: Connect with HMI via USB cable or
	Ethernet to check the number of history files in HMI. This feature is
	not supported by cMT Series.
	EasyAccess 1.0: Control remote HMI instantly and conveniently no
	matter which corners in the world you are. You can find more
	information at: <u>www.ihmi.net</u>
Data Conversion	Database Editor: Used to edit recipe data.
	Click the icon to download the document about Recipe
	Database.
	Easy Converter: Reads the data sampling file (.dtl) and event log file
	(.evt) in HMI and convert the files to Excel (.xls) format.
	🖙 See "25 EasyConverter".
	Recipe Editor: Used to create, view, and edit recipe data.
	🖙 See "24 Recipe Editor".
0	Minimize the window.
×	Close the window.
	Add the frequently used utilities to the toolbar at the bottom of the
G	window.
Run	window. Run the selected utility on the toolbar.



2.2. HMI IP, Password

Settings

When operating HMI via Ethernet or USB cable, please set the password for HMI to protect against unauthorized access.

Password :	*****
	V Mask

Set the download password. To use masking password, select [Mask] check box.



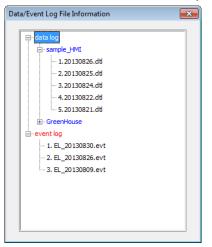
Please remember the password, otherwise, while restoring HMI default settings, the project files and data in HMI will be completely erased.

Reboot HMI

Reboot the HMI without unplugging. After reboot, the system returns to the initial state. Set the correct IP address when rebooting HMI via Ethernet.

Data/Event Log File Information

After setting, connect with HMI to check the number of history files in HMI.







2.3. Editing Tools

2.3.1. Build Download Data for SD/USB Disk

Select the folder to save download data :	
PLEASE INPUT DIRECTORY NAME !	Browse
	,
Sources ☑ Project	
PLEASE INPUT EXOB FILE NAME !	Browse
Recipe (RW)	
PLEASE INPUT RECIPE FILE NAME !	Browse
Recipe A (RW_A)	
PLEASE INPUT RECIPE_A FILE NAME !	Browse
✓ Data log	
PLEASE INPUT DATA LOG FILE NAME !	Browse
☑ User-defined startup screen	
PLEASE INPUT START-UP SCREEN FILE NAME !	Browse
Build	Exit

- 1. Insert an external device (SD card or USB drive) to PC.
- 2. Assign the directory to store data.
- 3. Select the directory of the source file.
- 4. Click [Build] to create files for downloading.

Files will then be store to the inserted device for users to download to HMI without connecting via a USB cable or Ethernet.

2.3.2. Steps to Download Project to HMI via USB Disk or SD Card

Assume we will download data in the folder named "123" (K:\123) on an USB disk.

- 1. Insert USB (in which the project is saved) to HMI.
- 2. In [Download / Upload] dialog box select [Download].
- 3. Enter Download Password.
- **4.** In [Download Settings] dialog box, select [Download project files] and [Download history files] check boxes.
- 5. Press [OK].
- 6. In [Pick a Directory] dialog box, select directory: usbdisk\disk_a_1\123.
- 7. Press [OK].

Project will then be updated.



Note

If only the history files are downloaded, it is necessary to reboot HMI to update files.

2.4. Transfer

2.4.1. Download

Download files to HMI via Ethernet or USB cable.

Firmware				
Project	PLEASE INPUT EXOB FILE NAME !			Browse
RW	PLEASE INPUT RECIPE FILE NAME	!		Browse
RW_A	PLEASE INPUT RECIPE_A FILE NAM	ME !		Browse
Recipe database	PLEASE INPUT RECIPE DATABASE	FILE NAME !		Browse
🖊 Data log	PLEASE INPUT DATA LOG FILE NA	ME !		Browse
Destination :	PLEASE INPUT DESTINATION FOL	DER NAME !		Browse
Startup screen	PLEASE INPUT LOGO FILE NAME !			Browse
 Ethernet 	O USB cable			
 Ethernet 	O USB cable			J
 Ethernet 	O USB cable			4
Ethernet IP Name	© USB cable			4
Ethernet IP Name	▼ <u>S</u> earch			4
₫ IP Name	•			4
● Ethernet <u>IP Name</u> <u>H</u> MI Name:	▼ <u>S</u> earch S <u>e</u> arch All			<u>4</u>
Ethernet IP Name	▼ <u>S</u> earch S <u>e</u> arch All		e startup screer	

Setting	Description
Firmware	Update HMI kernel programs. The firmware
	must be downloaded at the first time
	downloading data to HMI.
Project	Select an .exob project file.
RW / RW_A	Select a .rcp recipe file.
Recipe database	Select a .db file of Recipe Database.
Data log	Select the data sampling folder in HMI and then



	select a .dtl file.			
Startup screen	Download a .bmp bitmap file to HMI. After HMI			
	is rebooted, this .bmp file will be shown before			
	project starts.			
Reboot HMI after				
download	Automatically reboot after download.			
Port No. Setting	Select the port by which to download the			
	project file via Ethernet.			
Reset recipe / recipe database / event log / data log / operation log / Delete startup screen	Erase the selected files in HMI before download.			
•				

2.4.2. Upload

Upload files from HMI to PC via Ethernet or USB cable. Click [Browse] and assign the file path before uploading.

pload				
Project	PLEASE INPUT EXOB FILE NAME !			Browse
RW RW	PLEASE INPUT RECIPE FILE NAME !			Browse
RW_A	PLEASE INPUT RECIPE_A FILE NAME	E!		Browse
📝 Recipe database	PLEASE INPUT RECIPE DATABASE F	ILE NAME !		Browse
Operation log	PLEASE INPUT OPERATION LOG FIL	E NAME !		Browse
🔽 Data log	PLEASE INPUT DATA LOG FILE NAME	E!		Browse
Event log	PLEASE INPUT EVENT LOG FOLDER	NAME !		Browse
	Use CSV (Comma Seprated Value	s) format to save da	ta/event log files	
Extend Memory (EM)	PLEASE INPUT EM FOLDER NAME !			Browse
Connection	O USB cable			
	O USB cable			4
Ethernet	 ◯ USB cable ✓ <u>Search</u> <u>Search All</u> 			



Setting	Description
Event log	Upload the .evt file in HMI to PC.
Extended Memory	Upload the .emi file saved in SD card or USB disk
(EM)	to PC.

For information about [Project], [RW / RW_A], [Recipe database] or [Data log], see "2.4.1 Download" in this chapter.

Note

- The file will be uploaded to PC in .exob format. Please decompile it into editable .emtp file first and open the .emtp file in EasyBuilder Pro.
- To upload the historical files saved in the external device, please use FTP. See "32 FTP Server Application" for more information.

2.5. Simulation

2.5.1. Off-line Simulation / On-line Simulation

Off-line simulation: Simulate project operation on PC without any connection. On-line simulation: Simulate project operation on PC and PLCs are directly connected with PC.



When using [On-line simulation] on PC, if the target device is a local PLC (the PLC directly connected to PC), there is a 10 minutes simulation limit.

Before executing On-line/Off-line Simulation, please select the source .exob file.

When executing On-line/Off-line Simulation, right click to use these functions:

	Exit simulation Run EasyDiagnoser Screenshot
Setting	Description
Exit simulation	Stop simulating.
Run EasyDiagnoser	To monitor current communication status.
Screenshot	Capture and save current screen image as a
	picture file in the screenshot folder under the
	installation directory.



2.6. Pass-Through

This function allows the PC application to connect PLC via HMI. In this case, the HMI works like a converter.



Pass-through provides two modes: [Ethernet] and [COM port].

When using [Ethernet], please install the virtual serial port driver first.

For more detail, please refer to "Chapter 29 Pass-Through Function".



3. Create an EasyBuilder Pro Project

This chapter explains the basic steps to create an EasyBuilder Pro project.

3.1.	Overview	3-2
3.2.	Create a New Project File	3-2
3.3.	Save and Compile the Project File	3-4
3.4.	Run On-Line or Off-Line Simulation	3-5
3.5.	cMT Viewer	3-5
3.6.	Download the Project File to HMI	3-6



EasyBuilder Pro V5.02.01

3.1. Overview

The following is the process of creating a project.

- 1. Create a new project file.
- 2. Save and compile the project file.
- **3.** Run On-line or Off-line simulation.
- 4. Download the project file to HMI.

The following describes each process.

3.2. Create a New Project File

- **1.** Launch EasyBuilder Pro and open a new file.
- 2. Select [Model] and select [Use template] check box.

EasyBuilder Pro (Copyright c 2006 Weintek Lab., Inc.)	×
Welcome to EasyBuilder Pro. Please select your model.	
Model : eMT3105 (800 x 600) Display mode : Landscape VUse template (template_800x600.emtp)	•
OK Cancel	

3. Click [New].

Extended Me	d Memory Printer/Backup Server		Printer/Backup &			e-M	fail	Recipes
Device	Model	General		System Setting		g	Security	Font
)evice list :								
No.	Name		Loca	tion	Device	type	Interfa	ace I
Local HMI	Local I	нмі	Local		eMT31	05 (800	I	-

4. Configure parameters.



Device Properties
Name : Mitsubishi FX0S/FX0N/FX1S/FX1N/FX2
○ HMI
Location : Local
PLC type : Mitsubishi FX0S/FX0N/FX1S/FX1N/FX2
V.1.40, MITSUBISHI_FX0N.e30
PLC I/F : RS-485 4W 🔹
* Support communications between HMI and PLC in pass-through mode
\ast Set LW-9903 to 2 to enhance the speed of download/upload PLC program in pass-through mode
COM : COM1 (9600,E,7,1) Settings
PLC default station no. : 0
Default station no. use station no. variable
Use broadcast command
Interval of block pack (words) : 5 Max. read-command size (words) : 32 Max. write-command size (words) : 32
OK Cancel

5. A new device is added to the [Device List].

Extended Memory		Printer/Backup Server		e-Mail		Recipes	
Device	Model	General	System Setti	System Setting		Font	
evice list :							
evice list : No.	Name	e Lo	ocation Devic	e type	Interfac	e l	
				e type 105 (800		e l. -	

6. Create an object, take Toggle Switch 💙 as an example, and set the address.

New Toggle Switch Object	×
General Security Shape Label	
Comment :]
Read/Write use different addresses	
PLC name : Mitsubishi FX0S/FX0N/FX1S/FX1N/FX2 Setting	
Address : Y 🗸 🗸 🗸	
Invert signal	



- E casybuilder Pro: EMTP2-(10-VINDOW_010)

 Image: Second S
- 7. Place the object in the edit window. A project is now created.

3.3. Save and Compile the Project File

- 1. In EasyBuilder Pro toolbar click [File] » [Save] to save the .emtp file.
- 2. In EasyBuilder Pro toolbar click [Tools] » [Compile] to compile .emtp file as .exob file, which could be downloaded to HMI. This also checks if the project can run correctly.



- For cMT-SVR, the project file extension is .cmtp, and the compiled file extension is .cxob.
- 3. To use multiple languages, all languages must be configured in Label Tag Library first. When downloading the project to HMI, select the needed languages only. A successful compilation is shown in the following figure.

Compiling					X					
Project name :	ne : C:\Users\user\Desktop\EMTP2.emtp									
EXOB file name :	C:\Users\user\Desktop\EMTP2.exob									
EXOB password :	Setting	(used in decompiler)) 📃 Deco	mpilation is prohibited						
Select the languages used on the HMI										
Startup language after redownloading the project : Language 1										
Language 1	🔽 Language 2	🗹 Language 3	🗹 Language 4	🗹 Language 5	Language 6					
Language 7	Language 8	Language 9	Language 10	Language 11	Language 12					
Language 13	Language 14	Language 15	Language 16	Language 17	Language 18					
Language 19	Language 20	Language 21	Language 22	Language 23	Language 24					
Note : A maximum	n of 8 languages can	be selected simultane	ously.							
Double click error messages to modify the attributes of relative objects !										
Compile	Build font file	s			Close					



3.4. Run On-Line or Off-Line Simulation

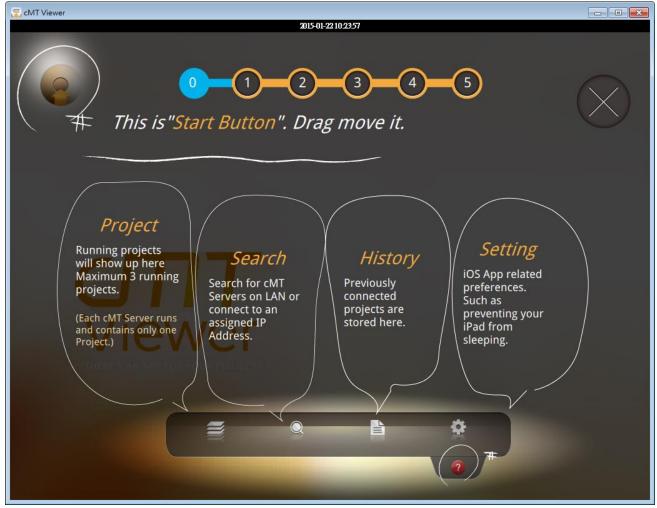
Off-line simulation: Simulate project operation on PC without connecting any device.
 On-line simulation: Simulate project operation on PC without downloading the project to HMI. The PLC is connected to PC, please set correct parameters.

Note

When using On-line Simulation on PC, if the target device is the PLC directly connected to PC, there is a 10-minute simulation limit.

3.5. cMT Viewer

This program connects to cMT Server (cMT-SVR) via network, similar to cMT Viewer App on iPad. To run this program, execute cMTViewer.exe under installation directory. Or, in EasyBuilder Pro toolbar, click [Tools] » [cMT Viewer].





Setting	Description
Start Button	Click to enter the main configuration screen, users
	can drag and move the button.
Project	Running projects will show up here. Maximum 3
	running projects. Each cMT Server runs and contains
	only one project.
Search	Search cMT Servers on LAN or connects to an
	assigned IP Address.
History	Previously connected projects are stored here.
System	Related preferences, such as using portrait mode.

3.6. Download the Project File to HMI

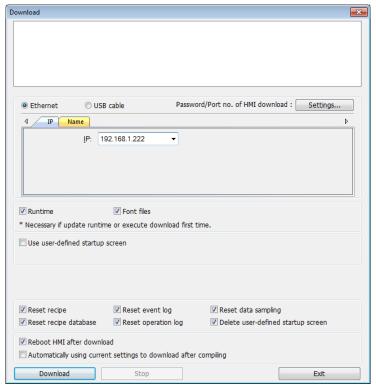
The following explains four ways to download the project file to HMI.

Note

For cMT-SVR, only the way described in 3.6.1 is available.

3.6.1. Configure in EasyBuilder Pro

- In EasyBuilder Pro toolbar, click [Tools] » [Download]. Make sure that all the settings are correct.
- 2. Select [Ethernet], set password and HMI IP.





Setting	Description
Firmware	Select the check box to update the HMI kernel programs. If this is the first time downloading file or
	EasyBuilder Pro version is updated, please download
	the firmware before downloading files to HMI.
Font files	Download the font used in the project.
Reset recipe/ event log/ data sampling/ recipe database/ operation log/ user-defined startup screen	The selected files will be erased before downloading.
Reboot HMI after download	HMI will reboot after the downloading process is done.
Automatically using current settings to download after compiling	The system will compile the project and download it to the latest target HMI. The way to enable this function is described in the following part.



For cMT-SVR, the [Firmware] and [Font files] check boxes are not available.

[Automatically using current settings to download after compiling]

If this check box is selected, EasyBuilder Pro will compile the project and download it to the latest target HMI.

- 1. In EasyBuilder Pro toolbar, click [Option] » [Preferences].
- 2. Select [Automatic save and compile when download and simulate] check box.

Preferences	×
Display : 📝 Object ID	
Display Common Window objects on Base Windows	
Using function key to make shape library object	
Automatic save and compile when download and simulate	
Generate backup of project file before save	
Automatically make used bitmaps and shapes into the project library	
OK Cancel	

- 3. In EasyBuilder Pro toolbar, click [Save] and then [Download].
- In the dialog box, select [Automatically using current settings to download after compiling] check box.



3-8

- 5. Click [Download].
- *6.* When finished, next time when [Download] is clicked, EasyBuilder Pro will automatically compile and download the project to the latest target HMI.

3.6.2. Use HMI Name

1. Go to [System settings] on HMI and then set HMI name first.

System settings 🛛 🔀
/ Network ¹ Time/Date ¹ Securit/ HMI name setting bg
HMI name test
Cancel Apply OK <

2. On PC, select the HMI name and start downloading. To use [Search], enter the HMI name first to search for the HMI. [Search all] searches for all HMIs in the same subnet network.

wnload			-
	JSB cable Pass	word/Port no. of HMI download :	Settings
4 IP Name			٥
<u>H</u> MI Name:	Default HMI <u>S</u> earch <u>S</u> earch All		
 Runtime Necessary if update rur Use user-defined start 	☑ Font files time or execute download firs up screen	st time.	
✓ Reset recipe ✓ Reset recipe database	✓ Reset event log ✓ Reset operation log	✓ Reset data sampling □ Delete user-defined start	tup screen
Reboot HMI after dow	nload		
Aucomatically using cu	rrent settings to download aft	.er compling	
Download	Stop		Exit



3.6.3. Use USB cable

Download		X
© Ethernet O USE	3 cable	Password : Settings
✓ Runtime	▼ Font files	
	ne or execute download first ti	me
Use user-defined startup		
 Reset recipe Reset recipe database 	 ✓ Reset event log ✓ Reset operation log 	✓ Reset data sampling ■ Delete user-defined startup screen
🗷 Reboot HMI after downlo		
Automatically using curre	nt settings to download after o	compiling
Download	Stop	Exit

Select USB cable to download project to HMI. The way of setting is same as "3.6.1 Configure in EasyBuilder Pro". Before downloading via USB cable, please make sure the USB driver is installed. Go to [Computer Management] » [Device Manager] to check if USB driver is installed. If it is not installed, please refer to installation steps to install manually.

3.6.4. Use USB Drive or SD Card

The following explains how to download project file by using USB drive or SD card.

1. On EasyBuilder Pro toolbar click [Tools] » [Build Download Data for SD / USB Disk].

USB Disk/CF Card/SD Card Data	×
Select the folder to save download data :	
H:\	Browse
☑ Use system setting (OS version 20131106 or later supports only) System setting file location :	
	Browse
Build System Settings	Exit



- 2. Insert an external device, such as SD card or USB disk to HMI.
- 3. Select [Download] and enter password.

Download/Upload	
Download	Download Settings
Upload	Password:
Restart project and exit	Download project files
Load System Setting	Download history files
Cancel	Clear history files
Stop after download/upload	Ok Cancel
Time remaining 4	

4. After password is confirmed, it will show the directories in the external device. (pccard: SD Card; usbdisk: USB Drive)

Pick a Directory				\mathbf{X}
Directory:	/usbdisk/disk_a_1			(*.7
e∜ pccard e∜ usbdis	k			
disk	a_1			
≜∛™t8	000ie			
1				
	OK	4	Cancel	

5. Select the directory that contains project, and then click [OK] to start downloading.

Note

- Please select the parent directory of the generated files when downloading. For the structure above, please select disk_a_1, not mt8000ie.
- You may click [System Settings] to save the hardware settings configured in EasyBuilder Pro into SD card or USB disk, and then download the settings file to HMI. See "4 Hardware Settings" for more information.



4. Hardware Settings

This chapter explains HMI settings.

4.1.	Overview	. 4-2
4.2.	I/O Ports	. 4-2
4.3.	LED Indicators	. 4-2
4.4.	System Reset	. 4-3
4.5.	System Toolbar	. 4-4
4.6.	EasySystemSetting	. 4-8



EasyBuilder Pro V5.02.01

4.1. Overview

This chapter discusses the HMI settings.

Lick the icon to download the cMT-SVR User Manual if needed.

Please confirm your internet connection before downloading the document.

4.2. I/O Ports

The I/O ports are different from one HMI type to another; please see the relevant datasheet for more information.

The I/O Ports include:

- SD card slot: Download / Upload project via SD card, including recipe transfer, event log, data log...etc and to backup or record history data.
- COM Port: Connects PLC or other peripheral devices. The type of serial port include: RS-232, RS-485 2W, RS-485 4W, and CAN Bus.
- Ethernet: Download / Upload project including recipe transfer, event log, data log...etc. Connects to Ethernet devices, such as PLC, laptop.
- USB Host: Supports USB devices, such as mouse, keyboard, USB disk, printer, or barcode device.
- USB Client: Download / Upload project including recipe transfer, event log, and data log...etc.

In addition, Weintek provides [FLZ232000 Multi-Connector Cable] and [FLZ485000 Multi-Connector Cable] to expand the COM port for easier operation.

For the first time operating HMI, please complete the following system settings. When finished, the project files designed using EasyBuilder Pro can be used on HMI.

4.3. LED Indicators

The LED indicators on the HMI indicate:

Models: MT8121XE, MT8150XE, MT8121iE, MT8150iE

LED	Description
PWR (Orange)	Indicates power status.
CPU (Green)	Blinks when read/write Flash Memory.
COM (Blue)	Indicates COM port communication status, blinks



EasyBuilder Pro V5.02.01

	during communication. When communication is	
	good, it may stay on. (Not including network	
	communication)	
er Models:		
LED	Description	
PWR (Orange)	Indicates power status.	
CPU (Green)	Indicates CPU status. If it blinks or goes out, there	
	may be a CPU error.	
COM (Blue/Red)	Indicates communication status, blinks during	
	communication. When communication is good, it	
	may stay on.	

4.4. System Reset

Each HMI is equipped with a reset button and a set of DIP switches. When using the DIP switches to change modes, the corresponding functions will be triggered. If system password is lost or forgotten, see the following steps to restore factory default.

- Flip DIP Switch 1 to ON and the rest to OFF, and then reboot HMI. HMI will switch to touch 1. screen calibration mode.
- 2. A "+" sign appears on the screen, touch the center of the sign, after all 5 signs are touched, "+" disappears and the touch screen parameters will be stored in HMI system.
- 3. After calibration, confirm to restore the default password, select [Yes].
- 4. Confirm to restore the default password again by typing [yes] and clicking [OK]. The project files and history records stored in HMI will all be removed. (The default Local Password is 111111. However, other passwords, such as Download/Upload passwords have to be reset.)

The following lists the DIP switch settings of different models. Please see the relevant installation instruction.

eMT / iE	SW1	SW2	SW3	SW4	Mode
Dip Switch	ON	OFF	OFF	OFF	Touch Screen Calibration Mode
DN 1 2 3 4	OFF	ON	OFF	OFF	Hide HMI System Setting Bar
	OFF	OFF	ON	OFF	Boot Loader Mode
	OFF	OFF	OFF	ON	Reserved
	OFF	OFF	OFF	OFF	Normal



Note

The state of DIP Switch 4 on each unit may be different. If it should be ON when out from factory, the Dip Switch 4 would be set ON and cut off. If it should be OFF, the Dip Switch 4 would be set OFF but the switch is not cut.

mTV	SW1	SW2	Mode
Dip Switch	ON	ON	Restore factory default
	ON	OFF	Hide system setting bar
	OFF	ON	Boot loader mode
	OFF	OFF	Normal mode
cMT-SVR	SW1	SW2	Mode
cMT-SVR Dip Switch	SW1 ON	SW2 ON	Mode Restore factory default
	ON	ON	Restore factory default
	ON ON	ON OFF	Restore factory default Restore Ethernet IP Settings

4.5. System Toolbar

After rebooting HMI, you can set the system with [System Toolbar] at the bottom of the screen. Normally, this bar is hidden automatically. Only by touching the arrow icon at the bottom-right corner of the screen will the System Toolbar pop up. From left to right the icons are: System Settings, System Information, Text Keyboard, and Number Keyboard.



How to hide HMI System Setting Toolbar:

- When [DIP Switch 2] is set ON, the system setting toolbar is disabled. When set OFF; the system setting toolbar is enabled. Please restart HMI to enable/disable the toolbar.
- For mTV Series, flip DIP Switch 1 to ON to hide system setting toolbar.
- System register [LB-9020] can also enable/disable system setting toolbar. When [LB-9020] is set ON, the toolbar is displayed, and set OFF to hide the toolbar.

4.5.1. System Setting

Set or modify system parameters. Confirm password for security first. The factory default password is 111111.



Hardware Settings

System settings	×				
Network\Time/Date\Security\y\me\\r setting\0					
Obtain an IP Address Automatically					
● IP address get f	from below				
	192 . 168 . 1 . 64 255 . 255 . 255 . 0				
GateWay:	192 168 1 254 8 8 8 8				
Prev Next	Cancel Apply OK (*				
System settings					
Network ⁾ Time/Dat	e\Security y ame setting hg \\				
Year: 2013 🖣 N	Mon: 8 Day: 8				
Week: 4					
Hour: 10 Min	n: 12 Sec: 28				

Cancel Apply OK <=

Network Time/Date Security y ame setting hg

Local Password

Upload Password

Download Password

Upload (History) Password

Network¹ Time/Date¹ Se History me\\r setting\0

Clear

Cancel Apply OK <

Clear Recipe Clear Operationlog Clear Eventlog Clear Data Log

Cancel Apply OK <-

Ne×t

Prev

Prev

Ne×t

Network

When downloading project file to HMI via Ethernet, set the correct IP of the target HMI. You can obtain an IP address automatically or enter the IP address manually. To use Email and EasyAccess2.0, please set correct DNS address.

Time / Date

Set HMI local time/date.

Security

Password protection, the default is 111111. Please click the buttons to set the passwords, and finish password confirmation. [Password for entering system] [Password for uploading project] [Password for downloading project] [Password for uploading history data]

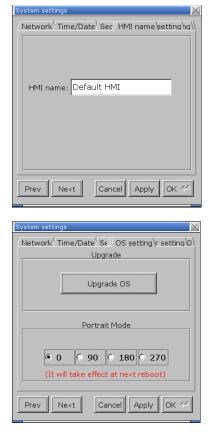
History

Clears history data in HMI.





Prev Ne×t



System settings Network¹ Time/Date¹ Sec/// VNC server setting Start VNC single-connection Start VNC multi-connection Stop VNC VNC login password Prev Next Cancel Apply OK

System settings 🛛 🗙
Network ^y Time/Date ^y Sec/// VNC/ Miscellaneous Backlight
Brightness
Download setting
Popup download window
Prev Next Cancel Apply OK <*

HMI name

Set HMI name to be used when download/upload project.

Firmware setting

Upgrade firmware and select the display mode. The display mode will take effect at next reboot.

VNC server

Remote HMI monitoring and controlling via Ethernet. [Start VNC single-connection]

Allows connection with one VNC client.

[Start VNC multi-connection]

Allows connection with multiple VNC clients.

Connecting more VNC clients may slow down the communication speed.

Please see the settings steps in the later part.

Miscellaneous

Rotary switch for adjusting LCD brightness.

[Popup download window]

If selected, after inserting USB disk or SD card to

HMI, the Upload / Download dialog box is displayed.

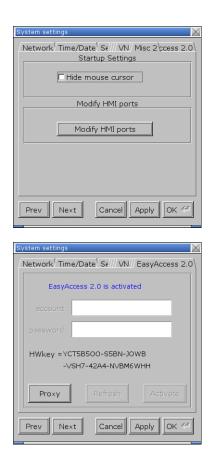
[Restart after download/upload]

If selected, restarts HMI automatically after

uploading / downloading project.



Hardware Settings



Misc 2

[Modify HMI ports] Change the port number for Upload/Download and FTP.

EasyAccess 2.0 Activate EasyAccess 2.0 service.

The following steps explain how to set VNC server.

- 1. Enable HMI VNC server, set password.
- 2. Install Java IE or VNC Viewer on PC.
- **3.** Enter remote HMI IP in Internet Browser. Or, in VNC Viewer enter remote HMI IP and password.



Note

If there is no activity of VNC client for more than one hour, HMI system will log out automatically.



4.5.2. System Information

Network: Displays network information & HMI IP.

System information (Default HMI)			
Network\Version\			
IP Address:	192.168.1.56		
Net Mask:	255.255.255.0		
Route Address:	192.168.1.254		
Mac Address:	00:0C:26:04:70:AC		
	Ok		

Version: Displays HMI firmware version and model type.

System information (Default HMI)
Network Version
MT8×××(iE) firmware build 20130719
Tupo (MT9070)E
Type :MT8070iE
Ok

4.6. EasySystemSetting

EasySystemSetting allows updating hardware system settings by using SD card or USB drive. The feature is available for HMI OS version 20131106 or later.

	🔅 [System Setting] Editor			
	Current local password : 111111			
	Password General Network Clear History VNC Server Download Back light : 15 • Time offset : 0 • hour(s) Portrait mode : 0 • degree(s) Import Export			
Setting	Description			
General	[HMI name] Enter HMI name.			
	[Back light] Adjust LCD backlight brightness.			
	[Time offset] Set the HMI RTC with offset.			
	For example, if the current RTC time is 15:00:00, and			



4-8



	the time offset is set to -3, the updated time will be
	12:00:00.
	[Protrait mode] Set the display mode.
Import	Import and edit an existing .conf file.
Export	Export the configured data to a .conf file.
Default	Restore default.

The following explains how to update HMI IP address by using SD card or USB drive.

1. On EasyBuilder Pro toolbar click [Tools] » [Build Download Data for SD / USB Disk], and then select [Use system setting] check box.

USB Disk/CF Card/SD Card Data	X
Select the folder to save download data :	
H:\	Browse
✓ Use system setting (OS version 20131106 or later supports System setting file location :	; only)
	Browse
Build System Settings	Exit

 Click [System Settings] button to open [System Setting] Editor dialog box. Specify HMI network information as shown in the following figure.

	5 5	
[System Setting] Edito	r	
Current local pas	sword : 111111	
Password General	I Enable	
Network	DHCP	
Clear History VNC Server Download	IP: 192 . 168 . 1 . 100	
	Mask: 0 . 0 . 0 . 0	
	Gateway: 0 . 0 . 0 . 0	
	DNS :	
Import	Export Default Exit	

- 3. Click [Export] to generate a "systemsetting.conf" file.
- 4. Click [Exit] to leave EasySystemSetting.
- Click [Build] button in [USB Disk/CF Card/SD Card Data] dialog box to generate the file for download by using SD card or USB disk.



EasyBuilder Pro V5.02.01

 Insert the storage device that stores the download file to HMI and the Download/Upload dialog box appears.



 Press [Load System setting] and then the [Download Config Settings] message appears. The project file will be updated after finishing system settings.



5. System Parameter Settings

This chapter introduces the system parameter settings.

Overview	
Device	
Model	
General	
System Setting	
Security	
Font	
Extended Memory	
Printer / Backup Server	
Time Sync./DST	
e-Mail	
Recipes	
	Device Model General System Setting Security Font Extended Memory Printer / Backup Server Time Sync./DST e-Mail



5.1. Overview

Launch EasyBuilder Pro, in the main menu select [Edit] » [System Parameters] to open the [System Parameter Settings] dialog box. System Parameter Settings are divided into several tabs as shown in the following figures. These tabs will be introduced respectively in this chapter.

eMT, iE, XE, mTV Series

Extended Memo	ry Printer	/Backup Server	Time Sync./DST	e-Mail	Recipes
Device	Model	General	System Setting	Security	Font
evice list :					
No.	Name	Location	Device type	Interface	I/F Protocol
Local HMI	Local HMI	Local	MT8071iE/MT8	-	-

cMT Series

	femory	Time Sync./I	ST	e-Mail	Recipes
Device	Model	General	System Setting	Security	Font Mapping
Device list :					
No.	Name	Location	Device type	Interface	I/F Protocol
Local HMI	Local HMI	Local	cMT-SVR (102	4	-

5.2. Device

Parameters in this tab determine the attributes of each device connected with HMI. The device can be a Local / Remote HMI / PLC. When creating a new project file, there is a default device "Local HMI" which indicates the HMI that will be updated and programmed. To change the relevant device settings, click [System Parameter Settings] » [Settings] to open [Device Properties] dialog box.



5.2.1. How to Control a Local PLC



"Local PLC" means the PLC is connected to the local HMI. To control/connect a Local PLC, add this type of device first. Click [System Parameter Settings] » [New] to open [Device Properties] dialog box. For example, when connecting MODBUS RTU as a Local PLC:

Name :	MODBUS RTU				
	© HMI	PLC			
Location :	Local	• Settings			
PLC type :		MODBUS R1	ហ	•]
	V.2.50, MODBU	JS_RTU.e30			
PLC I/F :	RS-232		•		
' Support communi	cations between	HMI and PLC in pas	s-through n	node	
* Set LW-9903 to 2	2 to enhance the	e speed of download	/upload PL	C program in pass-	through mode
COM :	COM1 (9600,E,	8,1)		(Settings
сом :	COM1 (9600,E, PLC default st			(Settings
сом :	PLC default st		o. variable		Settings
сом :	PLC default st	tation no. : 1	o. variable		Settings
COM :	PLC default st Default static Use broadca	tation no. : 1			Settings
	PLC default st Default static Use broadca	tation no. : 1 on no. use station no st command the station no. in object		Address Ran	
Inte	PLC default st Default static Use broadca: <u>How to designate</u> rval of block pack	tation no. : 1 on no. use station no st command the station no. in object	s address ?	Address Ran Convers	ge Limit
Inte Max. re	PLC default st Default static Use broadca: <u>How to designate</u> rval of block pack rval of block pack	tation no. : 1 on no. use station no st command the station no. in object k (words) : 5	<u>s address ?</u>		ge Limit

Setting	Description
Name	The name of the device.
HMI / PLC	In this example the device used is a PLC, so select [PLC].
Location	Select [Local] or [Remote]. In this example the PLC is
	connected to the Local HMI, so select [Local].
PLC type	Select the type of the PLC.
PLC I/F	The available PLC interface: [RS-232], [RS-485 2W], [RS-485
	4W], [Ethernet], [USB], and [CAN Bus].
	 If the interface used is [RS-232], [RS-485 2W], or
	[RS-485 4W], configure communication parameters
	by clicking [Device Properties] » [Settings] and then
	[Com Port Settings] dialog box opens.



COM Port Settings	
COM : COM 1 Baud rate : 9600 Data bits : 7 Bits Party : Even Stop bits : 1 Bit	Timeout (sec) : 1.0 Turn around delay (ms) : 0 Send ACK delay (ms) : 0 Send RESET when power on : Disable
* OS version 20120920 or later support 1	The number of resending commands : 0

Timeout

If the communication has been disconnected for more than preset time limit configured in [Timeout] (in sec), Window No. 5 will pop up and show "PLC No Response" message.

Turn around delay

While sending the next command to PLC, HMI will delay the sending according to the time interval set in [Turn around delay]. This may influence the efficiency of the communication between HMI and PLC. Default value is "0".

Note: If the PLC used is SIEMENS S7-200 Series, it is recommended to assign "5" to [Turn around delay] and "30" to [Send ACK delay].

 If the interface used is [Ethernet], click [Device Properties] » [Settings] and the [IP Address Settings] dialog box opens. Please set correct PLC IP address and port number.

IP Address Settings	
IP address : 192	2 . 168 . 1 . 100
Port no. : 500	
Timeout (sec) : 1.0	 Turn around delay (ms): 0
Send ACK delay (ms): 0	Parameter 1 : 0
Parameter 2 : 0	Parameter 3 : 0
	The number of resending commands : $\begin{bmatrix} 0 & \bullet \end{bmatrix}$
	OK Cancel

- If the interface is [USB], no further setting is required. Please check the settings in [Device Properties].
- If the interface is [CAN (Controller Area Network)



	Bus], please see "PLC Connect Guide" for "CANopen" and import the .eds device file.
PLC default	The default station number for PLC address if the PLC
station no.	station number is not included in the address. PLC station
station no.	no. can be set in PLC address. The address format:
	ABC#DEFGH
	ABC stands for PLC station number and ranges from 0 to
	255. DEFGH stands for PLC address. And the "#" sign
	separates the station number and the address. As shown
	in the following figure, the data is read from PLC station
	number 1, and address 0x-20.
	Toggle Switch/Bit Lamp Object's Properties General Security Shape Label Profile
	General Security Shape Label Profile Comment :
	Bit Lamp Toggle Switch
	Read address PLC name : MODBUS RTU
	Address : 0x 🔽 1#20
	Invert signal
	Address
	PLC name : MODBUS RTU
	Address : 1#20
	Address format : DDDDD [range : 1 ~ 65535]
Default	Use the station number variables as the default PLC
station no.	station number. Select one from LW-10000 to LW-10015
use station	(var0 to var15) as the station number variables. If the
no. variable	station no. is not specified in PLC address, the station
	number will be determined by the station no. variable.
	For example, if var3 is set for default station no:
	PLC default station no. : LW-10003 (16bit) : var3 🔹
	Default station no. use station no. variable
	Use broadcast command
	The followings demonstrate some examples:
	The PLC station number is "5".
	Read address
	PLC name : MODBUS RTU Address : 4x 5#111
	 The PLC station number is determined by var7
	(LW-10007)



Use

-Read address			
PLC name :	MODBUS RTU		Setting
Address :	4x 🔹	var7#111	

PLC address is set to "111", since PLC station no. is not specified, and the default station no. is var3, the PLC station no. is determined by var3 (LW-10003).

Read address			
PLC name :	MODBUS RTU		 Setting
Address :	4x	• 111	

When [Use broadcast command] check box is selected, broadcast please fill in [Broadcast station no.] according to the command broadcast station number defined by PLC. When HMI sends a broadcast command to the station number set here, PLC will only receive the command and not reply to HMI.

PLC default station no. :	0
🔲 Default station no. use	station no. variable
Vse broadcast comman	Broadcast station no. : 255 -

As shown in the following figure:

Read address		
PLC name :	MODBUS RTU 👻	Setting
Address :	4x ▼ 255#200	

When HMI sends a command to address 255#200, all the PLCs will receive this command and will not reply. Only PLCs that support broadcast command can use this feature.

Interval of	If the interval between read addresses of different
block pack	commands is less than this value, the commands can be
(words)	combined to one. The combining function is disabled if
	this value is set to "0".
	For example, the interval value is set to "5", to read 1 word
	from LW-3 and 2 words from LW-6 respectively (read from
	LW-6 to LW-7), since the interval of addresses between
	LW-3 and LW-6 is less than 5, these two commands can be
	combined to one. The result is to read 5 consecutive words
	from LW-3 to LW-7.
	Note: The maximum size of command combination data
	must be less than [Max. read-command size].



Max. read - command size (words)	The maximum data size to read from the device at one time. Unit: word
Max. write - command size (words)	The maximum data size to write to the device at one time. Unit: word.

After all settings are completed, a new device named "Local PLC 1" is added to the [Device list].

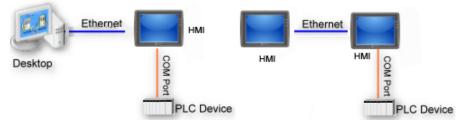
Extended Me	mory	Printer/Bac	kup Server	Time Sync./DST	e-Mail	Recipes
Device	Mod	lel	General	System Setting	Security	Font
Vevice List :						
evice list : No.	N	ame	Location	Device type	Interface	I/F Protocol
	1.5.5	ame Ical HMI	Location Local	Device type MT8071iE/MT8	1.00000000000	I/F Protocol

Note

When using cMT-SVR, select "Local HMI" in [System Parameter Settings] dialog box and then click [New PLC] to add a "Local PLC 1" under "Local HMI".

Device N	víodel	25				
	100.61	General	System	Setting	Security	Font Mapping
No.	Name	Loca	tion Devi	ce type	Interface	I/F Protocol

5.2.2. How to Control a Remote PLC



"Remote PLC" is a PLC being connected to a remote HMI. To control a remote PLC, add this type of device first. Please click [System Parameter Settings] » [New] to open [Device Properties] dialog box. For example, use MODBUS RTU as the Remote PLC:



Name :	MODBUS RTU					
	O HMI 💿	PLC				
Location :	Remote	▼ Settings	IP:192.168	3.1.100 (Port	= 8000)	
PLC type :		MODBUS RTU		•]		
	V.2.50, MODBUS	_RTU.e30				
PLC I/F :	RS-485 2W		-			
COM :	СОМ1				Settings	
COM :	PLC default stat	ion no. : 1 no. use station no. v	variable		Settings	
СОМ :	PLC default stat				Settings	
	PLC default stat	no. use station no. v			Settings	
Inter	PLC default stat	no. use station no. v station no. in object's ad words) : 5			Settings	

Setting Description HMI or PLC In this example, the device used is a PLC, so select [PI Location Select [Local] or [Remote]. In this example the PLC is connected to Remote HMI, so select [Remote]. Set the address and port number of the Remote HMI by click [Settings] next to [Remote]. Image: Provide the sector of	e IP
Location Select [Local] or [Remote]. In this example the PLC is connected to Remote HMI, so select [Remote]. Set the address and port number of the Remote HMI by click [Settings] next to [Remote]. IP Address Settings • Ethernet • COM port (use master-slave protocol) IP address : 192 . 168 . 1 . 10 • Port no. : 8000	e IP
connected to Remote HMI, so select [Remote]. Set the address and port number of the Remote HMI by click [Settings] next to [Remote].	
address and port number of the Remote HMI by click [Settings] next to [Remote].	
[Settings] next to [Remote].	ing
IP Address Settings • Ethernet • COM port (use master-slave protocol) IP address: 192 IP address: 10 Port no.: 8000	
Ethernet COM port (use master-slave protocol) IP address: 192 . 168 . 1 . 10 Port no.: 8000	
IP address : 192 . 168 . 1 . 10 Port no. : 8000	
Port no. : 8000	
OK Cancel	
PLC Type Select the type of the PLC.	
PLC I/F The interface used for Remote PLC. If the remote PLC	uses a
COM port, select [RS-232], [RS-485 2W], or [RS485 4\	V].
COM Set the correct COM port used by the Remote PLC.	
PLC default Set the station number of Remote PLC.	
station no.	

After all settings are completed, a new device named "Remote PLC 1" is added to the [Device list].



Extended Mem	ny	Frinter/Bac	kup Server	Time Sync./DST	e-Mail	Recipes
Device	Mode	1	General	System Setting	Security	Font
No.	Na	me	Location	Device type	Interface	I/F Protoco

Note

When using cMT-SVR, select the defined "Remote HMI 1" in [System Parameter Settings] dialog box and then click [New PLC] to add a "Remote PLC 1" under "Remote HMI 1".

Extended	Memory	Ti	me Sync./DSI	ſ	e-Mail	Recipes
Device	Model	Genera	al Sj	ystem Setting	Security	Font Mapping
	News			Device have	Interferen	UE Deste est
No.	Name	l	Location	Device type	Interface	I/F Protocol
No. Local HM				Device type cMT-SVR (1024		I/F Protocol
	Local HN	AI L		cMT-SVR (1024		I/F Protocol - TCP/IP

5.2.3. How to Control a Remote HMI



"Remote HMI" is the HMI other than "Local HMI", and PC is also a "Remote HMI". To control a Remote HMI, add this type of device first. Click [System Parameter Settings] » [New] to open [Device Properties] dialog box as shown in the following figure:

Name :	Dovico		
Name .	Device		
	HMI	© PLC	
Location :	Remote	✓ Settings IP : 192.168.1.10 (Port = 8000)	

Setting	Description
HMI or PLC	In this example, the device used is a HMI, so select [HMI].
Location	Select [Local] or [Remote]. In this example Remote HMI is
	used, select [Remote]. Set the IP address and port number
	of the Remote HMI by clicking [Device Properties] »
	[Settings].



IP Address Settings
Ethernet
IP address : 192 . 168 . 1 . 10 Port no. : 8000
OK Cancel

After all settings are completed, a new device named "Remote HMI 1" is added to the [Device list].

Extended Me	mory	Printe:	r/Backup Server	Time Sync./DST	e-Mail	Recipes
Device	Mo	del	General	System Setting	Security	Font
	N.		Location	Device type	Interface	I/F Protocol
No						
No. Local HM	125	ame ocal HMI	Local	MT8071iE/MT8		I/F FI010C0

Note

When using cMT-SVR, in [System Parameter Settings] dialog box click [New HMI] to add a "Remote HMI 1".

Extende	d Memory	Time S	ymc./DST		e-Mail	Recipes
Device	Model	General	System	Setting	Security	Font Mapping
evice list :						
No.	Name	Loca	ation Devi	ce type	Interface	I/F Protoco
				ce type SVR (1024		I/F Protoco



5.3. Model

Extended Mer	mory Pi	inter/Backup	Server	Time Sync./I	DST	e-Mail	Recipes
Device	Model	Gene	eral	System Setting	Se	curity	Font
HMI	model : MT807	0iE/MT8100i	E (800 x 4	30)			-
	tion no : 0	•					
P	ort no. : 8000		(used as M	ODBUS ærver's p	ort no.)		
	📝 Supj	ort cMT com	munication	protocol			
P	ort no. : 8010						
Timer	7,207	1010 IV					
Clock	source : Externa	al device	•				
PLC	C name : MODBL	IS RTU			▼ Set	ttings	
А	ddress : 4x		▼ 1		16-bit	: Unsigned	
Printer							
	Type : SP-M, D,	E, F		•			
(COM : COM 3	•					
Baud	l rate : 19200	•					
F	arity : None	•	Data	bits : 8 Bits	▼ St	op bits : 1 I	Bit 👻
Pixels of v	vidth : 100	pixel(s)		Sc	reen hard cop	y scale : 10	0% 🔻
	*100 p	ixels (for 161)	O type) or 2	20 pixels (for 240	7, 4004 type)		
Scroll bar							
Default	t Style						
Widt	۱						
	🖲 Small	🔘 Middle	C) Large			
Pass through ('	Virtual COM port)					
Р	ort no. : 2000		(2000~21	00)			

Configure the [HMI model], [Timer], [Printer] and [Scroll bar] settings.

Note

When using cMT-SVR, configure only [HMI model], [HMI station no.], and [Port no.].

Setting	Description
HMI model	Select the HMI model to use.
	If the HMI model is changed, the [Resize pop-up windows /
	objects] dialog box will pop up. Select required adjustment
	and click [OK]. In most cases, select all options.



	Resize pop-up windows Ørenaral windows
HMI station no.	Set the station number of current HMI. The default value is "0".
Port no.	Set the port number of current HMI. It is also used in MODBUS server. The default value is "8000".
Support cMT Viewer communication protocol	Supports cMT Viewer using this HMI as a Remote HMI. The Remote HMI port number in cMT-SVR project must be identical to the port number set here.
Timer	 Clock source Set the source device of the clock/time information. It is used by [Data Sampling], [Event Log], etc. If [HMI RTC] is selected, the time information comes from the internal clock of the HMI. If [External device] is selected, the time information comes from an external device. The address of the source device must set correctly. As shown below, the time information is from "TV" address type of the "Local PLC". The addresses of "TV" start from 0 and contain 6 consecutive words and each of them store the following information: TV 0 → Second (range: 0~59) TV 1 → Minute (range: 0~59) TV 2 → Hour (range: 0~23) TV 3 → Day (range: 1~31) TV 4 → Month (range: 1~12) TV 5 → Year (range: 1970~2037)



EasyBuilder Pro V5.02.01

Address			- X
PLC name :	Mitsubishi FX0S/FX0N/FX1S	5/FX1N/FX2	
Device type :	TV		•
Address :	0		
Address format :	DDD [range : 0 ~ 255]		
		Index register	
	16-bit Unsigned 🔹]	
Tag Library		ОК	Cancel

Printer	Туре
	A printer can be connected with the HMI. The HP PCL Series
	printer is connected through USB interface while other
	printers through a COM port.
	For more information, see "23 HMI Supported
	Printers".
	If the printer is connected through [COM], configure the
	parameters correctly. If the printer type is [SP-M, D, E, F],
	the [pixels of width] has to be set accurately, i.e. the set
	pixel(s) cannot exceed printer's default setting, or the HMI
	will fail to print data.
Scroll bar	Set the width of Scroll Bar, when the size of the object is too
	small to display the contents, a scroll bar is displayed in the
	object. This feature can be applied to objects that allow
	scrolling, such as Alarm Display, Event Display, History Data
	Display, and Option List.
Pass through	Set the port number for Pass-through communication.

5.4. General

Configure the properties related to screen display.



Extended Mer	nory	Printer	/Backup Server	Time Sync./DST	e-Mail	Recipes
Device	Mod	el	General	System Setting	Security	Font
Fast selection b	utton					
	Attrib	ute : Enat	ole 👻	Settings		
	Positi	on : Left		Hide button when	n HMI starts	
-						
Screen saver						
Ba	ick light sav	/er : Non/	• •	minute(s)		
*Notice :	mTV Serie	s HMI mu:	st use keyboard or	mouse to enable back light		
	Screen sav	/er : Non/	•	minute(s)		
		122				
Options						
	n window a	n · 10 1	WINDOW_010			•
	9.129935333.29 <u>9</u>			01:041		
Con	imon wind	ow : [Abo	ve base window	Ubject La	ayout : Nature	•
		🗸 R	W_A enabled			
Event						
Event		E 1	I∞ L W9450~9455	i as time tags of event logs		
	no. of eve		Jœ L ₩9450~9455	as time tags of event logs		
Extra.	no. of eve		J⊛ L ₩9450~9455	as time tags of event logs		
Extra. Keyboard 50. Keypa	d 11 - Intes	nts: 0	J⊛ L ₩9450~9455	as time tags of event logs External keyboard layo	nut (L W-9199)	
Extra. Keyboard 50. Keypa 51. Keypa	d <u>11 - Inte</u> s d 12 - Intes	nts : 0	Jæ L W9450-9455			FY (1)
Extra. Keyboard 50. Keypai 52. Keypai 53. Keypai	d <u>11 - Integ</u> d 12 - Integ d 3 - Intege d 4 - Intege	nts: 0 ger er	J⊛ L ₩94509455	External keyboard layo QWERTY (()) 💿 AZER 1	ry (1)
Extra. Keyboard 50. Keypai 52. Keypai 53. Keypai	d <u>11 - Inte</u> s d 12 - Intes d 3 - Intess	nts: 0 ger er	Jæ L W9450-9455	External keyboard layo QWERTY ((IY (1)
Extra. Keyboard 50. Keypa 51. Keypa 52. Keypa 53. Keypa 54. Keypa	d <u>11 - Integ</u> d 12 - Integ d 3 - Intege d 4 - Intege	nts: 0 ger er	Jæ L W9450~9455	External keyboard layo @ QWERTY ((Careto)) 💿 AZER 1	IY (1)
Extra. Keyboard 51. Keypa 53. Keypa 53. Keypa 54. Keypa	d 11 - Integ d 12 - Integ d 3 - Intege d 4 - Intege d 5 - Intege Add	nts: 0 ger er	•	External keyboard layo @ QWERTY ((Careto)) 💿 AZER I color : 🛄	IY (1)
Extra. Keyboard 50. Keypa 51. Keypa 53. Keypa 54. Keypa 54. Keypa	d 11 - Integ d 12 - Integ d 3 - Integ d 4 - Intege d 5 - Intege Add	nts: 0 ger er	•	External keyboard layo @ QWERTY ((Careto)) 💿 AZER I color : 🛄	IY (1)
Extra. Keyboard 50. Keypa 51. Keypa 53. Keypa 54. Keypa 54. Keypa	d 11 - Integ d 12 - Integ d 3 - Intege d 4 - Intege d 5 - Intege Add	nts: 0 ger er	•	External keyboard layo @ QWERTY ((Careto)) 💿 AZER I color : 🛄	IY (1)
Extra. Keyboard 50. Keypa 51. Keypa 53. Keypa 54. Keypa 54. Keypa	d 11 - Integ d 12 - Integ d 3 - Integ d 4 - Intege d 5 - Intege Add	nts: 0 ger er	•	External keyboard layo @ QWERTY ((Careto)) 💿 AZER I color : 🛄	IY (1)
Extra. Keyboard 50. Keypa 51. Keypa 53. Keypa 54. Keypa 54. Keypa	d 11 - Integ d 12 - Integ d 3 - Integ d 4 - Intege d 5 - Intege Add	nts: 0 ger er	•	External keyboard layo @ QWERTY ((Careto)) 💿 AZER I color : 🛄	IY (1)

Setting	Description
Fast	Setting the attributes for fast selection button for Window
selection	No. 3. To use the fast selection button, create Window No.
button	3 first.
	Attribute
	Enable or disable fast selection window. Select [Enable] and
	click [Settings] to set the attributes, including color and text
	of the button.
	Position
	Select the button position on the screen. If [Left] is chosen,
	the button will show up in at bottom left side of the screen;
	if [Right] is chosen, the button will show at the bottom
	right side of the screen.
Screen saver	Back light saver
	If the screen is left untouched and reaches the time limit
	set here, the back light will be turned off. The unit is
	minute. Back light will be on again once the screen is
	touched. If [none] is set, the back light will always be on.
	Screen saver



	If the screen is left untouched and reaches the time limit
	set here. The current screen will automatically switch to a
	window assigned in [Saver window no.].The setting unit is
	minute. If [none] is set, this feature is disabled.
	Saver window no.
	To assign a window for screen saver.
Option	Startup window no.
	Designate the window shown when start up HMI.
	Common window
	The objects in the common window (Window No. 4) will be
	shown in each base window. This determines that the
	objects in common window are placed above or below the
	objects in the base window.
	Object layout
	If [Control] mode is selected, when operating HMI,
	[Animation] and [Moving Shape] objects will be displayed
	above other kinds of objects neglecting the sequence that
	the objects are created. If [Nature] mode is selected, the
	display will follow the sequence that the objects are
	created, the first created will be displayed first.
	RW_A enabled
	Enable or disable recipe data RW A. Enable this, the
	objects can then control RW_A .The size of RW_A is 64K.
Event	Extra no. of events
	The default number of the events in the system is 1000. For
	additional number of events, modify this setting. The
	maximal is 10000.
Keyboard	The window number in which the keyboard is placed.
	When using Numeric Input or ASCII Input objects, the type
	of keyboards can be selected. Up to 32 keyboards can be
	added. To design a keyboard, a window should be
	designated for creating it. Press [add] after creating, and
	add the window to the list.
	🖙 See "12 Keyboard Design and Usage".
	Caret color
	Set the color of caret that appears when entering data in
	Numeric Input and Word Input objects.
Project	Projects can be restricted to be executed by a specific HMI.
Project protection	Projects can be restricted to be executed by a specific HMI.



cMT Series does not support [Fast selection button] and [Keyboard].



5.5. System Setting

Extended Me	emory P	rinter/Backup Server	Time Sync./D	ST e-Mail	Recipes
Device	Model	General	System Setting	Security	Font
	Startup lang	uage after redownloadir	ng the project : Lan	guage 1	•
Execut	e init. macro whe	n power on	Macro : [[ID:	:000] macro_0	•
uto logout					
		▼ minute(s) perate the HMI for longe		ne, the system will	
	vstem setting bar 9062 to open har	₩ Hide Ware setting dialog.	mouse cursor		
Sound contro					
🔘 With e	ach touch on a bu	utton, a sound is emitted	ł.		
With each of the second sec	ach operation fro	m a button, a sound is e	emitted.		
🔲 Disable	buzzer (or use L	B-9019 to disable/enable	e buzzer)		
Prohibi	t remote HMI con	necting to this machine			
Disable	upload function	(effective after rebootir	ng HMI)(or set LB903	3 on)	
🔲 Prohibi	t password remot	te-read operation (or se	t LB9053 on)		
Prohibi	t password remot	te-write operation (or se	t LB9054 on)		
Use a d	disconnection icor	n on relative objects whe	en PLC communicatio	n fails	
VNC server					
Passwo	ord free		Monitor mode		
V Passwo	ord from project	Password :	111111		
EasyAccess s	erver				
🔽 Login E	asyAccess 1.0 se	erver www.ihmi.ne	t		
		Location of EasyAcce	ess 2.0 server : Glob	pal	•

[System Setting] is used to configure different features of EasyBuilder Pro.

Some features are duplicated from system registers, such as, [Hide system setting bar (LB-9020)], [Hide mouse cursor (LB-9018)], [Disable buzzer (LB-9019)], [Prohibit remote HMI connecting to this machine (LB-9044)], and [Disable upload function (LB-9033)]. Users can also set these features via system tag.

To select a system tag, select [Address] » [System tag] check box when adding a new object and then select the [Device Type].

To browse all the system tags, Select [Library] » [Tag] » [System] from the main menu of EasyBuilder Pro.

Setting	Description
Startup language after redownloading the project	Set the language to use when HMI starts after the project is re-downloaded.
Execute init. MACRO when power on	Designate the macro to be executed when HMI power on.



5-16



Auto logout	If leaving HMI untouched for longer than the set time, the objects protected by security classes will not be able to operate. The user ID and password must be entered again to unlock it.			
Hide System Setting Bar	Hide the system setting bar in the bottom right corner of the HMI screen.			
Hide Mouse Cursor	Hide the mouse cursor in HMI screen.			
Sound control	 Disable Buzzer: Mute HMI. With each touch on a button, a sound is emitted: A sound is emitted when touching a button. With Each operation from a button, a sound is emitted: When the [Min. press time] is specified, there may be a time gap between touching the object and the action of the object. This setting can control the timing to emit a sound. 			
Prohibit remote HMI connecting to this machine	Prohibit the connection with a remote HMI. The remote HMI will not be able to control the local HMI.			
Disable upload function (effective after rebooting HMI) (or set LB9033 ON)	Disable HMI to upload project, after downloading, HMI must be rebooted to disable uploading project.			
Prohibit password remote-read operation (or set LB9053 ON)	Prohibit Remote HMI to read Local HMI.			
Prohibit password remote-write operation (or set LB9054 ON)	Prohibit Remote HMI to write Local HMI.			
Use a disconnection icon or relative objects when PLC communication fails	If selected, displays a disconnection icon on relevant objects when failing to communicate with PLC. This icon will be shown in the lower right corner of the object.			
VNC Server	If [Password free] check box is selected, the client can connect with HMI via VNC without entering a password.			



	If [Monitor mode] check box is selected, the
	HMI connected via VNC can only be monitored
	but not controlled.
	If [Password from project] check box is
	selected, set the password for VNC login.
LW protection	If select [Disable LW/RW remote-write] check
RW protection	boxes and set the protect range in [LW/RW
•	range], values within the protected range
	cannot be adjusted using Remote HMI.
Easy Access server	Login EasyAccess 1.0 server:
	Through this technology, users can easily
	access to any HMI connected to the internet
	and operate them on PC just like holding touch
	screen in hand.
	Easy Access does not transmit updated graphic
	images directly but only the real time data.
	This makes transmission really quick and
	efficient. Please refer to "EasyAccess Manual"
	for more information.
	Location of EasyAccess 2.0 server:
	Current location includes Global and China.



CMT Series does not support VNC Server.

5.6. Security

Parameters in this tab configure the user passwords and security classes. There are two authentication modes: General Mode and Enhanced Security Mode.

For more information, see "10 Security".

5.6.1. General Mode

Extended Memory		Printer/Backup Server		Time Sync./DST		e-Mail	Recipe	
Device		Model	General	System Setting		Security	Font	
Password	range : O	s for each user ~ 4294967295	C 14	Chu D	0	Cl. D	Cha	
-	range : O		Class A	Class B	Class C	Class D	Clas	•
Password	range : O	~ 4294967295	Class A	Class B	Class C	Class D	Clas	• III
Password	range : 0 - Enable	~ 4294967295 Password					Clas	-
Password No.	range : 0 - Enable	~ 4294967295 Password O	V				Clas	-
Password No. 1 2	range : 0 · Enable V	~ 4294967295 Password 0 222		V V			Clas	-



Up to 12 sets of user and password are available. Password should be one non-negative integer. Once the password is entered, the objects that the user can operate are classified. There are six security classes available: A to F.

If **[None]** is selected for an object, every user can access this object.

For example, when the security class of User No. 3 is set as the preceding figure, User No. 3 could only access objects of classes A, B, C and "none".

Project password (EMTP file)	
🔽 Enable	Setting

Users can set password to protect the project (.emtp) files. The password set here must be entered when editing the project file.

Select [Enable] then click [Setting] to set the password.

Before editing a project, a popup window is shown for entering the password. Only when the password is correct can the user edit this project.

5.6.2. Enhanced Security Mode

At most 11 users can be set here. An [Administrator] user is provided in this mode. An [Administrator] has all privileges and can operate all object classes. User passwords must be alphanumeric and each user can have up to 12 classes: A to L.

Extended Me	mory	Printer	/Backup Server	Time Sync./DST	e-Mail	Re	cipe	
Device	Mo	del	General	System Setting	Security F		Font	
elect operabl		r each user		Enhanced security model	le			
	1957-9917-991 1957-9917-9917-9917-9917-9917-9917-9917-	counts on H	0638	diama na ma	The second second		-	
Charles Int		ecret user	User name	Password	Class A	Class B	-	
1 2			user1 user2	1 2		V	Ε	
3			user2 user3	3		V		
4			user4	4				
5			119er5	5			Ŧ	
•	antestan este s	III				Þ		
Class	Descr	iption						
Class A		-						
Class B								
Class C							-	
Administra	ator							
Secret	user	U	ser name : admin	Pass	word : 111111			
Control ad	dress							
PLC r	name : Loc	al HMI		*	Settings			
	tress : LW		_] 90	10	5-bit Unsigned			
Aut	1 C33 . [LVV		* 03	50	o bit on signou			
roject passwo	ord (EMTP	file)						
Enabl			-	Settings				
		turkan inan	t an USB key into	UM				
		I WHEN DISEN		111411				
🔽 Enabl				105 J. J				
* Simulat	ion not sup	ports this fu	nction and L W-11	165s display expiration tim	e of USB key.			
Status add	ress							
	name : Loc	al HMT		_)[Settings			
	dress : LW							
	Trece . LIM		▼ 0	16	5-bit Unsigned			



Setting	Description		
Select operable classes for each user	When selecting [Use existing user accounts on HMI] check box, the operable objects for each user will be decided by the account settings in HMI.		
Administrator	Default administrator account, cannot be deleted, has all privileges, and cannot be changed. Enhanced Security Mode can be used with Option List object. It displays the account names and privileges. If [Secret user] is checked, the account names and privileges will be hidden in Option List.		
Control address	An address for users to manage the accounts directly on HMI.		
Execute auto. Login/logout when insert an USB key into HMI	This feature allows automatic login / logout using an USB security key. The login / logout status will be written into a designated address. Insert the USB disk to HMI to log in, and remove the USB disk to log out. The result codes of login / logout: 0x00: No action, 0x01: Login succeeds, 0x04: Login fails, 0x08: Logout succeeds, 0x10: Logout fails.		



cMT Series only supports Enhanced Security Mode, but does not support [Execute auto.
 Login/Logout when insert an USB key into HMI.]

5.7. Font

5.7.1. eMT, iE, XE, mTV Series

Parameters in [Font] tab determine the non-ASCII fonts.



Extended Memo	ory Pri	nter/Backup Server	Time Sync./DST	e-Mail	Recipes
Device	Model	General	System Setting	Security	Font
onts for non-as	cii strings				
Angsana Ne	W			bbA	-
Arabic Type Arial	esetting				
Arial Black				Dele	te
Arial Bold Arial Narroy	w			Delete	All
Arial Narrov Batang	w Bold		=]
Book Antiq			-		
Book Antiq Bookman C					
Bookshelf S Calibri					
Calibri Bold					
Century Go Century Go	thic thic Bold				
Comic Sans Comic Sans	MS				
Courier Nev	N				
Courier Nev Droid Sans					
	thic Medium				
Georgia					
Georgia Bo. Gulim	ld				
Impact Latha					
Lucida Con					
Lucida Sans	Demibold Rom	an	-		
	Add All Non-as	cii Fonts		Line spacing : 0	*
Support Aral	bic, Persian, Heb	rew, and Thai, alpha	bets (OS version 2013091	4 or later)	
	Fallback" font in	cludes support for Sim	plified and Traditional Ch	inese, Korean,	
Japanese.					

The non-ASCII fonts are listed here. When using non-ASCII characters or double byte characters (including Simplified or Traditional Chinese, Japanese, or Korean) which are not listed in [Fonts for non-ascii strings] table, EasyBuilder Pro will select a font from the list to substitute for it automatically.

The non-ASCII fonts in Windows can be added to the [Fonts for non-ascii strings] table. Decide the space between lines in the text in [Line spacing] field.

Select [Support Arabic, Persian, Hebrew, and Thai, ... alphabets] check box to correctly display these alphabets.

5.7.2. cMT Series

This tab lists the fonts used in WINDOWS and the corresponding fonts displayed on iPad.





Extended	Memory	Time Sync	./DST	e-Mail	Recipes	
Device	Model	General	System Setting	Security	Font Mappin;	
Support cM	T-iV5 Fonts					
Windows f	ont	▲ iOS font	cMT-i	V5 Font		
Agency FB		Helvetica Neue	Droid :	Sans	E	
Agency FB	Bold	Helvetica Neue	Droid :	Sans		
Aharoni Bo	old	Arial Hebrew	Droid 3	Sans		
Algerian		Helvetica Neue	Droid 3	Sans		
Andalus		Helvetica Neue	Droid 3	Sans		
Angsana N	ew	Helvetica Neue	Droid 3	Sans		
Angsana N	ew Bold	Helvetica Neue	Droid 3	Sans		
AngsanaUH	°C	Helvetica Neue	Droid 3	Sans		
AngsanaUH	'C Bold	Helvetica Neue	Droid 3	Sans		
Aparajita		Helvetica Neue	Droid 3	Sans		
Aparajita B	old	Helvetica Neue	Droid 3	Sans		
Arabic Typ	esetting	Helvetica Neue	Droid 3	Sans		
Arial		Arial	Droid :	Sans		
Arial Black	82	Helvetica Neue	Droid \$	Sans		
Arial Bold		Arial	Droid 3	Sans		
Arial Narro	W	Helvetica Neue	Droid 3	Sans		
Arial Narro	w Bold	Helvetica Neue	Droid \$	Sans		
Arial Roun	ded MT Bold	Helvetica Neue	Droid :	Sans		
Baskerville	Old Face	Helvetica Neue	Droid 3	Sans		
Batang		Helvetica Neue	Droid :	Sans		
Bauhaus 93	3	Helvetica Neue	Droid :	Sans		
Bell MT		Helvetica Neue	Droid 3	Sans		
Bell MT Be	0.000	Helvetica Neue	Droid 3			
Berlin Sans	Contraction of the second s	Helvetica Neue	Droid 3			
Berlin Sans		Helvetica Neue	Droid :			
Berlin Sans	FB Demi Bold	Helvetica Neue	Droid :	Sans	-	

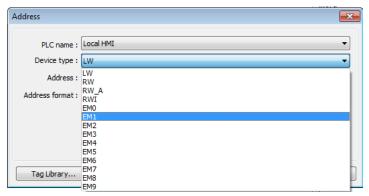
5.8. Extended Memory

Parameters in this tab determine the location of the extended memory.

Device	Model	l General	System Setting	Security	Font		
Extended Mem	iory	Printer/Backup Server	Time Sync./DST	e-Mail	Recipes		
EMO	101 07						
File name :	em0.emi			🕽 USB disk			
EM1							
File name :	em1.emi]	🕽 USB disk			
EM2							
File name :	em2.emi			🕽 USB disk			
EM3							
File name :	em3.emi			🕽 USB disk			
EM4							
File name :	em4.emi]	O USB disk			
EM5							
File name :	em5.emi		O USB disk				
EM6							
File name :	em6.emi			🔊 USB disk			
EM7							
File name :	em7.emi		O USB disk				
EM8							
File name :	em8.emi			🔊 USB disk			
EM9							
File name :	em9.emi			🕽 USB disk			



Extended Memory is numbered from EMO to EM9. It works in a way similar to other device types (i.e. LW or RW address). Users can simply select from [Device type] list while adding a new object. Size of each extended memory is up to 2G word.



Extended memories are saved as files in [SD card] or [USB disk]. [EM0] to [EM9] are saved as "em0.emi" to "em9.emi" respectively. Users can use RecipeEditor.exe to open these files and edit the data in the extended memory.

Data in extended memory will not be erased when power is cut, which means next time when start up HMI again, data in the extended memory remains the same as before power off. This is similar to recipe data (RW, RW_A). The difference is that users can specify the location to store data. (SD card, USB disk)

When the device of extended memory does not exist and to read data in it, the data content will be "0"; to write data to a device that does not exist, the "PLC no response" message will be shown in HMI.

Users can insert or remove the external device to or from HMI without cutting the HMI power to update or take data in extended memory.



5.9. Printer / Backup Server

Configure remote printer / backup server.

Device	Model	General	System Setting	Security	Font
Extended Mer	nory Pri	nter/Backup Server	Time Sync./DST	e-Mail	Recipes
* Use EasyPr Dutput settings Orient Printer		PC for printing screen ntal 🔺 💿 Ve	hardcopy and storing back ertical to printer margins mm 0 \Rightarrow mm		
		0 168 · 1 · 10			

Setting	Description
Output	Orientation
settings	Set how will words or pictures be printed out, [horizontal]
	or [vertical].
	Printer size
	Set to print out in [Original size] or to [Fit to printer
	margins].
	Margin
	Set the top, bottom, right and left margin width.
Communication	IP address
settings	Assign the IP address of the printer via network.
	[Port], [User name], [Password]
	Specify the data to log in printer.
	Port can be set from 1 to 65535.
	Maximum length of user name or password is 12
	characters.

For more information, see "26 EasyPrinter".



5.10. Time Sync./DST

Synchronize HMI time with NTP server.

	Model		General	System S	etting	Security	Font
Extended Men	nory	Printer/I	Backup Server	Time S	ync./DST	e-Mail	Recipes
🔽 Enab	le time sync}	uronization	n via NTP (Netwo	ork Time Proto	ol) server:		
🔽 Exec	ute time sync	hronizatio	on when HMI sta	rts			
Serve	er response ti	me has be	en adjusted in ac	cordance with I	DST		
	HMI	time zone	: (UTC+08:00)				•
	Server resp	onse time	: (UTC+00:00)	GMT Standard	Time		•
	Network tim	e server 1	:		(e.g. wwv.n	ist.gov or 24.56.17	78.140)
	Network tim	e server 2	:		í		
	Network tim	e server 3	:		, 1		
	Network tim) 1		
	interval (10			seconds			
opuate	11161/01 (10	~ 00400)	. 10	secontas			
			v settings on HMI		ON		
			v settings on HMI zation, the status		set to ON.		
* If failed to e	execute time :	synchroniz	-	of LB-12055 is	set to ON.		
* If failed to e	execute time :	synchroniz daylight s	zation, the status	of LB-12055 is	set to ON. nday	▼ 02:00 上午	- +
* If failed to e Auto: St	execute time : matically set	synchroniz daylight s	zation, the status aving time (DST)	of LB-12055 is		 ▼ 02:00 上4 ▼ 03:00 上4 	
* If failed to e Auto: St	execute time : matically set tart : March	synchroniz daylight s aber	aving time (DST)	of LB-12055 is	nday nday		
* If failed to e Auto Si E	matically set tart : March and : Noven	synchronis daylight s ber Daylig	aving time (DST) Second First ht bias : 01:00	of LB-12055 is	nday nday		
* If failed to e Auto: S: E * HMI enters:	matically set tart : <u>Manch</u> ind : <u>Noven</u>	synchroniz daylight s iber Daylig riod when	aving time (DST) Second First	of LB-12055 is) 	nday nday		
* If failed to e Auto: S: E * HMI enters:	matically set taut : <u>Manch</u> ind : <u>Noven</u> into DST per 260 ~ 11272	synchroniz daylight s nber Daylig iod when to modify	aving time (DST) Second First th bias : 01:00 (LB-12355 (read r settings on HMI	of LB-12055 is) 	nday nday		
* If failed to e Auto Si E * HMI enters * Use L W-112	matically set taut : <u>Manch</u> ind : <u>Noven</u> into DST per 260 ~ 11272	synchroniz daylight s nber Daylig iod when to modify	aving time (DST) Second First th bias : 01:00 (LB-12355 (read r settings on HMI	of LB-12055 is) 	nday nday		
* If failed to e Auto Si E * HMI enters * Use L W-112	matically set taut : <u>Manch</u> ind : <u>Noven</u> into DST per 260 ~ 11272	synchroniz daylight s nber Daylig iod when to modify	aving time (DST) Second First th bias : 01:00 (LB-12355 (read r settings on HMI	of LB-12055 is) 	nday nday		

Setting	Description				
Enable time	Execute time synchronization when HMI starts				
synchronization	Automatically synchronizes HMI time with the				
via NTP (Network	designated NTP server when HMI starts.				
Time Protocol)	HMI time zone				
server	Select HMI time zone.				
	Server response time				
	Select NTP server time zone.				
	Network time server				
	Provides four fields to fill in Network Time Servers for				
	user's device. If the synchronization with server 1 fails,				
	the system will try to synchronize with server 2, and so				
	on. If HMI time cannot synchronize with any of the				
	Network Time Servers, the system register LB-12055 will				
	change to ON status.				

	Update interval
	The frequency of synchronization, the range is from 10 to
	86400 seconds.
Automatically set	Start / End
daylight saving	Set the start/end of Daylight Saving Time.
time (DST)	The option [Last] may refer to the 5 th or 6 th week
	depending on the selected month.
	Daylight bias
	Set the time zone's offset during DST.

Note

- When DST ends, and the time falls back, the Trend Display object in Real-time mode will stop refreshing for the overlapping hour(s) when it just going back to standard time. However, historical data will not be affected.
- When DST ends, adjusting HMI time (manually or by network) back to the DST period will not be effective to the system. The system register LB-12355 stays OFF.
- When DST ends, and the time falls back, the system will not start DST for the overlapping hour(s). The system register LB-12355 stays OFF.
- Before DST starts, adjusting HMI time (manually or by network) into the DST period will start DST. The system register LB-12355 will be set ON. The HMI time will be the specified time without adding the offset.
- During DST, adjusting HMI time (manually or by network) to a time that is not within DST period will end DST. The system Register LB-12355 will be set OFF. The HMI time will be the specified time without subtracting the offset.
- This feature **does not yet support the DST in Southern Hemisphere**.

For more information on system registers relevant to DST, see "22 System Registers".

5.11. e-Mail

Select [Enable e-Mail function] check box to configure the parameters.

If [Use existing contact settings on HMI] check box is selected, the system will use the contact settings in HMI.



	ter Settings	- 1 - 2 - 1		1 2 3		×
Device Extended Me	emory Model	General Printer/Backup Server	System Setting Time Sync./DS	T Securi T e-M		Font Recipes
SMTP settings	-Mail function	Use ez	risting contact settings	on HMI		
		www.gmail.com		Port no	. : 25	
	User name :	eMT3000@weintek.com				
	Password :					
Confi	rm Password :	•••••		(17) A 3		
		Log on using Secure Parallel Log on using Secure Parallel Use the following type				
		TLS 👻				
Sender inform	nation					
	Name :	🔽 Use HMI name				
		Local HMI				
	Mail address :	eMT3000@weintek.com				
* LB-12053 o Open contact		ON when failed to send an	e-Mail			
		estation				
Setting		cription				
MTP Settings		ver: Set SMTF				
	Por	t: Set commu	nication po	ort.		
	Use	e r name: Set e	e-mail addro	ess.		
	Pas	sword: Set e-	mail passw	ord.		
	Cor	ifirm Passwoi	'd: Confirm	e-mail p	basswo	ord.
	Log	on using Sec	ure Passwo	ord Auth	enticat	tion (SPA):
	Dec	ide whether s	SPA is need	ed wher	n log in	e-mail.
		the followin				
	Use	the following	g type of e	ncrypteo	a conne	ection:
	Dec	ide whether	the encry	/pted_co		ection: ion (TLS, SSL)
ondor	Dec nee	ide whether ded when sei	the encry	/pted_co		
ender	Dec nee Nar	ide whether ded when sei ne:	the encry	/pted co il.	onnect	ion (TLS, SSL)
	Dec nee Nar Spe	ide whether ded when sei ne: cify the sende	the encry	/pted co il.	onnect	ion (TLS, SSL)
ender nformation	Dec nee Nar Spe Mai	ide whether ded when sei ne:	the encry nding e-ma er name or	/pted co il.	onnect	ion (TLS, SSL)

Click [Open Contact settings] to open the following dialog to edit contacts:





Contacts				×
Contacts list		Group information	No. of groups : 1	×
Contact Name Mail Address ▶ eMT3000 eMT3000@weintek.com		Contact Name eMT3000	Mail Address eMT3000@weintek.com	
	>>			
Add Remove		Current gro	oup : Group A ent :	•
		Import	Export OK	Cancel

Setting	Description				
Contact list	Add or remove contacts from the list.				
Group	Group up contacts.				
Information	No. of groups:				
	Set no. of contact groups, according to the number, the				
	groups are named from A to P and up to 16 groups can be				
	set.				
	Current group:				
	Displays the group that includes the contacts above.				
	Comment:				
	Enter a description for the current group.				

5.12. Recipes

Configure the recipe list for [Recipe Database].



Device	Mo	del	General	Sys	tem Setting	Security	_	Font
Extended Me	mory	Printer/I	Backup Server	T	ime Sync./DST	e-Mail		Recipes
ecipes List :								
Recipes 🚺		Item name	Data type	Size	Display width	Decimal Pt.	Align	iment
 1. test		NewItem	16-bit BCD	1	5	0	Left	
		NewItem1	16-bit BCD	- 1	5	0	Left	-
		New	Settings		elete			

Setting	Description
Recipes List	Add or delete a recipe.
New	Add a new item.
Settings	All the item information can be user-defined and modified, see the following description.
Delete	Delete the selected items.

Click [Settings]:

Recipes Item Information	
Name :	NewItem1
Display Type :	16-bit BCD 👻
Item size (WORD) :	1 ~
Display width (Chars) :	5 🔹
Right of decimal Pt. :	0 🗸
Alignment :	Align left 🔹
	OK Cancel

otion ecipe item name.
·
item data type.
the size of the item.
the number of characters of the item to be ed.
the decimal place when displaying data.
the alignment when displaying data. [Align

For more information, see "24 Recipe Editor".



6. Window Operations

This chapter describes different types of windows and how to create, set and delete a window.

6.1.	Overview	6-2
6.2.	Window Types	6-2
6.3.	Create, Set, and Delete a Window	6-4



6.1. Overview

A window is a basic element in a project. With a window, all kinds of information like objects, pictures, and texts can be displayed on HMI screen. In total, 1997 windows numbered from 3 ~ 1999 in EasyBuilder Pro can be built and edited.

6.2. Window Types

There are 4 types of windows, each with different functions and usages:

- Base Window
- Fast Selection Window
- Common Window
- System Message Window

6.2.1. Base Window

The most frequently used window, except for main screen, it can also be:

- A background of other windows.
- A keyboard window.
- A pop-up window of Function Key object.
- A pop-up window of Direct Window and Indirect Window objects.
- A screen saver.

Note

Base Window should be in same size as the HMI screen. Therefore, the resolution of the base window should be set to the resolution of HMI.

6.2.2. Fast Selection Window

Window no. 3 is the default Fast Selection Window. This window can co-exist with base window. Generally, it is used to place the frequently-used buttons on the lower-left side or the lower-right side on the screen. Please create window no. 3 first, and set the relevant properties in [System Parameter Settings] » [General] tab. Apart from showing or hiding fast selection window with the button on the screen, there are system registers to do so: [LB-9013] Fast Selection window control [hide(ON)/show(OFF)] [LB-9014] Fast Selection button control [hide(ON)/show(OFF)]

[LB-9015] Fast Selection window/button control [hide(ON)/show(OFF)]

Note

CMT-SVR Series does not support Fast Selection Window.



6.2.3. Common Window

Window no. 4 is the default Common Window. Objects in this window will be displayed in other base windows, not including pop-up windows. Therefore, the common objects in different windows are often placed in common window.

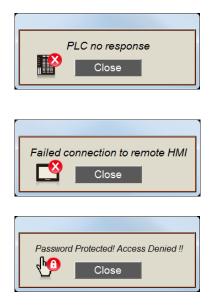
When operating HMI, select [Function Key] » [Change common window] to change the source of common window.

In menu [Option] » [Preferences] select whether or not to [Display Common Window objects on Base Windows] when editing a project. This can avoid overlapping objects in base window with objects in common window.

Preferences	×
Display Object ID	Display Object Address
Font size : 12 -	Text color :
Background	Background color :
	The number of displayed addresses : 3
☑ Display Common Window obje	cts on Base Windows
Display Underlay Window object	cts on Base Windows
Using function key to make sh	ape library object
Automatically save and compile	e the project when download and simulate
Generate backup of project fil	e before save
Automatically make used picture	res and shapes into the project library
Save AutoRecover information	
Every 3	minutes
	OK Cancel

6.2.4. System Message Window

Windows No. 5, 6, 7, 8 are the default System Message Windows:



Window No. 5: PLC Response

When the communication between PLC and HMI is disconnected, this message window will pop up automatically right on the base window currently opened. This window can be disabled by system registers. **Window No. 6: HMI Connection** When failing to connect HMI with a remote HMI, this

message window will pop up automatically.

Window No. 7: Password Restriction

When attempting to control an object without authorization, this window may pop up as a warning depending on the settings of the object.





Window No.8: Storage Space Insufficient

When HMI flash memory, USB disk or SD card run out of storage space, this message window will pop up automatically. (When the memory space is under 4 MB)

The following system registers can be used to check the free memory space in HMI, USB disk, or SD card:

[LW-9072] HMI current free space (K bytes)

[LW-9074] SD current free space (K bytes)

[LW-9076] USB disk current free space (K bytes)

To check if there is sufficient storage in the devices, the following system registers can be used.

These addresses will set ON when the space is under 4 MB.

[LB-9035] HMI free space insufficiency alarm (when ON)

[LB-9036] SD card free space insufficiency alarm (when ON)

[LB-9037] USB disk free space insufficiency alarm (when ON)

For more information, see "22 System Registers".

The text shown in windows no. $5 \sim 8$ can be edited for easier reference.

Note

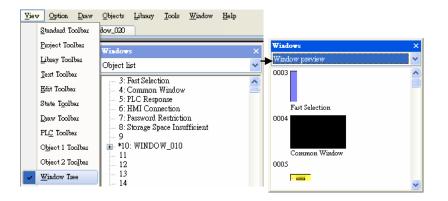
- A screen can display up to 16 pop-up windows simultaneous including System Message Window, Direct Window and Indirect Window.
- The system does not allow opening the same window with two Direct (or Indirect) Windows in one base window.
- Windows no. 3 to 9 are used by the system only, and windows no. 10 to 1999 can be edited based on actual usage.
- For cMT-SVR Series, only 1 pop-up window can be displayed simultaneously.

6.3. Create, Set, and Delete a Window

Check the existing windows in [View] » [Window Tree].

[Object list] displays window numbers and window names. Opened windows are marked with (*) sign. Press the (+) sign to see the object ID, address and description in this window. [Window preview] displays the thumbnails of windows.





6.3.1. Creating and Setting a Window

In window tree right click on a window number then select [New].

Name	
	WINDOW_010
Window no. :	10
Size	
Width :	800 Height : 480
Frame	
Width :	0 ▼ Color :
Background	
Color :	•
	Filled
Underlay window	
Bottom :	11. Window_011
Middle :	None
Top :	None
Popup window	
Start position	
x: 0	Y: 0 V Monopoly V Title bar
Macro	
Open :	[ID:000] macro_0
Cyde :	Disable 💌
Close :	Disable 🗸

Setting	Description
Name	The name appears on the title bar and also in window tree.
Window no.	Can be 3 to 1999.
Size	Set the window size in accordance with the HMI resolution.
Underlay	Underlay Window can be seen as an extra Common
window	Window. When designing the project, some commonly
	used objects are used in some windows but not all. These
	objects can be placed in underlay window.



	Each base window can set three underlay windows as background, from [Bottom] to [Top]. The objects in underlay windows are displayed in this order in base window.
Pop-up window	Base window can also be used as a pop-up window. Use [X] and [Y] to set the coordinates indicating where in the screen will this base window pop up. The origin of the coordinates is the upper-left corner of the window.
Monopoly	If the option is selected, when the base window pops up, no operations of other pop-up windows and background windows are allowed until the monopoly window is closed. If a base window is used as a keyboard window, "Monopoly" is automatically enabled.
Title bar	If the option is selected, a title bar appears on a system message window. This feature is only available for system message windows no. 5 to no. 8.
Macro	Select the macro to be executed when the window opens or closes, or the macro to be executed periodically (each 500 ms). Please build the macro in advance.



- The objects in underlay window cannot be edited from the base window that displays them. To edit those objects, please open the underlay window where they are located.
- When the window number of the underlay window used by the base window is identical to the pop-up window, the pop-up window is disabled.
- When base window and pop-up window use the same underlay window, the objects in the underlay window cannot be displayed in pop-up window.

Or, from the main menu click [Window] » [Open Window] and then click [New] and select the type of the window and click [OK].





No.	Window name	Size	
3	Fast Selection	100,450	New
4	Common Window	800,480	
5	PLC Response	320,100	Settings
6	HMI Connection	320,100	
7	Password Restriction	320,100	Delete
8 *10	Storage Space Insufficient	320,100	Delece
50	WINDOW_010 Keypad 11 - Integer	800,480 275,280	Open
50	Keypad 11 - Integer	275,280	Open
52	Keypad 3 - Integer	200,170	
53	Keypad 4 - Integer	304,213	
54	Keypad 5 - Integer	160,230	
55	Keypad 6 - Integer	241,331	
56	Keypad 7 - HEX	306,223	
57	Keypad 8 - Floating	198,220	
58	Keypad 9 - Numeric	248,248	
60	ASCII Middle	576,240	
61	ASCII Small	480,200	
62	ASCII Upper M	576,240	
63	ASCII Lower M	576,240	
64	ASCII Upper S ASCII Lower S	480,200	
65	ASCII LOWER S	480,200	

Ways to call up [Window Settings] dialog:

- Right click on the window number in the window tree and select [Settings].
- In [Window] » [Open Window] select the window then click [Settings].
- In the window, right click when no object is selected, and select [Attribute].

6.3.2. Open, Close, and Delete a Window

The ways to open an existing window:

- Double click on the window number in the window tree.
- In the window tree, select the window, right click, and then select [Open].
- In [Window] » [Open Window] select the window then click [Open].

The ways to close or delete an existing window:

- In the window tree, select the window; right click, then select [Close] or [Delete].
- In [Window] » [Open Window] select the window then click [Delete].
- To delete a window, please close it first.



7. Event Log

This chapter explains how to set and use Event Log.

7.1.	Overview	7-2
7.2.	Event Log Management	7-2
7.3.	Creating a New Event Log	7-6



7.1. Overview

The following are the basic steps to use Event Log:

- Define event content and trigger condition. 1.
- Trigger event according to the condition. 2.
- Save the event log to the specified device. 3.
- View the process of event by using the relevant objects. 4.

This chapter will explain how to set and use Event Log.

7.2. Event Log Management

Firstly, define the event content then use Alarm Bar 🗮 Alarm Display 🗳, Event Display 📓 objects to view the process of the whole event from triggering \rightarrow waiting to be processed \rightarrow return to normal.

7.2.1. eMT, iE, XE, mTV Series

C	Category :	All [2]		•					2
No.	Category	Text	Mode	Condition	Read address	Notification address	Buzzer	e-Mail	Save to histo
1	0	Event 0		ON	Local HMI : LB-0	Disable	Disable	Disable	Enable
2	0	Event 1	BIT	ON	Local HMI : LB-1	Disable	Disable	Disable	Disable
	Enable b tory files	ack light	when	alarm occur	III S				4
His		HMI men	nory	alarm occur		Save to U	SB disk		•
His	tory files V Save to	HMI men	nory	alarm occur		🗌 Save to U	SB disk)
His	tory files V Save to	HMI men	nory	·		Settings	SB disk		Import

Category

Classifies events by dividing them into $0 \sim 255$ categories.

Select one category to add or view event log. In the bracket

"[]", it shows the number of events are in this category.



History	To specify a location to save event log files. However, when
files	executing On-line or Off-line Simulation on PC, the files will
	be saved in the HMI_memory / SD_card / USB folder under
	the installation directory.
	Preservation limit
	This setting determines the length of the data can be
	preserved. For example, the [Days of preservation] is set to
	two days; the data of yesterday and the day before yesterday
	will be kept. The history data in other days will be deleted
	automatically for saving the storage space.
Save to	When a loacation to save event log files is specified in
history	[History files] group box, each event can be respectively
	defined whether to be saved to history. The events will still be
	displayed in Event Display object Real-time mode, but not in
	History mode.
Print	In [System Parameter Settings] » [Model], select a printer and
	set the printing format.
Paste	Overwrites the selected item with the new items. A message
	window will pop up to confirm this operation.
Paste	
(Add Mode)	Appends as a new entry.



7.2.2. cMT Series

C	ategory :	All [2]		-					2
No.	Category	Text	Mode	Condition	Read address	Notification address	Buzzer	e-Mail	
1	0	Event 0	BIT	ON	Local HMI : LB-0	Disable	Disable	Disable	
2	0	Event 0	BIT	ON	Local HMI : LB-0	Disable	Disable	Disable	
	ntrol addre: Enable PLC nam Addres	e : Local H	IMI		• 0	Setting 16-bit Unsigned			
Hist	Enable PLC nam	e : Local H	łMI		• 0				
Hist	Enable PLC nam Addres	e : Local H s : LW	IMI		• 0				
Hist	 Enable PLC nam Addres cory files Enable 	e : Local H s : LW	IMI						
Hist	 Enable PLC nam Addres cory files Enable 	e : Local H s : LW SD card	Insert.	Syr		16-bit Unsigned	Export		

6	
Setting	Description
Category	Classifies events by dividing them into 0 ~ 255 categories.
	Select one category to add or view event log. In the bracket
	"[]", it shows the number of events are in this category.
Control	There are two ways to save the event log file:
address	Automatically saved by system
	If [Enable] check box under [Control address] is not selected,
	the system will automatically save the event log file to
	HMI_memory folder. The maximum is 10000 records, when
	there are more than 10000 records, the earliest 1000 records
	will be deleted.
	If [Enable] check box under [Control address] is selected,
	selecting the [Enable] check box under [History file] can save
	data to HMI_memory / SD_card / USB folder.
	Manually saved by user
	If select [Enable] check boxes under both [Control address]
	and [History files], entering a specific value in the control
	address sends the corresponding command.
	When entering 1, clears the event log on cMT-SVR.



	When entering 2, synchronizes event log to the external
	device.
	When entering 3, synchronizes event log to the external
	device and then clears the event log on cMT-SVR.
	If none of these values is entered, the system will save the
	event log file in HMI_memory.
History	If enabled, saves the synchronized data to SD card or USB
files	disk.
Paste	Overwrite the selected items with the clipboard contents.
	A message window will pop up to confirm this operation.
Paste	
(Add Mode)	Append the clipboard contents to the end of the list.

Note

To remove SD card or USB disk, event log data can be synchronized by using control address first.

7.2.3. Excel Editing

Click on the Excel icon in Event Log setting dialog box to open the Excel template for a reference of editing. This template is under the installation directory, the file name is EventLogExample.xls. This template includes the ready-made dropdown lists and validation mechanism.

	A	В	С	D	Е	F	G	Н	Ι	J	K
1	Category	Priority level	Address type	PLC name	Device type	System tag	User-defined tag	Address	Index	Data Format	Enable
2	0	Middle	Word	Local HMI	LW	False	False	100	null	32-bit Signed	True
3	1	Low	Bit	Local HMI	LB-9009	True	False	9009	idx 5	16-bit BCD	▼ lse
4										16-bit BCD 32-bit BCD	
5										16-hit IInsigned	
6										16-bit Singed 32-bit Unsigned 32-bit Signed 32-bit Float	
7										32-bit Float	

Note

- [System tag] and [User-defined tag] cannot be set to true simultaneously, otherwise, the system will view the User-defined tag to be a System tag, and [User-defined tag] to be false. If setting [Device type] to [User-defined tag], please set [System tag] to false.
- When setting [User-defined tag] to true, if the system compares the [Device type] with the user-defined tag in the system, and no suitable tag is found, the system will set the [User-defined tag] in event log to false
- [Color] format is R:G:B, each should be an integer form 0 to 255.
- Before importing Label Library / Sound Library, please make sure the library names exist in the system.



7.3. Creating a New Event Log

General Tab

Click [New] in the [Event (Alarm) Log] dialog box.

Event (Alarm) Log
General Message Occurrence
Category : 0 Priority level : Low 🗸
Address type : Bit Word Delay time for event monitoring when HMI resets : 1 second(s)
Read address
PLC name : Local HMI
Address : LW
Notification Image: Constraint of the second seco
Follow
PLC name : Local HMI
Address : LB 🔹 0
Condition
Trigger if value is : <
Dynamic condition value
Read/Condition use different addresses
Condition value address
PLC name : Local HMI
Address : LW 👻 1
OK Cancel Help

Setting	Description
Category	Select event category, the range is from 0 to 255.
Priority level	When the number of events equals to the max number of the system (default 1000), the lower priority events will be deleted and new events will be added in.
Delay time for event monitoring when HMI resets	This feature is used to set the delay time of Event Log after HMI reboot, in order to avoid false alarm that occurs upon HMI reboot due to uninitialized values. This feature is often used with [Dynamic condition value]. The delay time only occurs once upon HMI reboot.
Read address	The system reads data from this address to check if the event matches the trigger condition.
Notification	When enabled, the system will set the specified address ON



	or OFF when the event is triggered.
	Follow
	The notification address will reset to the original state once
	the alarm condition returns to normal.
Condition	When [Bit] is selected, Event Log will detect the state of a Bit address.
	When [Word] is selected, Event Log will detect the value of a
	Word address to check if it is greater than, less than, or
	equals to a specified value. See Example 1 and Example 2.
	Dynamic condition value
	Allows online change of the comparison value for trigger
	condition when the condition is a Word address type. If
	[Read/Condition use different addresses] is not selected, the
	source of condition value will be the next consecutive address
	from [Read address].
	Read/Condition use different addresses
	Allows selecting the Word address type to be the source of
	condition value.

Example 1

Condition	
Trigger if value is : == 🔻	30
In tolerance : 1	Out tolerance : 2

The setting above indicates:

When [Read address] value is greater than or equals to 29 (= 30 - 1)

Or less than or equals to 31 (= 30 + 1), the event will be triggered. The trigger condition:

```
29 \leq [\text{Read address}] \text{ value } \leq 31
```

After the event is triggered, when [Read address] value is greater than 32 (= 30 + 2) or less than 28 (= 30 - 2) the system will return to normal condition:

[Read address] value < 28 or [Read address] value > 32

Example 2

Condition	
Trigger if value is : <> 🔻	30
In tolerance : 1	Out tolerance : 2

The setting above indicates:

When [Read address] value is less than 29 (= 30 - 1)

or greater than 31 (= 30 + 1), the event will be triggered. The trigger condition:

[Read address] value < 29 or [Read address] value > 31

After the event is triggered, when [Read address] value is greater than or equals to 28 (= 30 - 2)



EasyBuilder Pro V5.02.01

or less than or equals to 32 (= 30 + 2) the system will return to normal condition: $28 \le [\text{Read address}] \text{ value } \le 32$

Message Tab

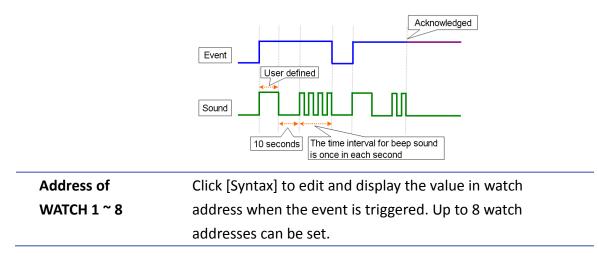
eneral Message e	e-Mail Occurrence
Text	
	Tank level low. level=%(WATCH1)d.0
	Use label library
Font :	Arial
Color :	
Acknowledge volue	for Event/Alarm Display object
Acknowledge	/alue: 11
Sound Enable	Sound Library
	Play
Addresses of WATC	
	Sans Fallback" font if the content of text is determined in run time. se HMI startup time.)

Setting	Description
Content	The text content displayed in [Alarm Bar], [Alarm Display],
	and [Event Display] objects. Use the formats in the
	following two examples of WATCH addresses to use
	register data in content. See Example 3, and Example 4.
Font / Color	The font and color can be set differently for each event.
	The setting determines the font and color shown in
	[Alarm Bar], [Alarm Display] or [Event Display] objects.
Write value for	When an event in [Event Display] or [Alarm Display] is
Event/Alarm	acknowledged, the value is written to the assigned [Write
Display object	address].
Sound	If enabled, the selected sound will be played when an
	event is triggered. Continuous beep can also be enabled,
	which only stops when the event is acknowledged or



recovered.

For continuous beep, a delay time can be set between triggering the alarm and the start of beeping.



Example 3

The data of the LW register can be used in the content displayed when an event is triggered:

Format: %#d (% -> initial sign, # -> address, d -> end sign)

When an event is triggered, if the value in LW-20 is 13:

```
Setting: "High Temperature = %20d" → Display: "High Temperature = 13"
```

Example 4

The data in the specified address when the event is triggered can be included in the content displayed. The address should be set to the [Read address] of Event Log, take MODBUS RTU 4x address as an example:

Format: **\$#d** (\$ -> initial sign, # -> address, d -> end sign)

When an event is triggered, if the value in MODBUS 4x-15 is 42:

Setting: "High Temperature = $\$15d" \rightarrow Display$: "High Temperature = 42"

e-Mail Tab

Please enable this function in [System Parameter Settings] » [e-Mail] first.



eneral Message e-M	ail Occurrence		
Enable Conditi	•	ents	
	nen event triggered nen event cleared Group .	*	
Send will Sen	ien eveni cieaten Gionh y	n	
Recipients			
	As recipients of triggered ma	ail settings	
То	Group A		
Cc]
Bcc			ĩ
Subject			
	Use event content as subject	t	
Subject :	Tank level low. level=%(WATC		
	4	*	
Message Opening :			
opening.			
	•	÷.	
	Use label library		
Ending :			
		-	
	•	Þ	
	Use label library		
	Label Library	Language 1	

Setting	Description
Recipients	Select the [To], [Cc], and [Bcc] recipients.
Subject	Enter the subject of the e-mail.
Message	Enter the [Opening] and [Ending] content of an E-mail.
Attach	If the [Contains a screenshot of window] check box is selected, the screenshot of the selected window will be attached.

Occurrence Tab

	Event (Alarm) Log
	General Message e-Mail Occurrence
	Write address
	PLC name : Local HMI Settings Address : LW 10 16-bit Unsigned
Setting	Description
Occurrence	If enabled, the number of events occurred after HMI startup will be written to the designated word address.



8. Data Sampling

This chapter explains how to set and use Data Sampling.

8.1.	Overview	. 8-2
8.2.	Data Sampling Management	. 8-2
8.3.	Creating a New Data Sampling	. 8-2
8.4.	Synchronizing cMT Viewer data and Saving to External Device	. 8-8
8.5.	Checking History Data of a Specific Date on cMT Viewer.	. 8-9



8.1. Overview

After defining how the data is sampled, by sampling time, address, or data length, the sampled data can be saved to the designated location, such as HMI memory, SD card, or USB disk. Trend Display and History Data Display objects can be used to display sampling records.

8.2. Data Sampling Management

Create a new Data Sampling object first by the following steps:

- 1. From the menu select [Objects] and click [Data Sampling].
- 2. Click [New] to finish relevant settings.

No.	Description	Read address	Sample mode	Trigger address	Clear address	Hold address	Auto. stop
1	_	Local HMI : LW-0	Periodical	Disable	Local HMI : LB0	Local HMI : LB1	
• [

8.3. Creating a New Data Sampling

The following introduces how to set a new Data Sampling. eMT, iE, XE, mTV Series

ata Sampling Object	
Comment :	PLC name : Local HMI
Sampling mode	Clear real-time data address
High priority (this may reduce refresh rate of screen components.)	Enable
Time-based Trigger-based Sampling time interval : 1 second(s)	
	Hold address
	Enable
Read address	
PLC name : Local HMI Settings	History files
Address : LW 🔻 0	Save to HMI memory
In prior to display or store the data log, you can use the conversion tag to check and modify the data log.	Save to USB disk
* When the Data record is converted by the user-defined conversion tag, the	Each file consists of all records of a day
GetCnvTagArraryIndex() function of [Read conversion] subroutine can get the relative array index.	Customized file handling
Data Record	Folder name : DataLog
Max. data records (real-time mode): 1000	File name example : 20150727.dt
Data Format Data length : 1 word(s)	✓ Preservation limit 7 day(s)
	OK Cancel



cMT Series

Comment :	
Sampling mode	Hold address
◎ Time-based	Enable Mode : OFF
PLC name : Local HMI Settings	PLC name : Local HMI
Address : LB 🔹 0	Address : LB 🔹 10
Read address	Control address
PLC name : Local HMI Settings	Enable
Address : LW 🔹 0	PLC name : Local HMI Settings
Data Record	Address : LW 🔹 11 16-bit Unsigned
Data Format Data length : 1 word(s)	History files
Data lengur, 1 word(s)	📝 Enable
	Name :
	Sync. to SD card Sync. to USB disk

Setting	Description
Sampling mode	High priority
	Data sampling processes with this feature enabled will be
	prioritized. Please note that too many priorities can slow
	down update rate of other objects.
	Time-based
	Samples data in a fixed frequency. The [Sampling time
	interval] can be set from "0.1 second(s) to 120 mins".
	Trigger-based
	Triggers data sampling by the status of a designated bit
	address.
	Mode Conditions to trigger Data Sampling:
	[OFF -> ON] Triggers sampling when the status of the
	address changes from OFF to ON.
	[ON -> OFF] Triggers sampling when the status of the
	address changes from ON to OFF.
	[OFF <-> ON] Triggers sampling when the status of the
	address changes.
	Set ON/OFF after triggered
	If selected, after triggering Data Sampling, the system wil
	set the designated bit address back to ON/OFF state.
Read address	Specify an address to be the source of Data Sampling.
Data Record	In Real-time Mode, the max. number of data records can
(Real-time)	be saved by one Data Sampling in one day is 86400. (1
	record per second for 24hours) If [sampling time interval]
	is set to "0.1 second", the max. number of data records is



Data Format

Data of different formats in consecutive registers can be sampled. As shown in the following figure: LW-0 (16-bit Unsigned), LW-1 (32-bit Float), and LW-3 (16-bit Unsigned).

ata Format		
1. "16-bit Unsigned" 2. "32-bit Float"	16-bit Unsigned 32-bit Float	New
3. "16-bit Unsigned"	16-bit Unsigned	Delete
		Settings
		Exit

Auto. stop

This function depends on the arrangement of different objects and modes.

	objects and modes.
	Gee "8.3.1 Demonstration of Auto. stop".
Clear real-time	Set when the bit address status changes from [OFF -> ON]
data address	or [ON -> OFF], clear the sampled data in Trend Display
	Real-time Mode. The number of data records returns zero
	but the data records that are already saved as history files
	will not be cleared.
Hold address	If the status of the designated address is set ON or OFF,
	sampling will be paused until the status of the designated
	address returns.
Control	Entering a value in the control address sends the
address	corresponding command.
(cMT Series)	Enter 1: Clears the sampled data in HMI.
	Enter 2: Synchronizes data to the external device.
	Enter 3: Synchronizes data to the external device and
	then clears the sampled data in HMI.
History files	eMT, iE, XE, mTV Series
	Save to HMI momery
	Saves Data Sampling to HMI only when its size reaches
	4KB. Or, use system register [LB-9034] to force storing
	data.



Save to SD card / USB disk

Saves Data Sampling to the specified external device.

Each file consists of all records of a day

The data sampling file will be saved on a daily basis into the specified folder, and the file name will be

yyyymmdd.dtl, indicating the date of the file.

Cutomized file handling

This feature can be used to customize naming and management of data sampling files (*dtl).

See "8.3.2 Customized File Handling".

Folder name

Specify Data Sampling file name which must be all in ASCII characters. The folder name will be written as: [Storage Location] \ [Folder Name] \ [File Name] Preservation limit

Determines the number of data sampling files to be

preserved.

cMT Series

History data can be saved to USB disk or SD card. When the sampled data reaches 10000 records, the sampled data is automatically saved to the selected external device, and the earliest 1000 records are deleted.

See "8.4 Synchronizing cMT Viewer data and Saving to

External Device".

Note

- A Data Sampling may include more than one type of records. Data Sampling can retrieve different types of records at the same time. For example, if define three types of data, 4 words in total, the system retrieves a 4-word data each time from the designated address to be the content in one Data Sampling.
- When using [Each file consists of all records of a day] and set [Preservation limit] to 2 files, the data of yesterday and the day before yesterday will be kept. Data that is not built in this period will be deleted to prevent the storage space from running out.
- When using [Customized file handling] and set [Preservation limit] to 2 files, not only the currently sampled file, another 2 newest files (3 files in total) will be kept. The rest of the data will be deleted to prevent the storage space from running out.
- When running simulation on PC, all data sampling will be saved to the datalog folder in C:\EBPro\ [Storage Location] \datalog. If you change the data format of data sampling, delete the previous data records in the installation directory to prevent the system from reading the old records.



8.3.1. Demonstration of Auto. stop

This feature depends on the arrangement of different objects and modes. (Set [Max. data records] to n.)

Object	[Auto. stop] not selected	[Auto. stop] selected
Trend Display- Real time	Deletes the earlier records and displays the latest number of records (n) in Trend Display. See the following figure.	Stops after reaching the specified number of data records (n).
Trend Display- Historical	Keeps on sampling data and displays all history data in Trend Display.	Irrelevant.
History Data Display	Keeps on sampling data and displays all history data in History Data Display.	Irrelevant.
Data Sampling	Keeps on sampling new data.	Irrelevant.

The figure illustrates how the data is sampled in Trend Display – Real Time mode when [Auto. stop check] box is not selected. Set the number of data records to 10, when the 11th data is generated, the earliest record is deleted and the newest record is added.

Record Number	Data	Not selecting [Auto. stop]
1	101	102
2	102	103
3	103	104
4	104	105
5	105	106
6	106	107
7	107	108
8	108	109
9	109	110
10	110	111
11	111	

8.3.2. Customized File Handling

This feature can be used to customize naming and management of data sampling files (*.dtl).



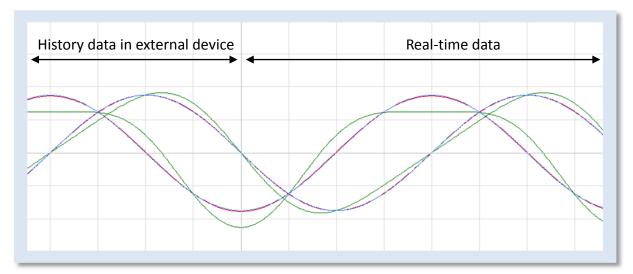
le creation	
Automatic mode (A	new file will be created when the file name is changed.)
Trigger mode () ise	the trigger method to create a new file.)
g mgger mode (ose	are algger method to create a new methy
rigger method	
,	er of data records
Max.	data records in a file : 1000
🔽 Register statı	S
Mo	de : OFF->ON V Set OFF after triggered
PLC name : Local H	MI Settings
Address : LB	▼ 0
e name	
Dynamic form	at
Dynamic Torm	
Year (1911-2048)	Year (00-99) Month (01-12) Day (01-31)
Week (00-53)	Weekday (0-6)
Hour (00-23)	Minute (00-59) Second (00-59) %
Format : %	.dt
Example : 20)141121.dt
%Y Year (1	911-2048)
	ist two digits (00-99)
	as a decimal number (01-12)
	the month (01-31) umber (00-53)
	ay, 0 is Sunday (0-6)
	24h format (00-23)
%M Minute	
%S Second %% %	(00-59)
7070 70	
A filename cannot co	ontain any of the following characters: $\/: *? < >$

L	
Setting	Description
File creation	Automatic mode
	A new file will be created when the name of an existing
	file is changed.
	Trigger mode
	A new file will be created according to the [Trigger
	method] settings.
Trigger method	Limit by number of data records
	A new file will be created when the number of data
	sampling records reaches the specified "Max. data
	records in a file"
	Register status
	A new file will be created when the status of a designated
	bit address meets the specified condition. The condition
	is specified in Mode field.
	Set ON/OFF after triggered
	If selected, after the new file is created, the system will
	set the designated bit address back to ON/OFF state.

File name	The file name can be an alphanumeric name, and certain
	half-width symbols are allowed. The file name can also be
	specified by a file name syntax.
	Dynamic format
	The file names can be set by a designated word address,
	or by a file name syntax indicating the current system
	time. The file name syntax can be specified by selecting
	time buttons or entering the syntax in Format field. The
	length limit is from 1 to 25.
	The following half-width characters are not allowed:
	\/:*?"<>



- If both [Limit by number of data records] and [Dynamic format] check boxes are selected, before startup HMI, please enter the name in the designated register for Dynamic Format, otherwise, it is impossible the reach the "Max. data records in a file", and the data sampling file will not be generated.
- A data sampling file (*.dtl) cannot be written when it's size exceeds 4MB.
- When a new file is generated, the systm will first detect if the filename already exists. If the file name does exist, the newly sampled data will be appended to the existing file.



8.4. Synchronizing cMT Viewer data and Saving to External Device

For other series, when displaying the sampled data in Trend Display object, it is necessary to select from Real-time mode or History mode and the two modes cannot simultaneously be displayed in one object.

cMT Series allows displaying history data and at the same time updates real-time data in one Trend Display or History Data Display object. The data saved in the external device can be



updated.

The rule of synchronizing the data saved in the external device:

- **1.** When the sampled data reaches 10000 records, HMI will automatically saves data to the external device and deletes the earliest 1000 records in HMI.
- 2. If the external device is removed from HMI, and inserted back again at the time when the sampled data is under 9000 records, the data generated during the time the external device is removed is saved in HMI and is not cleared. If the data exceeds 9000 records during the time the external device is removed, the earlier data is cleared and cannot be synchronized even to insert the external device back to the HMI.
- 3. If there already exists sampled data in the external device, the new data is appended without overwriting the original data each time in synchronization.

8.5. Checking History Data of a Specific Date on cMT Viewer.

To check the history data, see the following steps (Use Trend Display object as example).

- 1. Tap the icon in the upper-right corner of the Trend Display object.
- 2. The following dialog box appears.

incel	Option	Do
Begin Date		OFF
Ended Date		OFF
Trend Display S	etting	
Channel Visibili	ty	>
Y Scale		On 👂
	Reset to project default	

3. Specify the [Begin Date] and [Ended Date].



4. Tap [Done] button to finish setting.



9. Object General Properties

This chapter explains the basic settings of an object.

9.1.	Overview	9-2
9.2.	Selecting PLC and Setting Read/Write Address	9-2
9.3.	Using Shape Library and Picture Library	9-3
9.4.	Setting Label Text	9-6
9.5.	Adjusting Profile Size	9-8



9.1. Overview

The basic steps to create an object:

- 1. Selecting the PLC device and setting the read/write address.
- 2. Using Shape Library and Picture Library.
- 3. Setting label text.
- 4. Adjusting profile size.

This chapter explains the basic settings of an object.

9.2. Selecting PLC and Setting Read/Write Address

Most objects read data from PLC devices, so a properly configured PLC address is needed. Select the PLC to control at [PLC name] which comes from [System Parameters Settings] » [Device List].

Write address				
PLC name :	Local HMI		•	Setting
Address :	LW	▼ 0	IDX 0	16-bit Unsigned
-	_			
Address				×
PLC name :	Local HMI			•
Device type :	LW			•
Address :	0	System tag		
Address format :	DDDDD [range : 0 ~ 10799]			
Index :	INDEX 0 (16-bit)	V Index register		
	16-bit Unsigned 🔹			
Tag Library			ОК	Cancel

Setting	Description
PLC name	Select the PLC type.
Device type	Different PLCs have different device types.
Address	Set the read/write address.
System tag	Address tags include [System Tag] and [User-defined Tag].
	This option allows users to use [System Tag]. [System Tag]
	consists of the preserved addresses by system for
	particular purposes. The address tags include bit registers
	or word registers (LB or LW).
	After selecting [System tag], not only will the [Device type]
	field display the chosen tag, but [Address] field will also
	display the chosen register.

9-2

- ☞ For more information about System Tag, see "22 System Reserved Words and Bits".
- For more information about Index Register, see "11 Index Register".
- For more information about Tag Library, see "16 Address Tag Library".

9.3. Using Shape Library and Picture Library

Shape Library and Picture Library are used to add visual effects on objects. Select [Shape] tab to use the libraries when creating an object.

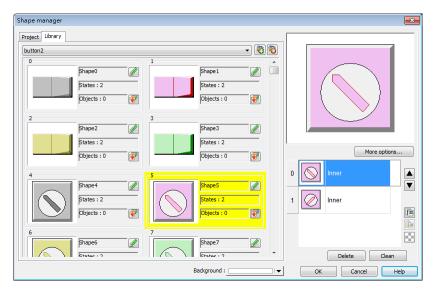
Vew Bit Lamp Object 🥃
General Security Shape Label
Shape
Shape Library
🖉 Inner
▼ Frame
Interior pattern : Pattern Style
Duplicate these attributes to every state
Picture Picture Library
Set to original dimensions
Preview
0 1 State : 0 -
Picture :
Ficure -
Background :
OK Cancel Help



Setting	Description
Shape Library	Select [Use shape] check box and select a shape from the
	library. See next page.
Inner	Select this check box to set the inner part of a shape. Click
	the drop down button to select a color or customize a
	color and click [Add to Custom Colors]. EasyBuilder Pro will
	save this color.
	Color
	Basic colors: Qustom colors: Define Custom Colors >> Define Custom Colors >> Define Custom Colors >> Add to Custom Colors
Frame	Select this check box to set the frame of a shape. Click the
	drop down button to select a color.
Inner Pattern	Set the color of the interior pattern of the shape.
	Pattern Style
	Click this button to select a pattern.
Duplicate these	
Duplicate these attributes to	Duplicate all attributes of the current state to other states.

9.3.1. Shape manager

Click [Shape Library...] button to open the [Shape manager] dialog box. The currently selected shape is highlighted yellow.



The illustration above provides information of one of the Shapes in the Shape Library:

Shape5 Name of the shape.

States: 2 Number of states of the shape.

Objects: 1 This shape is used by 1 object in the project.

The illustration above shows the shape has two states, State 0 and State 1, and contains only "inner" but not "frame." When finished, click [OK], and the object will use the selected shape.

9.3.2. Pictures manager

Click [Picture Library...] button to open the [Pictures manager] dialog. The currently selected picture is highlighted yellow.

Pictures manager		
Project Library		
button_type14	- 10 10	
0 States : 2 Objects : 0	States : 3 Objects : 0	
2 5tates : 1 Objects : 0	States : 1 Objects : 0	Export Modify
4 5 6		0 5975 bytes
States : 2 Objects : 0	States : 2 Objects : 0	1 0 5948 bytes 2 0 PNG 126x185 6002 bytes
6 France - 7	Ktater + 7	2 6002 bytes
More picture libraries	Background :	OK Cancel Help

For more information, see "14 Shape Library and Picture Library".



9.4. Setting Label Text

Bit Lamp Object's Properties
General Security Shape Label Profile
Use label Use label library Convert labels to bitmap images (Use bitmap font) Label Library
State : 0 V O 1
Font : Arial Color : Size : 16
Align : Left Blink : None Italic Underline
Duplicate these attributes to Every state
Movement Direction : No movement
Content
Tracking Duplicate this label to every state
OK Cancel Help

Setting	Description
Use label	Select this check box to use labels for the object.
Use label	Select this sheet, but to sheeps a label in Label Library
library	Select this check box to choose a label in Label Library.
Use bitmap	Select this check box to convert the label text into bitmap
font	format.
Label Library	Browse Label Library
	For more information, see "15 Label Tag Library
	and Multi Language".
Font	Select a font from the list. EasyBuilder Pro supports
	Windows True-type fonts.
	For cMT-SVR Series, if the font is followed by another font
	enclosed by parentheses, the enclosed font is the one
	used in iPad.

	Font : Arial Color : Arial Black (Helvetica Neue) Arial Bold Arial Narrow (Helvetica Neue) Arial Narrow Helvetica Neue) Arial Narrow Bold (Helvetica Neue) Arial Unicode MS (Heiti TC) AmoPro-Bold (Helvetica Neue) AmoPro-Bold Caption (Helvetica Neue) AmoPro-BoldSuppin (Helvetica Neue) AmoPro-BoldSmText (Helvetica Neue) AmoPro-BoldSubhead (Helvetica Neue) AmoPro-BoldSubhead (Helvetica Neue) AmoPro-Caption (Helvetica Neue)
Color	Select the font color.
Size	Select the font size.
Align	The text aligned
	[Left] [Center] [Right]
	111 111 111 222222 222222 222222 33333333 333333333
Blink	Specify the way the text blinks. Choose [None] to disable
	this feature or set the blinking interval to [1 second] or
	[0.5 seconds].
Italic	Use Italic font. Italic Label
Underline	Use Underline font. Underline Label
Movement	Direction
	Set the direction of the marquee effect. The directions
	include: [No movement], [Left], [Right], [Up], [Down].
	Continuous
	Specify how the marquee effect is displayed.
	If not selecting this check box, the next text appears only
	when the previous text disappears completely.
	Alarm Irm
	Alarm Irm If selecting this check box, the text will be displayed
	If selecting this check box, the text will be displayed
	If selecting this check box, the text will be displayed continuously.
	If selecting this check box, the text will be displayed continuously.
Content	If selecting this check box, the text will be displayed continuously.



Tracking	If this check box is selected, when changing the position of
	the text in one state, the text position in the other states
	will also change in accordance.
Duplicate this	
label to other	Duplicate the current text to the other states.
state	

9.5. Adjusting Profile Size

When an object is created and placed in the editing screen, double click it and select the [Profile] tab to adjust the position and size of the object.

	Bit Lamp Object's Properties
	Position Pinned X: 217 Size
	Keep width/height ratio Width : 50 Width : 50 Width (%) : 100 Height (%) : 100
Setting	Description
Position	Pinned When this check box is selected, the position and the size
	of the object cannot be changed.
	[X] and [Y]
	The coordinates of the position of the object in the editing screen.
Size	Adjust the [width] and [height] of the object.





10. User Password and Object Security

This chapter discusses the protection for operations provided by setting up user passwords and security classes.

10.1.	Overview	10-2
10.2.	User Password and Operable Object Classes	10-2
10.3.	Enhanced Security Mode and Control Address	10-4
10.4.	Enhanced Security Mode Usage	10-7
10.5.	Object Security Settings	10-12
10.6.	Example of Object Security Settings	10-13



10.1. Overview

This chapter discusses the protection for operations provided by setting up user passwords and security classes. There are two authentication modes:

- General Mode
- Enhanced Security Mode

Each mode will be introduced later.

To set up the protection system, please:

- 1. Set user password and operable classes.
- 2. Set object class for objects.

An object belongs only to one security class. Setting the object class to "None" means any user can operate this object.

10.2. User Password and Operable Object Classes

The security parameters can be found in [Edit] » [System Parameter Settings] » [Security]. Two modes are available: General Mode and Enhanced Security Mode.

10.2.1. General Mode

Up to 12 sets of user and password are available. A password should be one non-negative integer. There are six security classes: A to F.

Once the password is entered, the objects that the user can operate are classified. As shown below, "User 1" can only operate objects with class A or class C.

Note

General Mode is not available for cMT-SVR Series.

Exter	nded Memor	y 🛛	Printer/Back	up Server		e-	Mail	Recipes
Device Model		odel	General	System Setting		g	Security	Font
	(0) G	eneral mode		0) Enhand	ed sect	urity mode	
elector	erable classe	s for each u	ser				-	
-		~ 42949672						
1035900	nu muge . o	~ 44545074						
N	o. Enable	Password	c	lass A	Class	В	Class C	Cla: 🔺
N	o. Enable	Password 111	c	lass A	Class	B	Class C	Cla:
			c		Class [B		
▶ 1		111	C		Class [B		
▶ 1 2		111 222	c		<u>Class</u> [[[B		
▶ 1 2 3		111 222 333	C		<u>Class</u> [[[[
▶ 1 2 3 4		111 222 333 0	c	▼ ▼ ▼				



10.2.2. Enhanced Security Mode

Up to 11 users can be set here. In addition, [Administrator] setting is provided. Administrator has all privileges and can operate all object classes. User passwords must be alphanumeric characters and each user can have up to 12 operable classes: A to L. (Up to 127 users can be set in Administrator Tools. Please see "10.4 Enhanced Security Mode Usage" for more details.)

Enhanced Security Mode provides a [Control address] for users to manage the accounts directly on HMI. Please see "10.3 Enhanced Security Mode and Control Address" for more details. Alternatively, use USB Security Key to log in automatically. Insert the device in which the key is saved to log in. Please see 10.4.3 Login / Logout Automatically with USB Security Key" for more details.

Extende	d Memory	/ F	rinter/Back	up Server	e-h	4ail	F	Recipes
Device	Mo	odel (Jeneral	System Set	tting	Security		Font
	🔘 Ge	eneral mode		💿 Enh	anced secu	ritv mode		
lectoper	able classe:	s for each user -		Ŭ		-		
		r accounts on Hi	MI					
No.	- Enable	Secret user	User na	me	assword	(Class A	
1			user1	1			V	
2			user2	2			V	=
3			user3	3			1	
4			user4	4			~	
5			user5	5			1	
6			user6	6			1	
7			user7	7				Ψ.
•	11	1					•	
Class	s D	escription						
Class	A							
Class	В							
Class								-
-Admini	strator —	_						
		📃 Secret user						_
Us	er name :	admin		Pass	word : [11]	1111		
-Control	address —							
PLC n	ame : Loo	cal HMI			*	Setting		
	ress : LW			1000		16-bit Unsid		
Auu	ress : LW	1		1000		TO-DIC ON SIG	yncu -	
oject pass	word (EM	(TP file)						
	,	Enable		Setting				
œcute aut	o.login/lo	gout when inser	t an USB ke	ey into HMI —				
		🗸 Enable	* Simu	lation not suppo	rts this fund	ction		
– Status a								
						Califica		
PLC n	ame : Loo	cal HMI			•	Setting	••••	

10-3



10.3. Enhanced Security Mode and Control Address

The Control Address is used for login and account management. The Control Address can only be assigned to LW register on Local HMI, and 20 consecutive registers will be used. To log in using Control Address, please select to log in by [user name] or [user index]. Please set [user name] and [password] in [System Parameter Settings] » [Security] » [Enhanced security mode] in advance.

10.3.1. Control Address Settings

When control address is set to LW-n, where n is an arbitrary number, the following addresses will be designated:

Tag Name	Description
command	Commands to be executed: Login, Logout, Add/Setting/Delete Accounts, etc.
command execution result	Displays the result of executing commands.
user index	The index of accounts (used with Option List Object).
user privilege	Binary value. Level A = bit0, Level B = bit1,
user name	Account name (Only alphanumeric characters, "-" or "_", case-sensitive).
password	Account password (Only alphanumeric characters, "-" or " ", case-sensitive).
	command command execution result user index user privilege user name

After setting the [Control address], the relevant addresses can be found in [Address Tag Library] » [Customized].For example, setting [Control address] to LW-0: (UAC stands for User Account Control)

- LW-0 \rightarrow [UAC command]
- LW-1 \rightarrow [UAC command execution result]
- LW-2 \rightarrow [UAC user index]
- LW-3 \rightarrow [UAC user privilege]
- LW-4 ~ LW-11 \rightarrow [UAC user name]
- LW-12 ~ LW-20 \rightarrow [UAC password]

Note

In Enhanced Security Mode, if cMT-SVR Series model is used, the Control Address can only be assigned to PLW register of Local HMI.



10.3.2. Commands

Setting different values in LW-n [command] enables different commands:

Set Value	Command	Corresponding Address
1	Log in by user name	Set [user name] and [password] first. After entering the user name and password, the system will check if they are valid in [System Parameter Settings] » [Security] » [Enhanced security mode].
2	Log in by user index	Set [user index] and [password] first. Please refer to 10.4.4 Enhanced Security Mode with Option List Object.
3	Log out	
4	Change the password of current logged-in user	Set [user name] and [password] first. Please fill in the original password in [user name] and new password in [password].
5	Add an account	Set [user name], [password] and [user privilege] first.
6	Add a temporary account	Set [user name], [password], [user privilege], and [user index] first. The index is for specifying an effective time period (minutes). If 0 is specified, this account is valid until HMI is turned off.
7	Delete an existing account by user name	Set [user name] first.
8	Delete an existing account by user index	Set [user index] first.
9	Setting the privilege of an existing account by user name	Set [user name] and [user privilege] first.
10	Setting the privilege of an existing account by user index	Set [user index] and [user privilege] first.
11	Setting the password of an existing account by user name	Set [user name] and [password] first.
12	Setting the password of an existing account by user index	Set [user index] and [password] first.
13	Read the privilege of an existing account by user name	Set [user name] first. If the command succeeds, [user privilege] can be displayed.
14	Read the privilege of an existing account by user index	Set [user index] first. If the command succeeds, [user privilege] can be displayed.



Note

- Add a temporary account: The difference between general accounts and temporary accounts is that temporary accounts will not be stored in the system; therefore, it is invalid after HMI turned off. The temporary accounts will be automatically deleted when they are expired.
- Delete the existing account: The currently logged in account cannot be deleted.
- Offline/Online Simulation: Simulate using the account settings in the program. Any modifications of the account during simulation will not be reserved for next simulation.
- admin: Default administrator account, cannot be deleted, has all privileges and cannot be changed.
- System Register LW-10754: Displays current user name.
- System Register PLW-10754: Displays current user name. (Only available for cMT-SVR)

10.3.3. Command Execution Results

After the command is executed, the system will store the result code to control address LW-n + 1. The listed result codes below are shown in hexadecimal format.

Result Codes	Command execution result
(0x001)	Succeeds
(0x002)	Invalid command
(0x004)	Account exists (when adding a new account)
(0x008)	Account not exists
(0x010)	Password error
(0x020)	Deny command
(0x040)	Invalid name
(0x080)	Invalid password character exists
(0x100)	Invalid import data
(0x200)	Out of validity range (when log in by USB Security Key). The [Effective Time] can be set in Administrator Tools.

Note

Users can add a new event in Event (Alarm) Log, and designate the [Read address] to LW-n
 + 1 [command execution result]. Open [Message] tab » [Text] » [Content] and specify the message to be displayed in Event Display Object for showing command execution result.



10.4. Enhanced Security Mode Usage

10.4.1. Importing User Accounts

The user accounts can be set using other tools we provide, apart from the settings in [System Parameter Settings] » [Security] tab. Administrator Tools can also be used to set user accounts. Administrator Tools can be found in the installation directory. After the program starts, select the [User Accounts] check box. Up to 127 accounts can be added.

Save	to USB	Contents of the	USB data							
•	V	User Accounts								
	[f***]	USB Security K	ey							
	[77]	e-Mail SMTP Se	erver Settings							
		e-Mail Contacts								
	int Setting									
No.	Secret	User name	Password	Class A	Class B	Class C	Class D	Class E	Class F	Cla
1		001001	001001	V						
2		002002	002002	V	V			V		
3	V	003003	003003	V		V				
4		004004	004004	V			V			
5	V	005005	005005	V					V	
6		006006	006006	V			V			
7		007007	007007	V	V	V				
8		008008	008008	V				V		
9		009009	009009	V						
10		010010	010010	\checkmark	V		V		V	
(III							4
	Add		Remove			Impo	ort		Export	
ffective	Time						a			
Restr	ict the usi							N	Save to USI	3
	2013/	六月 /03 14:43	- 2013	/ 六月 /03 14:	43 🔻 🗮 👘					

For more information, see "36 Administrator Tools".

The added accounts can be stored in USB disk or SD card and imported in HMI by a Function Key Object. To do so, create a Function Key Object, and select [Import user accounts].



New Fund	ction Key Object
General	Security Shape Label
	Comment :
	C Activate after button is released
	Function mode
	Import user accounts
	◎ Use [USB Security Key] to Login
-AS	Data position
	○ SD card
	Account import mode
	Overwrite Append
	Delete file after importing user accounts
He	
114	OK Cancel
	Import user data/Use [USB Security Key]

When finished, insert the external device to HMI, and press Function Key to import accounts. If [Overwrite] is selected, the existing accounts will be overwritten with new accounts and automatically log out after importing. If select [Delete file after importing user accounts] check box, the system will delete the account data saved in the external device after importing. If the [Effective Time] in Administrator Tools is specified, the importing can only be done in the time limit specified. The imported accounts will not be deleted by system when the effective time ends.

10.4.2. Login with USB Security Key

Instead of entering user name and password to login, a key can be used to do so. In EasyBuilder Pro installation directory, launch Administrator Tools, select [USB Security Key] check box. The account information uses the predefined data in [System Parameter Settings] » [Security].



) Adn	ninistrator Too	ls		×
	Save to USB	Contents of the USB data		
		User Accounts		
•		USB Security Key		
		e-Mail SMTP Server Settings		
		e-Mail Contacts		
USB	Security Key			
		User name :	admin	
		Password :	•••••	
		Confirm :	•••••	
	fective Time			
			(
Ľ	Restrict the usi	ng terms	m ² m	
	2013/	六月 /03 14:43 🚽 🚔 🔷	2013/ 六月 /03 14:43 🖃	Save to USB
Helt	p Topics			
ner	p ropus			



Please note that the user accounts used for USB Security Key must already exist in HMI.

For more information, see "36 Administrator Tools".

USB Security Key can be stored in USB disk or SD card, and create a Function Key to log in by USB Security Key as shown below:



General Security Shape Label Comment : Activate after button is released Function mode
Activate after button is released
Function mode
Function mode
Import user accounts
Use [USB Security Key] to Login
A: Data position
○ SD card
H4
OK Cancel
Import user data/Use [USB Security Key] Settings

When finished, insert the external device to HMI, and press Function Key to log in using USB Security Key. If the [Effective Time] in Administrator Tools is specified, the login can only be done in the time limit specified. The system will log out automatically when the key expires.

10.4.3. Login / Logout Automatically with USB Security Key

As shown below, in [System Parameter Settings], select [Enable] check box for [Execute auto. Login / logout when insert an USB key into HMI].

-E>	ccute auto.logii	n/logout when ins	ert an USB kej	y into HMI ——			
		🔽 Enable	* Simula	ation not supports	this fun	ction	
	– Status address						
	PLC name :	Local HMI			-	Setting	
	Address :	LW	•	0		16-bit Unsigned	

This function allows automatic login / logout using an USB security key. Insert the USB disk in which the key is saved to HMI to log in, and remove the USB disk to log out. The login / logout status will be written into a designated address, the result codes of login / logout:

0x00: No Action 0x01: Login Succeeds

0x04: Login Fails

0x08: Login Succeeds

0x10: Logout Fails



For more information about USB Security Key, see "36 Administrator Tools".

Note

- When Auto Login / Logout is enabled, log in by [Function Key] object is not possible, but it is still possible to log in / out with a designated control address.
- This function does not support On-line / Off-line simulation.
- Only the USB Security Key saved in USB disk is valid.

Llick the icon to download the demo project that explains how to use USB Security Key to

log in / out. Please confirm your internet connection before downloading the demo project.

10.4.4. Enhanced Security Mode with Option List Object

Enhanced Security Mode uses Control Address LW-n + 2 as account index. With Option List Object, account names and privileges can be displayed. Users can select whether or not to display the account privileges and secret users in Option List. Secret users are set to be hidden in [System Parameter Settings] » [Security] » [Enhanced Security Mode]; their account names will be hidden in Option List if [Secret user] check box is not selected. If the control address is set to LW-0, the monitor address for index of Option List is designated to LW-2.

New Option List Object
Option list Mapping Security Shape Label
Comment :
Mode : Drop-down List Background :
Selection :
Direction : Down
Source of item data : User account 🗸 🗸
Sort
Ascending Descending Vivilege Secret user
Monitor address
PLC name : Local HMI
Address : LW 👻 2 16-bit Unsigned

Click the icon to download the demo project that explains more about Enhanced Security Mode. Please confirm your internet connection before downloading the demo project.

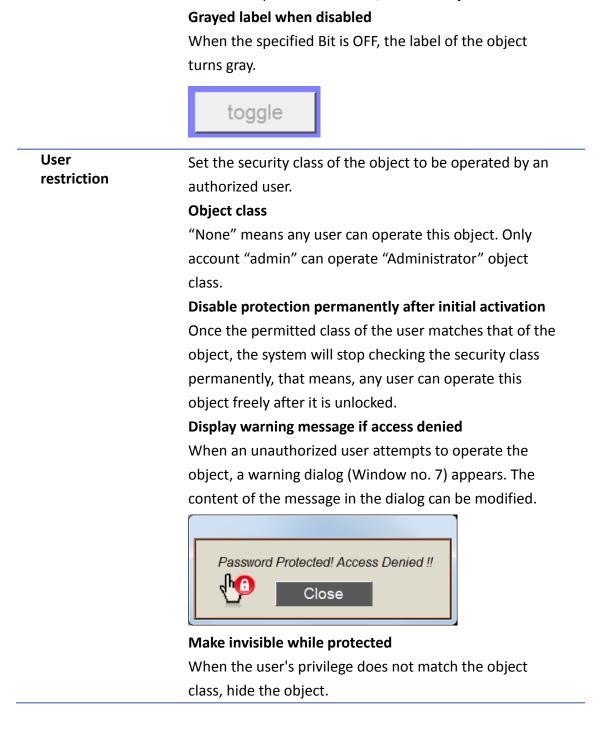


10.5. Object Security Settings

New Function Key Object
General Security Shape Label
Safety control Min. press time (sec) : 0
Display confirmation request Max. waiting time (sec) : 10
Interlock Image: Interlock function
 Hide when disabled Grayed label when disabled
Enable when Bit is ON Enable when Bit is OFF
PLC name : Local HMI Address : LB O
User restriction
Object class : Class : A
 Display warning message if access denied Make invisible while protected

Setting	Description			
Safety control	[Min. press time (sec)]			
	Press and hold the object for longer than the [Min. press			
	time] set here to activate the object.			
	[Display confirmation request] After pressing the object,			
	a dialog appears for operation confirmation. If the			
	response to this dialog comes later than the set [Max.			
	waiting time (sec)], this dialog disappears automatically			
	and the operation will be canceled.			
	Please confirm the operation OK Cancel			
Interlock	When this check box is selected, the specified Bit address			
	is used to enable or disable the object. As shown, if LB-0			
	is ON, the object is enabled.			





Hide when disabled

When the specified Bit is OFF, hide the object.

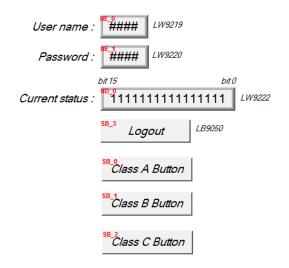
10.6. Example of Object Security Settings

The following shows an example of setting object security class:

 Create a project, go to [System Parameter Settings] » [Security] » [General] to enable 3 users:



- User 1 = Operable class: A
- User 2 = Operable class: A, B
- User 3 = Operable class: A, B, C
- **2.** Design Window no. 10 as shown:



Create two [Numeric Input] objects:

[LW-9219] User no. (1~12), Length = 1word

[LW-9220] For entering user password. Length = 2 words

Create a [Numeric Display] object:

[LW-9222] Displays the status of current user. (16-bit Binary)

Create a [Set Bit] object

[LB-9050] logout

Create three [Set Bit] objects:

Each set to different classes but all select [Made invisible while protected].

3. After setting, please save and compile the project and execute off-line simulation. The below shows how it works when simulating.

User name : 1 LW9219
Password : 0 LW9220
bit 15 bit 0
Current status : 00000000000000000000000000000000000
Logout L89050

Before entering the password, it displays "000000000000000, which means that the user operable object class is "None". [Class A Button] ~ [Class C Button] objects are classified from "A" to "C" and selected [Made invisible while protected]; therefore they are hidden at this moment.



User name : 1 LW9219 Password : 111 LW9220 bit 15 bit 0 Current status : 00000000000001 LW9222 Logout LB9050 Class A Button	Enter User 1 password "111". Since User 1 is only allowed to operate class A objects, [Class A Button] object appears for operating. [LW-9222] bit 0 turns to "1" means that user can operate class A objects.
User name : 3 LW9219 Password : 333 LW9220 bit 15 bit 0 Current status : 000000000001111 LW9222 Logout LB9050 Class A Button Class B Button Class C Button	Enter User 3 password "333". Since User 3 is allowed to operate class A, B, C objects, [LW-9222] bit 0 ~ bit 2 turns to "1", means that user can operate class A ~ C objects.
User name : 3 1.W9219 Password : 333 1.W9220 bit 15 bit 0 Current status : 00000000000000 1.W9222 Logout 1.B9050	Click [Logout] button to log out, the system will return to the initial state, and current user can only operate class "None" objects.

Note

- Password input: If the password is incorrect, [LB-9060] will be ON; if the password is correct, [LB-9060] will be OFF. All user passwords (User 1 to User 12) can be obtained from system registers [LW-9500] ~ [LW-9522], 24 words in total.
- Changing password directly on HMI: When [LB-9061] is set ON, the system will read data in [LW-9500] ~ [LW-9522] to update user password. The new password will be used in future operations. Please note that the user operable object classes will not be changed due to the change of password.



11. Index Register

This chapter explains how to use Index Register.

11.1.	Overview1	1-2
11.2.	Examples of Index Register	1-2



11.1. Overview

EasyBuilder Pro provides Index Registers for changing addresses flexibly. With Index Registers, user can change the object's read/write address directly on HMI without changing its settings. There are 32 Index Registers, divided into 16-bit and 32-bit.

Address		
PLC name :	Local HMI	•
Device type :	LW-9002 (32bit-float) : input high limit	-
	LW-9200 (16bit) : address index 0 LW-9201 (16bit) : address index 1	^
Address format :	LW-9202 (16bit) : address index 2 LW-9203 (16bit) : address index 3	
	LW-9204 (16bit) : address index 4 LW-9205 (16bit) : address index 5	
	LW-9206 (16bit) : address index 6 LW-9207 (16bit) : address index 7	Ξ
	LW-9208 (16bit) : address index 8 LW-9209 (16bit) : address index 9	
	LW-9210 (16bit) : address index 10	
Tag Library	LW-9211 (16bit) : address index 11 LW-9212 (16bit) : address index 12	
	LW-9213 (16bit) : address index 13 LW-9214 (16bit) : address index 14	
	LW-9215 (16bit) : address index 15 LW-9219 (16bit) : user no. (1~12)	
	LW-9220 (32bit) : password	
	LW-9222 (16bit) : classes can be operated for current user (bit 0:A, bit 1:B,bit LW-9230 (32bit) : address index 16	
	LW-9232 (32bit) : address index 17 LW-9234 (32bit) : address index 18	
	LW-9236 (32bit) : address index 19 LW-9238 (32bit) : address index 20	
	LW-9240 (32bit) : address index 21	
	LW-9242 (32bit) : address index 22 LW-9244 (32bit) : address index 23	
	LW-9246 (32bit) : address index 24 LW-9248 (32bit) : address index 25	
	LW-9250 (32bit) : address index 26	$\overline{\mathbf{v}}$

The corresponding address of 16-bit Index Register 0 to 15: LW-9200 (16bit) to LW-9215 (16bit) The maximum offset range is 65536 words.

The corresponding address of 32-bit Index Register 16 to 31: LW-9230 (32bit) to LW-9260 (32bit)

The maximum offset range is 4294967296 words.

When using [Index register], the address is designated by the following equation:

The constant set in [Address] + the value in the chosen Index Register.

Note

Index Registers work for the Word registers. For Bit registers, adding 1 to the value in the Index Register, the offset is 16 bits.

11.2. Examples of Index Register

The following explains the way to designate the register while Index Register is used. If **not** selecting **[Index register]** check box and set address to [LW-10]. The system will directly



read / write LW-10						
	Address				— ×	
	PLC name : Device type :				•	
	Address :	10	System tag			
	Address format :	DDDDD [range : 0 ~ 10799]				
			Index register			
	Tag Library			ОК	Cancel]

If select **[Index register]** check box and set [Index] to [INDEX 0 (16-bit)]. The system will read / write [LW(10 + value in Index Register 0)].

If the data in [LW-9200] is "5", the designated address is [LW(10+5)] = [LW-15].

Address	×
PLC name :	Local HMI 👻
Device type :	LW
Address :	10 System tag
Address format :	DDDDD [range : 0 ~ 10799]
Index :	INDEX 0 (16-bit) 🔻 🔽 Index register
Tag Library	OK Cancel

Here's a demo project shown as an example:

Index Function of Word			
0000 Index 0 (LW-9200)	0000 LW-0 + Index 0 (LW-9200)	0003 LW-10	0006 LW-20
Index Function of Bit		L	
0000			
Index 6 (LW-9206)	LB-0 + Index 6 (LW-9206)	LB-16	LB-32





Example 1

The following shows an example of using a Word register and select [Index register]. If the value in [LW-0] is 0, in [LW-10] is 3, and in [LW-20] is 6, the result is:

Index Function of Word			
0000	0000	0003	0006
Index 0 (LW-9200)	LW-0 + Index 0 (LW-9200)	LW-10	LW-20

If the value in Index 0 (LW-9200) is 0, then [LW0 + Index 0] = read [LW-0].

Index Function of Word			
0010	0003	0003	0006
Index 0 (LW-9200)	LW-0 + Index 0 (LW-9200)	LW-10	LW-20

If the value in Index 0 (LW-9200) is 10, then [LW0 + Index 0] = read [LW-10] = 3.

Example 2

The following shows an example of using a Bit register and select [Index register].

If the state of [LB-16] is ON, and the state of [LB-32] is OFF.

Since 1 Word = 16 Bit, adding 1 in Index Register, the offset is 16 bits.



If Index 6 (LW-9206) is set to 1, then switch [LB-0 + Index6] reads LB-16 which is in ON state.



If Index 6 (LW-9206) is set to 2, then switch [LB-0 + Index6] reads LB-32 which is in OFF state.



Note

When using Index Registers for Bit register, the offset is 16 bits. For example, if the Bit register is LB-0, and set the value in Index Register to 1, then LB-16 will be activated. If set the value in Index Register to 2, then LB-32 will be activated.

Lick the icon to download the demo project. Please confirm your internet connection.



12. Keyboard Design and Usage

This chapter explains how to design and use keyboard in EasyBuilder Pro.

12.1.	Overview	12-2
12.2.	Steps to Design a Popup Keyboard	12-2
12.3.	Steps to Design a Keyboard with Direct Window	12-5
12.4.	Steps to Design a Fixed Keyboard on Screen	12-6
12.5.	Steps to Design a UNICODE Keyboard	12-7



12.1. Overview

Numeric Input and ASCII Input objects need keyboard as an input tool. Both numeric keyboard and ASCII keyboard are created with Function Key object. Apart from the keyboards provided by EasyBuilder Pro, you can create the keyboard if needed.

The types of the keyboards are:

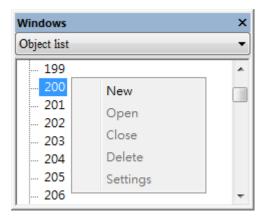
- Popup Keyboard (with or without title bar)
- Fixed Keyboard
- UNICODE Keyboard



 cMT Series uses the built-in keyboard of iPad which cannot be customized. Please skip this chapter if cMT Series is used.

12.2. Steps to Design a Popup Keyboard

1. Create and open a window for the new keyboard. For example, set to "window no. 200".



2. Adjust the height and width of "window no. 200" and create a variety of Function Key objects in [ASCII/UNICODE mode].

SP_0			
FK_0) ^{FK_1}	FK_2	
FK_4	FK_5	FK_6	CR
FK_8	FK_9	FK_10	ES
FK_12	FK 13	FK_14	INT



Set one of the Function Key objects as the [Esc] key.

ASCII/UNICODE mode					
🔘 [Enter]	🔘 [Backspace]	🔘 [Clear]	(E∞)		
🔘 [Delete]	🔘 [Left]	🔘 [Right]			
○ [ASCII] / [UNICODE]					

Set another Function Key object as the [Enter] key.

_A	SCH/UNICOL)E mode			
	💿 [Enter]	🔘 [Backspace]	🔘 [Clear]	🔘 [Ex:]	
	🔘 [Delete]	🔘 [Left]	🔘 [Right]		
) [ASCII] ([UNICODE]			

The rest are mostly used to enter numbers.

-ASCII/UNICODE	mode			
🔘 [Enter]	🔘 [Backspace]	🔘 [Clear]	🔘 [Ex:]	
🔘 [Delete]	🔘 [Left]	🔘 [Right]		
(ASCII) / [U]	NICODE] 1			

3. Select a suitable picture for each Function Key object and place a picture object at the bottom layer as a background.



 Select [System Parameter Settings] » [General] » [Keyboard] » [Add] to add "window no. 200". Up to 32 keyboards can be added.

Ad	d a keyboard window 🧮	٢
	Window no. : 200. Keyboard 🗸	
	OK Cancel	

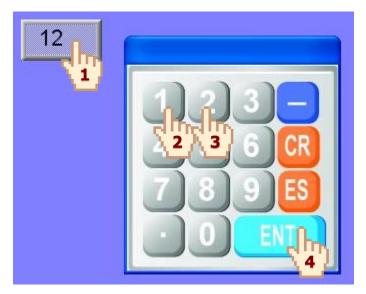
5. After the keyboard window is added, when you create Numerical Input and ASCII Input objects, "200. Keyboard" can be found in [Data Entry] » [Keyboard] » [Window no.]. The



[Popup position] is for designating the display position of the keyboard on the HMI screen. The system divides the screen into 9 areas.

Keyboard Use a popup keyp Hide title bar Restart the keypad		ut val	ue is o	ut of range
Window no. : 200. Keyboard Popup position :	0	0	0	_
{relative to HMI screen}	0) ()	0	

6. Select "200.Keyboard". When you press Numerical Input or ASCII Input objects on the screen, "window no. 200" will pop up. You can press the keys on the keyboard to enter data.





12.3. Steps to Design a Keyboard with Direct Window

- **1.** Create a Direct Window object and set a read address to activate it.
 - In [General] » [Attribute] select [No title bar] and the correct [Window No.].

Direct Window Object's Properties
General Profile
Comment :
Trigger: ON
Read address
PLC name : Local HMI Setting
Address : LB 🔻 0
Attribute Style : No title bar
Window No.: 200. Keyboard

2. Open the [Profile] tab to set the same size as the created keyboard window.

Direct Window Object's	Properties			×
General Profile				
Position				
📃 Pinned	X : 233	÷	¥: 134	\$
Size				
📃 Keep width/heigh	t ratio			
	Width : 300	*	Height : 200	-
	Width (%) : 100	÷ F	Height (%) : 100	-

3. Create a Numeric Input object, and don't select [Use a popup keypad] check box.

Numeric Input Object	ct's Properties	×
General Data Entry	Numeric Format Security Shape Font Profile	
Mode :	Touch •	
Input order Enable		
Keyboard		
	🔲 Use a popup keypad	



4. Create a Set Bit object, set address to [LB-0] and set [Set style] to [Set ON]. Overlay it on the Numeric Input object. Pressing the Numeric Input object will open the keyboard window.

New Set General	Bit Object Security Shape Label	×
	Comment :	
Write	address	
PLC	name : Local HMI	
Ad	idress : LB v 0	
	Write after button is released	
Attrib	ute Set style : Set ON 🔹	

5. Add Set Bit objects on the [Enter] and [ESC] Function Key objects respectively. Set address to [LB-0] and [Set style] to [Set OFF]. In this way when pressing either [Enter] or [ESC] key will close the keyboard window.

12.4. Steps to Design a Fixed Keyboard on Screen

You can also place a fixed keyboard on the screen instead of popup keyboard or Direct Window. This type of keyboard can't be moved or closed.

- Create a Numeric Input object, in [Data Entry] » [Keyboard] don't select [Use a popup keypad] check box.
- 2. Use Function Key objects to design the keyboard and place it on the screen.
- 3. Press the Numeric Input object and enter a value with Function Key objects directly.



12.5. Steps to Design a UNICODE Keyboard

The following steps explain how to create a UNICODE keyboard with Function Key objects.

- 1. Place an ASCII Input object on the window and select [Use UNICODE] check box.
- 2. Create Function Key objects as shown in the following figure, and an [Enter] key. A simple UNICODE keyboard is created.

New ASCII Input Object E3 General Data Entry Security Shape Font Description :	ASCII/UNICODE mode
Mask V Use UNICODE	UNICODE
Reverse high/low byte Read address PLC name : Local HMI	FK_0 α ASCII/UNICODE α ASCII/UNICODE mode
Address : LW	$\begin{array}{c c} FK_1 & & & \\ \beta & & & \\ \hline \end{array} \\ \hline & & & \\ \hline \hline \\ \hline & & & \\ \hline \hline \\ \hline \\ \hline$
Address	FK_2 γ ASCII/UNICODE mode ASCII/UNICODE mode Control Cont
Device type : LW Address : 0 System tag	FK_3 S [Backspace] [Delete] [Left] @ [ASCII] / [UNICODE] Y
Address format : DDDDD [range : 0 ~ 10799]	FK_4 Enter
	© [Enter] © [Backspace] © [Delete] © [Left]
No. of word : 8 Tag Library OK Cancel	● [ASCII] / [UNICODE] 8

Note

You can group up the elements of the designed keyboard and save to Group Library for future use.



13. Objects

This chapter explains how to use different objects.

Bit Lamp
Word Lamp 13-5
Set Bit
Set Word 13-11
Function Key13-18
Toggle Switch
Multi-State Switch
Slider
Numeric
ASCII
Indirect Window
Direct Window13-50
Moving Shape 13-53
Animation
Bar Graph
Meter Display 13-65
Trend Display
History Data Display13-86
Data Block Display
XY Plot
Alarm Bar and Alarm Display 13-106
Event Display
Data Transfer (Trigger-based)
Backup
Media Player
Data Transfer
PLC Control
Scheduler
Option List
Timer
Video In and Video In (USB Camera)
System Message



13.33.	Recipe View	
13.34.	Flow Block	13-174
13.35.	Operation Log	
13.36.	Combo Button	
13.37.	Circular Trend Display	
13.38.	Picture View	
13.39.	File Browser	
13.40.	Recipe Import/Export	
13.41.	Pie Chart	
13.42.	QR Code	



13.1. Bit Lamp

13.1.1. Overview

Bit Lamp object displays the state of a designated bit address. If the bit state is OFF, the State 0 shape will be displayed. If the bit state is ON, the State 1 shape will be displayed.



13.1.2. Configuration



Click the Bit Lamp icon on the toolbar to open a Bit Lamp object property dialog box. Set up the properties, press OK button, and a new Bit Lamp object will be created.

General Tab

	Security	Shape	Label					
	Commen	t :	AC 55					
		Bit	t Lamp	C	Toggle S	witch		
PLC	address name : Lo idress : LE	(eccan	vert signal	• 0		-	Setting	5
Blink	ing							
Blink		e : None						•
Blink:	Mod	100	0.000	onding picture	for currer	ut state		•
	Mod	100	0.000	onding picture	for currer	it state		·
	Mod	100	0.000	onding picture	for currer	ut state		•



Setting	Description			
Comment	User can describe the information of the object. Bit Lamp / Toggle Switch			
	Switch between Bit Lamp and Toggle Switch features.			
Read address	Click [Setting] to select the [PLC name], [Address], [Device type],			
	[System tag], [Index register] of the bit device that controls the [Bit			
	Lamp] object. Users can also set address in [General] tab while			
	adding a new object.			
	Invert signal			
	Reverses the display of ON / OFF states. For example, if [Invert			
	signal] check box is selected, when the designated bit is OFF, the			
	object displays ON state.			
Blinking	The appearance of the object may alternate between states when			
	the bit is ON or OFF.			
	Mode:			
	None			
	No blinking.			
	Alternating image on state 0			
	The appearance of the object alternates between State 0 and 1			
	when the bit is OFF.			
	Alternating image on state 1			
	The appearance of the object alternates between State 0 and 1			
	when the bit is ON.			
	Blinking on state 0			
	The State 0 appearance of the object will blink when the bit is OFF.			
	Blinking on state 1			
	The State 1 appearance of the object will blink when the bit is ON.			
	Hide picture/shape if no corresponding picture for current state			
	If selected, when there are not enough pictures to represent all the			
	states, hides the picture. Otherwise, displays the last state.			

Note

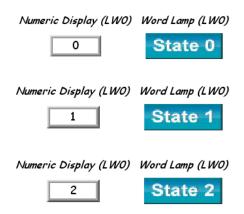
In [Label] tab, if select [ON=OFF (use state 0)] check box, both state 0 and 1 follow the settings of state 0.



13.2. Word Lamp

13.2.1. Overview

Word Lamp object displays the state according to the value of a designated word register. Up to 256 states are available. When the value of the register is 0, State 0 appearance of the object is displayed, and with the register value being 1 the object displays State 1, and so on.



13.2.2. Configuration



Click the Word Lamp icon on the toolbar to open a Word Lamp object property dialog box. Set up the properties, press OK button, and a new Word Lamp object will be created.



General Tab

Jenera	and the second se	
	Comment :	1
50	Word Lamp O Multi-State Switch	
	Mode : Value Offset : 0	
Rea	d address	
	Address : LW 🗸 0 16-bit Unsigned	d
Attr	ibute No. of states : 2 🗸	•
- A ftr	NEW YORK AND A CONTRACT OF A C	
	No. of states : 2	
	No. of states : 2 -	
De	No. of states : 2 Hide picture/shape if no corresponding picture for current state OK Cancel Help	2
De Us	No. of states : 2 Hide picture/shape if no corresponding picture for current state OK Cance Help Escription	2

	Word Lamp / Multi-State Switch				
	Switch between Word Lamp and Multi-State Switch features.				
Mode / Offset	Word Lamp object offers the following three modes:				
	Value				
	The state is displayed according to the value in the designated word				
	address and plus the [Offset].				
	As shown below, if the value within LW-200 is 3, since the offset is				
	set to 3, the shape of state 6 is displayed. (value 3 + offset 3)				
	New Word Lamp Object				
	General Security Shape Label				
	Comment :				
	Mode : Value Offset : 0				
	Read address				
	PLC name : Local HMI				



Setting Comment

LSB

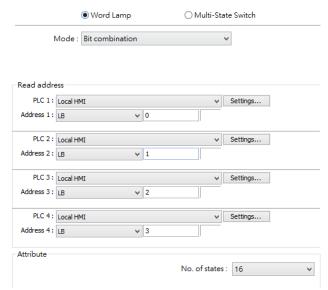
Convert the value from decimal to binary. The least significant

Decimal	Binary	Displayed state
0	0000	State 0 displayed. All the bits are 0.
1	0001	State 1 displayed. The least significant active bit is bit 0.
2	0010	State 2 displayed. The least significant active bit is bit 1.
3	0011	State 1 displayed. The least significant active bit is bit 0.
4	0100	State 3 displayed. The least significant active bit is bit 2.
5	0101	State 1 displayed. The least significant active bit is bit 0.
6	0110	State 2 displayed. The least significant active bit is bit 1.
7	0111	State 1 displayed. The least significant active bit is bit 0.
8	1000	State 4 displayed. The least significant active bit is bit 3.

active bit in a binary data word selects the state displayed.

Bit combination

Lamp state depends on the states of bit combinations, where PLC 1 represents the least significant bit (LSB), PLC 2 represents the next LSB, and so on. Maximum number of bit is 4, for a total of 16 states. Changing [No. of states] in Attribute group box changes the number of read addresses.



Change state by time

The state displayed changes on a time basis. The frequency can be set.

Read address	Click [Setting] to select the [PLC name], [Address], [Device type],
	[System tag], [Index register] of the word device that controls the



	[Word Lamp] object. Users can also set address in [General] tab while adding a new object.
Attribute	No. of states
	The number of states is utilized by the object. The state is
	numbered from 0, so the number of states minus 1 will be the state
	number. If the value within the word register is ≥ [No. of states]
	defined in Attribute, the highest state will be displayed.
	If the number of states is set to 8, the valid states will be 0, 1, 2,,
	7. In this case if the word value is 8 or higher, the system will
	display the state 7 shape.
	Hide picture/shape if no corresponding picture for current state
	If selected, when there are not enough pictures to represent all the
	states, hides the picture. Otherwise, displays the last state.



In [Label] tab, Language 1 determines the relevant settings of the font. For Language 2~8, only the font size can be changed and other settings follow Language 1.

New Word Lamp Object	New Word Lamp Object
General Security Shape Label	General Security Shape Label
Use label Use label library Label tag: Label_0 Label Library	Use label Use label library Label tag : Label_0 Label Library
Language : 1 State : 0 V V 0 1 Attribute Font : Arial	Language : 2
Color :	Color:
Italic Uuplicate these attributes to Every state Every language All Movement Direction : No movement	Italic Duplicate these attributes to Every state Every language All Movement Direction : No movement
Content text	Content
Tracking Duplicate this label to every state	Tracking Duplicate this label to every state
OK Cancel Help	OK Cancel Help



13.3. Set Bit

13.3.1. Overview

The Set Bit object provides two operation modes: manual or automatic. Manual mode can trigger a designated bit address to change the state between ON and OFF when the object is touched. In automatic mode, the bit is automatically activated when a pre-defined condition occurs; touching the button will not be effective.

13.3.2. Configuration



Click the Set Bit icon on the toolbar to open a Set Bit object property dialog box. Set up the properties, press OK button, and a new Set Bit object will be created.

General Tab

General Security	/ Shape Label
Commer	it :
Write address	
PLC name : L	ocal HMI 🔹 Setting
Address :	в то
	Write after button is released
Attribute	
	e : Set ON
occocy.	Set ON
	Set OFF
	Toggle
Macro	Momentary Periodic togale
Execute r	naci Set ON when window opens
	Set OFF when window opens
	Set ON when window closes Set OFF when window closes
	Set ON when backlight on
	Set OFF when backlight on
	Set ON when backlight off
	Set OFF when backlight off

Setting	Description
Write address	Click [Setting] to select the [PLC name], [Address], [Device type],
	[System tag], [Index register] of the bit device that controls the Set
	Bit object. Users can also set address in [General] tab while adding
	a new object.



Write after button is released

If this function is selected, the action is delayed till button is released; otherwise, the action is executed once the button is pressed. This function does not work with momentary buttons.

Mode / Offset	Set style	Description		
···· ·	Set ON	Set ON the designated bit of the device.		
	Set OFF	Set OFF the designated bit of the device.		
	Toggle	Alternates the bit state each time pressed.		
	Momentary	Holds the bit ON only while button is pressed.		
	Periodical toggle	Set a designated bit ON and OFF at a set time interval. Time interval can be selected; the range is from 0.1 to 25.5 seconds.		
	Set ON when	Set ON the bit within the window when the		
	window opens	window opens.		
	Set OFF when	Set OFF the bit within the window when the		
	window opens	window opens.		
	Set ON when window closes	Set ON the bit within the window when the window closes.		
	Set OFF when	Set OFF the bit within the window when the		
	window closes	window closes.		
	Set ON when backlight on (N/A for cMT)	Set the bit ON when the backlight is turned ON.		
	Set OFF when backlight on (N/A for cMT)	Set the bit OFF when the backlight is turned ON.		
	Set ON when backlight off (N/A for cMT)	Set the bit ON when the backlight is turned OFF.		
	Set OFF when backlight off (N/A for cMT)	Set the bit OFF when the backlight is turned OFF.		
Macro	Set Bit object can trigger the start of a Macro routine when the			
	Macro has been created in advance.			
	1	mation, see "18 Macro References".		
Trigger mode	If [Set style] is set to [Toggle], there is a further selection to make of			
	whether the macro operates after Off to ON, ON to OFF transition,			
	or at both of the changes of state.			

Note

In [Label] tab, if select [ON=OFF (use state 0)] check box, both state 0 and 1 follow the settings of state 0.



13.4. Set Word

13.4.1. Overview

The Set Word object provides two operation modes: manual or automatic. Manual mode can change the value in a designated word address when the object is touched. In automatic mode, the word register is automatically activated when a pre-defined condition occurs; touching the button will not be effective.

13.4.2. Configuration



Click the Set Word icon on the toolbar to open a Set Word object property dialog box. Set up the properties, press OK button, and a new Set Word object will be created.

General Tab

Jaw Sat Ward Ob	inst			
lew Set Word Obj				
General Security	Shape Label			
Commen	t:			
⊂Write address –				
PLC name :	ocal HMI		•	Setting
Address :		• 0		16-bit Unsigned
				1
	🔲 Write after button i	s released		
Notification				
	📝 Enable	Set ON	Se 🔘	t OFF
	Before writing	After writing		
PLC name :	ocal HMI		•	Setting
Address :		• 0		
Attribute				
Set Style	e : Write constant value			•
Set value	e: 0			
	ОК	Cancel		Help



Setting	Description		
	•		
Write address	Click [Setting] to select the [PLC name], [Address], [Device type],		
	[System tag], [Index register] of the word device that controls the		
	Set Word object. Users can also set address in [General] tab while		
	adding a new object.		
	Write after button is released		
	If this function is selected, the action is delayed till button is		
	released; otherwise, the action is executed once the button is		
	pressed.		
Notification	If this check box is selected, it will notify a designated bit address		
	(setting ON or OFF).		
	Before writing / After writing		
	Set the state of the designated bit address before or after the		
	manual operation.		
Attribute	Set Style		
	Select the button action from the drop down list, see Example 2.		
	Dynamic limits		
	Set the [Bottom limit] and [Upper limit] by a designated register,		

Example 1

Set the [Bottom limit] and [Upper limit] by a designated register. When Dynamic Address is LW-n, where n is an arbitrary number, the rule of setting Upper / Bottom limit is:

	Content	16-bit	32-bit
	Dynamic address	LW-n	LW-n
	Bottom limit	LW-n	LW-n
	Upper limit	LW-n+1	LW-n+2
When Dynamic Ac	dress is LW-100, the rule	of setting Upper	/ Bottom limit is:
	Content	16-bit	32-bit
	Dynamic address	LW-100	LW-100
	Bottom limit	LW-100	LW-100
	Upper limit	LW-101	LW-102

Example 2

The available button actions are:

• Write constant value

Preset a register with the value entered. Each time when the button is pressed, it writes the [Set value] to the designated register. Data format is as set by the [Write address]; it can be 16-bit BCD, 32-bit BCD, ...32-bit float. As shown below, when the button is pressed, preset the register with 12.



Attribute		
Set Style :	Write constant value	-
Set value :	12	

Increment value (JOG+)

Increase value in register by a set amount in [Inc. value], each time when the button is pressed, up to the [Upper limit]. As shown below, each button press increases the value in the register by 1 until the value is 10.

Attribute					
Set Style :	Increment value (JOG+)				
Inc. value :	1	Upper limit :	10		

Decrement Value (JOG-)

Decrease value in register by a set amount in [Dec. value], each time when the button is pressed, down to the [Bottom limit]. As shown below, each button press decreases the value in the register by 1 until the value is 0.

Attribute			
Set Style : Decrement value (JOG-)			
Dec. value :	1	Bottom limit :	0

Press and hold increment (JOG++)

When the button is held longer than a set time in [JOG delay], it will increase the value in a register by a set amount :[Inc. value] at a set rate :[JOG speed], to the [Upper limit]. As shown below, when the button is pressed, it increases the value in the designated register by 1. When the button is held longer than 1 second, it increases the value in register by 1 every 0.5 second, till the value is 10.

Attribute			
Set Style :	Press and hold incremen	t (JOG++)	•
Inc. value :	1	Upper limit :	10
JOG delay :	1.0 second(s)	JOG speed :	0.5 second(s)

• Press and hold increment (JOG--)

When the button is held longer than a set time in [JOG delay], it will decrease the value in a register by a set amount: [Dec. value] at a set rate: [JOG speed], to the [Bottom limit]. As shown below, when the button is pressed, it decreases the value in the designated register by 1. When the button is held longer than 1 second, it decreases the value in register by 1 every 0.5 second, till the value is 0.



Attribute			
Set Style : Press and hold decrement (JOG)			
Dec. value :	1	Bottom limit :	0
JOG delay :	1.0 second(s)	JOG speed :	0.5 second(s)

Periodic JOG++

This automatic function increases the value in the register by a set amount: [Inc. value], at a set rate: [Time interval], to the [Upper limit]. As shown below, the system will automatically increase the value in the register by 1 every 0.5 second, till the value is 10. Then the value returns to 0 and add 1 every 0.5 second again.

Set Style :	Periodic JOG++ (up->0)->up->)	
Inc. value :	1	Upper limit :	10
Time interval :	0.5 second(s)]	

Automatic JOG++

This automatic function increases the value in the register by a set amount: [Inc. value], at a set rate: [Time interval], to the [Upper limit].then holds this value. As shown below, the system will automatically increase the value in the register by 1 every 0.5 second, till the value is 10, and then stop.

Attribute Set Style :	Automatic JOG++ (up	to high limit)	
Inc. value :	1	Upper limit :	10
Time interval :	0.5 second(s)		
nine interver.	U.5 SECOND(S) +		

Automatic JOG--

This automatic function decreases the value in the register by a set amount: [Dec. value], at a set rate: [Time interval], to the [Bottom limit].then holds this value. As shown below, the system will automatically increase the value in the register by 1 every 0.5 second, till the value is 10, and then stop.



Attribute			
Set Style :	Automatic JOG (dow	n to low limit)	•
Dec. value :	1	Bottom limit :	10
Time interval :	0.5 second(s)	·	

Periodic bounce

Increases the word address value to the [Upper limit] by a [Inc. value] at a set rate in [Time interval], then decreases to the [Bottom limit] by the same value at the same rate. As shown below, the system will increase the value in the designated register by 1 every 0.5 second, till the value is 10, and then decrease the value by 1 every 0.5 second till the value is 0 whenever the screen is active.

Attribute			
Set Style : (Periodic bounce (up->do	own->up->)	•
Bottom limit :	0	Upper limit :	10
Inc. value :	1		
Time interval : (0.5 second(s)		

Periodic step up

Step up to the [High limit] by [Inc. value] at a set rate in [Time interval], then reset immediately to the [Low limit]. The action repeats whenever the screen is active. As shown below, the system will increase the value in the designated register by 1 every 0.5 second, till the value is 10, and then reset to 0 and increase again, and the action repeats.

Attribute			
Set Style :	Periodic step up (low to	high)	•
Low limit :	0	High limit :	10
Inc. value :	1		
Time interval :	0.5 second(s)		

Periodic step down

Step down to the [Low limit] by [Dec. value] at a set rate in [Time interval], then reset immediately to the [High limit]. The action repeats whenever the screen is active. As shown below, the system will decrease the value in the designated register by 1 every 0.5 second, till the value is 0, and then reset to 10 and decrease again, and the action repeats.



Attribute				
Set Style :	Periodic step down (high to low)			
Low limit :	0	High limit :	10	
Dec. value :	1			
Time interval :	0.5 second(s)			

• Set when window opens / Set when window closes

Automatic function occurs whenever the screen is active. The value entered in [Set value] is set into the word address when the action occurs. If [Set value] is set to 5, when the window opens / closes, the system enters 5 into the designated register.

• Set when backlight on / Set when backlight off

Automatic function occurs whenever the backlight is active. The value entered in [Set value] is set into the word address when the action occurs. If [Set value] is set to 5, when the backlight turns ON / OFF, the system sets 5 into the designated register.

Cyclic JOG+

Each time when the button is pressed, increases the word address value to the [Upper limit] by [Inc. value] then reset to the [Bottom limit]. As shown below, each time when pressing the button, the system will increase the value in the designated register by 1, till the value is 10, and then reset to 0 and increase again by pressing the button.

Attribute			
Set Style :	Cyclic JOG+		•
Bottom limit :	0	Upper limit :	10
Inc. value :	1		

Cyclic JOG-

Each time when the button is pressed, decrease the word address value to the [Bottom limit] by [Dec. value] then reset to the [Upper limit]. As shown below, each time when pressing the button, the system will decrease the value in the designated register by 1, till the value is 0, and then reset to 10 and decrease again by pressing the button.

Attribute			
Set Style :	Cyclic JOG-		▼
Bottom limit :	0	Upper limit :	10
Dec. value :	1		

Cyclic JOG++

When the button is held longer than a set time in [JOG delay], it increases the value in a register by a set amount in [Inc. value] at a set rate in [JOG speed], to the [Upper limit], then reset to the [Bottom limit]. As shown below, when the button is held longer than 0.5 second, increase the value in the designated register by 1 every 0.1 second, till the value is 10, and then



reset to 0 and increase again by holding the button.

Attribute	
Set Style : Cyclic JO	S++ ▼
Bottom limit : 0	Upper limit : 10
Inc. value : 1	
JOG delay : 0.5 seco	nd(s) \checkmark JOG speed : 0.1 second(s) \checkmark

• Cyclic JOG- -

When the button is held longer than a set time in [JOG delay], decrease the value in a register by a set amount in [Dec. value] at a set rate in [JOG speed], to the [Bottom limit], then reset to the [Upper limit]. As shown below, when the button is held longer than 0.5 second, decrease the value in the designated register by 1 every 0.1 second, till the value is 0, and then reset to 10 and decrease again by holding the button.

Attribute		
Set Style :	Cyclic JOG	•
Bottom limit :	0	Upper limit : 10
Dec. value :	1	
JOG delay :	0.5 second(s)	JOG speed : 0.1 second(s)



13.5. Function Key

13.5.1. **Overview**

The Function Key object can be used for several tasks, such as switching between windows, keypad design, Macro execution, screen hardcopy, and setting USB security key.

13.5.2. Configuration



Click the Function Key icon on the toolbar to open a Function Key object property dialog box. Set up the properties, press OK button, and a new Function Key object will be created.

General Tab

cMT Series

	General Security Shape Label
	Comment :
New Function Key Object	Activate after button is released
General Security Shape Label	Change full-screen window Change common window
Comment :	Display popup window
Change full-screen window Change common window	
Display popup window	Window no. : 50. Keypad 11 - Integer 💌
Window no. : 10. WINDOW_010	Return to previous window Olose window
Return to previous window	ASCII/UNICODE mode
	○ [Enter] ○ [Backspace] ○ [Clear] ○ [Esc]
Execute macro	[Delete] (Left] (Right]
Screen hard copy	CASCII] / [UNICODE]
Screen nard copy	Execute macro
Admowledge all events(alarms) Notification	◎ Window title bar
Notification Enable	Hard copy screen to USB disk, SD card or printer
	Screen hard copy
	Notification
	Enable
OK Cancel He	OK Cancel Help

eMT, iE, XE, mTV Series

New Function Key Object



23



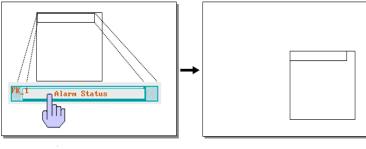
Setting	Description	
Activate	If this function is selected, the action is delayed till button is	
after button	released; otherwise, the action is executed once the button is	
is released	pressed.	
Change	Change full-screen window: Change to another base window.	
window	Change common window: Change common window.	
	Display popup window: A pop-up window displays in the base	
	window. If [Close this popup window when parent window is closed]	
	check box is selected, the pop up window will be closed when	
	change the base window to another window. Otherwise, a function	
	key in the pop up window is needed to close it.	
	Display popup window	
	☑ Close this popup window when parent window is closed	
	Style : With title bar	
	Window no. : 11. Window_011	
	Return to previous window: If this is selected, the Function Key will	
	change from the current screen to the previous one displayed. For	
	example, when window no. 10 is changed to window no. 20, press	
	the function key to return to window no. 10. This function is only	
	available for base window.	
	Close window: Close any active pop-up windows, message windows	
	included.	
ASCII/	Configures the button as a keypad key, and the character it enters,	
UNICODE	via [Numeric] or [ASCII] objects.	
mode	Enter: Same as the keyboard's "Enter" function.	
	Backspace: Same as the keyboard's "Backspace" function.	
	Clear : Clear the value in the word register.	
	Esc: Same as the [Close window] function; it is used to close the	
	keyboard window.	
	Delete: Same as the keyboard's "Delete" function, deletes the	
	number or character on the right side of the text cursor.	
	Left: Same as the keyboard's " \leftarrow " key moves the text cursor to the	
	left side of the previous number or character.	
	Right: Same as the keyboard's " $ ightarrow$ " key moves the text cursor to the	
	left side of the next number or character.	
	ASCII/UNICODE: Specify the character to be entered by this key.	
Execute	Select this check box to execute one of the Macros from the drop	
Macro	down list that has already been configured by users.	



bar

	For more information, see "18 Macro References".	
	Execute macro	Macro : [ID:000] macro_0
Window title	Function Key defined car	n be used to move a pop-up window which

has no [window title bar] to a preferred position on screen. Select the pop-up window and then click on a preferred position, the window will be moved.



Select the window title bar firstly.

Touching the screen for the new position the popup window will be moved.

Hard copy	Print the current window. Before using this function, choose a printer
screen to	model in [System Parameter Settings] » [Model] » [Printer].
USB disk, SD	If a monochrome printer is used, selecting [grayscale] can provide a
card or	better print result, but the text may not be clearly printed. To
printer	improve text printing, avoid using [grayscale].

Hard copy screen to USB disk, SD card o	or printer –		
Screen hard copy	Printer :	HP PCL Series (USB)	
Rotate image 90 degrees	Mode :	color 👻	J

Screen hard	
сору	Output current screen to iPad Photos.
(cMT Series)	
Acknowledge	
all events	A duranted as all events as as by pressing the Europtics Key
(alarms)	Acknowledge all events once by pressing the Function Key.
(cMT Series)	
Import user	
data / Use	A Function Key can be used to import the e-mail contacts or user
[USB	accounts set, also, to log in using USB Security Key.
Security Key]	



Function mode	
 Import e-mail s Import user ac Use [USB Secu 	
Data position	
○ SD card	OUSB disk
Account import mo	ode
Overwrite	Append App
🔲 Delete file afte	r importing user accounts
	OK Cancel

Data Position

Select the external device to store data from [SD card] or [USB disk].

Account import mode

If [Overwrite] is selected, the existing accounts will be overwritten with new accounts in the external device after importing. If [Append] is selected, HMI will append more accounts while the old accounts still exist.

Delete file after importing user accounts

If select this check box, the system will delete the account data saved in the external device after importing, this can prevent the account data from leaking out.

Notification	If this selection is enabled, it will notify a designated bit address to
	set ON or OFF, each time the button is pressed.

Note

[Overwrite] is the only option when importing the e-mail contacts. This means that all existing contacts will be removed first, and then the new contacts are added.

For more information, see "6 Window Operations", "12 Keypad Design and Usage", "36 Administrator Tools".



13.6. Toggle Switch

13.6.1. Overview

Toggle Switch object is a combination of Bit Lamp object and Set Bit object. The appearance of the object is controlled by the ON / OFF state of the read bit address. As well, pressing the button sets the value in the bit address according to the settings.

13.6.2. Configuration



Click the Toggle Switch icon on the toolbar to open a Toggle Switch object property dialog. Set up the properties, press OK button, and a new Toggle Switch object will be created.

General Tab

eneral	Security	Shape	Label				
	Comment	:[]	JN 50				
		🔘 Bit	Lamp		Toggle	Switch	
PLC	address name : Loc idress : LB	al HMI	ad/Write use	different an	ldresses	-	Settings
Write	address :						
		IN r	ite when hut	ton is releas	ed		
Attrib		E Wr	ite when but	ton is releas	æd		
				ton is releas	æd		
	ute Switch style			ton is releas	ed		
Macro	ute Switch style	: Set OI		ton is releas	ed		



Setting	Description			
Comment	User can describe the information of the object.			
	Bit Lamp / Toggle Switch Switch between Bit Lamp and Toggle Switch features.			
Read address	Click [Setting] to select the [PLC name], [Address], [Device type],			
	[System tag], [Index register] of the bit device that controls the			
	[Toggle Switch] o	bject. Users can also set address in [General] tab		
	while adding a n	ew object.		
	Invert signal			
	Reverses the display of ON / OFF states. For example, if [Invert			
	signal] check box is selected, when the designated bit is OFF, the			
	object displays ON state.			
Write address	Click [Setting] to	select the [PLC name], [Address], [Device type],		
	[System tag], [In	dex register] of the bit device that controls the		
	[Toggle Switch] object. Users can also set address in [General] tab			
	while adding a new object. The address can be the same or			
	different from [Read address].			
	Write after button is released			
	If this function is selected, the action is delayed till button is			
	released, otherwise, the action is executed once the button is			
	•	nction does not work with momentary buttons.		
Attribute	Set style Set ON	Description		
		Set ON the designated bit of the device.		
	Set OFF	Set OFF the designated bit of the device.		
	Toggle	Alternates the bit state each time pressed.		
	Momentary	Holds the bit ON only while button is		
		pressed.		
Macro	Toggle Switch ob	pject can trigger the start of a Macro routine wher		
	the Macro has b	een created in advance.		
	👉 For more in	formation, see "18 Macro References".		

13.7. **Multi-State Switch**

13.7.1. **Overview**

Multi-state Switch object is a combination of Word Lamp object and Set Word object. The appearance of the object is controlled by the value of the read word address. As well, pressing the button sets the value in the word address according to the settings.

13.7.2. Configuration



Click the Multi-State Switch icon on the toolbar to open a Multi-State Switch object property dialog box. Set up the properties, press OK button, and a new Multi-State Switch object will be created.

General Tab

New Ger

cMT Series

eral Security Shape Label	General Security Shape Label
Comment :	Comment :
Word Lamp Word Lamp	🕥 Word Lamp 💿 Multi-State Switch
Mode : Value Offset : 0 Read/Write use different addresses	Mode : Value Offset : 0 Read/Write use different addresses
ead address	Read address
PLC name : Local HMI	PLC name : Local HMI
Address : LW O 16-bit Unsigned Error handling	Address : LW O
	🕅 Write when button is released
ttribute	Write when button is released
ttribute Switch style : [JOG+ Vo. of states : 2 V	
ttribute Switch style : JOG+ No. of states : 2 Cyclical : Disable	Attribute
Switch style : JOG+ No. of states : 2	Attribute Switch style : JOG+ Vo. of states : 2 V
Switch style : JOG+ No. of states : 2 Cyclical : Disable	Attribute Switch style : [JOG+ No. of states : 2 Cyclical : [Disable
Switch style : JOG+ Vo. of states : 2 Cyclical : Disable User-defined mapping	Attribute Switch style : [JOG+ - No. of states : 2 Cyclical : [Disable User-defined mapping
Switch style : JOG+ Vo. of states : 2 Cyclical : Disable User-defined mapping end notification after writing successfully	Attribute Switch style : JOG+ No. of states : 2 Cyclical : Disable User-defined mapping Send notification after writing successfully

eMT, iE, XE, mTV Series



Setting	Description			
Comment	User can describe the information of the object. Word Lamp / Multi-State Switch			
	Switch between Word Lamp and Multi-State Switch features.			
Model /	Different modes can be selected: [Value], [LSB].			
Offset	👉 For more information, see "13.2 Word Lamp".			
Read address	Click [Setting] to select the [PLC name], [Address], [Device type],			
	[System tag], [Index register] of the word device that controls the			
	Multi-state Switch object. Users can also set address in [General]			
	tab while adding a new object.			
Write address	Click [Setting] to select the [PLC name], [Address], [Device type],			
	[System tag], [Index register] of the word device that controls the			
	Multi-state Switch object. Users can also set address in [General]			
	tab while adding a new object.			
	Write after button is released			
	If this function is selected, the action is delayed till button is			
	released; otherwise, the action is executed once the button is			
	pressed.			
Attribute	Switch style			
	Select the object's operation mode, see Example 1.			
	User-defined mapping			
	The value placed in the write register of each selection, the action			
	taken when an illegal value is entered, and error notification to a			
	designated bit address can be set.			
	Mapping			
	State Value 0 0			
	1 1 2 (error)			
	ок			
	Input illegal			
	Remain current state O Jump to error state Error notification			
	Image: PLC name : Local HMI Set ON Set Unit Setting			
	Address : LB v 0			

Remain current state

If an illegal value is entered, Multi-state Switch will remain at the



	current state.
	Jump to error state
	If an illegal value is entered, Multi-state Switch will jump to the
	error state.
	Error notification
	If an illegal value is entered, automatically set the value placed in
	the designated register.
Send	
notification	After the system successfully writes data to PLC, the designated bit
after writing	address will be set On/Off.
successfully	
Error handling	The action taken when an illegal value is entered or notify a
(cMT-SVR	designated bit address. This is similar to [User-defined mapping];
Series)	the difference is the value corresponding to each state need not to
-	be preset.

Example 1

JOG+

Increase the value of a designated register by 1 each time when pressing the button, till the value equals to [No. of states]. A cyclic action can be enabled. As shown below, each time when pressing the button, the state number will add 1 start from state 0, till state 4 ([no. of state]-1), and returns to 0 and step up again.

Attribute		
Switch style :	JOG+ 👻	No. of states : 5
Cyclical :	Enable	•
User-defined	d mapping	

JOG-

Decrease the value of the designated register by 1 each time when pressing the button, till the value equals to 0. A cyclic action can be enabled. As shown below, each time when pressing the button, the state number will minus 1 start from state 4 ([no. of state]-1), till state 0, and returns to state 4 and step down again.

Attribute Switch style : JOG-	•	No. of states : 5
Cyclical : Enable		•
User-defined mapping		



13.8. Slider

13.8.1. Overview

Slider object is used to change the value in a designated word register address by moving the slide on the screen.

13.8.2. Configuration



Click the Slider icon on the toolbar to open a Slider object property dialog box. Set up the properties, press OK button, and a new Slider object will be created.

General Tab

New Slider Object
General Outline Security Shape
Write address
PLC name : Local HMI Setting
Address : LW 🔹 0 16-bit Unsigned
Notification
🗹 Enable 💿 Set ON 💿 Set OFF
Before writing
PLC name : Local HMI
Address : LB 🗸 0
Enable
PLC name : Local HMI
Address : LW 🗸 0 16-bit Unsigned
OK Cancel Help

Setting	Description	
Write address	Click [Setting] to select the [PLC name], [Address], [Device type],	
	[System tag], [Index register] of the word device that controls the	



	Slider object. Users can also set address in [General] tab while
	adding a new object.
Notification	If enabled, the state of a designated bit address will be set to ON or
	OFF.
	Click [Setting] to select the [PLC name], [Address], [Device type],
	[System tag], [Index register] of the bit device that controls the
	notification settings. Users can also set address in [General] tab
	while adding a new object.
	[Before writing] / [After writing]
	Change the state of a designated bit register before, or after the
	slider is slid.
Watch	When moving the roller, the new value written to the word register
address	address can be displayed in real time.

Outline Tab

New Slider Object
General Outline Security Shape
Attribute
Direction : Right - Resolution : 1
Low/High limit : Constant Address
Low limit : 0 High limit : 100
Coarse increment Increment : 10
Slider button type
Shuel buildh type
Width: 20
Frame : Transparent
Background : 🔄 🔽 Transparent
Slot : Transparent
OK Cancel Help

Setting	Description
Attribute	Direction
	Select the direction of the slider. (Right, Up, Left, Down)



	right left up down
	Resolution
	Sets the value change in the word register for each step of the
	Slider. For example, if set to 10, the register value changes by 10
	points for each increment or decrement on the Slider.
	Constant
	Sets the range of the Slider. For example, If set [Low limit] to 5, and [High limit] to 100, the Slider will enter values between 5 and 100. Address
	Set the [Low/High limit] by a designated register, see Example 1.
	Coarse increment
	Apart from moving the roller to change the value as in [Resolution],
	if this option is selected, the word value will increase / decrease by the [Increment] value each time the object is touched.
Slider button	Use picture (cMT Series only)
type	If this checkbox is selected, a picture can be selected from the
type	If this checkbox is selected, a picture can be selected from the Picture Library to be the slider button.
type	
type	Picture Library to be the slider button.
type	Picture Library to be the slider button. Four default styles are offered here, and the width of the Slider
type	Picture Library to be the slider button. Four default styles are offered here, and the width of the Slider may be set.
type	Picture Library to be the slider button. Four default styles are offered here, and the width of the Slider may be set. Color



Objects

Example 1

Set the low or high limit by a designated register. When write address is LW-n, where n is an arbitrary number, the rule of setting limits is:

	Content	16-bit	32-bit
	Address	LW-n	LW-n
	Low limit	LW-n	LW-n
	High limit	LW-n+1	LW-n+2
When address is L	W-100, the rule of setting lin	nits is:	
	Content	16-bit	32-bit
	Address	LW-100	LW-100
	Low limit	LW-100	LW-100
	High limit	LW-101	LW-102
			6



13.9. Numeric

13.9.1. Overview

Numeric object can be used to input or display the value of a designated word register.

13.9.2. Configuration



Click the Numeric icon on the toolbar to open a Numeric object property dialog box. Set up the properties, press OK button, and a new Numeric object will be created.

General Tab

neral	Data Entry Numeric Format Security Shape Font
inci di	Data End y Humener ofmat Security Shape Fond
	Description :
	Allow input
	Read/Write use different addresses
Read a	address
PLC	name : Local HMI 🔹 Settings
Ad	ldress : LW 🗸 0
	address
	name : Local HMI Settings
Ad	dress : LW 🔹 1
Notific	ation
	✓ Enable ○ Set ON ③ Set OFF
	Before writing
PLC	name : Local HMI 🔹 Settings
Ad	dress : LB 🔹 0
Notific	ation on invalid input
	Tenable Set ON Set OFF
PLC	name : Local HMI 🔹 🗸 Settings
	dress : LB 🔹 1



Setting	Description
Allow input	If selected, the input features and relevant settings are enabled.
Read / Write use different address	Set [Read address] and [Write address] differently.
Read address	Click [Setting] to select the [PLC name], [Address], [Device type],
	[System tag], [Index register] of the word device that displays the
	value. Users can also select a tag defined in Address Tag Library
Write address	Select the [PLC name], [Device type], [Address] of the word device
	that system writes to.
Notification	If this check box is selected, it will notify a designated bit address
	(setting ON or OFF).
	Before writing / After writing
	Set the state of the designated bit address before or after the
	manual operation.
Notification	If an illegal value is entered, automatically set the state of a
on invalid input	designated register.

Data Entry Tab

cMT Series

eMT, iE, XE, mTV Series

New Numeric Object	New Numeric Object
General Data Entry Numeric Format Security Shape Font Mode : Touch	General Data Entry Numeric Format Security Shape Font Mode : Touch
Input order	□ Input order ② Enable
Stop sequential input function after input	Stop sequential input function after input
Input order : 1 👘 🗐 Group	Input order : 1 Group
	Image: Constraint of the set of the
OK Cancel Help	OK Cancel Help



Setting	Description
Mode	Touch
	Used when data entry is initiated by touching the screen object.
	Bit control
	Used when data entry is enabled by turning ON a designated bit,
	and entry ends when the bit goes OFF.
Allow input	Specify a bit address that enables or ends data entry. The order of
bit address	data entry is specified in [Input order] and an external USB
	keyboard is needed for data entry. For cMT-SVR, use iPad's
	keyboard.
Input order	Perform continuous input by setting [Input order] and [Group].
	The criterion of searching the next input object:
	The range of [Input order]: 1 ~ 511, range of [Group]: 1 ~ 15.
	 If [Group] is not selected, its input order is 0.
	 The system only searches for the objects within the same
	Group.
	 The lower number of order is entered before the higher
	number of order.
	 For multiple objects within the same group and with the same
	input order, the object placed in the lower layer is entered first.
Keyboard	Use a popup keypad
(For eMT, iE,	If selected: A pre-designed pop-up keypad can be chosen by
XE, mTV	selecting a check box, and selecting the relative position on the
Series)	HMI screen. When data entry is enabled, the pop-up keypad
	displays in the selected position, and closed when data entry ends.
	If not selected: When data entry is enabled, the pop-up keypad is
	not displayed. Users may:
	 Create a custom design on the same screen window.
	 Use a USB keyboard.
	Hide title bar
	Use a keypad without the title bar.
	Restart the keypad if input value is out of range
	When entering data, if the value entered is not within the valid
	range, the system will automatically restart the keypad.

Note

To enter data for cMT-SVR, the iPad's keyboard is used.



To create a keyboard in current window, see "12 Keypad Design and Usage".

Example 1

This example demonstrates how to use [Input Order] and [Group] to perform continuous input in several Numeric objects. After entering data in one object, entry will be passed to the next input order object which is in the same group.

 Create three Numeric objects, and set [Input order] to 1, 2, and 3 respectively. Include the three objects in [Group 1] as shown in the following figure. LW-0

_W-0

LW-1

LW-2

🔽 Enable				
Stop seque	ential input function	n after input		
Input orde	er: 1	Group	1	×
⊂ Input order				
🔽 Enable				
Stop seque	ential input functio	n after input		
Input orde	er: 2 🚔	Group	1	÷
Input order Enable Stop seque	ntial input function	after input		
Enable		after input	1	
Stop seque			1	
Enable			1	<u>↓</u>
Tinput orde		Group		
Enable Stop seque Input orde		Group		0

 When finish entering data in the last object, to end data entry of all objects, please select [Stop sequential input function after input] check box.



Input order Enable		
Stop sequential input function after	er input	
Input order : 3	📝 Group	1

Numeric Format Tab

cMT Series

eMT, iE, XE, mTV Series

New Numeric Object	New Numeric Object
General Data Entry Numeric Format Security Shape Font	General Data Entry Numeric Format Security Shape Font
Display	Display
Data format : 16-bit Unsigned 👻 🔲 Mask	Data format : 16-bit Unsigned 👻 🥅 Mask
Number of digits	Number of digits
Left of decimal Pt. : 4 🚔 Right of decimal Pt. : 0 🚔	Left of decimal Pt. : 4
Scaling	Display format
Method : Interpolation	🖉 Enable Truncated digits : 0 🔿
Test <= Preview conversion result	Format : **** Examples
Dynamic scales Engineering low : 0 Engineering high : 9999	Use star "*" to represent each digits. Number of "*" = [Left of decimal Pt.] - [Truncated digits] Scaling
	Method : None
Other options	
Display lower and upper limits	
Display previous value	
Limits	
O Direct O Dynamic limits	
PLC low : 0 PLC high : 9999	Limits
Input low : 0 Input high : 9999	Direct Opmamic limits
	PLC low : 0 PLC high : 9999
🔲 Use alarm color	Input low : 0 Input high : 9999
	Use alarm color
OK Cancel Help	OK Cancel Help

Setting	Description
	•
Display	Data Format
	Set the data format of a designated word register. The selections
	include: BCD, HEX, Binary, Unsigned, Signed, Float. 16-bit uses 1
	word where 32-bit uses two words.
	Mask
	If selected, any values entered will be hidden by displaying them as

Number of	Left of decimal Pt.
digits	The number of digits before the decimal point.
	Right of decimal Pt.
	The number of digits after the decimal point.

Display If [Enable] check box is selected, the [Allow input] feature in General tab will be displayed in the Numeric object. Apart from "*" signs, extra texts can be entered in the [Format] field, for example: kg. The numeric value represented by "*" sign will be displayed from the highest digit to the lowest in the Numeric object. Truncated digits Specify the number of digis to be truncated, from the lowest digit to the highest. The number of "*" signs = [Left of decimal Pt.]-[Truncated digits] Please see the following examples: Please see the following examples: Please see the following examples: Display format [left of decimal Pt.]-[Truncated digits] Please see the following examples: Please see the following examples: Display format [left of decimal Pt.] = 5, [Truncated digits] = 0 => Number of "** 5 stars are required. We on use ""sign"s", and the reautival like "Big2050" Emaple 2 : Change (Truncated digits] = 0 => Number of "** 5 stars are required. We on use ""sign"s", and the reautival like "Big2050" Emaple 2 : Change (Truncated digits] = 0 => Number of "** 5 stars are required. Scaling Interpolation If this check box is selected, [Engineering low] and [Engineering high] boxes appear. Values entered in these boxes correspond to the display range required. The setting also requires [Input low] and [Input high] in the limits section. See Example 2. Scaling Interpolation State or computed by and sergister. See Example 4.		
If this check box is selected, [Engineering low] and [Engineering high] boxes appear. Values entered in these boxes correspond to the display range required. The setting also requires [Input low] and [Input high] in the limits section. See Example 2. Test: Preview the result of Interpolation. See Example 2. Dynamic scales: Set the [Engineering low] and [Engineering high] by a designated register. See Example 4. Macro subroutine (Not available for cMT Series) The value read from or written to the register can be computed by macro subroutines selected in [Read conversion] and [Write		<pre>tab will be disabled. Each "*" sign represents each digit that will be displayed in the Numeric object. Apart from "*" signs, extra texts can be entered in the [Format] field, for example: kg. The numeric value represented by "*" sign will be displayed from the highest digit to the lowest in the Numeric object. Truncated digits Specify the number of digis to be truncated, from the lowest digit to the highest. The number of digis to be truncated, from the lowest digit to the highest. The number of "*" signs = [Left of decimal Pt.]-[Truncated digits] Please see the following examples: DisplayFormat Example 1 Read data = 31255 (5 digits, unit : grams) [Left of decimal Pt.] = 5, [Truncated digits] = 0 => Number of "*" = 5 stars are required. We can use "*kg***g", and the result will be "31kg255g" Example 2: Change [Truncated digits] = 2 => Number of "*" = 3 stars are required and last two digits will be truncated. We can use "*kg*00g", and the result will be "31kg200g" Example 3 Read data = 255 (5 digits, unit : grams) [Left of decimal Pt.] = 5, [Truncated digits] = 0 => Number of "*" = 5 stars are required. We can use "*kg*00g", and the result will be "31kg200g" Example 3 Read data = 255 (5 digits, unit : grams) [Left of decimal Pt.] = 5, [Truncated digits] = 0 => Number of "*" = 5 stars are required. We can use "*kg*1**g", and the result will be "00kg255g" Nute : The text alignent only uses [Leading zero] mode when enables [Display format] function.</pre>
 high] boxes appear. Values entered in these boxes correspond to the display range required. The setting also requires [Input low] and [Input high] in the limits section. See Example 2. Test: Preview the result of Interpolation. See Example 2. Dynamic scales: Set the [Engineering low] and [Engineering high] by a designated register. See Example 4. Macro subroutine (Not available for cMT Series) The value read from or written to the register can be computed by macro subroutines selected in [Read conversion] and [Write 	Scaling	Interpolation
		 high] boxes appear. Values entered in these boxes correspond to the display range required. The setting also requires [Input low] and [Input high] in the limits section. See Example 2. Test: Preview the result of Interpolation. See Example 2. Dynamic scales: Set the [Engineering low] and [Engineering high] by a designated register. See Example 4. Macro subroutine (Not available for cMT Series) The value read from or written to the register can be computed by macro subroutines selected in [Read conversion] and [Write



_

Objects

Function Library. To use this feature, see "13.9.2.1 The rule of using		
Macro subroutine".		
Display lower and upper limits		
is displayed near the object.		
Display previous valueRange: 0 to 9999If selected, when entering a value, the valueCurrent value: 5		
before update is displayed near the object.		
This section allows users to apply display limits to the values held in		
the input register. The color when the register value is outside limits		
can be set.		
Direct		
Sets the limits by entering values in [Input low] and [Input high]. If		
the value entered is outside the limits, the value in the register		
cannot be changed.		
Dynamic limits		
Set the limits by a designated register, see Example 5.		
Low limit		
When the value in the register is outside the [Low limit], display		
digits by the color set.		
High limit		
When the value in the register is outside the [High limit], display		
digits by the color set.		
Blink		
When the value in the register is outside either limit, the digits flash.		

13.9.2.1. The rule of using Macro Subroutine

• There must be a return value and exactly one parameter.

Examples:

```
sub char test (short a) // (Correct)
```

```
sub test (char a) // (Incorrect, no return value.)
```

sub char test (char a, char b) // (Incorrect, two parameters.)

• Use the Macro data type that corresponds to the object's data format.

The mapping is as follows:

Macro Data Type	Numeric Object Data Format
short	16-bit Signed
Int	32-bit Signed



Objects

unsigned short	16-bit BCD, 16-bit HEX, 16-bit Binary, 16-bit Unsigned				
unsigned int	32-bit BCD, 32-bit HEX, 32-bit Binary, 32-bit Unsigned				
float	32-bit Float				

For example, if the data format of the numeric object is 16-bit Unsigned, only the corresponding Macro data type: unsigned short, is available.

```
Examples:
```

```
sub char test(unsigned short a) // (Correct)
sub char test(char a) // (Incorrect)
```

```
    Supports only the local HMI address.
    Examples:
    GetData(var, "Local HMI", LB, 0, 1) // (Correct)
    GetData(var, "MODBUS RTU", 0x, 0, 1) // (Incorrect)
```

 The following system defined functions are unable to be invoked: ASYNC_TRIG_MACRO, SYNC_TRIG_MACRO, DELAY, FindDataSamplingDate, FindDataSamplingIndex, FindEventLogDate, FindEventLogIndex, INPORT, INPORT2, OUTPORT, PURGE, TRACE

```
• The following statements are not supported:
For-Next, While-Wend
```

Example 2

If [Interpolation] is selected, the scaling equation is as the following:

If A indicates the original data and B indicates the displayed data:

B = [Engineering low] + (A - [PLC low]) × ratio

```
where, ratio = ([Engineering high] - [Engineering low]) / ([PLC high] - [PLC low])
As shown below, the original data is 15, after conversion, 40 will be displayed.
```



Scaling								
Method : Interpolation	•							
Test <= Preview conversion result								
Dynamic scales	;							
Engineering low : 10	Engineering high : 50							
Limits								
 Direct Dynamic limits 								
PLC low : 0	PLC high : 20							
Input low : 10	Input high : 50							

Click [Test] button to preview the result of Interpolation. Enter a value in [PLC] field as shown in the following figure, for example, enter value 15, and the result, which is 40, will be displayed.

Left of d	lecimal Pt. : 4	×	Right of decimal	Pt. : 0	×	
HMI 40	Engineering low = 10 + (PLC 15	PLC low) × —		Engineering low 10 0 PLC low
40 HMI		•	15 PLC		FLC night	PLC IOW

Example 3

If the numeric format selected is not Float and decimal point is used, the decimal place of the converted result will not be adjusted automatically, please adjust [Engineering high] to correctly place the decimal point of the result gained in [Interpolation] mode. Please see the illustration below.

1. Create two Numeric objects, set [Right of decimal Pt.] to 1 and select [Interpolation] method for one of the objects as shown in the following figure.



New Numeric Object	X
General Data Entry Numeric Format Security Shape Font	
Display Data format : 16-bit Unsigned	
Left of decimal Pt. : 4 Right of decimal Pt. : 1	* *
Scaling Method : Interpolation	
Dynamic scales	
Engineering low : 0 Engineering high : 200	
Limits	
Direct Opynamic limits	
PLC low : 0 PLC high : 100	
Input low : 0 Input high : 200	

2. Enter value "123", the object set to [Interpolation] displays "246.0" instead of "24.6".

Original Value	123
Converted Value	246.0

3. To move the decimal point one place to the left, adjust [Engineering high] as shown in the following figure.

lew Numeric Ob	ject			X
General Data E	ntry Numeric Forma	at Security	Shape Font	
Display				
	at: 16-bit Unsigned	d → [Mask 📃 Mask	
Number of dig Left of de	ts cimal Pt. : 4	F	Right of decimal Pt. : 1	×
Scaling				
	Method : Interpola	ition	•	
Dyr	namic scales			
Engine	ering low : 0		Engineering high : 20)
		Orig	inal Value	123
		Conve	rted Value	24.6

Example 4

If [Interpolation] is selected, set the [Engineering low] and [Engineering high] by a designated register. When Dynamic Address is LW-n, where n is an arbitrary number, the rule of setting [Engineering low] and [Engineering high] is:



	Content	16-bit	32-bit
	Dynamic address	LW-n	LW-n
	Engineering low	LW-n	LW-n
	Engineering high	LW-n+1	LW-n+2
When address is L	W-100, the rule of setting	limits is:	
	Content	16-bit	32-bit
	Dynamic address	LW-100	LW-100
	Engineering low	LW-100	LW-100
	Engineering high	LW-101	LW-102

Example 5

Set the limits by a designated register. When [Address] is LW-n, where n is an arbitrary number, the rule of setting limits is:

	Content	16-bit	32-bit
	Address	LW-n	LW-n
	Low limit	LW-n	LW-n
	High limit	LW-n+1	LW-n+2
When address is L	W-100, the rule of set	ting limits is:	
	Content	16-bit	32-bit
	Address	LW-100	LW-100
	Low limit	LW-100	LW-100
	High limit	LW-101	LW-102

Example 6

The following demonstrates how to use [Macro subroutine] for scaling when configuring Numeric object.

The following two macros are used, one for [Read conversion] and one for [Write conversion].

Read Conversion	
sub short read(unsigned short a)	
short b b = a + 10 return b	
end sub	

Write	Conversion

sub short write(unsigned short a)
short b
b = a - 10
return b
end sub

 Create two Numeric objects: NE_0 and NE_1 and use the same control address. Select [Macro subroutine] for NE_1.





Numeric Object

Write conversion : write

umeric (Object's Prop	perties					23
General	Data Entry	Numeric Format	Security	Shape	Font	Profile	
Display							 -
Da	ata format :	16-bit Unsigned	• [Mask			
	er of digits eft of decimal	Pt. : 4	* *				
Scaling	,						51
	Met	hod : Macro subro	utine	•			
	Read convers	sion : read		•			-

Enter 0 in NE_0 then NE_1 will execute [Read conversion]. The value gained will be 10. 2.

NE_0 (Without Macro conversion)	NE_1 (With Macro conversion)
Read Conversion	Write Conversion
sub short read(unsigned short a)	sub short write(unsigned short a)
short b b = a + 10 return b	short b b = a - 10 return b
end sub	end sub

Enter 80 in NE_1, [Write conversion] is executed and the value gained will be 70. NE_0 3. displays 70.

NE_0 (Without Macro convers 70	NE_1 sion) (With Macro conversion) 80
Read Conversion	Write Conversion
sub short read(unsigned short a	a) sub short write(unsigned short a
short b b = a + 10 return b	short b b = a - 10 return b
end sub	end sub

Note

If executing [Read conversion] and [Write conversion] by the same numeric object, the value entered in this object is computed by the Macro subroutine of [Write conversion] first, and then the result is computed by the Macro subroutine of [Read conversion]. In



13-42



Example 5, if the subroutine of [Write conversion] is set to b=a-20, then entering 80 in NE_1 will get 60 after [Write conversion] and then the object displays 70 after [Read conversion].

Font Tab

-	Data Catav	Numeric Format	Convibu	Change	Font	
lerai	Data Entry	Numeric Pormac	Security	snape		
\ttribu						
	Font :					•
	Color :				Size : 16	• • •
	Align :	Left	•			
onten						
		OK		Cancel	L	Help

Setting	Description
Color	When the value is within the limits, display digits using color set in
	this tab.
Align	Left: Align the number to the left.
	Center: Align the number to the center.
	Right: Align the number to the right.
	Leading zero: The number is preceded with leading zeros when the
	number of digits is less than that set.
	Left 66
	Center 66
	Right 66
	Leading zero 0066
Size	Set the font size.



13.10. ASCII

13.10.1. Overview

ASCII object can be used to input or display ASCII or UNICODE characters held in designated word registers.

13.10.2. Configuration



Click the ASCII icon on the toolbar to open an ASCII object property dialog box. Set up the properties, press OK button, and a new ASCII object will be created.

General Tab

	I Objec								L
ieneral	Data E	intry	Security	Shape	For	it			
	Descripti	on :							
	V Allo	ow inp	ut						
	Ма	ek		🔲 Use L		DE			
			high/low b		, acco				
			nign/iow c	Jyte					
	address name :		LINT				_	Settin	
	ldress :	_	HMI		•	0	 •	Setun	iys
		LVV			•	•			
Notific	ation								
-Notific	ation		Enable Before	writing		Set ON	Se	t OFF	
		8	✓ Before	writing		Set ON			
PLC	name :	Local	✓ Before	writing		After w	● Se	t OFF	lgs
PLC		Local	✓ Before	writing	•				ıgs
PLC	name :	Local	✓ Before	writing	•	After w			ıgs
PLC	name :	Local	✓ Before	writing	•	After w			igs
PLC	name :	Local	✓ Before	writing	•	After w			igs
PLC	name :	Local	✓ Before	writing	•	After w			igs
PLC	name :	Local	✓ Before	writing	×	After w			igs
PLC	name :	Local	✓ Before	writing	•	After w			igs



Setting	Description
Allow input	If selected, the input features and relevant settings are enabled.
Mask	If selected, any values entered will be hidden by displaying them as ****.
Use UNICODE	Select this check box to display data in UNICODE format. If not selected, the characters are displayed in ASCII format. This feature can be used with the [Function Key] object that uses [ASCII/UNICODE].
Reverse high/low byte	Normally an ASCII code is displayed in "high byte", "low byte" order. Reverse selection makes the system display ASCII characters in "low byte", "high byte" order. ABCD BADC The left object is in normal form, and
Read address	another is high/low byte reversed. Click [Setting] to select the [PLC name], [Address], [Device type],
	[System tag], [Index register] of the word device that displays characters. Users can select a defined address tag from Address Tag Library, or set address in [General] tab while adding a new object.
	Address PLC name : Local HMI Device type : LW Address : 0 Address : 0 Address format : DDDDD [range : 0 ~ 10799] Index register No. of word : 1
	Tag Library OK Cancel
	No. of words Select the maximum number of words to be displayed.

Note

An UNICODE character uses 1 word, and an ASCII character uses 1 byte. Therefore 1 word can be used as 1 UNICODE character or 2 ASCII characters. (1 word equals to 2 bytes)



Objects

Font Tab

New ASCII Object	8
General Data Entry Security Shape Font	
Attribute Font : Arial	•
Content AA	
OK Cancel Help	

Setting	Description
Attribute	The font, size, color, and alignment can be set.
	Align
	Left: Align the text to the left.
	Center: Align the text to the center.
	Right: Align the text to the right.



13.11. Indirect Window

13.11.1. Overview

Indirect Window object opens the pop-up window assigned by a designated word register. There are two ways to use Indirect Window object: The first is to use the profile of Indirect Window object, and let the pop-up window be resized and displayed in the defined profile; the second is to automatically resize the window according to the size of the pop-up window to be displayed. To close the pop-up window, assign 0 to the designated word register. The difference between Direct Window and Indirect Window is that Direct Window is controlled by a bit register, while Indirect Window is controlled by a word register.

13.11.2. Configuration



Click the Indirect Window icon on the toolbar to open the object property dialog box. Set up the properties, press OK button, and a new Indirect Window object will be created.

General Tab

cMT Series

New Indirect Window Object	New Indirect Window Object
General	General
Comment :	Description :
Read address PLC name : Local HMI Address : LW IG-bit Unsigned	Read address PLC name : Local HMI Address : LW 0 16-bit Unsigned
	Attribute Style : No title bar
✓ Use window no. offset Offset : -10 ✓ Auto. adjust window size Alignment : ④ ● {relative to object display region} ●	✓ Use window no. offset Offset: -10 ▼ ✓ Auto. adjust window size Alignment :
OK Cancel Help	OK Cancel Help

eMT, iE, XE, mTV Series



Setting	Description
Read address	Click [Setting] to select the [PLC name], [Address], [Device type],
	[System tag], [Index register] of the word device that controls the
	pop-up window. Users can also set address in [General] tab while
	adding a new object.
Attribute	Style
	Set the display style of the pop-up window. There are two styles:
	 No title bar
	The pop-up window has no title bar and cannot be dragged.
	WINDOW 11
	• With title bar
	The pop-up window has a title bar that can be dragged to move the
	window.
	WINDOW 11
Use window	Sets the offset of the window number for selecting the pop-up
no. offset	window. The window number of the pop-up window is calculated
	by the value in the word register added to the offset. For example,
	assume the value in the register is 20 and offset is 5, the pop-up
	window number will be 25.
Auto. adjust	Automatically resize the Indirect Window and align the pop-up
window size	window to the preset region.
	Alignment
	Sets a reference point of the pop-up window from one of the five
	positions on the screen. For example, if the lower-right region is
	selected, the lower-right corner of the pop-up window is aligned to

Example 1

Here is an example of using Indirect Window. The setting is shown in the following figure, set the address to LW-0 which assigns the window number. Create window no. 11 and 12 first.

1. Create an Indirect Window object, set address to LW-0, and select [Auto. adjust window size].



2. Select the region where the window is to be displayed.

Indirect Window Object's Properties
General Profile
Comment :
Read address PLC name : Local HMI ✓ Address : LW ✓ 16-bit Unsigned
Attribute Style : With title bar
Use window no. offset Alignment : Alignment : (relative to object display region) (oregion) (
OK Cancel Help

- **3.** Enter value 11 in LW-0, the pop-up window displayed is window no. 11.
- 4. Enter value 12 in LW-0, the pop-up window displayed is window no. 12.
- 5. Enter value 0 in LW-0, the pop-up window is closed.

Popup Window No. 11	Popup Window No. 12
<- The frame of Indirect Window	<- The frame of Indirect Window
Window11	Window12

To close the pop-up window, apart from entering 0 in the designated word register, another way is to place a Function Key object in the pop-up window, and set the key to [Close window].

Note

- At most 24 windows can be displayed simultaneously at run time.
- The system does not allow opening the same window with two Direct (or Indirect) windows in one base window.
- If the pop up window has monopoly property enabled, then when the window pops up, all background windows may not be operated until the monopolizing window has been closed.



13.12. Direct Window

13.12.1. Overview

Direct Window object defines the position and size of a pop-up window location on a window. When the content of the bit register is changed, the window will pop up at the predefined location. The display area for the pop-up window is limited by the size of predefined location. Restoring the value of the bit register closes the pop-up window. The difference between the Direct Window and the Indirect Window is that Direct Window object has a predefined window number, and is controlled by a bit register, while Indirect window is controlled by a word register, and the value of the word register determines the window displayed.

13.12.2. Configuration



Click the Direct Window icon on the toolbar to open a Direct Window object property dialog box. Set up the properties, press OK button, and a new Direct Window object will be created.

General Tab

cMT Series	eMT, iE, XE, mTV Series
New Direct Window Object	New Direct Window Object
General	General
Comment : Trigger: ON	Comment : Trigger: ON
Read address	Read address
PLC name : Local HMI	PLC name : Local HMI
Address : LB 0	Address : B 0
Attribute Window No. : 11. Window_011	Attribute Style : No title bar Window No. : 11. Window_011
OK Cancel Help	OK Cancel Help

oMT if XF mTV Sories



Setting	Description
Read address	Click [Setting] to select the [PLC name], [Device type], [Address],
	[System tag], [Index register] of the bit device that control the
	window popup. Users can select a defined address tag from
	Address Tag Library, or set the address in [General] tab while
	adding a new object.
Attribute	Style
	Define the pop-up window style. Two styles are available, [No title
	bar] and [With title bar].
	Window no.
	Set the pop-up window number.

Example 1

Here is an example to explain how to use the Direct Window object. The following figure shows the settings of the Direct Window object. In the example, use LB-10 to call up window no. 35.

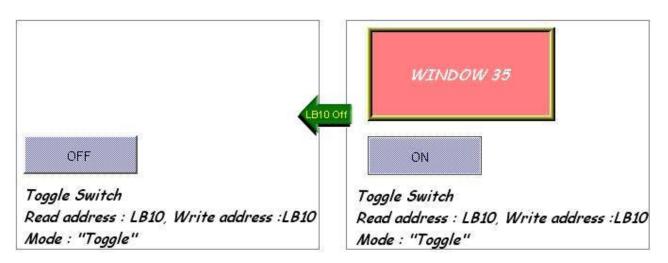
WC_0	
	PLC name : Local HMI Setting
	Address : LB 🗾 10
TS_0	Attribute Style : No title bar
Toggle Switch	Window No. : 35. Window_035
Read address : LB10, Write address :LB10	
Mode : "Toggle"	

If the state of LB-10 turns to ON, window no. 35 will pop up; if the state of LB-10 turns to OFF, window no. 35 will be closed, as shown in the following figure.

	WINDOW 35
OFF	ON
Toggle Switch	Toggle Switch
Read address : LB10, Write address :LB10	Read address : LB10, Write address :LB10
Mode : "Toggle"	Mode : "Toggle"







Note

- A screen can simultaneously display up to 24 pop-up windows including System Message Window, Direct Window and Indirect Window.
- The system does not allow opening the same window with two Direct (or Indirect) Windows in one base window.
- If the pop up window has monopoly property enabled, then when the window pops up, all background windows may not be operated until the monopolizing window has been closed.



13.13. Moving Shape

13.13.1. Overview

Moving Shape object defines the states and moving distance of an object. The state and the location of the object depend on consecutive registers.

13.13.2. Configuration



Click the Moving Shape icon on the toolbar to create a Moving Shape object. Set up the properties, press OK button, and a new Moving Shape object will be created.

General Tab

General Shape L				
PLC name	Local HMI			•
Read address				
PLC name : Loo	al HMI		-	Setting
Address : LW	•	100		16-bit Unsigned
Attribute				
Mode	X axis only			•
No. of states	1 -			
Display ratio State	0		Ratio: 1	
	C O ▼		Ratio : 1	
State	☑ Limit from register		Ratio : 1	Setting
State Limit address	✓ Limit from register tal HMI	0	Ratio : 1	Setting 16-bit Unsigned

Setting	Description
Read address	Click [Setting] to configure the [PLC name], [Device type],
	[Address], [System tag], or [Index register] of the word devices that
	control the display of object's state and moving distance. Users can



Objects

	also set the	e address in [G	eneral] tab while	adding a new object.
Attribute	Select the object's movement mode and range. See "13.13.2.1			
	Illustration	Illustration of Modes" in the following part.		
Display ratio	The size of	The size of shape in different states can be set individually as		
	shown in tł	shown in the following figure.		
	Ratio : 1	Ratio : 1.2	Ratio : 1.4	Ratio : 1.6
	State 0	State 1	State 2	State 3
Limit address	The object	's moving rang	e can be set by a	djusting the data in the
	designated	register, see E	xample 1.	

Example 1

Supposed that the object's moving range is limited by register LW-n, the addresses in the following table are used to limit the moving range.

Data format	16-bit	32-bit
[Min. X] address	LW-n	LW-n
[Max. X] address	LW-n+1	LW-n+2
[Min. Y] address	LW-n+2	LW-n+4
[Mas. Y] address	LW-n+3	LW-n+6

13.13.2.1. Illustration of Modes

Available modes are: (Assume Read Address is LW-n)

• X axis only

The object is only allowed to move along the X-axis. The moving distance ranges from [Min. X] to [Max. X].

Attribute			
Mode :	X axis only		•
No. of states :	8 🔹		
Min. X :	0	Max. X : 600	
Data format		16-bit	32-bit
Data format Object state		16-bit LW-n	32-bit LW-n

• Y axis only

The object is only allowed to move along the Y-axis. The moving distance ranges from [Min. Y] to [Max. Y].



Objects

Attribute	r		
Mode :	Y axis only		
No. of states :	8 🔹		
Min. Y :	0	Max. Y: 480	
Data format	t	16-bit	32-bit
Data format Object state		16-bit LW-n	32-bit LW-n

• X & Y axis

The object is allowed to move along the X-axis and Y-axis. The moving range in X and Y directions is defined by [Min. X], [Max. X] and [Min. Y], [Max. Y] respectively.

Mode :	X & Y axis		
No. of states :	8 -		
Min. X :	0	Max. X :	600
Min. Y:	0	Max. Y:	480

Data format	16-bit	32-bit
Object state	LW-n	LW-n
Moving distance on X-axis	LW-n+1	LW-n+2
Moving distance on Y-axis	LW-n+2	LW-n+4

For example, if the object's read address is LW-100 and the data format is [16-bit Unsigned], LW-100 is used to control the object's state, LW-101 is used to control the object's moving distance on the X-axis, and LW-102 is used to control the object's moving distance on the Y-axis. The following figure shows that the object's read address is LW-100 and initial position is (100, 50). To move the object to the position (160,180) and change its state to State 2, assign 2 to LW-100, 160-100 = 60 to LW-101, 180-50 = 130 to [LW102].

(100, 50)	L W100	LW101	LW102
State 0	0	0	0
	State	X Offset	Y Offset
(160, 180) next	position		
(100, 50)	LW100	LW101	LW102
(100, 50)	2	<i>LW101</i> 60	<i>LW102</i> 130
			130



• X axis w/ scaling

The object moves in X-axis only with scaling. Suppose that the value of the designated register is DATA, the system uses the following equation to calculate the moving distance on the X-axis.

Displacement (Data [Input low])	[Scaling high]-[Scaling low]
Displacement=(Data-[Input low])×	[Input high]-[Input low]

Data format	16-bit	32-bit
Object state	LW-n	LW-n
Moving distance on X-axis	LW-n+1	LW-n+2

• Y axis w/ scaling

The object is for Y axis movement with scale, and the equation to calculate the moving distance on the Y-axis is the same as the one in [X axis w/ scaling].

Data format	16-bit	32-bit
Object state	LW-n	LW-n
Moving distance on Y-axis	LW-n+1	LW-n+2

• X axis w/ reverse scaling

This works in the way as [X axis w/ scaling], but the moving direction is in reverse.

• Y axis w/ reverse scaling

This works in the way as [Y axis w/ scaling], but the moving direction is in reverse.



13.14. Animation

13.14.1. Overview

Animation object is defined by a pre-defined point set and states. Animation object will then move to a given point in a given state defined by designated registers. The object state and position depend on current value of two consecutive registers. The first register controls the state of the object and the second register controls the position along the predefined path.

13.14.2. Configuration



Click the Animation icon on the toolbar. First, create the pre-defined path. Move the mouse to each moving position, and click the left button to define positions one by one. When it is done, right click on the screen, set up the properties, press OK button, and a new Animation object will be created.



To change the object's attributes, double click on the object to open Animation Object's Properties dialog box.



General Tab

	Animation Object's Properties
	General Shape Label Profile
	Comment :
	Attribute No. of states : 8
	Position : O Controlled by register O Based upon time interval
	Read address
	PLC name : Local HMI
	Address : LW 0 16-bit Unsigned
	OK Cancel Help
tting	Description
_	Description
_	No. of states
_	No. of states Configure the number of states for this object.
_	No. of states Configure the number of states for this object. Controlled by register
_	No. of states Configure the number of states for this object.
_	No. of states Configure the number of states for this object. Controlled by register
_	No. of states Configure the number of states for this object. Controlled by register Use the designated registers to control the object's state and
_	No. of states Configure the number of states for this object. Controlled by register Use the designated registers to control the object's state and position. See Example 1.
_	 No. of states Configure the number of states for this object. Controlled by register Use the designated registers to control the object's state and position. See Example 1. Based upon time interval The object's state and position will change from time to time. [Time
_	 No. of states Configure the number of states for this object. Controlled by register Use the designated registers to control the object's state and position. See Example 1. Based upon time interval The object's state and position will change from time to time. [Time interval attributes] is used to set the time interval for states and
_	 No. of states Configure the number of states for this object. Controlled by register Use the designated registers to control the object's state and position. See Example 1. Based upon time interval The object's state and position will change from time to time. [Time interval attributes] is used to set the time interval for states and positions.
etting ttribute	 No. of states Configure the number of states for this object. Controlled by register Use the designated registers to control the object's state and position. See Example 1. Based upon time interval The object's state and position will change from time to time. [Time interval attributes] is used to set the time interval for states and positions.
_	 No. of states Configure the number of states for this object. Controlled by register Use the designated registers to control the object's state and position. See Example 1. Based upon time interval The object's state and position will change from time to time. [Time interval attributes] is used to set the time interval for states and positions.

Supposed that [Speed] is set to 10, the object's position will change each second.



Image state change: Determines how state changes, either
[Position dependent] or [Time-based]. If [Position dependent] is
selected, the object state will change when position changes. If
[Time-based] is selected, the object position will change based on
[Position speed] and the object state will change based on [Image update time].
Backward cycle: Assumed the object has four positions: position 0,

position 1, position 2, and position 3, and [Backward cycle] is not selected. When the object moves to the last position (position 3), the next position will be back to the initial position 0, and repeat. The moving path is shown as follows:

position $0 \rightarrow \text{position } 1 \rightarrow \text{position } 2 \rightarrow \text{position } 3 \rightarrow \text{position } 0 \rightarrow \text{position } 1 \rightarrow \text{position } 2...$

If [Backward cycle] is selected, when the object moves to the last position (position 3), it will move backwards to position 2, position 1 and then the initial position 0, and start over again. The moving path is shown as follows.

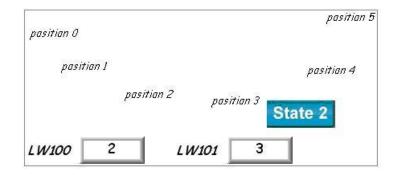
position $0 \rightarrow \text{position } 1 \rightarrow \text{position } 2 \rightarrow \text{position } 3 \rightarrow \text{position } 2 \rightarrow \text{position } 1 \rightarrow \text{position } 0...$

Example 1

The object's state and position are determined by the registers, and the addresses must be configured correctly, as in the following table:

Data format	16-bit	32-bit
Object state	LW-n	LW-n
Object position	LW-n+1	LW-n+2

For example, if the designated register is LW-100 and the data format is [16-bit Unsigned], then LW-100 represents object's state, LW-101 represents position. In the picture below, LW-100 = 2, LW-101 = 3, so the object's state is 2 and position is 3.





Profile Tab

General Shape Label	Profile				
Position Pinned	х:	191	Y:	56	*
Size	ight ratio				
	Width :	404	Height :	131	*
	Width (%) :	100 ≑	Height (%) :	100	* *
Shape rectangle size					
	Width :	84 🚖	Height :	33	* *
Trajectory		Position 0			•
	х:	191	Y:	147	•

Setting	Description
Shape rectangle size	Set the size of the shape.
Trajectory	Set the position of each point on the moving path.



Since multiple pictures might be used by an [Animation] object, [Set to original dimension] will not return all pictures to the original size.



13.15. Bar Graph

13.15.1. Overview

Bar Graph object displays data as a bar graph for visualization.

13.15.2. Configuration



Click Bar Graph icon on the toolbar to open Bar Graph dialog box. Select properties, click OK button, a new Bar Graph object is created.

General Tab

New Bar Graph	-				
General Outlin	e Security Sh	ape			
Comm	ent :				
Read addres	s				
PLC name :	Local HMI			•	Setting
Address :	<u></u>	•	0		16-bit Unsigned

Setting	Description
Read address	Click [Setting] to Select the [PLC name], [Device type], [Address],
	[System tag], and [Index register] of the word devices that controls
	how the bar graph displays.



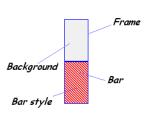
Outline Tab

Attribute			
	Normal -	Direction :	Up 👻
Zero :	0	Span :	10
	В	ar width ratio (%) :	100 🗘
Bar color/style	Transparent		
	M Haisparen		
Bar :	•	Bar style :	
Target indicator	_		
	Enable	Color :	
Target value :	LW:0+2	Tolerance :	0
Alarm indicators			
Low limit :	LW:0	High limit :	LW:0+1
Low color :	•	High color :	
Dynamic target/al	arm/zero(span)		
	Enable		
PLC name : Loc	al HMI		▼ Setting
Address : LW	•	0	16-bit Unsigned
	Dynamic zero/span		

Description Setting Attribute Туре Choose either [Normal] or [Offset]. If [Offset] is selected, an original value [Origin] must be entered for reference. Direction Determine the bar graph direction. Available options are [Up], [Down], [Right], and [Left]. Zero / Span The percentage of filling can be calculated by the formula, see Example 1. Bar width ratio (%) It is the ratio of bar to object width. The figure below shows two ratios, 100% and 50%. 100% 50%



Set the bar's frame and background color, bar style, and bar color. Bar color/style See the picture below.



Target	When the register value meets the condition, the color of filled
indicator	area will change to the target color, see Example 2.
Alarm	If the register value is larger than [High limit], the color of filled
indicators	area will change to [High color]. If the register value is smaller than
	[Low limit], the color will change to [Low color].
Dynamic	When [Enable] is selected, the [Low limit] and [High limit] of [Alarm
taget/alarm	indicator] and the [Target Value] of [Target indicator] will use
/zero(span)	designated registers, which is shown in their respective fields see
	Example 3.

Example 1

The percentage of filling can be calculated by the following formula:

Percentage of filling =
$$\frac{\text{Register value} - [\text{Zero}]}{[\text{Span}] - [\text{Zero}]} \times 100\%$$

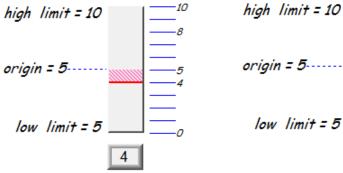
Assume [Offset] is selected. If (Register value - [Zero]) is greater than 0, the bar will fill up from [Origin]. If (Register value – Zero) is less than 0, the bar will be drawn below [Origin].

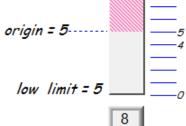
For example, [Origin] is 5, [Span] is 10, and [Zero] is 0.

For different value in read address, it will display as below:

If the value at read address is 4:

If the value at read address is 8:







Example 2

When the register value meets the following condition, the color of filled area will change to the target color.

[Target Value] - [Tolerance] ≤ Register value ≤ [Target Value] + [Tolerance]

Assume [Target Value] is 5 and [Tolerance] is 1. As shown below, if the register value is equal to or larger than 4 (=5-1) and equal to or less than 6 (=5+1), the filled area's color of the bar will change to the target color.



Example 3

If [Dynamic target/alarm] is enabled, [Low limit] and [High limit] of [Alarm indicator] are defined by designated registers as shown in the following table. Furthermore, if [Dynamic zero/span] is used, [Zero], [Span] and [Origin] will be defined by designated registers. Assume the address is LW-n, the limits are:

Data format	16-bit	32-bit
Alarm Low Limit	LW-n	LW-n
Alarm High Limit	LW-n+1	LW-n+2
Target	LW-n+2	LW-n+4
Zero	LW-n+3	LW-n+6
Span	LW-n+4	LW-n+8
Origin	LW-n+5	LW-n+10



13.16. Meter Display

13.16.1. Overview

Meter Display object displays the value of word register with a meter.

13.16.2. Configuration



Click the Meter Display icon on the toolbar to open the Meter Display dialog box. Set the object's attributes and then click OK to create a new Meter Display object.

13.16.2.1. eMT, iE, XE, mTV Series

General Tab

[New Meter Display Object	×
	General Outline Limits Security Shape	
	Comment :	
	Read address PLC name : Local HMI Address : LW 0 16-bit Unsig	
Setting	Description	
Read address	Click [Setting] to select the [PLC name], [D	evice t
	[System tag], and [Index register] of the we	ord dev

the Meter Display object.

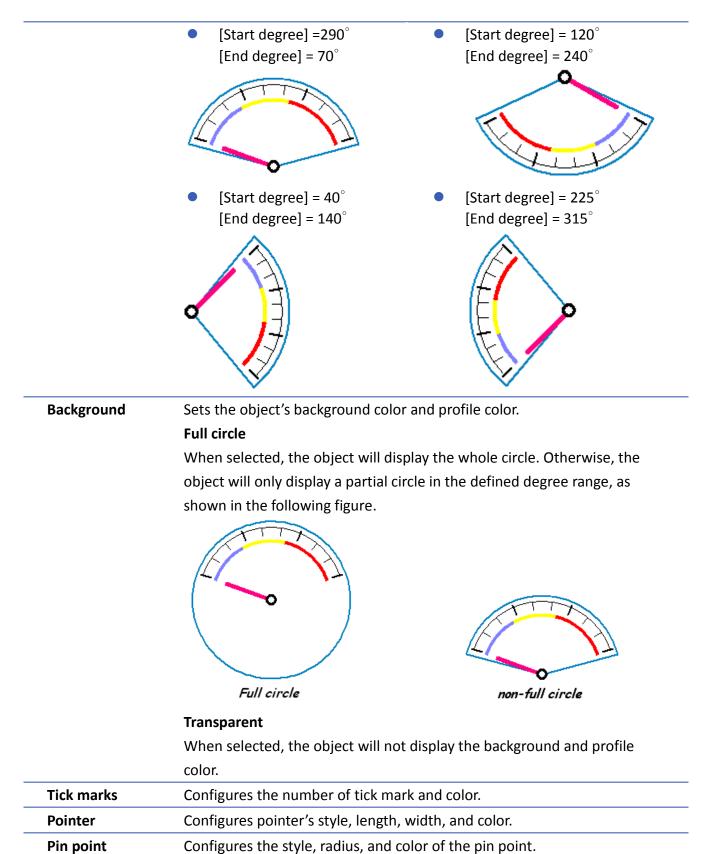


Outline Tab

New Meter Display O	bject			×
General Outline L	imits Security Shape			
Degree				
Start degree :	0	End degree : 3	360	×
Background				
Background :		Profile :		•
	V Full circle	Transparent		
Tick marks				
	C Enable		_	
			Coordinate	
Main scale :		Sub. scale : 2	2	* *
Length :	10			
Pointer				
	Arm style	Frame : [
		Inner :		• •
	Width : 4 🗸	Length : (50	
Pin point Radius :	7			
Inner :		Eramo I		
		Traile .		
	le 🔘 Rectangle			
	ОК	Cancel		Help
ange mark		Tick	mark	
		\sim		Corrdina
14				
		1 11		
4		11		
$\left \right $				
H	•			
r f (•			
÷{{	8			
÷	0	Pin poi	'nt	

Setting	Description
Degree	Sets the object's start degree and end degree measured clockwise
	from the 12 o'clock position. The angle range is 0 to 360 degrees.
	The following shows meters of different settings.





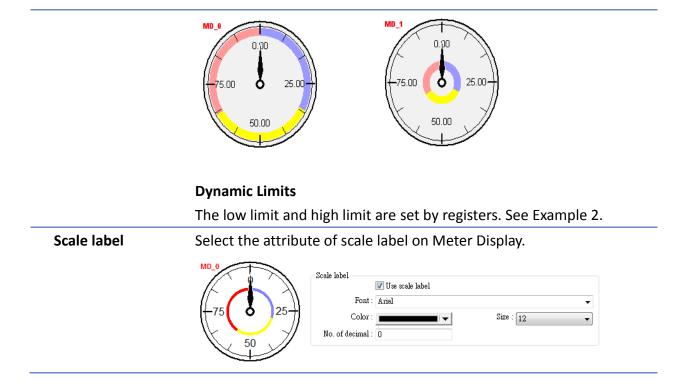


Limits Tab

General C	Outline Limits	Security Sha	аре			
Value						
	Zero : 0		-	Span :	100	×
-Range lin	nits					
	V E	nable				
	Low :		Mid :	_ ▼	High :	
	Width: 3		×			
	🔳 U	se user-defined	radius			
	V D	ynamic limits				
PLC na	me : Local HMI	-			-	Setting
	ess : LW	•	→ 0			it Unsigned
Scale lab	el					
	V U	se scale label				
	Font : Aria	l				•
	Color :		-	Size :	16	•
			Right of decima	al point :	0	

Setting	Description
Value	Sets the object's display range. Meter Display object will use the
	value of [Zero] and [Span] and the value of register to calculate the
	pointer's position. See Example 1.
Range limits	Configures the values of [Low limit], [High limit], their
	corresponding display colors, and the width.
	30 ⁶⁰ 0 100
	Use user-defined radius
	Configures the radius to display range limits.
	For example, set to 80: Set to 30:





Example 1: Pointer position calculation

Set object's display range. Meter Display object will use the value of [Zero] and [Span] and the value of register to calculate the pointer's position. For example, supposed that [Zero] is 0, [Span] is 100, when the value of register is 30, [Start degree] is 0, and [End degree] is 360, then the degree indicated by the pointer is:

{ (30 - [Zero]) / ([Span] - [Zero]) } * ([End degree] - [Start degree]) =

 $\{(30-0) / (100-0)\} * (360-0) = 108$

Pointer will be pointing at 108 degrees.

Example 2: Dynamic Limits

The low limit and high limit are set by the register.

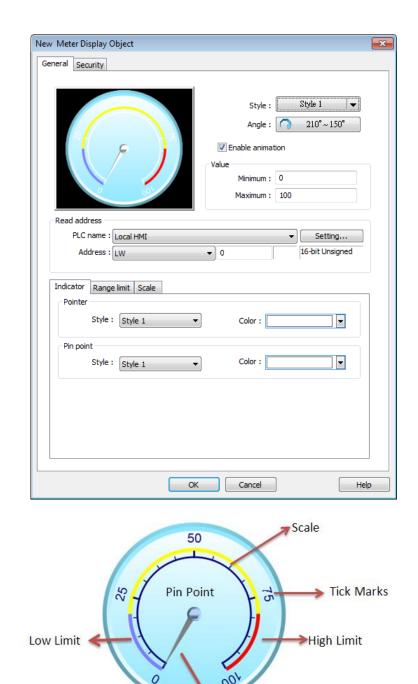
Suppose the address is LW-n, the following table shows the read address of low limit and high limit:

	Content	16-bit	32-bit	
	Low limit	LW-n	LW-n	
	High limit	LW-n+1	LW-n+2	
For instance, when	n address is LW-100, the rule of setting limits is:			
	Content	16-bit	32-bit	
	Low limit	LW-100	LW-100	
	High limit	LW-101	LW-102	



13.16.2.2. **cMT** Series

General Tab



Setting	Description
Style	Three options are available: [Custom], [Style 1], and [Style 2]. Click on the text to configure the properties. If [Custom] is selected, set
	the properties such as [Pointer], [Pin point], [Background picture], etc.

00

Indicator



Setting

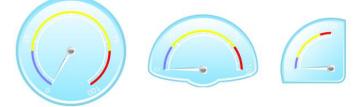
13-71

Click the button of [Style 1] or [Style 2] to set the style of meter.

Settings		—
Background Outlin	e : Quarter circle 🔹	Color :
Rotatio	n : 0 degree 🔻	
		OK Cancel

Outline

The following is the outline of Style 1 when [Full circle], [Half circle], or [Quarter circle] is selected.



The following is the outline of Style 2 when [Full circle], [Half circle], or [Quarter circle] is selected.



	Rotation
	Rotates the background picture clockwise according to the angles
	set.
	Color
	Sets the color of the background picture of meter.
Angle	Sets the range to label the scale, 0° to 360 $^\circ$ clockwise or
	counterclockwise from the twelve o'clock position.
	Full circle
	If selected, the full circle is drawn according to the selected
	direction and the start angle. The limits are determined by the
	value set in [Minimum] and [Maximum] field under [Value].
Enable	If selected, the pointer slides to the designated position when the
animation	read value changes; if not selected, the pointer directly points to
	the designated position when the read value changes.
Value	Sets the lower and upper limits of the meter.
Read address	Displays the value in meter according to the value in the designated
	word register.
Indicator	Sets the style of pointer and pin point. If [Custom] is selected, the
	direction of the pointer must points upward to correctly display.



Range limit Sets the colors to indicate different ranges.	
	Dynamic limits
	The low limit and high limit are decided by the register. See
	Example 2 above.
Scale	Sets the number of main and sub scale, the color of tick marks and
	scale label.



13.17. Trend Display

13.17.1. Overview

Trend display objects draw curves of the data recorded by Data Sampling object.

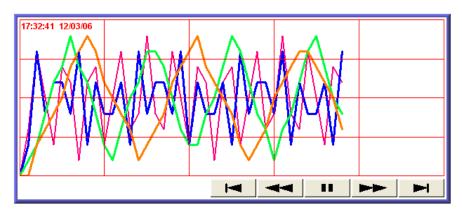
13.17.2. Configuration



Click the Trend Display icon on the toolbar to open a Trend Display object property dialog box. Set up the properties, press OK button, and a new Trend Display object will be created.

13.17.2.1. eMT, iE, XE, mTV Series

General Tab



Button	Description
	Go to the earliest sampling data.
	Go to the previous time interval.
	Click to stop auto-scrolling. When the new sampling data is
	generated, the display does not scroll, nor is the new data outside
	the display range displayed.
	Click to start auto-scrolling. The display scrolls as the new sampling
	data is generated.
	Go to the next time interval.
	Go to the latest sampling data.





TT	rend Display Object's Properties				
	General Trend Channel Shape Profile				
	Comment :				
	Data Sampling Object index : 3.				
	Trend type : History Note : if no. of channels is changed, you must reset HMI's data samplings !!				
	X axis time range : Pixel Time Default distance : 60 second (s)				
	🕼 Dynamic X axis time range				
	PLC name : Local HMI Setting				
	Address : LW 🔹 10				
	History control				
	PLC name : Local HMI				
	Address : LW				
	Watch line				
	PLC name : Local HMI				
	Address : LW v 20				
	Time stamp output				
	PLC name : Local HMI Setting				
	Address: LW V 30				
Sotting	Description				
Setting	Description				
Data					
Sampling	Select a Data Sampling object as the source data.				
Object index					
Trend type	Select the mode of data source, either [Real-time] or [History].				
	Real-time				
	In this mode, the display object shows all sampled data since the				
	HMI started. The maximum number of records that can be sampled				
	is set in [Max.data records] (Real-time mode) of the Data Sampling				
	object. When the sampling data exceed this setting, the earlier				
	data will be deleted.To show older data, use [History] mode.				
	[Hold control]: Suspends the update of Trend Display. However, It				
	does not stop the sampling process of Data Sampling object.				
	History				
	In this mode, the data comes from the history data files stored on				
	HMI The history data files are sorted by dates and each is given				
	an index. The system uses [History control] to select the history				
	data files that are created on different dates.				

The system sorts the history data of sampling data by date; the



latest file is record 0 (typically the data sampled today), the second latest file is record 1, and so on. If the value of designated register in [History control] is n, the Trend Display object will display data record n.

Here is an example to explain [History control]. If the designated register is LW-0, and the sampling data files available are pressure 20061120.dtl, pressure 20061123.dtl,

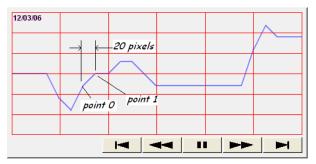
pressure 20061127.dtl, and pressure 20061203.dtl, and it is 2006/12/3 today, based on the value of LW-0, the sampling data file which will be selected by [Trend Display] is shown as follows:

Value of LW-0	Selected sampling history data
0	pressure_20061203.dtl
1	pressure_20061127.dtl
2	pressure_20061123.dtl
3	pressure_20061120.dtl

If use with Option List object and select data source as [Dates of historical data], the history data will be sorted by date and displayed in Option List object, see "13.29 Option List".

Pixel

[Distance] is used to set the distance between two sampling points, as shown in the following figure.



Time

[Distance] is used to set the X-axis in unit of time, as shown in the following figure.

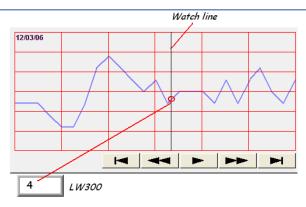


Select [Time] for [X axis time range] and go to [Trend] » [Grid] and



	enable [Time scale]. Please refer to [Time scale] in the later section.
Dynamic distance between data samples/ Dynamic X axis time range	Designate a 32-bit word register for adjusting the distance between two sampling points (select Pixel), or the time unit represented by X-axis (select Time). If no value is entered, the default value will be used.
Refresh data automatically	 If selected, every time when opening the window on which the Trend Display object in history mode is placed, the display is automatically refreshed once per second. Please note that: The refresh status can be observed from the control button of Trend Display object. Showing is button: The automatic refresh feature is enabled. Showing is button: The automatic refresh feature is disabled. When scrolling to the previous data, the automatic refresh feature is disabled, the is button is shown. If [Refresh data automatically] check box is selected, when change back to the window, the display is refreshed, ignoring the control buttons. For example, select [Refresh data automatically], and scroll to the previous data, the automatic refresh feature is disabled. In this case, changing to another window and then change back will still refresh the display. If [Refresh data automatically] check box is not selected when building the project, the feature can still be enabled by pressing is button on HMI. In this case, the automatic refresh feature is disabled, that is, even when change back to the current window, the display will not be refreshed.
Hold control	When the register is set ON, suspend the update of Trend Display. It does not stop the sampling process of Data Sampling object. This setting is available only in Real-time mode.
Watch line	Use the [Watch line] function to display a "watch line" when user touches the Trend Display object. It will also export the sampling data at the position of watch line to the designated word device and use Numeric objects to display the results, as shown in the following figure.





[Watch line] can also export sampling data with multiple channels. The system will consecutively write each channel to the specified address and the following addresses, in the same order as in [Data Sampling] object. The address assigned to [Watch line] is the start address, and sampling data for each channel will be exported to the word devices starting from "start address." If the data format of each channel is different, the corresponding address of each channel is arranged from the first to the last. If the watch register is LW-300, watch function will export each channel's data to the following addresses:

	Register	Channel	Data format
	LW-300	0	16-bit Unsigned (1 word)
	LW-301	1	32-bit Unsigned (2 words)
	LW-303	2	32-bit float (2 words)
	LW-305	3	16-bit Signed (1 word)
me s	stamp outpu	ıt	

Time stamp	Time stamp output
output	Suppose the address is set to LW-n, then:
	If enabled, the system will use the time of the first sampling data as
	"time origin", and write the time stamp of the most recent sampled
	data (relative to "time origin") to [LW-n+2].
	When clicking on the curve, the time stamp of the closest sampled
	point will be written to [LW-n].
	Clear real-time data address (Data Sampling object) will clear the
	time origin as well.
	Time stamp is recorded in seconds.

Note

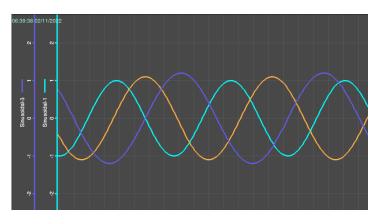
LW-n and LW-n+2 are both in 32-bit format.

- LW-n is for both real-time and history mode, whereas LW-n+2 only applies to real-time mode.
- This function is available upon enabling [relative time mode] in [Trend] tab.

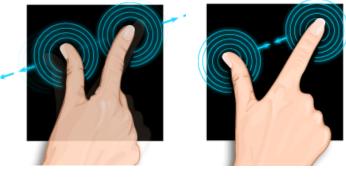


13.17.2.2. cMT Series

General Tab



The Trend Display on cMT Series combines Real-time mode and History mode. Drag left to scroll the Trend Display to view history data and drag right to view the latest sampling data. Pinch two fingers together to zoom out Trend Display or spread them apart to zoom in.



Zoom In Zoom Out For more information about how sampling data is saved, see "8 Data Sampling".





Comment : Data Sampling Object index : 1. Note : if no. of channels is changed, you must reset HMI's data samplings !! X axis time range : Millimete Image Time Distance : 60 second(s) Watch line PLC name : Local HMI Address : LW V 0	Data Sampling Object index : 1. Note : if no. of channels is changed, you must reset HMI's data samplings !! X axis time range : Millimete Image Time Distance : 60 second(s) Watch line The Distance : Local HMI The Setting	•
Note : if no. of channels is changed, you must reset HMI's data samplings !! X axis time range : Millimete Time Distance : 60 second(s) Watch line Image: Im	Note : if no. of channels is changed, you must reset HMI's data samplings !! X axis time range : Millimete Distance : 60 Watch line Image: Enable PLC name : Local HMI	•
Note : if no. of channels is changed, you must reset HMI's data samplings !! X axis time range : Millimete Time Distance : 60 second(s)	Note : if no. of channels is changed, you must reset HMI's data samplings !! X axis time range : Millimete Time Distance : 60 second(s)	
Distance : 60 second(s) Watch line Vatch line PLC name : Local HMI Setting	Distance : 60 second(s) Watch line U Enable PLC name : Local HMI Setting	
Distance : 60 second(s) Watch line Vatch line PLC name : Local HMI Setting	Distance : 60 second(s) Watch line U Enable PLC name : Local HMI Setting	
Watch line V Enable PLC name : Local HMI	Watch line V Enable PLC name : Local HMI	
Enable PLC name : Local HMI Setting	Enable PLC name : Local HMI Setting	
PLC name : Local HMI Setting	PLC name : Local HMI Setting	
Address . Lw V		
		Help

Setting	Description
Data	
Sampling	Select a [Data Sampling] object as the source data.
Object index	
Millimeter	See 13.1.2.1.
Time	See 13.1.2.1.
Watch line	See 13.1.2.1.





Objects

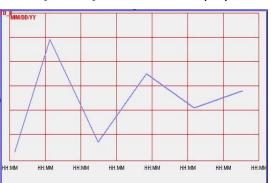
Trend Tab

	Trend Display Object's Properties
	General Trend Channel Shape Profile
	Transparent Frame : ■ Background : ■ ▼ Background : ■ ▼
	Grid Vertical Color: X-axis interval: X-axis interval: A the second(s) Time scale Vertical
	Ime HH:MM:SS HH:MM Image: SSSSS SSSSS (Leading zero) Image: Date MM/DD/YY DD/MM/YY DD.MM.YY Color : Image: Date Image: Date
	OK Cancel Help
Setting	Description
Frame / Background	Select the color of frame and background.
Show scroll	Enable or disable the scroll control as shown in the following figure.
controls	
Grid	Set the number of dividing lines and the line color. The number of divisions depends on the setting in General tab.
	X-axis interval
	The number of vertical grid lines.
	 Select [Pixel] / [Millimeter] in General tab:
	Select how many sampling point will be included between two
	vertical grid lines.
	 Select [Time] in General tab:
	Select the time range between two vertical grid lines.
	Y-axis interval
	The number of horizontal grid lines.



Time scale

Select [Enable] check box to display the time scale along the x axis.



Format

The available time scale formats: HH:MM or HH:MM:SS.

Font / Color / Size

Select the font, font color, and font size of the time scale. The default font size is 8.

Time / DateThe time of latest sampling data will be marked on the top left
corner of the object. This group box is used to set the time display
format and font color.

Channel Tab

lew Trend Display Object
General Trend Channel Shape
Data sampling object
Channel Display Description Data type Y scale
▶ 0 📝 16-bit Unsigned 16-bit Unsigned None
Channel Pen property Color : Width : 2
Dynamic limits Zero: 1 Span: 100
Channel visibility control
🕼 Enable
PLC name : Local HMI
Address : LW 🗸 0 16-bit Unsigned
Display channel when the corresponding bit is :
OK Cancel Help



13-81

Setting	Description							
Y scale	Set Y-axis to be Main A	kis or Aux. Axis. See	"Y Scale Tab" for more					
	information.							
	Channel Display Descr.							
	1 True channe 2 True channe		t Unsigned Aux. Axis t Unsigned Main Axis					
	3 True channel C 16-bit Unsigned Aux. Axis							
Channel	Configure each samplin	Configure each sampling line's format and color. At most 64						
	channels could be configured.							
	Dynamic limits							
	Not selected:	 Not selected: 						
	[Zero] and [Span] are used to set the low limit and high limit of							
	sampling data. If the low limit is 50 and the high limit is 100 for one							
	sampling line, [Zero] and [Span] must be set as [50] and [100], so							
	that all the sampling data can be displayed in the trend display							
	object.							
	Selected							
	The low limit and the high limit are read from the designated word							
	devices, as shown below. When address is LW-n , the register's							
	address:							
	Data Format	16-bit						
	Data i Offiat	10-010	32-bit					
	Low limit	LW-n	32-bit LW-n					
	Low limit	LW-n LW-n+1	LW-n LW-n+2					
	Low limit High Limit	LW-n LW-n+1	LW-n LW-n+2					
	Low limit High Limit For example, if LW-100	LW-n LW-n+1	LW-n LW-n+2					
	Low limit High Limit For example, if LW-100 will be read from:	LW-n LW-n+1 is used here, the lo	LW-n LW-n+2 w limit and the high lir					
	Low limit High Limit For example, if LW-100 will be read from: Data Format	LW-n LW-n+1 is used here, the lo 16-bit	LW-n LW-n+2 w limit and the high lin 32-bit					
	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102					
	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit High Limit	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101 s to zoom in and zoo	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102 om out of Trend Displa					
Channel	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit High Limit A typical usage of this is	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101 s to zoom in and zoo Series). See Example	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102 om out of Trend Displa					
	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit High Limit A typical usage of this is (Not available for cMT S	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101 s to zoom in and zoo Series). See Example he bits of the assign	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102 om out of Trend Displa e 1.					
visibility	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit High Limit A typical usage of this is (Not available for cMT S If [Enable] is selected, t	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101 s to zoom in and zoo Series). See Example he bits of the assign n channel. The first	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102 om out of Trend Displa e 1. ned word register will b bit controls the first					
Channel visibility control	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit High Limit A typical usage of this is (Not available for cMT S If [Enable] is selected, t used to show/hide each	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101 s to zoom in and zoo Series). See Example he bits of the assign n channel. The first d bit controls the se	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102 om out of Trend Displa e 1. ned word register will k bit controls the first econd channel, and so					
visibility	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit High Limit A typical usage of this is (Not available for cMT S If [Enable] is selected, t used to show/hide each channel, and the secon	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101 s to zoom in and zoo Series). See Example he bits of the assign channel. The first d bit controls the se se there are 5 chan	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102 om out of Trend Displa e 1. hed word register will b bit controls the first econd channel, and so nels and LW-0 is used,					
visibility	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit High Limit A typical usage of this is (Not available for cMT S If [Enable] is selected, t used to show/hide each channel, and the secon on. For example, suppor	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101 s to zoom in and zoo Series). See Example he bits of the assign channel. The first d bit controls the se se there are 5 chan	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102 om out of Trend Displa e 1. hed word register will b bit controls the first econd channel, and so nels and LW-0 is used,					
visibility	Low limit High Limit For example, if LW-100 will be read from: Data Format Low limit High Limit A typical usage of this is (Not available for cMT s (Not available for cMT s If [Enable] is selected, t used to show/hide each channel, and the secon on. For example, suppor channels which will be are:	LW-n LW-n+1 is used here, the lo 16-bit LW-100 LW-101 s to zoom in and zoo Series). See Example he bits of the assign n channel. The first d bit controls the se se there are 5 chan shown given the sta	LW-n LW-n+2 w limit and the high lin 32-bit LW-100 LW-102 om out of Trend Displa e 1. hed word register will b bit controls the first econd channel, and so nels and LW-0 is used,					



2	LW_bit-001	ON	NO
3	LW_bit-002	ON	NO
4	LW_bit-003	OFF	YES
5	LW_bit-004	OFF	YES

Note on using this feature: Each control bits are not reserved for the channel. If a particular channel is not displayed, the control bit is assigned to the next displayed channel. For example, if the third channel of the 5 channels is not displayed, only 4 channels will be displayed in Trend Display, and the used control bits will only be: LW_bit-000~003.

Example 1

The example explains how to zoom in or zoom out Trend Display. The feature described is not available for cMT Series.

In Channel tab select [Dynamic limits] check box. If the [Address] is set to LW-n, then LW-n controls the low limit where LW-n+1 controls the high limit.

Dynamic limits

PLC name :	Local HMI	-	Setting
Address :	LW 🔻	0	16-bit Unsigned

Set [Address] to LW-0 and create two Numeric objects for entering the low / high limit. The address that controls the low limit is LW-0; the address that controls the high limit is LW-1. Let's suppose the data is between 0 and 30; set the [Low limit] to 0 and the [High limit] to 30, the trend curve is displayed as shown in the following figure.



To zoom out the Trend Display, enter a value greater than 30 in [High limit] as shown in the following figure.





EasyBuilder Pro V5.02.01



Y Scale Tab

eMT, iE, XE, mTV Series

Jeneral	Trend	Channel	Y Scale	Security	Shape	Profile	
Data sa	unpling o	bject		<u> 2</u>			
C	hannel	Display	Description	a	Da	ata type	Y scale
1			channel A			-bit Unsigned	Aux. Axis
2		True	channel B		16	-bit Unsigned	Main Axis
3		True	channel C		16	-bit Unsigned	Aux. Axis
	Fo	nt : Arial		-			
	nic Y-sca	or : le visibility V Ens	ble			Size : 12	•
	nic Y-sca	le visibility	blej			Size : 12	▼ Settings
PLC	nic Y-sca	e visibility Ens Local HMI	ble	■ ▼ ▼ 50			• Settings
PLC	nic Y-sca name : [ddress : [Local HMI	when the co)		▼ Settings
PLC Ac Disple Dynan	nic Y-sca name : [ddress : [ay chann nic main	e visibility Ens Local HMI LW el's Y-scale ON axis V Ens	when the co	orrespondi)		•
PLC Ac Disple Dynan	nic Y-sca name : [ddress : [ay chann nic main	le visibility Ens Local HMI LW el's Y-scale ON axis	when the co	orrespondi)		

Setting	Description
Y scale	Set Y-axis to be Main Axis or Aux. Axis, or set none to hide Y-axis.
Scale font	Select the font, font color, and font size of the scale.
Dynamic	To show or hide Y-scale. If the control address is LW-50, then the
Y-scale	first axis is controlled by LW_Bit 5000, and the second axis is
visibility	controlled by LW_Bit 5001, and so on.
Dynamic	To change the main axis. If writing 1 into LW-80, the main axis will
main axis	be Channel 1; if writing 2 into LW-80, the main axis will be Channel,
	and so on.





13-85

cMT Series

The scale along the Y axis of a specific channel can be displayed. To enable Y Scale, [Grid] should first be enabled in [Trend] tab. Y Scale can be configured on the iPad as shown in the following steps.

1. Tap the

button on the upper right corner of Trend Display object.

2. Tap [Trend Display Setting] » [Y Scale].



3. Select the channels.

Y Scale	ON
🚖 Channel 1	ON
☆ Channel 2	OFF
☆ Channel 3	OFF
☆ Channel 4	OFF
☆ Channel 5	OFF



13.18. History Data Display

13.18.1. Overview

History Data Display object displays data stored by Data Sampling object. It differs from Trend Display in that History Data Display object uses a table to display data. In case when the trend display shows history data from today, the display will refresh once per second. The following is an example of a history data display object.

No.	Time	Date	Ch.0	Ch.1	Ch.2
3577	21:52	16/09/07	0	0	0
3576	21:52	16/09/07	0	0	0
3575	21:52	16/09/07	0	0	0
3574	21:52	16/09/07	0	0	0
3573	21:52	16/09/07	0	0	0
3572	21:52	16/09/07	0	0	0
3571	21:52	16/09/07	0	0	0
3570	21:52	16/09/07	0	0	0
3569		16/09/07	0	0	0
3568	21.22	16/00/07	0	0	

13.18.2. Configuration



Click the History Data Display icon on the toolbar to open a History Data Display object property dialog box. Set up the properties, press OK button, and a new History Data Display object will be created.



General Tab

lew History Data Display Object	×
General Data Format Title Shape	
Data Sampling Object index : 1.	•
Grid Color: Column interval: 5	÷
Profile color	
Frame : Background :	•
Text Font : Arial Size : 12	•
Time	•
Date DD/MM/YY Color :	•
Sequence no. Color :	•
 Time ascending Time descending 	
History control	
PLC name : Local HMI Address : LW 200	
OK Cancel Hel	lp

ampling bject indexSelect a Data Sampling object as the source data.Shows grids between rows and columns.ColorChange the color of grids.Column intervalChange the width of each column. The figures below are the examples.No <th>Setting</th> <th>Description</th>	Setting	Description						
Shows grids between rows and columns. Color Change the color of grids. Column interval Change the width of each column. The figures below are the examples. No. Time Date \frown 3667 21:57 16/09/07 1 0 0 3665 21:57 16/09/07 1 0 0 3665 21:57 16/09/07 1 0 0 3665 21:57 16/09/07 3665 21:57 16/09/07 1 0 0 3665 21:57 16/09/07 3665 21:57 16/09/07 3665 21:57 16/09/07 3662 21:57 16/09/07 3662 21:57 16/09/07 3662 21:57 16/09/07 3662 21:57 16/09/07 3661 21:57 16/09/07 3661 21:57 16/09/07 3661 21:57 16/09/07 3661 21:57 16/09/07 3661 21:57 16/09/07 3661 21:57 16/09/07 3661 21:57 3661 3661 21:57 16/09/07 3661 3661 21:57 16/09/07 3661 </th <th>Data Sampling object index</th> <th colspan="7">Select a Data Sampling object as the source data.</th>	Data Sampling object index	Select a Data Sampling object as the source data.						
Change the color of grids. Column interval Change the width of each column. The figures below are the examples. No. Time Date Chi	Grid	Shows grids between rows and columns.						
$\begin{array}{c} \hline \textbf{Column interval} \\ \hline \textbf{Change the width of each column. The figures below are the examples.} \\ \hline \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Ch.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Ch.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Oh.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Oh.2 \textbf{A}} \\ \hline \textbf{No.} \underline{\textbf{Time} Date Ch.0 Oh.1 Oh.2 Oh.2 $								
Change the width of each column. The figures below are the examples. $\underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.0 Ch.1 Ch.2}}{3667 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3666 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3666 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3666 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Time } \text{ Date } \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 \text{ 21:57 16/09/07 1 0}} \underbrace{\frac{No. \text{ Ch.09/07 1 0 0}}{3664 21:57 16/09/07 1 $		Change the color of grids.						
$\underbrace{\frac{No. \text{ Time } \text{ Date } (\text{Ch.0 } (\text{Ch.1 } \text{Ch.2})}{3667 \ 21:57 \ 16/09/07 \ 1 \ 0 \ 0}}_{3666 \ 21:57 \ 16/09/07 \ 1 \ 0 \ 0}$		Column interval						
No. Time Date Ch.0 Ch.1 Ch.2		Change the width of each column. The figures below are the						
$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$		examples.						
3660 21:56 16/09/07 0 0 0 3659 21:56 16/09/07 0 0 0 3659 21:56 16/09/07 0 0 - 3658 21:56 16/09/07 - - - 3658 21:56 16/09/07 - - -		$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$						



- (1)	
Profile color	Change the color of frame and background. Use [Transparent] to
	hide frames and background.
Time / Date	Enable or disable showing the time and date and configure its
	format.
	Time ascending
	Put earlier data at the top and the latest data at the bottom.
	Time descending
	Put the latest data at the top and the earlier data at the bottom.
History Control	The history files are sorted by date and each file is given an index.
(eMT, iE, XE,	The latest one is assigned index 0 (in most cases: today), the
mTV Series)	second latest file is assigned index 1, and so on. [History Control] is
	used to specify the history data to be shown.



When using cMT-SVR, use the filter icon in the upper-right corner of History Data Display object on iPad to select the date and display the data.



Data Format Tab

General Data Format Title Shape
Channel : Channel 0 ~ Channel 7 Channel 0 [16-bit Unsigned] Left of decimal Pt. : 4 Right of decimal Pt. : 0
☑ Display Center □ Leading zero
Channel 1 [16-bit Unsigned]
Display
Channel 2 [16-bit Unsigned]
Display
Channel 3 [16-bit Unsigned] Left of decimal Pt. : 4 💌 Right of decimal Pt. : 0 🖈
☑ Display Center □ Leading zero

SettingDescriptionChannelEach History Data Display object can display up to 64 channels.
Check [Display] to select the channels to be shown on the screen.
In the figure above, there are 4 channels (channel 0 to channel 3) in
the Data Sampling object, and only Ch.0 and Ch.3 are selected. The

data formats are shown next to channel name. The data format of each channel is decided by the corresponding Data Sampling objects. The result is shown below:

No.	Time	Date	Ch.0	Ch.3	
5272	22:43:09	16/09/07	4	1	
5271	22:43:08		2	0	
5270	22:33:42		0	0	
5269	22:33:41	16/09/07	0	0	
5268	22:33:40		0	0	
5267	22:33:39		0	0	
5266	22:33:38		0	0	
5265	22:33:37		0	0	
5264	22:33:36		0	0	
5263	22.33.32	16/00/07	0	0	

When display [String] format in History Data Display object, there are two additional options:





• Reverse high byte and low byte data and then display.

Title Tab

1	lew History Data Display (Object			—		
	General Data Format Tit	tle Shape]				
ſ	✓ Use title						
	Title background						
	Transparent		Color :				
	Title name	Title	Label library	Label tag	<u>^</u>		
	Sequence no.	No.	None				
	Time	Time	None		=		
	Date [Date	None				
	Channel 0 d	ch.0	None				
	Channel 1	ch.1	None				
	Channel 2	ch.2	None		_		
	Channel 3	ch.3	None		_		
		ch.4	None		_		
		ch.5	None		_		
		ch.6	None		_		
		ch.7	None		_		
		ch.8	None		_		
		ch.9	None		_		
		ch.10	None		- 1		
		ch.11	None		_		
		ch.12	None		- 1		
	Channel 13	ch.13	None		-		
				Setting			
				octangin			
		ОК	Cancel		Help		
	Description						
е	Enable or dis	ahle tit	le which is	marked a	s shown he	olow.	
				markea a	5 5110 0011 50	21010	
	No. Time 5272 22:43:0	Date 9 16/09/					
		8 16/09/					
	Transparent						
und	-						
	When selecte	ed, hid	e the backg	round for	title area.		
	Color						
	Set the back	ground	color of titl	e.			
		,					
	Defines the t	ext to I	he shown o	n the title			
	Defines the t					n n + + + + + + + + + + + + + + + + + +	o in
	Defines the t Label Tag Lib multiple lang	rary ca	n be used fo	or title in c	order to dis		



Titl	e Setting				
	Title : Label tag :			↓	
		🔽 Use label library			
			ОК	Cancel	



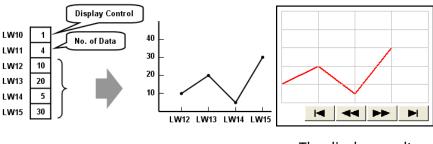
During off-line simulation, if the format of sampling data has been changed, please delete previous data records in C:\[EasyBuilder Pro directory]\HMI_memory\datalog to prevent the system from misinterpreting the old data records.



13.19. Data Block Display

13.19.1. Overview

Data Block is a combination of several word devices with continuous address, where the X axis of Data Block Display object represents the address and the numbers on the Y axis represent the data values in the corresponding address. Data Block Display object can display multiple data blocks. For example, it can display two data blocks LW-12~LW-15 and RW-12~RW-15 in trend curves simultaneously. It is very useful to observe and compare the difference of trend curves.



The display result

13.19.2. Configuration



Click the Data Block Display icon on the toolbar to open the property dialog box. Set up the properties, press OK button, and a new Data Block Display object will be created.



General Tab

eneral Displa	y Area Shape		
Comm			
No. of chan			
Cursor line	nel: 1		
Carbor Inc	V Enable	Color :	
PLC name :	Local HMI	•	Setting
Address :	LW 🔻 0		
	nel : 0 🔻		
Control addre			
PLC name :		•	Setting
Address :	LW 🔻 10		
No. of	data address : LW : 10 + 1	Offset to	start address
Data storage	start address :		
PLC name :	Local HMI	•	Setting
	LW - 12		16-bit Unsigned
Address :			
Address :			7
Limit	4in.: 0	Max.: 3276	
Limit	1in. : 0	Max. : 32/6	
Limit	/lin.: 0	Max. : 3276	
Limit	/in.: 0	Max. : 3276	
Limit	/in.: 0	Max. : 3276	
Limit	/in.: 0	Max. : 3276	

Setting	Description
Comment	Description of the object.
No. of	Set the no of channel for this object. Each channel represents one
channel	data block. The maximal number of channels is 12.
Cursor Line	If enabled, when user touches the [Data Block Display] object, it will display a vertical cursor line on it, and store the data on the line to the designated registers. See Example 1.
Channel	Select the channel to be configured.
Control	Specify the control address also the data source.
address	Control address is used to control and clear the drawn curve. After
	executing the operation below, the system will reset the control
	word to zero.
	Enter "0" = No action (default)
	Enter "1" = Draw (Without clear first)
	Enter "2" = Clear
	Enter "3" = Redraw
	No. of data address
	If control address is LW-n, then LW-n+1 stores the number of word

	devices in each data block, i.e. the number of data. The maximum
	value is 1024.
	Data storage start address
	If [Offset to start address] is enabled, the [Offset value storage
	address] will be set as [Control address] + 2.
	If select 16-bit data format, the address for each data will be start
	address, start address + 1, start address + 2 and so on.
	If select 32-bit data format, the address for each data will be start
	address, start address + 2, start address + 4 and so on.
	For more information about control address, see Example 2 to 5.
Limit	Set the minimum and maximum limit for the curve.

Note

The system can draw at most N curves, where N = 32 divided by the number of channel.

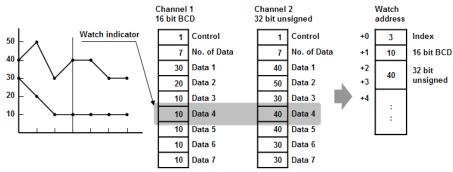
Example 1

How to use watch (Cursor Line) feature

Use "Watch" function to check the value of any point of the curve. When the user touches [Data Block] object, it will display a "cursor line", and the system will write the index and value of that data on the cursor line to the designated address.

Data Format	Index Value	Channel 1 Value	Channel 2 Value
16-bit	Address	Address + 1	Address + 2
32-bit	Address	Address + 2	Address + 4

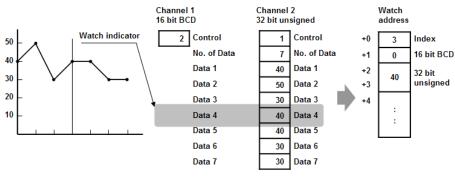
When watch address is set to LW-n, the value written into LW-n represents the channel index number to be called up. (Start form 0)

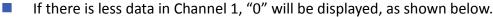


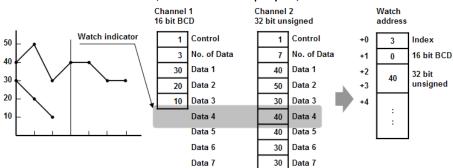
Note

- [Data Index] is a 16 bit unsigned integer. When the designated register of cursor line is 32 bit device, it will be stored in the bit 0-15.
- If the channel to be viewed has no data, "0" will be displayed, as shown below. In the example, there is no data in channel 1, when the cursor points at Data 4, "0" will be displayed as shown below.





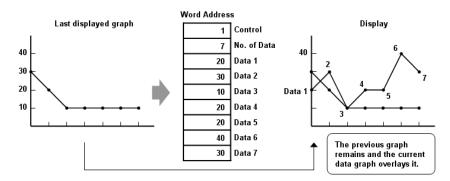




Example 2

How to show a data block

- 1. Write the number of data to [No. of data address], i.e. "Control address+1"
- 2. Store the data consecutively beginning at [Data storage start address].
- **3.** Write "1" to [Control address] to draw the curve without cleaning the plot. All previous curves will not be erased.
- 4. The system will write "0" to [Control address] after marking the plot.



Note

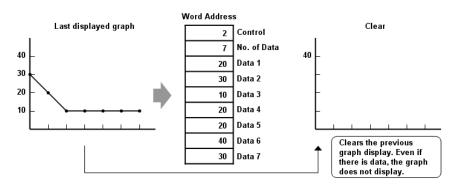
Do not change the content of [Control address], [No. of data address] and [Data storage start address] between step 3 and step 4 above as doing so might cause error for the trend curve plot.



Example 3

How to clear the graph

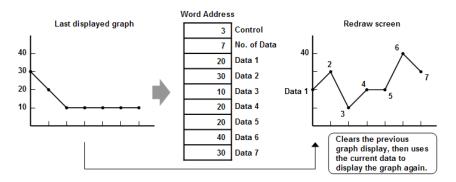
- **1.** Write "2" to [Control address], all the trend curves will be cleared.
- 2. The system will write "0" to [Control address] after the trend curve is cleared.



Example 4

How to clear the previous trend curve and display new one

- 1. Write the number of data to [No. of data address], i.e. "control address+1"
- 2. Store the data consecutively beginning at [Data storage start address].
- **3.** Write "3" to [Control address], the previous trend curves will be cleared and the new content in data block will be plotted on the screen.
- 4. The system will write "0" to [Control address] after the trend curve has been plotted.



Example 5

How to use offset mode

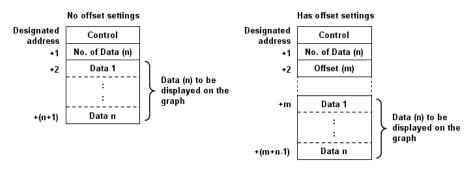
If [Offset to start address] is selected, [Control address], [No. of data address], and [Offset value storage address] will use 3 consecutive addresses.

For example, assume the total number of channels is 3 (start from 0 to 2), and the [Control address] are LW-0, LW-100, and LW-200, respectively. Then, the other addresses will be set as follows: (In the example, format 16-bit Unsigned is used and [Offset value storage address] are all m).



ltem	Channel 0	Channel 1	Channel 2
Control Address	LW-0	LW-100	LW-200
No. of data	LW-1	LW-101	LW-201
address			
Offset value	LW-2 (=m)	LW-102 (=m)	LW-202 (=m)
storage address			
Data 1	LW-0+m	LW-100+m	LW-200+m
Data 2	LW-1+m	LW-101+m	LW-201+m

The following figure on the left shows the result when offset mode is not used while the figure on the right shows the result when offset mode is used.

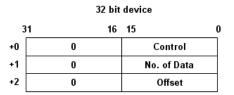


Note

When [Control address] is set to LW-n, [No. of data address] and [Offset value storage address] are as follows:

Data Type	16-bit	32-bit
Control address	LW-n	LW-n
No. of data address	LW-n+1	LW-n+2
Offset value storage address	LW-n+2	LW-n+4

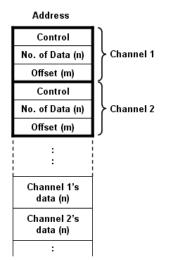
If the control registers are 32-bit devices, only bit 0-15 will be used for control purpose, bit 16-31 will be ignored. (as illustrated below)



- When the value in [Control address] is not zero, the system will read [No. on data address] and [Offset value storage address].
- It is recommended to use [Offset to start address] for data block display with multiple channels and the same device type. As shown in the following figure, The control words of



channel 1 is located from LW-n, the control words of channel 2 is located from LW-n+3, and so on.



Display Area

Data Block Display Object's Properties General Display Area Shape Profile
Data samples : 50 - Samples to scroll : 10 -
Profile color
□ Transparent Frame : Background : □
Grid
✓ Enable Color:
Horiz. : 5 🚔 division(s) Verti. : 5 👘 division(s)
Channel Channel Pen property Color : Width : 3
OK Cancel Help

Setting	Description
Description	Data samples
	Configure the maximal number of data samples (points) to be
	displayed.



	Samples to scroll		
	Configure the number of data samples being scrolled.		
	Enable scroll switch		
	Clicking displays the previous or next data point.		
	Clicking displays the first or the last data point.		
Profile	Set the color of the frame and background of the object.		
	Transparent		
	Hides the background. Color selection will not be available.		
Grid	Set the number of horizontal and vertical divisions shown by grid.		
Channel	Set the color, width and style of each curve.		



13.20. XY Plot

13.20.1. Overview

XY Plot object is used to display values for two variables (x,y) for a set of data, where the data comes from word registers. Up to 32 channels can be displayed simultaneously. This object facilitates data observation and analysis. Additionally, negative numbers can be displayed as well.

13.20.2. Configuration



Click the XY Plot icon on the toolbar to open a [XY Plot] object property dialog box.

General Tab

(Y Plot Object's Properties	X
General Display Area Shape Profile	ile
Comment :	
Direction : Right	▼ No. of channels : 2
Control Address	
PLC name : Local HMI	▼ Setting
Address : LW	▼ 10
No. of data address : [LW : 10	J+1
Channel : 0	•
PLC name : Local HMI	
	· · · · · · · · · · · · · · · · · · ·
✓ Separated add	ldress for X and Y data
PLC name : Local HMI	✓ Setting
Address : LW	▼ 100 16-bit Unsigned
Y data	
PLC name : Local HMI	- Setting
Address : LW	▼ 200 16-bit Unsigned
Limits	
X axis	3
Low : 0	High : 32767
Yaxis	
Low: 0	High: 32767
OK	Cancel Help



Setting	Description					
Direction	There are	four sele	ctions, right, le	eft, up or down.		
	Right Y P origin →		Left Left direction	Up Up direction × ↓ origin → v	Down origin → ↓ × Down direction	
No. of channels	Set the number of channels for observation.					
Control	Controls t	he opera	tion of all chan	inels simultaneou	usly. When the	
address	[Control a	ddress] is	s LW-n, assignii	ng values to LW-r	n will issue	
	command	s to XY pl	lot according to	o the table below	v. Meanwhile,	
	LW-n+11	controls t	he number of	data points plotte	ed. After	
	operation	, the [Cor	ntrol address] v	will be reset to 0.		
	Control address	Value	Result			
	LW-n	1		on XY curve.		
		2	· · ·	(The plotted points are kept.) Clears all XY curves.		
		3		plots new XY cu	rve.	
	LW-n+1					
	No. of data address					
	Controls the number of data points. Each channel can plot up to					
	1023 points.					
Channel	Select a channel to configure.					
Read Address	PLC name					
	Select a PLC which will be the source of [X data] and [Y data] and					
	designate a read address.					
	The format of the data register blocks used for the display channels					
	depends on whether [Separated address for X and Y data] and/or					
	[Dynamic limits] has been selected. See Example 1.					
Dynamic	 Whe 	n not sele	ected (See Exa	mple 2)		
limits	The Low and High limits can be set by entering constants. The Low					
	and High limits are used for calculating X and Y range in					
	percentage.					
	 When selected (See Example 3) 					
	Whe	n selecte	d (See Example	23)		



Example 1

The format of the data register blocks used for the display channels depends on whether [Separated address for X and Y data] has been selected, and if [Dynamic limits] has been selected. The following explains the situations where 16-bit register is used:

If [Separated address for X and Y data] is not selected, and set [Read address] to LW-n:

	Select [Dynam	Select [Dynamic limits]		Not select [Dynamic limits]	
	X data	Y data	X data	Y data	
Low Limit	LW-n	LW-n+2	Constant	Constant	
High Limit	LW-n+1	LW-n+3	Constant	Constant	
1 st data	LW-n+4	LW-n+5	LW-n+0	LW-n+1	
2 nd data	LW-n+6	LW-n+7	LW-n+2	LW-n+3	
3 rd data	LW-n+8	LW-n+9	LW-n+4	LW-n+5	
4 th data	LW-n+10	LW-n+11	LW-n+6	LW-n+7	

• If [Separated address for X and Y data] is selected, and set [X data] to LW-m, [Y data] to LW-n:

	Select [Dynam	Select [Dynamic limits]		Not select [Dynamic limits]		
	X data	Y data	X data	Y data		
Low Limit	LW-m+0	LW-n+0	Constant	Constant		
High Limit	LW-m+1	LW-n+1	Constant	Constant		
1 st data	LW-m+2	LW-n+2	LW-m+0	LW-n+0		
2 nd data	LW-m+3	LW-n+3	LW-m+1	LW-n+1		
3 rd data	LW-m+4	LW-n+4	LW-m+2	LW-n+2		
4 th data	LW-m+5	LW-n+5	LW-m+3	LW-n+3		

Example 2

When [Dynamic limits] is not selected, the Low and High limits can be set. The Low and High limits are used for calculating X and Y range in percentage.

Scale (%) =
$$\frac{\text{Read Address Value} - \text{Low Limit}}{\text{High Limit} - \text{Low Lmit}}$$

If [Separated address for X and Y data] is **not** selected and the address is LW-n, the corresponding limits are retrieved from the addresses as shown in the following table.

Data format	16-bit	32-bit
X axis low limit	LW-n	LW-n
X axis high limit	LW-n+1	LW-n+2
Y axis low limit	LW-n+2	LW-n+4
Y axis high limit	LW-n+3	LW-n+6

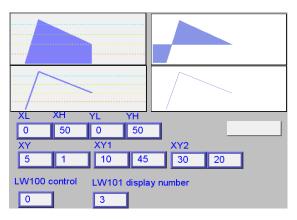


Objects

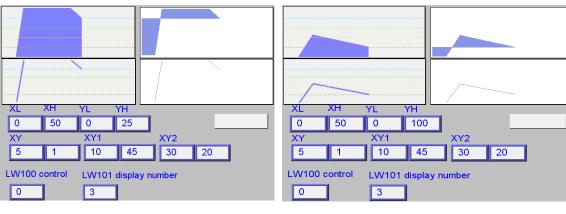
Example 3

If **[Dynamic limits]** is selected, a zoom effect can be created by changing the setting of Low / High Limits.

In the following example, XL=X low limit, XH=X high limit, YL=Y low limit, YH=Y high limit, and XY, XY1, XY2 are three XY data. When changing the high limits of X and Y axis, the result is shown below:



Original



Change the high limit of Y axis to 25. (zoom in)

Change the high limit of Y axis to 100 (zoom out)

For more information, see "13.17 Trend Display".

Note

- For cMT Series, on the screen of the visualization device directly pinch two fingers together to zoom out or spread them apart to zoom in.
- X and Y data can be set to different formats. For example: If X data uses 16-bit unsigned, Y data uses 32-bit signed, please note the address setting.
- When using a Tag PLC, such as AB tag PLC, X and Y must be in the same format. When using different formats a warning will be shown.



Display Area Tab

(Y Plot Object's P	roperties		×
General Display	Area Shape Profile		
-Profile color			
	Transparent		
Fram	ie:	Background :	
Curve			
Chann	el: 0 🔻		
Pen property			
Colo	or :	Width : 1	-
Maker O Line	Point	Point width : 1	▼ /-axis projection
-Reference line	☑ Limit from PLC		
PLC name :	.ocal HMI		▼ Setting
Address :	.w 👻	500	16-bit Unsigned
	Reference line 1	20	
	Reference line 2	40	
	Reference line 3	60	•
	📝 Reference line 4	80	
	ОК	Cancel	Help

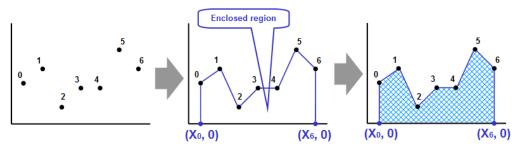
Setting	Description				
Profile color	Select the color of the frame and the background, or select				
	[Transparent] check box to hide the frame and background.				
Curve	For each channel select the properties of color, width, and line				
	style.				
Maker	There are four different types of XY plot. The result is shown below:				
	Line Dot				

	X-Axis Projection Y-Axis Projection See Example 4.
Reference line	Up to 4 horizontal reference lines can be shown on the graph. Fill in high, low limits and Y axis percentage values. Different colors can be selected for each reference line. If [Limit from PLC] is selected, designate a register to be the read address of reference line.

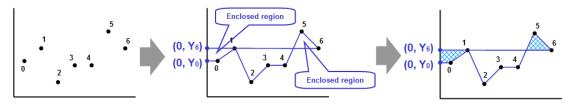
Example 4

The curve shown in the following figure is drawn with 7 points numbered from P0 to P6. The steps the system draws the X-axis Projection are:

- **1.** Calculates the two points in X-axis $(X_0, 0)$ and $(X_6, 0)$.
- 2. Link all the points in the order of $(X_0, 0)$, P0... P6, $(X_6, 0)$ and returns to $(X_0, 0)$ at last.
- 3. Fill out all enclosed areas.



Similarly for Y-axis projection:



Note

- XY Plot can be drawn repeatedly up to 32 times:
 - 1 channel→32 times
 - 2 channels → 16 times

The way to calculate: 32 divided by the number of channels.



13.21. Alarm Bar and Alarm Display

13.21.1. Overview

Alarm Bar and Alarm Display objects are used to display alarm messages which are defined in Event (Alarm) Log objects. When the trigger conditions are met, events or alarms will be displayed as they occur in chronological order in Alarm Bar or Alarm Display object. Alarm Bar scrolls all alarm messages in one single display line, whereas Alarm Display shows alarm messages in multiple lines.

For more information, see "7 Event Log".

1 (When LW 1 >= 10) 13:21:06 Event 0 (when LW0

Alarm Bar - Displays alarm messages in one scrolling line.

3/12/06	13:21:38	Event 2 (when LB10 = ON)
3/12/06	13:21:38	Event 3 (when LB11 = ON)
3/12/06	13:21:38	Event 0 (when LW0 == 100)
13/12/06	13:21:38	Event 1 (When LW 1 >= 10)

Alarm Display – Displays alarm messages in multiple lines.

13.21.2. Configuration



Click the Alarm Display or Alarm Bar icon on the toolbar to open the object property dialog box. Set up the properties, press OK button, and a new object will be created.

General Tab

The difference between these two objects is that Alarm Display allows an [Acknowledge address] and a [Scrolling control address] to be set.



neral Alarm	n Security Shape Font
Comr	ment :
Acknowledge	e address
PLC name :	Local HMI Settings
Address :	LW 🗸 0 16-bit Unsigned
	Enable acknowledge function
crolling con	ntrol address
crolling con	urol address ✓ Enable
	The second secon
PLC name :	Enable
	Enable

Setting	Description			
Enable	If selected, the [Acknowledge value] selected for the associated			
acknowledge	event, specified in Alarm (Event) Log » Message tab will be written to			
function	the [Acknowledge address] designated in Alarm Display. For more			
	information, see "7 Event Log".			
	Acknowledge value for Event/Alarm Display object			
	Acknowledge value : 11			
Scrolling	If enabled, the value in the designated control address indicates the			
control	number of lines to be scrolled down. The minimal value is 0, which			
address	indicates the first line.			

Note

When using cMT-SVR, press and hold the event on the screen to acknowledge an event; drag a finger on the screen to scroll.

The following are general settings of these two objects:



Alarm Tab

	rm Ba	ar Object		— ×
Alarm	Shap	e Font		
Inc	lude o	ategories : 0 thru 0	{see Event (A	larm) Log object}
	Scr	roll speed : Speed 5	•	
Color ·		Transparent		
			-	
		Frame :	 Backg 	round :
Format				
Sort				
() T	ïme a	scending Time desc 	ending	
Order	& Ch	aracters		
		Display items		Display order
	V	Event trigger date		Event trigger time
	V	Event trigger time		Event message
	1	Event message		Event trigger date
			•	
Date :	MM/	DD/YY Time :	HH:MM:SS	
Date :	[MM/	DD/YY Time:	HH:MM:SS	•
Date :	[MM/	DD/YY Time :	HH:MM:SS	•
Date :	[MM/	DD/YY Time :	HH:MM:SS	•

Setting	Description			
Include	Events in the selected category will be displayed. The categories are			
categories	set in Event (Alarm) Log object.			
-	For example, if the category is set to "2 to 4" here, only events in			
	categories 2, 3, 4 will be displayed. For more information, see "7			
	Event Log".			
Scroll speed	This selection is only available for Alarm Bar. Select one of the			
	speed settings at which the messages scroll.			
Format	Time ascending			
	Latest alarm is placed last in the list (the bottom).			
	Time descending			
	Latest alarm is placed first in the list (the top).			
	Display order			
	Select the items to be displayed and use the up and down arrow			
	buttons to adjust the display order of the alarms.			
	Date			
	Displays the date tag with each alarm message. The four formats of			
	date tag:			
	MM/DD/YY、DD/MM/YY、DD/MM/YY、YY/MM/DD			



Time Displays the time tag with each alarm message. The four formats of time tag: HH:MM:SS 、 HH:MM 、 DD:HH:MM 、 HH

Font Tab

Set the font size or select [Italic].

New Alarm Bar Object	- ×
Alarm Shape Font	
⊂ Attribute	
Size : 16	•
✓ Italic	

The font, color, and content of the alarm messages displayed in Alarm Bar and Alarm Display objects are set in Alarm (Event) Log object:

Event (Alarm) Log		x
General Message		
Text		П
Content :	Event 1: press once to acknowledge	
	<u>v</u>	
	Use label library Label Library	
Font :	Arial	
Color :		



13.22. Event Display

13.22.1. Overview

Event Display object is used to display event messages which are defined in Event (Alarm) Log and have met a trigger condition. The triggered events are displayed in the chronological order. Event Display object displays not only the date and time the event occurs, but also the time the event is acknowledged, the time the event returns to normal, and the event message. Multi-lined messages can also be displayed.

8	12/13/06	22:03:15		Event 3 (when LB11 = ON)	
7	12/13/06	22:03:14	22:03:17	Event 2 (when LB10 = ON)	
6	12/13/06	22:03:13		Event 1 (When LW 1 >= 10)	
5	12/13/06	22:03:12		Event 0 (when LW0 == 100)	
4	12/13/06	22:02:57		Event 3 (when LB11 = ON)	
3	12/13/06	22:02:56	22:03:04	Event 2 (when LB10 = ON)	
2	12/13/06	22:02:56	22:02:58	Event 1 (When LW 1 >= 10)	

1	07/27/10	14:32:56	14:32:57	14:32:59	Event 0 LW 0< 2 Multi-text	Multi-te

13.22.2. Configuration



Click the Event Display icon on the toolbar to open an Event Display object property dialog box. Set up the properties, press OK button, and a new Event Display object will be created.



General Tab

13.22.2.1. eMT, iE, XE, mTV Series

eneral	Event Display Security Shape Font
	Brown Display Rooding Rispo Tom
	Comment :
	Mode : Real-time 🗸
	Real-time History
Ackno	wledge address
PLC	name : Local HMI Settings
Ad	dress : LW 🗸 0 16-bit Unsigned
Contro	ol address
PLC	name : Local HMI 🔹 Settings
	dress : LW V V 0 16-bit Unsigne
~~~	
	Enable event management Usage
	Uugu
	Uugu
	Uugg
	Uddge
	Uddge
	Uddgr
Secoli	
Scrolli	ing control address
Scrolli	
	ing control address I Enable
PLC	ing control address
PLC	ing control address I Enable
PLC	ing control address
PLC	ing control address
PLC	ing control address

Setting	Description	
Mode	The available modes are: [Real-time] and [History].	
	Real-time	
	All the events triggered since HMI starts up are displayed.	
	History	
	The system reads the event log in HMI memory and displays	
	them. The content can be updated by changing window. In case	
	when the trend display shows history data from today, the	
	display will refresh once per second.	
Acknowledge	When in Real-time mode, and an event is acknowledged by touching	
address	an active display line, the [Acknowledge value] specified in Event	
	(Alarm) Log object, Message tab, is output to the [Acknowledge	
	address] of Event Display object. For more information, see "7 Event	
	Log".	



	Acknowledge value for Event/Alarm Display object Acknowledge value : 11
History	When in History mode, and if:
Control	<ul> <li>[Enable reading multiple histories] is <b>not</b> selected</li> </ul>
	Daily event log files can be displayed. A history control address car
	be designated. The value in the designated register is used as a
	index to select historical files.
	Index value 0 displays the latest file.
	Index value 1 displays the second latest file.
	Index value 2 displays the third latest file, and so on.
	If control address to LW-100, and four data log exist with dates:
	EL_20100720.evt, EL_20100723.evt, EL_20100727.evt, and
	EL_20100803.evt.
	Each index value in the control word corresponds to a recor
	according to the table below:
	Value in LW-100 The corresponding record
	0 EL_20100803.evt
	1 EL_20100727.evt
	2 EL_20100723.evt
	3 EL_20100720.evt
	<ul> <li>[Enable reading multiple histories] is selected</li> </ul>
	Displays a list of events triggered in multiple days. If [History contro
	address is set to LW-n, the LW-n to LW-n+1 form a range of lo
	selection.
	Number of days
	The data range starts from the number in LW-n. The value in LW-n+
	represents how many days to be included from the start to day
	before.
	Example: As illustrated below, if LW-n = 1, and LW-n+1 = 3, then th
	range of data will start from 20100609, and include the data of
	days before (20100609 included). However, since the data of
	20100607 does not exist in this example, the data displayed will onl
	include 20100609 and 20100608.
	EL_20100604         No.4         1 KB         EVT           EL_20100605         No.3         6 KB         EVT
	EVEL_20100608 No.2 17 KB EVT
	EL_20100609 No.1 4 KB EVT
	EL_20100610 No.0 12 KB EVT

#### Index of the last history

The range of data will start from the number in LW-n and end in LW-n+1. If LW-n = 1, and LW-n+1 = 3, the data displayed will include data No.1, No.2, No.3. If the number entered in LW-n+1 is greater than the number of event log files, LW-n+1 will not be effective and only the data specified by LW-n is displayed.



	EL_20100608No.217 KBEVTMo.14 KBEVTNo.012 KBEVTThe maximum size of data that can be displayed is 4MB; the exceeding part will be ignored.The following shows how data will be stored when the data size is too big.5 history data, each 0.5MB $\rightarrow$ Data displayed: 8 x 0.5MB5 history data, each 1MB $\rightarrow$ Data displayed: 4 x 1MB5 history data, each 1.5MB $\rightarrow$ Data displayed:2 x 1.5MB+1 x 1MB (partial)
Control	Enable event management
address	If this check box is selected, writing a specific value into register LW-n and LW-n+1, where n is an arbitrary number, will control [Event Display] object with different commands as shown below:
	Address Value Command
	LW-n 0 Display all events.
	1 Hide [Confirmed] events.
	2 Hide [Recovered] events.
	3 Hide [Confirmed] or [Recovered] events.
	4 Hide [Confirmed] and [Recovered] events.
	LW-n+11Delete a single selected event.
Scrolling	If enabled, the value in the designated control address indicates the
control	number of lines to be scrolled down. The minimal value is 0, which
address	indicates the first line.
	In the following figure, there are 10 events recorded in the object,
	and value 3 is set at the control address. The upper event object
	displays the events in time ascending order, and begins at the 4 th
	event; on the other hand, the lower one displays the events in time
	descending order, and begins at the 7 th event.
	event LB0         5         18:48:19         Test Event           6         18:48:20         Test Event           7         18:48:20         Test Event           8         18:48:20         Test Event           Scrolling         7         18:48:20         Test Event
	0003         6         18:48:20         Test Event           5         18:48:19         Test Event           4         18:48:19         Test Event           3         18:48:19         Test Event
	If [Scrolling control address] is enabled, the scroll bar cannot be used

for scrolling control address is enabled, the scroll bar cannot be used for scrolling, but still shows the relative position of the content. If the control address holds a value that is larger than the total number of lines, the display will stroll to the end.



#### 13.22.2.2. cMT-SVR Series

Event Displa	y Object's Properties	×
General g	Event Display Shape Font Profile	
	Comment :	
Acknow	edge address	_
	me : Local HMI    Setting	_

For cMT-SVR, all the events occur are displayed and updated in real-time.

Press the filter icon in the upper-right corner of the object and set the start and end date. If the dates are not set, all the events are displayed.

					<		Ju	ly 20	13		
					MON	TUE	WED	THU	FRI	SAT	SUN
					1	2	3	4	5	6	7
2	11:55:04 Event 2	$\nabla$			8	9	10	11	12	13	14
1	11:55:04 Event 0		Cancel	ď	15	16	17	18	19	20	21
			✓ starts	>	22	23	24	25	26	27	28
			06/07/2013		29	30	31	1	2	3	4
			ends 08/07/2013	>	5	6	7	8	9	10	11

eMT, iE, XE, mTV Series

#### **Event Display Tab**

#### cMT Series

Include categories : 0 thru 255 {see Event (Alarm) Log object} or Transparent Frame : Background :  Acknowledge : Return to normal :  Enable Color :	Color Transparent Frame : Backgrow Acknowledge : Return to nom Select box : Return to nom Grid Ø Enable Co	style : Click at no. : 200
Transparent Frame : ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	Color Transparent Frame : Backgrou Acknowledge : Return to nom Select box : Return to nom Grid Ø Enable Co	und :
Transparent Frame : ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	Color Transparent Frame : Backgrou Acknowledge : Return to nom Select box : Return to nom Grid Ø Enable Co	und :
Transparent Frame : ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	Color Transparent Frame : Backgrow Acknowledge : Return to nom Select box : Return to nom Grid Ø Enable Co	und :
Transparent Frame : ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■	Transparent Frame : Backgrou Acknowledge : Return to nom Select box :  Grid Enable Co Format	mal :
Frame : Background : Acknowledge :	Frame : Backgrou Acknowledge : Return to norm Select box : V Grid Format	mal :
Acknowledge : Return to normal :	Acknowledge : Return to nom Select box : Select box : Co Grid Format	mal :
Enable Color :	Grid Select box : The s	
Color :	Grid Co Format Co	lor :
Color :	Format Co	lor :
	Format	olor :
natut	Sort	
Time ascending <ul> <li>Time descending</li> </ul>	Time ascending	
der & Characters	Order & Characters	
Display items Display chars Display order	Display items Display chars	Display order
Sequence no. 0 Event trigger time		vent trigger time
Event trigger date 0		ent message
Event trigger time     Acknowledge time     O	Event trigger time     Acknowledge time     O	
Return to normal time 0     Event message 20	Return to normal time     V     Event message     20	
	Occurrence count 0	
		- NA
"Display chars" is 0, it means that the system will display all of characters.	If "Display chars" is 0, it means that the system will display all of	í characters.
te : MM/DD/YYYY V Time : HH:MM:SS V	Date : MM/DD/YY	
MWWDDALAAA A A A A A A A A A A A A A A A A	Date : MM/DD/YY	<b>•</b>
OK Cancel Help	OK Cancel	_



#### EasyBuilder Pro V5.02.01

Setting	Description			
Include	Events in the selected category will be displayed. The categories are			
categories	set in Event (Alarm) Log object.			
-	For example, if the category is set to 2 to 4 here, only events in			
	categories 2, 3, 4 will be displayed. For more information, see "7			
	Event Log".			
Acknowledge	Select [Click] or [Double Click] to acknowledge each single event.			
style	When an event occurs the user can tap the event line once or twice			
-	to acknowledge the new event.			
	When acknowledged, the text color of the event will change to the			
	selected color, and the acknowledge value associated with that			
	event will be sent to the register designated in [Acknowledge			
	address]. If the address is set to LW-100, and the acknowledge			
	value is set to 31, when user acknowledges the event, value 31 is			
	written to LW-100.			
	This can be used in conjunction with Indirect Window object so			
	that when an event is acknowledged, the corresponding message			
	window is displayed.			
Max. event	The maximum number of events to be displayed in this Event			
no.	Display object. When the number of the displayed events equals to			
	the set maximum number, the new coming event will overwrite the			
	latest event.			
Color	Different colors indicate different event states, such as			
	acknowledged, returns to normal, or selected. The system draws a			
	highlight box around the latest selected event. Acknowledge			
	6 13:12:19 Event 1 (When LW 1 >= 10) 5 13:12:18 Event 2 (when LB10 = ON)			
	(4) 19:12:16 (19:12:20 Event 3 (when (19:11 - (9)))			
	3     13:12:15     Event 2 (when LB10 = ON)       2     13:12:14     Event 1 (When LW 1 >= 10)			
	1 13:12:14 Event 0 (when LW0 == 100)			
	Sequence no. Return to normal Select box			
Grid	Displays a grid of rows and columns in the object. The color of the			
	grid lines can be selected.			
Format	trigger date trigger time notification time return to normal time			
	0 12/14/06 15:26:21 15:26:31 15:26:36 Event 0 (when LV 1 12/14/06 15:26:47 15:26:50 Event 1 (When LV			
	2 12/14/06 15:26:48 Event 2 (when LE			
	Time ascending			
	Latest event is placed last in the list (the bottom).			
	Eatest event is placed last in the list (the bottolin).			
	Time descending			

Order & Characters
Select the items to be displayed and use the up and down arrow
buttons to adjust the display order of the events.
Date
Displays the date tag with each event message. The four formats of
date tag:
MM/DD/YY、DD/MM/YY、DD/MM/YY、YY/MM/DD
Time
Displays the time tag with each event message. The four formats of
time tag:
 HH:MM:SS、HH:MM、DD:HH:MM、HH

#### Font Tab

In Real-time mode: Users may select Italic font and set the font size. The font is displayed according to the setting in Event Log object.

In History mode: Users may select Italic font and set the font size, font and color, or tick the [Font from label library] check box.

(	Color :			Size :	16	~
	÷					
	L	Font fro	m label library			
	F	Italic				



## 13.23. Data Transfer (Trigger-based)

#### 13.23.1. Overview

Data Transfer (Trigger-based) object can transfer values from the source register to the destination register. The data transfer operation can be activated by changing the state of the designated bit register, or by manually pressing the object. For cMT Series, only touch trigger mode is available.

#### 13.23.2. Configuration



Click the Data Transfer (Trigger-based) icon on the toolbar to open the property dialog box. Set up the properties, press OK button, and a new Data Transfer (Trigger-based) object will be created.

#### **General Tab**

#### **cMT** Series

#### eMT, iE, XE, mTV Series

New Data Transfer (Trigger-based) Object	New Data Transfer (Trigger-based) Object
General Security Shape Label	General Security Shape Label
Comment : Source address PLC name : MODBUS RTU	Comment : Source address PLC name : MODBUS RTU
No. of word : 1	No. of word : 1 Mode : External trigger ▼ Trigger address External trigger PLC name : Local HMI Address : IB ▼ 0
OK Cancel Help	OK Cancel Help



Setting	Description	
Source	Data Transfor object roads the data from [Source Address]	
address	Data Transfer object reads the data from [Source Address].	
Destination	Data Transfer object writes the data to [Destination Address]	
address	Data Transfer object writes the data to [Destination Address].	
Attribute	No. of words	
	The number of words to be transferred from [Source Address] to	
	[Destination Address]. The unit is word.	
	Mode	
	Touch trigger	
	Press the object to activate data transfer operation.	
	External trigger	
	The data transfer operation is activated when the state of the	
	designated bit address changes.	
	There is a further selection to make of whether the data transfer	
	operation is activated after Off to ON, ON to OFF transition, or at	
	both of the changes of state.	
Trigger address	Specify a bit address for [External trigger] mode.	



When using Data Transfer Trigger Based object, place the control bit addresses in the same window in order to trigger Data Transfer. If the Data Transfer Trigger Based object is placed in the common window, when the state of the control bit addresses placed in any window changes, Data Transfer is triggered.



## 13.24. Backup

#### 13.24.1. Overview

Backup object can transmit recipe data (RW, RW_A), event log, recipe database, sampling data, and operation log to an external device (SD card, USB disk), in a specified time range or format. For example, when the event log is saved in a SD card, a USB disk can be inserted when HMI power is still ON, and use Backup object to copy the data into USB disk from SD card, and then remove USB disk without turning off HMI power. The data saved in USB disk can be used on PC for analyzing. When the system is backing up, the state of system register [LB-9039] is set ON. With [e-Mail] option, information can be sent to configured email contacts.

#### 13.24.2. Configuration



For eMT, iE, XE, mTV Series, click Backup icon on the toolbar to open a [Backup] object property dialog box. Set up the properties, press OK button, and a new Backup object will be created.

For cMT Series, click Backup icon on the toolbar to open a [Backup] object managing dialog box, click [New] to open a [Backup] object property dialog box. Set up the properties, press OK button, and a new Backup object will be created.



## **General Tab**

cMT	Seri	ies
0		

## eMT, iE, XE, mTV Series

		New Backup Object	22
		General Security Shape Label	
		Comment :	
		RW      RW_A     Recipe database	
ickup	X	Historical event log     Historical data sampling	
General		Operation log	
Comment :		Backup position	
Source © RW © RW_A	Recipe database	SD card     O USB disk     e-Mail	
Historical event log	Historical data sampling	Remote printer/backup server	
		Note : Use LW-9032~9039 to change the backup folder name. Note : Use [Remote printer/backup server] to store data to a remote PC. Enable the	
Backup position		server in [System Parameter][Printer/Backup Server] settings.	
SD card OUSB		Save format Format : eMT3000 SERIES Event Log File (*.evt)	
	kup server] to store data to a remote PC. Enable the	Range	
server in [System Parameter][Pr	inter/backup Server] settings.	Start :      Today     Vesterday	
		Within : All (max. 90 days)	
		Trigger Mode : Touch trigger	
		· · · · · · · · · · · · · · · · · · ·	
PLC name : Local HMI	▼ Setting		
Address : LB	• 0		
		*LB-9039 indicates the status of file backup activity (backup in process if status is ON)	
0	K Cancel Help	OK Cancel Help	
Catting	Description		
Setting	Description		
Source	[RW], [RW_A], [Rec	cipe database], [Historical event log],	
	[Historical data sam	npling], [Operation log]	
	Select one from the	above for the source. When backing up	
	[Historical data log]	, use [Data Sampling object index] to select tl	he
	one to back up.		
Backup	· · · ·	on where the source files will be saved to.	
position	SD card / USB disk		
	The external device	connected to HMI.	
		SD card and USB disk can only save [RW],	
	[RW_A], and [Recip		
		ckup server (eMT, iE, XE, mTV Series)	
	•	le MT remote printer/backup server at: [Men	าน]
		arameters] » [Printer/Backup Server].	
		peration log] backup can only be saved to	
		ckup server. To save into a SD card or USB driv	
			ve,



	please use the control address of Operation Log object.					
	For more information, see "26 EasyPrinter".					
	E-mail					
	To use e-mail, go to [System Parameters] » [e-Mail] tab to configure					
	first. And then go to Backup object » [e-Mail] tab to configure the					
	recipient address, subject, and message.					
Save format	Select the desired format to back up the file.					
	eMT, iE, XE, mTV Series:					
	<ul> <li>HMI Event Log File (.evt) / HMI Data Log File (.dtl)</li> </ul>					
	<ul> <li>Comma Separated Values (.csv)</li> </ul>					
	When back up event log in .csv format, open the csv file in EXCEL.					
	The BOM (Byte Order Mark) can be added to the file header so that					
	the .csv file containg non-ASCII strings can directly be opened in					
	EXCEL. The milliseconds field can be included or omitted in the					
	backed-up .csv file.					
	Storage format Format : Comma Separated Values (*.csv)					
	Add BOM (Byte Order Mark) to file header for EXCEL can interpreting non-ASCII strings correctly.					
	✓ Include millisecond (ms) information					
	The [Event] column is included in the backup file to indicate the					
	type of the event.					
	A     B     C     D     E       1     Event     Category     Date     Time     Message					

	A	В	C	D	E
1	Event	Category	Date	Time	Message
2	0	1	2013/7/4	16:12:11	Event A
3	2	1	2013/7/4	16:12:12	Event A
4	0	0	2013/7/4	16:12:33	Event B
5	2	0	2013/7/4	16:12:36	Event B
6	0	0	2013/7/4	16:12:37	Event B
7	1	0	2013/7/4	16:12:37	Event B
8	2	0	2013/7/4	16:12:39	Event B
9	0	0	2013/7/4	16:12:40	Event B

0 = Event is triggered

1 = Event is acknowledged

2 = Event returns to normal

EasyConverter can be used to easily convert HMI Event Log File

(.evt) and HMI Data Log File (.dtl) to .xls or .csv format.

• SQLite Database File (.db)

#### cMT Series:

- SQLite Database File (.db)
- Comma Separated Values (.csv)

Range

Within

Select the number of days. For example, [Yesterday] is selected at [Start], and [2 day(s)] is selected here, which means the files



	obtained yesterday and the day before yesterday will be backed up				
	Select [All] to save all files in the system, the maximum is 90 days.				
Trigger	Mode				
(eMT, iE, XE,	There are three ways to activate Backup function.				
mTV Series)	Touch trigger				
	Touch the Backup object to activate backup operation.				
	External trigger (bit)				
	Register a bit device to trigger the backup operation.				
	Select whether the backup operation is activated after Off to ON,				
	ON to OFF transition, or at both of the changes of state.				
	External trigger (word)				
	Users can specify the number of days to backup data using [Trigger				
	address]. [Trigger address] usage (suppose LW-n is used):				
	LW-n: Will start to back up when the value changes from 0 to 1.				
	LW-n+1: The start date of backup.				
	LW-n+2: The number of days for backup. (The maximum: 90 days).				
	Syntax				
	LW : 0 Set 1 to trigger backup activity				
	LW:0+1				
	Define backup start day				
	0:today				
	1 : yesterday				
	2 : the day before yesterday n : and so on				
	2 : the day before yesterday				
	2 : the day before yesterday n : and so on				
	2 : the day before yesterday n : and so on LW : 0 + 2				
	2 : the day before yesterday n : and so on LW : 0 + 2 Define backup range				

Trigger	When the state of the designated register is set ON, the backup
address	operation is activated. When the backup operation is done, the
(cMT Series)	state of the designated register is set OFF.

- All history files should have been saved in memory, either HMI memory, USB disk or SD card. Otherwise, the Backup object will not work.
- The maximum number of days for backup is 90 days. (Not including cMT Series)

For cMT Series, see "7 Event Log" and "8 Data Sampling" that explain the mechanism of

synchronizing data to external device.



#### Advance Tab

eneral Advance	Security Shape	Label	Profile	
	nge : 00000 ~ 6553.			
	nge . 00000 ~ 0003.	<i>)</i>		
📝 Enable				
Source			ALC:	
PLC name : Lo	ocal HMI		▼ Se	ettings
Address : L	N	▼]0		
* For example, w	hen serial number is	123 :		
A data samulin	z file - 20140407 dtl	will he ba	cked up to 2014040700123.dtl	
				6
An event log fi	le - 20140407.evt w	ill be backe	d up to 2014040700123.evt.	
Options				
2003 2003				
🔽 Remove c	ld files after backup			

Setting	Description
Serial number	If enabled, when backing up history files, a user-defined, 5-digit serial number can be appended to the end of the file name of the history data backup. The serial number is determined by the value in the designated source address. After backup, the value of this LW address will automatically increment by 1. The range of the serial number is 0~65535
	For example, if the serial number is 123, the appended 5 digits will be 00123.
	A data sampling file -20140407.dtl will be backed up as 2014040700123.dtl.
	An event log file -20140407.evt will be backed up as 2014040700123.evt.
Options	Remove old files after backup
	If selected, the old history files will be removed after backup.

# Note

CMT Series does not support Advance settings.



#### e-Mail Tab

eneral	Advance	Security	Shape	Label	Profile	e-Mai	L		
🔽 Add	l.txt exten	sion to the	filename	to skip the	anti-virus	detection	1		
Recipie	ents								
	To								
	Cc Bcc								
Subject									
		. 8							
	Sub	ject:						F	÷
			Jse label	library					
Messag		-						_	_
	Open	ing :							*
		*						E	
	End		lse label l	library				_	
	15110.	uug .							*
		* 	Jse label I	librarv					
				N 6803-1123					
				Labe	l Library		angua;	ge 1	

Setting	Description			
Add .txt				
extension to	If selected, when sending backup data as an email attachment, the			
the filename	filename extension .txt will be added to the file name. This			
to skip the	prevents the mail server or anti-virus software from blocking			
anti-virus	emails.			
detection				
Recipients,				
Subject <i>,</i>	Back up the email address of the recipients, the subject of the			
Message	email, and the message content.			



## 13.25. Media Player

#### 13.25.1. Overview

At the first time using Media Player object in the project, download the project to HMI via Ethernet. EasyBuilder Pro installs Media Player driver automatically.

Media Player object plays video files with controls such as seek, zoom, and volume adjustment to provide maintenance instructions or procedures on video so as to enable on-site operators to perform tasks efficiently.

#### 13.25.2. Configuration



Click the Media Player icon on the toolbar to open a Media Player object property dialog box. Set up the properties, press OK button, and a new Media Player object will be created.

#### **General Tab**

Comment :	
Control address	
🔽 Enable	
PLC name : Local HMI	▼ Setting
Address : LW	• 0
Command : LW : 0	Status : LW : 0 + 3
Parameter 1: LW:0+1	File index : LW : 0 + 4
Parameter 2: LW:0+2	Start time : LW : 0 + 5
	End time : LW : 0 + 6
Update video p	aving time
Update period : 5 second	Playing time : LW : 0 + 7
Ext. device	
SD card OUSB disk	Folder name : video
Attribute	
Auto. repeat	Background :
* OS version 2012.11.12 or later suppo	ort media player only !

Setting	Description
Control	<ul> <li>Selected</li> </ul>



address	<ul> <li>Designate a word register to control the object operations.</li> <li>Not selected</li> </ul>				
	No manual control. Video will be played automatically when the designated window opens.				
	<b>Command</b> (control address + 0)				
	Enter a value in the Command register to designate which action is				
	executed.				
	Parameter 1 (control address + 1)				
	Enter a value in Parameter 1 associated with each command action				
	Parameter 2 (control address + 2)				
	Enter a value in Parameter 2 associated with each command action				
	Status (control address + 3)				
	Indicates the status or errors.				
	File index (control address + 4)				
	The file number in the designated folder. It is recommended to file				
	the video name with a number.				
	Start time (control address + 5)				
	The start time of the video (second). 0, normally.				
	End time (control address + 6)				
	The end time of the video (second). (The time length of the video)				
	Update video playing time				
	If enabled, the elapsed playing time of video will be written into				
	[Playing time] register at a rate set by [Update period] in seconds.				
	Update period				
	Update period of [Playing time], range from 1 to 60 (second).				
	Playing time (control address + 7)				
	The elapsed playing time of video (Second). Normally between				
	start time and end time.				
Ext. device	Play video files in SD card / USB disk.				
	Folder name				
	The folder name of video files stored in SD card or USB disk. Files				
	must be stored in root directory. Subdirectories won't be accepted.				
	(For example, "example\ex" is an invalid directory.) [Folder name] cannot be empty, must be alpha-numeric, and all in				
	ASCII character.				
Attribute					
	Auto. repeat				
	When finish playing all the video files, replay from the first file. Ex: Video 1 > Video 2 > Video 1 > Video 2				
	Background				
	The background color of the object.				



#### Objects

The data format for control address is 16-bit Unsigned or 16-bit Signed. If using 32-bit Unsigned or 32-bit Signed, only the previous 16 bits will be effective.

#### **Control command**

The following are the settings of different commands.

Play index file [Command] = 1 [Parameter 1] = file index [Parameter 2] = ignore (set 0)

# Note

- Files are stored with file names in ascending order.
- If the file cannot be found, [Status] bit 8 is set ON.
- Please stop the playing video before switching to another.

```
Play previous file
```

[Command] = 2 [Parameter 1] = ignore (set 0) [Parameter 2] = ignore (set 0)



- If [File index] is zero, the same file is replayed.
- If the file cannot be found, [Status] bit 8 is set ON.

#### Play next file

[Command] = 3 [Parameter 1] = ignore (set 0)

- [Parameter 2] = ignore (set 0)
- If there are no more files, the index 0 file is played.
- If the file cannot be found, [Status] bit 8 is set ON.

Pause / Play Switch [Command] = 4 [Parameter 1] = ignore (set 0)

- [Parameter 2] = ignore (set 0)
- Stop playing and close file
   [Command] = 5
   [Parameter 1] = ignore (set 0)



[Parameter 2] = ignore (set 0)

 Start playing from the designated time [Command] = 6 [Parameter 1] = target time (second) [Parameter 2] = ignore (set 0)

# Note

Parameter 1 (target time) must be less than the ending of time or it plays the last second.

```
Forward
```

[Command] = 7
[Parameter 1] = target time (second)
[Parameter 2] = ignore (set 0)

# Note

- Going forward to the designated second in [Parameter 1]. If the video is paused, the forwarding action will be started by playing.
- When the designed time is later than the end time, it plays the last second.

```
Backward
```

[Command] = 8 [Parameter 1] = target time (second) [Parameter 2] = ignore (set 0)

# Note

- Going Backward to the designated second in [Parameter 1], if the video is paused, the backward action will be started by playing.
- When the designed time is earlier than the beginning time, it plays from beginning.

```
Adjust volume
```

[Command] = 9 [Parameter 1] = volume (0 ~ 128) [Parameter 2] = ignore (set 0)



Default volume is 128.





[Command] = 10 [Parameter 1] = display size (0 ~ 16) [Parameter 2] = ignore (set 0)

# Note

- [Parameter 1 = 0] : Fit video image to object size.
- [Parameter 1 = 1 ~ 16]: Magnification from 25% ~ 400% in 25% increments where 1 = 25%, 2 = 50%, 3 = 75% and so on.

## Status (control address + 3)

When playing a video the system sets [File Open (bit00)] and [File Playing (bit01)] to ON. If the file cannot be found, or an invalid command is entered, the Command Error bit 08 is set ON. If the file format is not supported, or a disk I/O error occurs, during playback (for example, USB disk unplugged), the File Error bit 09 is set ON.

15	09	08	02 01	1 00	bit
Reserved (all 0)	0	0	0	0	

00: File Opened / Closed	(0 = closed, 1 = opened)
01: File Playing	(0 = not playing, 1 = playing)
08: Command Error	(0 = accepted, 1 = incorrect)
09: File Error	(0 = accepted, 1 = incorrect)

# Note

The figure shows the status value associated with each state:
 Stop = 0, Pause = 1, Playing = 3



[Command], [Parameter 1], and [Parameter 2] are write addresses. All others are read only.



#### **Preview Tab**

Users can test whether the video format is supported by using the preview function.



Description	
Go forward or backward of the video. (in minutes)	
do for ward of backward of the video. (in finitutes)	
Select to start playing video or pausing.	
Stop playing and close the video file. To test another video, please	
stop playing the current video first.	
Select a video to preview.	

# Note

- Only one video file can be played at one time.
- If [control address] is not enabled and [Auto. repeat] is not selected, after finish playing the first file, the system will stop playing and close the video file.
- If [control address] is not enabled, the system will find the first file in the designated folder and start to play (in ascending order of the file name).
- If the file can be previewed, the format is supported. If the video image quality is poor, please adjust the resolution.
- The supported formats: mpeg4, xvid, flv...etc.

Lick the icon to download the demo project. Please confirm your internet connection

before downloading the demo project.



## 13.26. Data Transfer

#### 13.26.1. Overview

Data Transfer object is similar to Trigger-based Data Transfer object. They transfer the data from source to destination register. The difference is that Data Transfer object transfers data based on time schedule, and is able to transfer data in bits.

When using cMT Series, Data Transfer object is divided into two modes: [Time-based] and [Bit trigger]. In these two modes, the system automatically detects the state of the designated bit register and executes data transfer. [Time-based] mode is the same as described earlier, where [Bit trigger] mode transfers data when the state of the designated bit register changes. For the detail of [Bit trigger] mode, see "13.26.2.2 Data Transfer Bit Trigger".

#### 13.26.2. Configuration



Click Data Transfer icon on the toolbar to open the Data Transfer management dialog box. Click [New] and configure the properties. All the defined Data Transfer can be viewed from the dialog box as shown in the following figure.

eMT, iE, XE, mTV Series

Data Transfer
Time-based
O:       [MODBUS RTU : 0x-1] -> [Local HMI : LB-0], Mode : Bit, Time interval=3.0 second(s), transfer length=1 bit(s)         1:       [MODBUS RTU : 3x-1] -> [Local HMI : LW-0], Mode : Word, Time interval=3.0 second(s), transfer length=1 word(s)
New Delete Settings Exit

**cMT** Series

lata Transfer
Time-based Bit trigger
0: [MODBUS RTU : 0x-1] -> [Local HMI : LB-0], Mode : Bit, Time interval=3.0 second(s), transfer length=1 bit(s) 1: [MODBUS RTU : 3x-1] -> [Local HMI : LW-0], Mode : Word, Time interval=3.0 second(s), transfer length=1 word(s)
New Delete Settings Ext



#### 13.26.2.1. Data Transfer Time-based

#### **General Tab**

Click the [New] button in the Data Transfer management dialog box.

General	
Comme	ent :
Attribute	
Address ty	pe : Bit Thterval : 3.0 second(s)
No. of	bit: 1
	Active only when designated window opened
Source addres	55
	ss MODBUS RTU
	MODBUS RTU   Setting
PLC name :	MODBUS RTU   Setting
PLC name : Address : Destination ad	MODBUS RTU   Setting
PLC name : Address :	MODBUS RTU   Setting
PLC name : Address : Destination ac	MODBUS RTU   Setting  Ox  I  Gal  Gal  HMI  Setting  Setting  Setting

Setting	Description
Attribute	Address type
	Select the data type, either [Bit] or [Word].
	No. of bits /No. of words
	When [Bit] is selected in [Address type], set the number of bits
	transferred each time when data transfer is triggered.
	When [Word] is selected in [Address type], set the number of
	words transferred each time when data transfer is triggered.
	Interval
	Select the time interval of data transfer, for example, when 3
	seconds is set, the system will transfer data every 3 seconds.
	Specifying a short time interval or a big number of data to transfer
	may cause an overall performance of system decrease. Therefore, it
	is recommended that users choose a longer time interval and a
	smaller amount of data to transfer.
	When a short interval is inevitable, be aware of the interval must
	be longer than the data transfer operation. For example, if the data
	transfer operation takes 2 seconds, set the interval longer than 2
	seconds.
Source	
address	Data Transfer object reads the data from [Source Address].

Destination	Data Transfer object writes the data to [Destination Address].
address	Data fransier object writes the data to [Destination Address].

## 13.26.2.2. Data Transfer Bit Trigger

#### **General Tab**

Click the [New] button in the Data Transfer management dialog box, and open [Bit trigger] tab.

Comme	nt :			
Source addres	s			
PLC name :	MODBUS RTU		•	Setting
Address :	3x	▼ 1		
Destination ad	dress			
PLC name :	Local HMI		•	Setting
Address :	LW	▼ 0		
Attribute				
No. of wo	ord: 1			
Trigger addres	s			
Trigger mo	de : ON->OFF	•		
PLC name :	Local HMI		•	Setting
Address :	LB	• 0		

Setting	Description	
Source address	Data Transfer object reads the data from [Source Address].	
Destination address	Data Transfer object writes the data to [Destination Address].	
No. of word	Set the number of words transferred each time when data transfer is triggered.	
Trigger	Set the register that controls data transfer and select the trigger	
address	mode. Trigger mode	
	Trigger data transfer when the state of the designated register changes from Off to ON, ON to OFF, or at both of the changes of	
	state.	



## 13.27. PLC Control

#### 13.27.1. Overview

PLC Control object can execute commands when it is triggered.

#### 13.27.2. Configuration



Click the PLC Control icon on the toolbar to open the PLC Control Object management dialog box. To add a PLC Control object, click [New], set up the properties, press OK button and a new PLC Control object will be created.

PLC Control Object		
2 : [Local HMI : 3 : [Local HMI : 4 : [Local HMI : 5 : [Local HMI :	LB-8999] => Sound control : OFF->ON, PLC no response (67 k) LW-100] => Change window (clear data after window thanged) LW-110] => Write data to PLC (current base window ID) LW-120] => General PLC control LU-120] => Change window (clear data after window changed) LB-10] => Execute macro program : [ID:000] macro_0 (OFF->ON) (active on Window 4)	
New	Delete Settings	Exit

Click [New] and the following dialog box appears. See "13.27.2.1 Type of Control".

PLC Control	
Comment :	
PLC name :	Local HMI 🔹
Attribute	
Type of control :	Change window 🔹
Active only w	Change window Write data to PLC (current base window ID)
Turn on back	General PLC control Back light control (write back)
Use window r	Back light control Sound control Execute macro program
Trigger address	Screen hardcopy
PLC name : Loca	I HMI
Address : LW	▼ 10 16-bit Unsigned
	OK Cancel

# Note

The [PLC Control] and [Backlight Control] options are not available for cMT Series.



## 13.27.2.1. Type of Control

#### • Change window

	PLC Control	
	Comment :	
	PLC name : Local HMI	
	Attribute	
	Type of control : Change window	
	Active only when designated window opened 4. Common Window	
	Turn on back light V Clear data after window changed	
	Use window no. offset	
	Trigger address	
	PLC name : Local HMI Setting	
	Address : LW   0 16-bit Unsigned	
	OK Cancel	
Setting	Description	
Active only when		
designated	Allow this operation only if a particular screen is displayed.	
window opened		
Turn on back light	The backlight is turned ON when the window object is	
	changed. (Not available for cMT Series)	
Clear data after	Reset the value at trigger address to zero when the window	
window changed	object is changed.	
Use window no.	Select the check box and select a window offset, the new	
offset	window no. to change to will be the value in [Trigger address]	
	plus the offset. For example, if [Trigger address] is LW-0 and	
	offset is set to -10. When the value in LW-0 is 25, the system	
	will change to window no. 15 (25-10=15). The range of the	
	offset is -1024 to 1024. The [Clear data after window changed]	
	check box is not available if [Use window no. offset] is selected.	

# Note

If [LB-9017] is set ON, the write-back function will be disabled, the new window number is not written back into a designated address.

Place a valid window number in the designated trigger address to change the base screen to the new window number. The new window number is written back into the designated address.

For example, if current window is window no. 10, and [Trigger address] is set to LW-0, When LW-0 is changed to 11, the system will change the current window to window no. 11,



Objects

#### and then write 11 to LW-1.

When the window is changed, the new window number is written back into the address that is calculated by [Trigger address] and the data format, as shown in following table.

Data Format	Trigger address	Write address
16-bit BCD	Address	Address + 1
32-bit BCD	Address	Address + 2
16-bit Unsigned	Address	Address + 1
16-bit Signed	Address	Address + 1
32-bit Unsigned	Address	Address + 2
32-bit Signed	Address	Address + 2

#### • Write data to PLC (current base window)

Each time the base window is changed, the new window number will be written into the [Trigger address].

#### • General PLC Control (eMT, iE, XE, mTV)

Transfer word data blocks from PLC to HMI, and vise-versa, and the transfer direction is controlled by the value in the [Trigger address].

Value in	Action
[Trigger address]	
1	Transfer data from PLC register $ ightarrow$ HMI RW register
2	Transfer data from PLC register $ ightarrow$ HMI LW register
3	Transfer data from HMI RW register $ ightarrow$ PLC register
4	Transfer data from HMI LW register $ ightarrow$ PLC register

Four consecutive word registers are used as described in the following table:

ose	Description
	Description
mine the	The valid values are listed in the
tion of data	above table. When a new control
fer	code is written into the register,
	HMI will start to transfer. After data
	transfer is finished, the value will
	be set to 0.
ize of data	The unit is word.
nsfer.	
t to the	Assume the value is "n", where n is
address of	an arbitrary number, the start
egister	address of PLC register is [Trigger
	mine the tion of data fer ize of data nsfer. t to the address of egister



	address + 4 + n].
	Take an OMRON PLC as an
	example:
	If [Trigger address] uses DM-100,
	[Trigger address + 2] will be
	DM-102. If the value in DM-102 is
	5, the start address of data source
	would be DM-109 (100 + 4 + 5 =
	109).
Offset to the	Take OMRON PLC as an example:
start address of	If set [Trigger address] to DM-100,
LW or RW	[Trigger address + 3] will be
memory in HMI	DM-103. If the value in DM-103 is
	100, the start address of memory
	in HMI is RW-100 or LW-100.
	start address of LW or RW

## Example 1

To use PLC Control object to transfer 16 words data in OMRON PLC, starting from address DM-100, to the HMI address, starting from RW-200. The setting is shown below:

- Firstly, create a PLC Control object, set [Type of control] to [General PLC control], and set [Trigger address] to DM-10, that is, to use the four sequential registers start from DM-10 to control data transfer.
- Confirm the data size and the offset addresses.
   Set DM-11 to 16, since the number of words to transfer is 16 words.
   Set DM-12 to 86, which indicates the address of data source is DM-100 (100=10+4+86).
   Set DM-13 to 200, which indicates the destination address is RW-200.
- Set DM-10 according to the direction of data transfer.
   If set DM-10 to 1, the data will be transferred from PLC to HMI RW register,
   If set DM-10 to 3, the data will be transferred from HMI RW register to PLC.
- Back light control (write back) (eMT, iE, XE, mTV)

When [Trigger address] is turned ON, HMI backlight will be turned ON/OFF and [Trigger address] will be turned OFF. Any touch on the screen will turn the backlight on.

Back light control (eMT, iE, XE, mTV)

When [Trigger address] is turned ON, HMI backlight will turn ON/OFF and the state of [Trigger address] will not be changed.



#### Sound control

When the state of the designated [Trigger address] changes, the HMI will play the sound selected from the sound library. There is a further selection determines whether the sound is played after Off to ON, ON to OFF transition, or at both of the changes of state.

#### • Execute macro program

Select a pre-defined Macro from the drop-down list. When the state of the designated [Trigger address] changes, the selected Macro is executed. There is a further selection determines whether the Macro is executed after Off to ON, ON to OFF transition, or at both of the changes of state. If select [Always active when ON], the macro will be executed repeatedly. (The shortest time interval between runs is 0.5 second.)

Screen hardcopy

PLC Control
Comment :
PLC name : Local HMI 🔹
Attribute
Type of control : Screen hardcopy 🔹
Active only when designated window opened
Rotate image 90 degrees
Trigger address
PLC name : Local HMI
Address : LB 🗸 0
Screen hardcopy Trigger mode : OFF->ON
Source window for print
O Current base window
Printer : SD card
Customized filename handling
OK Cancel

When the state of the designated [Trigger address] changes, print the selected screen. If select [SD card] or [USB disk] as [Printer], a "hardcopy" folder will be generated in the selected external device for saving the printed screen in JPG format. The name of the JPG files starts from yyymmdd 0000.

To print the screen using a printer, go to System Parameter Settings » Model tab and set the printer.

To print the screen using a remote printer, go to System Parameter Settings » Printer/Backup Server tab and configure the parameters.

There are three options to specify the source window for hardcopy:



#### **Current base window**

Print the base window currently opened.

#### Window no. from register

Print the window designated by the value in a designated word address.

#### Designate window no.

Directly select a window to be printed.

#### **Customized File Handling**

This feature can be used to customize naming of the folders and the JPG files.

	-		
Setting	Description		
Folder name	The folder name can be an alphanumeric name, and		
	certain half-width symbols are allowed:		
	!@#\$%^&()_+{}`-=;',.		
	The folder name can also be specified by a naming syntax.		
	Dynamic format		
	The folder names can be set by a designated word		
	address, or by a naming syntax indicating the current		
	system time. The syntax can be specified by selecting		
	time buttons or entering the syntax in Format field. The		
	length limit is from 1 to 25.		
	Note: Up to 10 layers of folders can be created. The		
	exceeding layers will be ignored.		
File name	The way to specify a file name is similar to specifying a		
	folder name.		
	Note: If the file name already exists, the system will add		
	"_0001" to the file name as a serial number. The numbers		
	of the later files add up until "_9999". The files after		
	"_9999" will be ignored.		
	For example, if the three file names exist: "A.jpg",		
	"A_0001.jpg", "A_0003.jpg". When trigger screen		
	hardcopy with the same file name, the coming files will		
	be named in this order: "A_0002.jpg", "A_0004.jpg",		
	"A_0005.jpg", and so on.		

# Note

The [Printer] setting is not available for cMT Series. The window hardcopy file is saved in iPad Photo folder.



- A background printing procedure is performed when the printed window is not the current base window.
- If the hard-copied window is a background window, its [Direct Window] and [Indirect Window] objects will not be printed.
- When using a dynamic format name, the system will use a "_" sign as a substitute for invalid symbols.
- When using a dynamic format name, if screen hardcopy is triggered without specifying the format first, the system will save the file in the default directory, which is: hardcopy\yymmdd_0000.JPG



# 13.28. Scheduler

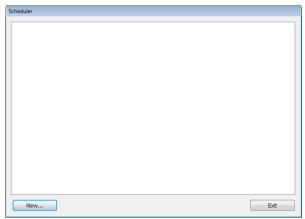
#### 13.28.1. Overview

Scheduler object turns bits ON/OFF, or writes values to word registers at designated start times. It works on a weekly basis.

#### 13.28.2. Configuration



Click the Scheduler icon on the toolbar to open the Scheduler management dialog box, click [New] to open the Scheduler property setting dialog box.

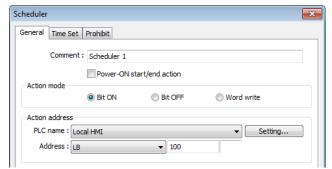


The following two demonstrations explain the usage of Scheduler.

## **Example 1**

A motor is scheduled to power - ON at 9:00 and power – OFF at 18:00, Monday to Friday. We are using LB-100 to control the motor state. LB-100 will be set ON at 9:00 and OFF at 18:00.

- Click the Scheduler icon on the toolbar to open the Scheduler management dialog box, click [New].
- 2. In [General] tab, select [Bit ON] in [Action mode] and set [Action address] to LB-100.



3. In [Time Set] tab, select [Constant].



	_
Scheduler	-
General Time Set Prohibit	_
Constant     O Address	
Setting on individual day	
9 💌 : 0 💌 : 0 💌 (HH:MM:SS)	
Sun 🖉 Mon 🖉 Tue 🕼 Wed 🗭 Thu 🖉 Fri 📄 Sat	
End	
Caracteria Enable termination action	
18 🔹 : 0 🔹 : 0 🔹 (HH:MM:SS)	

- Enter [Start] time as 9:00:00 and select Monday to Friday. Do not select [Setting on individual day].
- 5. Enter [End] time as 18:00:00 and select [Enable termination action] check box.
- 6. Click [OK], a new Scheduler object will be created on the [Scheduler] list.

## Example 2

A thermal heater is scheduled to heat up to 90°C at 08:00 and cool down to 30°C at 17:00, Monday to Friday. LW-100 is used to store the set point value.

- Click the Scheduler icon on the toolbar to open the Scheduler management dialog box, click [New].
- 2. In [General] tab, select [Word write] in [Action mode] and set [Action address] to LW-100.
- 3. Select [Constant] for [Word write value settings] and enter 90 in [Start value].

eneral	Time Set	Prohibit			
(	Comment :	Scheduler 2			
		Power-ON s	tart/end action		
Action n		Bit ON	Bit OFF	() Word	write
Action a	ddress				
	ddress ame : Loca	il HMI		•	Setting
PLC n		I HMI	▼ 100	<b></b>	Setting 16-bit Unsigned
PLC n Add	ame : Loca ress : LW		▼ 100	<b></b>	
PLC n Add	ame : Loca			<b></b>	
PLC n Add Word w	ame : Loca ress : LW	ettings Constant		-	

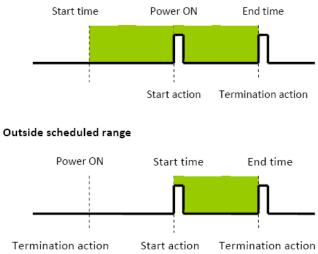
- 4. In [Time set] tab select [Constant].
- 5. Enter [Start] time as 8:00:00 and select Monday to Friday. Do not select [Setting on individual day].
- 6. Enter [End] time as 17:00:00 and select [Enable termination action] check box.
- 7. Return to [General] tab and enter 30 in [End value].
- 8. Click [OK], a new schedule object will be created on the [Scheduler] list.



#### **General Tab**

	t Prohibit		
Commen	t : Scheduler 2		
	V Power-ON st	art/end action	
Action mode	Bit ON	Bit OFF	Word write
Action address			
PLC name :	ocal HMI		▼ Setting
Address :	W	▼ 100	16-bit Unsigned
Word write valu	e settings		
Word write valu	-	Address	
Word write valu	Onstant	Address	

Setting	Description			
Power ON	Execute the defined action when the HMI is powered ON.			
start/end	<ul> <li>Enabled</li> </ul>			
action	When HMI is powered ON within the scheduled time range, the			
	start action will be performed automatically. When HMI is powered			
	ON outside the scheduled time range, th	e termination action will		
	be executed.			
	Inside scheduled range			
	Start time Power ON End	time		

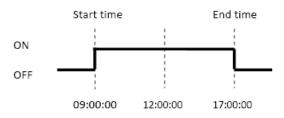


Disabled

When the HMI is powered ON at a time later than the start time, the start action will not be performed, but the termination action will be performed. When the termination action is not defined, the scheduled range is not recognized and no action is performed.



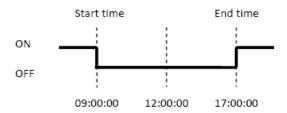
# Action modeChoose the action to do at the given time.Bit ONAt the start time, set the designated bit ON. At the end time, set<br/>the designated bit OFF.Example: Start time : 09:00:00 End time : 17:00:00



#### Bit OFF

At the start time, set the designated bit OFF. At the end time, set the designated bit ON.

Example: Start time: 09:00:00 End time: 17:00:00



#### Word write

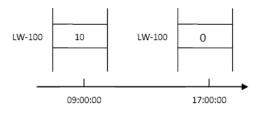
The [Write start value] entered here is transferred to the designated [Action address] word register at the start time. At end time, the [Write end value] entered here is written to the [Action address]. The valued can be entered manually or be set by using [Address] mode. In [Address] mode, the value in the specified address is the start value where the value in [Address + 1] is the end value.

Example: Device address: LW-100

Start time: 09:00:00 End time: 17:00:00

Write start value: 10 Write end value: 0

Use register: If control address is LW-n, then enter 10 in LW-n and enter 0 in LW-(n+1).





Only is an [End time] is set in the [Time set] tab will the [Write end value] box appear.

#### Time Set

Specify start time and end time. [Constant] allows specifying a date or period and time. [Address] allows controlling the time by the designated address.

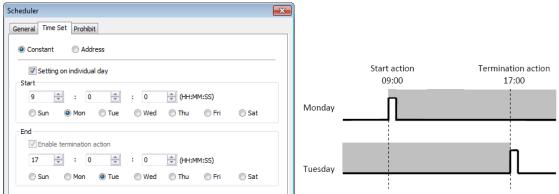
Scheduler	Scheduler
General Time Set Prohibit	General Time Set Prohibit
Constant     O     Address	Constant   Address
Setting on individual day	Time setting address
Start 9 🐨 : 0 🐨 : 0 🐨 (HH:MM:SS)	PLC name : Local HMI   Address : LW  Address
Sun V Mon V Tue V Wed V Thu V Fri Sat	Audress : LW V
	Control: LW:0
Enable termination action	Status: LW:0+1
	Action mode : LW : 0 + 2
	Start time (day) : LW : 0 + 3
	Start time (hour) : LW : 0 + 4
	Start time (minute) : LW : 0 + 5
	Start time (second) : LW : 0 + 6
	End time (day) : LW : 0 + 7
	End time (hour) : LW : 0 + 8
	End time (minute) : LW : 0 + 9
	End time (second) : LW : 0 + 10
OK Cancel Hel	p OK Cancel Help

#### Constant

## Setting on individual day

If [Setting on individual day] is selected

The same start time and end time can be assigned to different days of the week.





- Start and end time must be entered.
- Start and end time must be on a different time, or same time but different day.

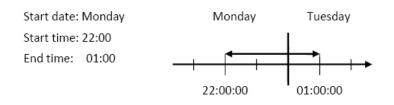
## If [Setting on individual day] is not selected

Start time and end time entered must start and end within 24-hours.

Scheduler			
Constant     Address		Start action	Termination action
Setting on individual day Start 9 - 0 - : 0 - (HH:MM:SS) Sun V Mon V Tue Wed Thu Fri Sat	Monday –		
End Finable termination action 17 A : 0 A : 0 A (HH:MM:SS)	Tuesday _		h

# Note

- Start time and end time must be on a different time, different day.
- If an end time is earlier than a start time, the end action will occur in the next day.



## Address

The scheduler object retrieves the start/end time and day of week information from word registers, enabling all parameters to be set and changed under PLC or user control. Designated as the top address in a block of 11 sequential registers which are used to store time

setting data. The format of the 11 word registers should normally be 16-unsigned integer. If a 32-bit word

address is chosen, only bits 0-15 are effective, and bits 16-31 should be written as zero. The following describes each register.

## Control (Time setting address + 0)

When [Control] bit is ON, the HMI will read and update [Action mode], [Start time], and [End time] values.

15	0	Bit
Reserved (0 fixed)	0	

Bit 0: no action 1: read times/action mode



HMI will not regularly read the data from [Action mode] (address + 2) to [End time] (address + 10). Please turn [Control] ON when the settings are changed.

## Status (Time setting address + 1)

When the read operation is completed, Bit00 of this register turns ON. If time data read is out of range or incorrect in any way Bit01 turns ON.

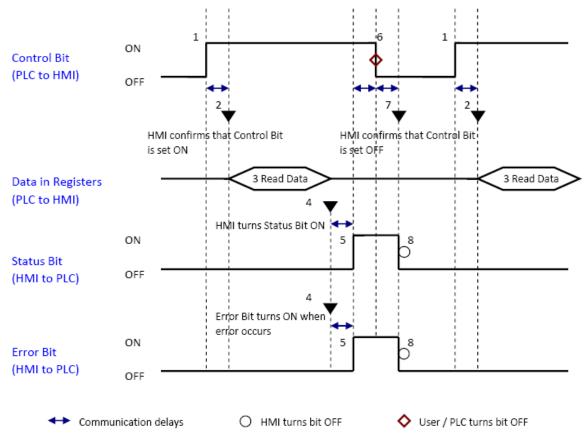
15	02	01	00	Bit
Reserved (0 fixed)		0	0	

Bit 00: Status bit: Read operation completed. (0: reading or reading not started. 1: reading completed.)

Bit 01: Error bit: Start or end time format incorrect. (0: corrected 1: error)

# Note

After the scheduler reads the data and the status is turned ON (The value in [Address + 1] = 01), the control bit must be turned OFF (address = 0). The status bit and error bit will be turned OFF  $(1 \rightarrow 0)$  at the same time.





## Action mode (Time setting address + 2)

Enable/disable [Enable termination action] and [Setting on individual day]. Whatever the [Enable termination action] bit is, all the time data, from [Control] to [End time (second)], will be read.



Bit 00 Enable termination action (0: Disabled 1: Enabled) Bit 01 Setting on individual day (0: Disabled 1: Enabled)



- If [Enable termination action] is OFF, all 11 registers are still read but end time is ignored.
- If [Setting on individual day] is ON, make sure that all start end times are entered. If more than one start / end day bit is ON, and error will occur.

Start/End Day (Start Day: Time setting address + 3, End Day: Time setting address + 7)

Designates which day of week is used to trigger the start or end action.

15		07	06	05	04	03	02	01	00	Bit
Reserved (0 fixed)			Sat	Fri	Thu	Wed	Tue	Mon	Sun	
Bit 00 Sunday	(0: not used 1: used)									
Bit 01 Monday	(0: not used 1: used)									
Bit 02 Tuesday	(0: not used 1: used)									
Bit 03 Wednesda	ay (0: not used 1: used	)								
Bit 04 Thursday	(0: not used 1: used)									
Bit 05 Friday	(0: not used 1: used)									
Bit 06 Saturday	(0: not used 1: used)									

Start/End Time (Start Time: Time setting address + 4 to + 6, End Time: Time setting address + 8 to + 10)

Hour: 0 – 23 Minute: 0 – 59 Second: 0 - 59

Values outside these ranges will cause error.

# Note

- 16-bit unsigned integer format must be used; BCD format is not supported here.
- End time depends on [Action mode] (address + 2). [Enable termination action] (Bit 00) and [Setting individual day] (Bit 01) are related:

Setting individual day	Enabled	Disabled	
Enable termination action	Enabled	Enabled	Disabled



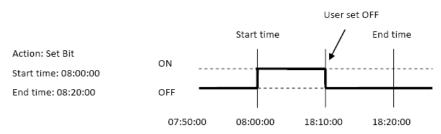
## Prohibit tab

heduler					X
General Time Set Prohibit					
Prohibit					
Use prohibit function					
					-
PLC name : Local HMI			-	Setting	
Address : LB	-	0			

Before the scheduled action is performed, the HMI will read the specified bit state. If it is ON, the scheduled start/end action will be skipped. Otherwise, it will be performed normally.

## Note

- Up to 32 scheduler objects are allowed.
- A time schedule applies one action only when the start time is reached.



- [Write start/end value] and [Prohibit] bit is read only once before start action. After that, even to change the state of [Prohibit] bit or [Write start/end value], the end action and the value written will not be affected. Also, to read data of [Write start/end value] and [Prohibit] bit, there is a delay of start action due to the communication.
- Each time RTC data is changed, scheduler list entries that possess both start and end times will be checked for in-range or out-range conditions. For in-range, the start action will occur. If the end action is not set, the new range is not recognized, the action will not occur.
- If several Scheduler objects are set to the same start time or end time, the action is performed in ascending order of the schedule number.
- In [Time Set] » [Address] mode, the system will read [Control] word regularly. The length of the period depends on the system.
- In [Time Set] » [Address] mode, when start time and end time is out- range, error occurs in the set action time. (Note: BCD is not an acceptable format)
- In [Time Set] » [Address] mode, the action will not start up until the first time the time data is successfully updated.

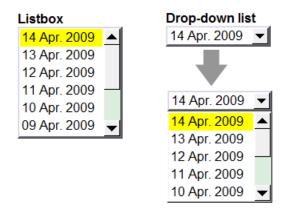


## 13.29. Option List

## 13.29.1. Overview

Option List object displays a list of items that the user can view and select. Once the user selects an item, the corresponding data will be written to a word register.

There are two forms of this object – [List box] and [Drop-down list]. The [List box] lists all items and highlights the selected one. The [Drop-down list] normally displays only the selected item. Once the object is pressed, the system will display a list (which is similar to list box) as shown in the following figure.



## 13.29.2. Configuration



Click the Option List icon on the toolbar to open an Option List object property dialog box. Set up the properties, press OK button, and a new Option List object will be created.



### Objects

## **General Tab**

	New Option List Object	
	Option list Mapping Security Shape Label	
	Attribute	
	Mode : List box  Background :  V Item no. : 1 Selection :  V	
	Source of item data : Predefine	
	Monitor address PLC name : Local HMI	
	Address : LW	
	Write when button is released	
	Send notification after writing sucessfully	
	Setting LB-0	
	OK Cancel Help	
Setting	Description	
Attribute	Mode: The list style, either [List box] or [Drop-down list].	
	Item no.: Set the number of items for the object. Each item	
	represents a state displayed in the list and the corresponding value	5
	will be written to the [Monitor address].	
	Background: Set background color.	
	Selection: Set background color for the selected item.	
	Source of item data: There are 4 sources available: [Predefine],	
	[Dates of historical data], [Item address], and [User account]. See	
	13.29.2.1	
Monitor	The corresponding value of the selected item will be written to	
address	[Monitor address].	
	Write when button is released	
	If this check box is selected, the selected item value will be written	ı
	to [Monitor address] after the button is released.	
	נס נאוסווונסי מעמיבשן מונרי נווב שעננטוו ש ובובמשבע.	



Send	
notification	Set On/Off the designated bit address after successfully writing
after writing	data to PLC.
successfully	

# Note

For cMT Series, the [Dates of historical data] and the [write when button is released] selections are not available.

## 13.29.2.1. Source of item data

## • Predefine

The list is manually defined in [Mapping] tab.

The number of items can be adjusted by [Item no.], and each item represents one state. Each item has a corresponding value which will be written to [Monitor address].

## Dates of historical data

This selection is not available for cMT Series.

New Option List Object
Option list Mapping Security Shape Label
Comment :
Mode : List box   Background :
Selection :
Source of item data : Dates of history data 🔹
📝 Enable [delete history files] function
Monitor address
PLC name : Local HMI    Setting
Address : LW 🔽 19 16-bit Unsigned
Write when button is released
Item data from dates of historical data
Type : Data Sampling 👻 Date : MM/DD/YY 💌
Data Sampling object : 1.
Control address
PLC name : Local HMI
Address : LW 🗸 0
[Address] : set 1 to delete the selected history data
Send notification after writing successfully
Enable

Option List object can be used with historical data display objects, such as Trend Display object, History Data Display object and Event Display object to control which history file should be



$ \longrightarrow $	
06/11/09 👻	p6/11/09.
06/11/09	᠘᠕ᡗᡐᢦ᠕᠕ᡣᡐ᠕᠕᠉ᡐ᠕᠕᠕ᡐᢦ᠕᠕ᡣᡐ᠕᠕᠕ᡐᢦ᠕᠕
06/10/09	KANNENAKENAKENA
06/09/09	A BARRY VIEWAAAAA NA DADAY VIRLAADAY SU AAAAY DUDLAADA
06/08/09	NATE OF ANY AND A FAMILY AND A DAMAGE A MANAGE AND A DAMA AND A FA
06/06/09	<b>1</b> 1 / 1
	$13{:}14{:}18  13{:}14{:}38  13{:}14{:}58  13{:}15{:}18  13{:}15{:}38  13{:}15{:}58  13{:}16{:}18$
	H H I H

shown. The figure below is an example of Option List used with Trend Display.

Setting	Description
Туре	Two options are available: [Event (Alarm) log] and [Data sampling].
Date	Set the date format. YYYY means a four digits year (EX: 2012), YY
	means a two digits year (EX: 12), MM means month and DD means
	day.
Data	Select which Data Sampling object is displayed when [Type] is [Data
Sampling	Sampling], and it should be the same as the [Data sampling object
object	index] configured in [Trend Display] or [History Data Display].
Enable	
[delete	If selected, a control address can be set. Writing "1" to this address
history data]	will delete the history data of the specified date.
function	

# Note

- In [Dates of historical data] mode, since the system automatically reads the historical data and finds the date information, it is not necessary to configure in the [Mapping] tab.
- The error message displayed in Option List can be modified in [Mapping] tab.

Item	Value	Item data
0 (error)		Error!!

## Item address

The list will be read from the given [Item address] and controlled by [Control address]. The following options will be available:



Option list	Mapping	Security	Shape	Label	Profile			
		nment:						
Attribute		List box		•	Backgroun	nd :		<b>_</b>
					Selectio	on : 🗖		
	Source o	fitem dat	a : Item	address				•
-Monitor a	ddress							
PLC na	ame : Loca	al HMI				•	Setting.	
باداده								
Addr	ress : LW						16-bit Unsigr	ned
Addr	ess : LW	Write	when but				16-bit Unsigr	ned
Control a		Write	when but				16-bit Unsig	ned
Control a			when but				16-bit Unsign	
Control a PLC na	ddress		when but		eased			
Control a PLC na Addr	ddress ame : Loca	al HMI		ton is rel	eased			
Control a PLC na Addr	ddress ame : Loca ress : LW : set 1 to	al HMI		ton is rel	eased		Setting.	
Control a PLC na Addr [Address] Item addi	ddress ame : Loca ress : LW : set 1 to	al HMI update ite		ton is rel	eased	ss] + 1 : i	Setting.	
Control a PLC na Addr [Address] Item addi Rev UNI	ddress ame : Loca ress : LW : set 1 to ress erse high/	al HMI update ite		ton is rel	eased	ss] + 1 : i	Setting.	 RD(s)

Setting	Description
Control	[Address]: If the value at this address is changed to 1, the option
address	list would be replaced by items defined at [Item address]. After
	updating, the value will be restored to 0.
	[Address + 1]: Define the number of items in [Item address].
Item address	Assign the item address
	UNICODE
	The item will use UNICODE characters, such as Chinese characters.
	The length of each item
	Define the number of letters for each item, the unit is Word.

# Note

- The UNICODE characters used here should be used by Text object, so that EasyBuilder Pro will compile the needed fonts and download these fonts to HMI, then the UNICODE letters can be correctly displayed.
- [The number of items] multiplied by [The Length of each item] must be less than 1024 words.
- The system automatically disables [Mapping] tab in [Item address] mode.



## • User account

If [Enhanced Security] mode is enabled, [User account] would appear in the [Source of item data] and it lists the names of users.

ſ	Option List Object's Properties
	Option list       Mapping       Security       Shape       Label       Profile         Comment :       Attribute       Mode :       List box       Background :       Image: Comment :       Image
	Monitor address PLC name : Local HMI Address : LW 0 16-bit Unsigned Write when button is released
Setting	Description
<b>a</b> .	

Sort	Select the sorting method from [Ascending] of [Descending].	
Display If [Privilege] is selected, the privileges for each user will be		
	displayed in option list.	
	If [Secret user] is selected, even though it is defined to be hidden in	
	[System parameter settings] » [Security] » [Enhanced Security], the	
	users will still be displayed in [Option List].	

## Note

The address that controls user index is [Control Address +2 (LW-n+2)] which is set in [System Parameters] » [Security] » [Enhanced Security].

## **Mapping Tab**

This table displays all available states/items, their item data and values. To change the number of available items, please go to [Option list tab] » [Attribute] » [Item no.].



ion list Map	ping Security Shap	e Label	
Joi i i i i i i i i i j	pang becurry prinap		
Item	Value	Item data	-
0	0	Black Coffee	-
1	1	Latte	
2	2	Black Tea	
3	3	Cappuccino	
4	4	Vanilla Latte	
5	5	Mattya	
6	6	Green Tea	
7	7	OoLong Tea	
8	8	Fruit Tea	
9	9	Scented Tea	
10 (error)			
			Set defau
Impo	rt item data from recipe	e record	

Setting	Description
Item	The system lists all available items. Each item represents a state that will be displayed in the list. This field is read-only.
Value	Here user can assign value for each item, basing on the following two criteria:
	For reading: If the value in [Monitor address] is changed, the object selects the first-matched item. If no item is matched, the status goes to error state and signals the notification bit register (if requested).
	For writing: The system writes this value to [Monitor address] when user selects an item.
Item data	Text displayed for each item. The Option List object displays the text of all items in the list for users to review and select.
Import item	This feature is enabled when select [Recipe-Selection] as [Monitor
data from	address]. Click [Import item data from recipe record] to open the
recipe record	[Recipe Records] setting dialog box. Select [Item data source], the data belonging to the selected column will all be imported to Option List object.

ecipes : Drinks (10)	Ac	d Delete	It	em data source :	Item	-
ninks (10)		Item	Coffee	Tea	Barcode Calories	
	1	Black Coffee	225	0	Coffee	
	2	Latte	150	0	Item	
	3	Black Tea	0	130	LH Price	1
	4	Cappuccino	150	0	Protein	
	5	Vanilla Latte	150	0	Sugar	
	6	Mattya	0	0	Tea	
	7	Green Tea	0	130	13	
	8	OoLong Tea	0	130	8	
	9	Fruit Tea	0	100	15	
	10	Scented Tea	0	120	36	
		m				

Before importing, the number of items in Option List changes



	according to the number of items defined in Recipe Records.		
	After importing, modifying Recipe Records will not change the		
	content of Option List.		
Error state	On error state, the list box removes the highlight to represent no item is selected and the drop-down list displays the data of error state. Only the drop-down list uses error state, list box is not able to use error state.		
	For example, item number 8 is the error state when specifying 8 in [Item no.]. (The first item number is 0)		
Set default	Reset all values or states to default. That is, set 0 for item 0, 1 for item 1, and so on.		
Error	The system will set ON/OFF to the specified bit register when error		
notification	is detected. The signal of the bit register could be used to trigger a procedure for correcting the error by using objects such as Event Log, Alarm Bar, or pop-up window.		



## 13.30. Timer

#### 13.30.1. Overview

Timer object is a switch that can be used to control the mode to count time. The modes are explained later. Timer object uses the following 6 variables:

Timer Variable	Туре	Description
Input bit (IN)	Bit	The main switch of Timer.
Measurement bit (TI)	Bit	Turns ON when the Timer begins
		counting time.
Output bit (Q)	Bit	Activated when the Timer finishes
		counting time.
Preset time (PT)	Word	Presets a time before the Timer
		begins counting time.
Elapsed time (ET)	Word	Displays the elapsed time.
Reset bit (R)	Bit	Resets the elapsed time (ET) to 0.

## 13.30.2. Configuration



Click the Timer icon on the toolbar to open the property dialog box as shown in the following figure.

Timer	8
Description :	
Mode : On delay	Input bit (IN)         PLC name : Local HMI          Setting           Address : LB         0         0         0
	Output bit (Q) PLC name : Local HMI   Address : LB  Measurement bit (TI) PLC name : Local HMI  Address : LB  2
Elapsed time (ET)	Preset time (PT)
PLC name : Local HMI	PLC name :         Local HMI         ▼         Setting           Address :         LW         ▼         10         16-bit Unsigned
L	OK Cancel Help





# Note

[Constant preset time] is only available for cMT series.

If use cMT Series, clicking the Timer icon on the toolbar will open the Timer managing window, click [New] to configure.

1	Timer
	0: Mode : On delay, IN : [Local HMI : LB-0], TI : [Local HMI : LB-2], Q : [Local HMI : LB-1]
	New Delete Settings Exit

• On delay

Mode	Register
	Input bit (IN): The main switch of Timer.
IN	Measurement bit (TI): Turns ON when the Timer
	begins counting time.
	Output bit (Q): Turns ON when the Timer finishes
	counting time.
1 2 3 4 5	Preset time (PT): Presets a time before the Timer
	begins counting time.
	Elapsed time (ET): Displays the elapsed time.
	Description

**Period 1**: When the IN turns ON, TI turns ON and the ET starts counting. The Q remains OFF. **Period 2**: When the ET equals to the PT, the TI turns OFF and the Q turns ON.

**Period 3**: When the IN turns OFF, the Q turns OFF and the ET is reset to 0.

**Period 4**: When the IN turns ON, the TI turns ON and the ET starts counting. The Q remains OFF.

**Period 5**: Turns IN OFF before the ET reaches the PT, the TI turns OFF, and the ET is reset to 0. Since the ET doesn't reach the PT, the Q remains OFF.



## Off delay

Mode	Register	
	Input bit (IN): The main switch of Timer.	
IN	Measurement bit (TI): Turns ON when the Timer	
	begins counting time.	
	Output bit (Q): Turns OFF when the Timer finishes	
Q	counting time.	
123456	Preset time (PT): Presets a time before the Timer	
	begins counting time.	
	Elapsed time (ET): Displays the elapsed time.	
Description		
Period 1: When the IN turns ON, the TI r	emains OFF and the Q turns ON, the ET is reset to 0.	
Period 2: When the IN turns OFF, the TI turns ON and the Q remains ON, the ET starts		
counting.		
<b>Period 3</b> : When the ET equals to the PT, the Q and TI turn OFF.		
Period 4: When the IN turns ON, the TI remains OFF and the Q turns ON, the ET is reset to 0.		
Period 5: When the IN turns OFF, the TI turns ON and the Q remains ON, the ET starts		

counting.

**Period 6**: Turns the IN to ON before the ET reaches the PT, the TI turns OFF, the Q remains ON, and the ET is reset to 0,.

Pulse		
Mode	Register	
	Input bit (IN): The main switch of Timer.	
	Measurement bit (TI): Turns ON when the Timer	
	begins counting time.	
	Output bit (Q): Turns ON when the Timer begins	
	counting time and turns OFF when the Timer	
1 2 3 4	finishes counting time.	
	Preset time (PT): Presets a time before the Timer	
	begins counting time.	
	Elapsed time (ET): Displays the elapsed time.	
	Description	

**Period 1**: When the IN turns ON, the TI and Q turn ON simultaneously, and the ET starts counting.

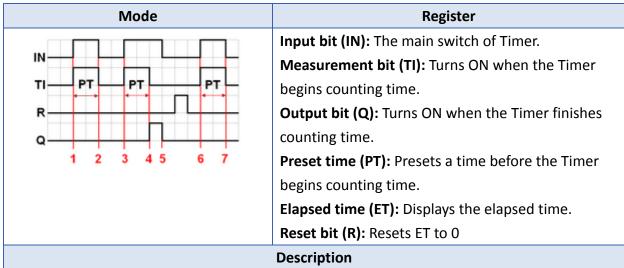
**Period 2**: When the ET equals to PT, the TI and Q turn OFF simultaneously. Since IN is turned OFF when counting time, the ET is reset to 0.

**Period 3**: When the IN turns ON, the TI and Q turn ON simultaneously, and the ET starts counting.

**Period 4**: When the ET equals the PT, the TI and Q turn OFF simultaneously.



## • Accumulated ON delay



**Period 1**: When the IN turns ON, the TI turns ON and the elapsed time ET starts counting, the Q remains OFF.

**Period 2**: When the IN turns OFF, if the ET doesn't reach the PT, the TI turns OFF, and at the same time the Q remains OFF. The ET is in the retentive state.

**Period 3**: When the IN turns ON, the TI turns ON. The timer measurement starts again and the ET starts counting from the kept value. The Q remains OFF.

**Period 4**: When the ET reaches the PT, the TI turns OFF and the Q turns ON.

**Period 5**: When the IN turns OFF, the Q turns OFF. Turning ON the reset bit R will reset the ET to 0, and then the reset bit turns OFF.

Accumulated OFF delay		
Mode	Register	
	Input bit (IN): The main switch of Timer.	
	Measurement bit (TI): Turns ON when the Timer	
	begins counting time.	
	Output bit (Q): Turns OFF when the Timer finishes	
	counting time.	
1 2 3 4 5 6 7 8 9 10	Preset time (PT): Presets a time before the Timer	
	begins counting.	
	Elapsed time (ET): Displays the elapsed time.	
	Reset bit (R): Resets ET to 0	
	Description	

**Period 1**: When the IN turns ON, the TI remains OFF and the Q turns ON.

**Period 2**: When the IN turns OFF, the TI turns ON and the Q remains ON. The ET starts counting.

**Period 3**: When the IN turns ON, the TI and Q remain ON, and the ET is in the retentive state. **Period 4**: When the IN turns OFF again, the ET starts counting from the kept value.

**Period 5**: When the ET equals to the PT, the TI and Q turn OFF simultaneously. Turning ON the reset bit R will reset the ET to 0, and then the reset bit turns OFF.



## 13.31. Video In and Video In (USB Camera)

#### 13.31.1. Overview

The eMT, XE, and mTV Series HMI models provide the Video Input feature. By installing a surveillance camera, user can monitor the site on HMI. The video images can be stored in external devices and then analyzed on PC. This feature can be utilized in different places for monitoring, such as vehicles or buildings.

eMT, XE, and mTV Series: Support USB camera video input.

eMT3120A/eMT3150A: Not only support USB camera video input but also NTSC and PAL analog video systems. For hardware, HMI provides 2 channels for Video Input. User can freely switch the channel, and capture real-time images even when Video Input is paused.

#### 13.31.2. Configuration



Click the Video In icon on the toolbar to open the property dialog box. Set up the properties, press OK button, and a new Video In object will be created.

**USB** Camera

#### **General Tab**

#### Analog Video Systems

New Video In Object	Video In (USB Camera) Object's Properties
General	General Profile
Comment :	Comment :
Input channel : 1   Encode format : NTSC	
Capture address	Capture address
Use capture function	V Use capture function
PLC name : Local HMI    Setting	PLC name : Local HMI
Address : LB V 0	Address : LB 🗸 0
Storage medium	Storage medium
Storage meaium	SD card
So card So bab usk	
Record time	
Before : 1 🚖 seconds After : 1 🚖 seconds	
	Control address
Control address	Use control function
Use control function	PLC name : Local HMI
	Address : LW 🔹 0 16-bit Unsigned
	Start/stop input : LW : 0
	Pause : LW : 0 + 1
	* OS version 2014.01.16 or later support USB camera only !
OK Cancel Help	OK Cancel Help



Setting	Description	
Input channel	Select the Video	Input channel from channel 1 or channel 2.
	(Analog video sy	/stems)
Encode format	Select the form	at from NTSC or PAL. (Analog video systems)
Capture	Select [Use capt	ure function] check box and configure the settings.
address	Capture addres	s
	Designate the a	ddress that triggers image capturing.
	Storage mediur	n
	Designate the st	orage device.
	System	Storage Device
	Analog video	Select SD card or USB disk to save the captured
	system	images. The images of channel 1 will be saved in
		"VIP1" folder in the chosen storage and so on.
	USB	Only saves the captured images in SD card.
	Camera	
	<b>Record time</b>	
	Set a period of t	ime to capture the images.
	System	Method
	Analog video	<ul> <li>The longest period can be set from 10</li> </ul>
	system	seconds before triggering [Capture address]
		to 10 seconds after triggering.
		• The time interval of image capturing is once
		every second.
		<ul> <li>The captured .jpg file will be named in the following format:</li> </ul>
		Before or after [Capture address] is
		triggered: YYYYMMDDhhmmss.jpg
		The moment that [Capture address] is
		triggered: YYYYMMDDhhmmss@.jpg
		For example, set [Record time] "Before" and
		"After" to "5" seconds. When the state of [Capture
		address] changes from OFF to ON, the system wil
		start capturing one image per second, from 5
		seconds before the triggering time to 5 seconds
		after the triggering time, which is 11 images in
		total including the one captured at the triggering
		moment.

## 13-164

	USB	Only t	the image of the triggering moment is
	Camera	captu	red. The name format:
		YYYYN	MMDDhhmmss.png.
Control	Use control fu	nction	
address	If enabled, ent	er certair	n value to the control address and the
			n control Video Input object. For example,
	_		address is LW-n (n is any address), enter
	-		ignated addresses will execute commands
	as the followin		
	Address	Value	Command
		-	Command
	LW-n	0	Stop displaying image
		1	Open channel 1 and display the
		2	image on HMI
		2	Open channel 2 and display the
		3	image on HMI Open channel 1 but don't display the
		5	image on HMI (Capture function
			operable)
		4	Open channel 2 but don't display the
			image on HMI (Capture function
			operable)
	LW-n+1	1	Pause / resume the video
	LW-n+2	1~100	Adjust the contrast ratio
			(Analog video systems)
	LW-n+3	1~100	Adjust the brightness
			(Analog video systems)
	<ul> <li>After cha</li> </ul>	inging the	e value in [Control address (LW-n)], the
	system w	vill keep t	he new value.
	<ul> <li>After cha</li> </ul>	inging the	e value in [Control address + 1 (LW-n+1)],
			ecute the command and then reset the
	value to (		
			ction] check box is not selected, the syste
	will play	une mage	e of the selected channel.

- If [Display adjustment] check box is selected, the contrast ratio and brightness can be adjusted. (Analog video systems)
- The USB Camera is controlled by LW-n (value 0 or 1) and LW-n+1.



# Note

## About analog video systems:

- Only one channel can be opened at a time.
- Real-time images can still be captured when Video In is paused.
- Recommended analog video systems and resolutions:

	1:1	50%
NTSC	720 x 480	360 x 240
PAL	720 x 576	360 x 288

## About USB Camera:

- If the USB Camera is removed during video playing, the image will not be loaded even though the USB Camera is returned. If [Control address] is selected, please stop and then restart video input. If [Control address] is not selected, please switch to another screen and then return, or restart HMI.
- The maximum size of Video In object of eMT3070A is 340*240, as for eMT3105P, eMT3120A, eMT3150A, XE, and mTV Series, the maximum size of Video in object is 640*480.
- When using USB Camera, the resolution of the run-time video image is determined by the resolution supported by the USB Camera that is closest to the size of Video In object. The resolution supported by the USB Camera may not be identical to the size of the object. The same resolution of Video In object and USB Camera image is recommended.
- When using USB Camera, the right and bottom edge of the Video In object will keep a distance of 50 pixels away from the window edge to prevent the run-time video image from exceeding the window.
- When using USB Camera, the background color of Video In object is black. If the resolution of the run-time video image is smaller than Video In object, the empty area is colored black. The same resolution of Video In object and USB Camera image is recommended.
- The tested and available USB Cameras are: Logitech C170, Logitech C310, Logitech C910, LifeCam VX-2000.
- The OS versions that support USB Camera:

Model	OS Version (or later)
eMT3070A	20140116
eMT3105P, eMT3120A, eMT3150A	20140701
XE Series	20140624
mTV	20140807



## 13.32. System Message

#### 13.32.1. Overview

If objects use [Display confirmation request] or [local HMI supports monitor function only] is turned on/off, the corresponding messages configured here will be displayed in pop-up message boxes.

#### 13.32.2. Configuration

Click the System Message icon riangleap on the toolbar to open the setting dialog box.

#### System Message

stem Message Confirmation require	he		
		-	_
Dialog size :	Small	Middle	Carge
Message :	Please confirm the	operation	
		Us	e label library
ОК:	ОК		
		Us	e label library
Cancel :	Cancel		
		Us	e label library
Font :	Arial		•
Deny write-commar	nd		
Message :	The system is being	g prohibited from writi	ing device registers!
Font :	Arial		•
		Us	e label library
Allow write-comma	nd		
Message :	The system is now	allowed to write devi	ce registers.
Font :	Arial		•
		Use	e label library
		ОК	Cancel

Setting	Description
Dialog Size	Select the size for pop-up window and texts.
Confirmation	If an object uses [Display confirmation request], this message
required	would pop up when the object is used. [Message] shown on
	confirmation dialog box, and the text label of the 2 buttons, [OK]
	and [Cancel], can be set. Please use the same font for the labels of
	[Message], [OK] and [Cancel]. Additionally, only when selecting

	[Label Library] for [Message], the use of Label Library for [OK] and
	[Cancel] buttons can be enabled.
Deny	Displays when system tag LB-9196 (local HMI supports monitor
write-command	function only) is turned ON.
Allow	Displays when system tag LB-9196 (local HMI supports monitor
write-command	function only) is turned OFF.



CMT-SVR does not support adjusting dialog size and using system tag LB-9196.



## 13.33. Recipe View

#### 13.33.1. Overview

Recipe View object can be used to display a specific recipe. All items and values of the recipe can be viewed by using this object.

### 13.33.2. Configuration



Click the Recipe View icon on the toolbar to open a Recipe View object property dialog box. Set up the properties, press OK button, and a new Recipe View object will be created.

#### **General Tab**

## cMT Series

## eMT, iE, XE, mTV Series

New Recipe View Object	8	New Recipe View Object
General Shape Font		General Shape Font
Comment :		Comment :
Recipe table	-	Recipe table
Recipe Name : Recipe		Recipe Name : Recipe
Title Transparent		Tite
Color :		Color :
Profile	511	Profile
Transparent		Transparent
Frame : Background : Background :		Frame : Background :
		Grid Finable
		Color :
		Selection control
		Color :
Default sort method		Default sort method
✓ Enable		V Enable
Sort by : NewItem		Sort by : NewItem
Ascending     Oescending		Ascending     Oescending
OK Cancel Help		OK Cancel Help



## The name of each part of the Recipe View object is shown in the following figure.



Setting	Description
Recipe table	Choose the recipe name or look for other recipes from the drop-down list.
Title	The item name assigned in [System Parameter Setting] » [Recipe]. Transparent
	If selected, the title row has no shading; the color selection is not available.
Profile	The frame and background color of the object can be set.
	Transparent
	Select to hide the background, the color selection is not available.
Grid	The dividing lines between columns and rows.
(N/A for cMT)	Enable
	Select to show the grid.
Selection	
Control	Change the shading color of the selected row.
(N/A for cMT)	
Default sort	Configure how the records are sorted. [Ascending] and
method	[Descending] can be selected.



## There are 4 system registers that can be used to view/update/add/delete recipe database: Selection

Current selection of record in Recipe View object, and it is numbered from zero. If the first record is chosen, the value of Selection will be 0. When the value of Selection is changed, the corresponding values will be updated, such as "No", "Timer_1", "Timer_2", as shown in the following figure.



Numeric Input Object's Properties			×
General Data Entry Numeric Format Security Shape Font P	rofile		
Description :			
Read/Write use different addresses			
Read address			
PLC name : Local HMI	Se	tting.	
Address : RECIPE	1		
✓ Recipe_Solar	•	$\checkmark$	Selection
			Count
			Command
			Result
			No
Notification			
Enable			Timer_1
			Timer_2
			Timer_3
			Timer_4
			Speed

#### Count

Show the number of records in current recipe.

#### Command

Enter certain value will send command to the selected record.

Enter "1": Add a new recipe record next to the currently selected one.

Enter "2": Update the selected recipe record.

Enter "3": Delete the selected recipe record.

Enter "4": Delete all recipe records.

### Result

View the result of executing commands.

Displays "1": Command successfully executed.

Displays "2": The selected record does not exist.

Displays "4": Unknown command.

Displays "8": Records reach limit (10000 records), no new records can be added.

Please go to [System Parameter Settings] » [Recipes] tab to create the recipe data

before using Recipe View object. See "5 System Parameter Settings".

About creating recipes, see "24 Recipe Editor".



## Example 1

In this example, a recipe database is created to be displayed by Recipe View object. When you select a recipe record on Recipe View object, the value of [Selection] and the corresponding values will change accordingly. When finish designing, you can modify the recipe database by entering a value in [Command].

No	Name	Timer_1	Timer_2 T	ïmer_3 T	imer_4	Speed	
0	Mercury	10	1	11	12	26.500	
1	Venus	20	1	21	22	33.500	
2	Mars	30	2	32	35	41.500	
3	Jupiter	50	3	53	56	50.500	
4	Saturn	80	5	85	90	60.500	
Jvstem Re	zisters:						
	gisters: action: 2	Count:	: 5	Comman	d: 0	Result: 1	
Sele	- -		5	Comman	d: O	Result: 1	
	ection: 2	lify here)	5	Comman	d: 0	Result: 1	

**1.** Create a recipe as shown in the following figure.

Device	Model	Gen	General System Setting		Security	/ For	nt	
Extended Memory		Prir	nter/Backup	Server		e-Mail	Recipe	s
Recipes List :								
Recipes		Item name	Data type	Size	Display wi	Decimal Pt.	Alignm	
1. Recipe_Solar	lar	No	16-bit U	1	5	0	Align left	
		Name	ASCII	10	8	0	Align left	
		Timer_1	32-bit Si	2	8	0	Align right	
		Timer_2	32-bit Si	2	8	0	Align right	
		Timer_3	32-bit Si	2	8	0	Align right	
		Timer_4	32-bit Si	2	8	0	Align right	
		Speed	32-bit Fl	2	8	3	Alien right	

2. Use Recipe Records to create a number of records as shown in the following figure.



tecipe_Solar (5)		dd	Dele	te					
		No	Name	Timer_1	Timer_2	Timer_3	Timer_4	Speed	
	1	0	Mercury	10	1	11	12	26.5	
	2	1	Venus	20	1	21	22	33.5	
	3	2	Mars	30	2	32	35	41.5	
	4	3	Jupiter	50	3	53	56	50.5	
	5	4	Saturn	80	5	85	90	60.5	

- 3. Create a Recipe View object and use the recipe database created in the preceding steps.
- 4. Create 4 Numeric objects using registers "Selection", "Count", "Command", and "Result".
- Create corresponding input objects for "No", "Name", "Timer_1", ..., "Timer_4", "Speed".
   For example, "Name" is an ASCII item with size "10". Create an ASCII object and set device type to "RECIPE" » "Name".

Read address			
PLC name :	Local HMI		Setting
Address :	RECIPE 🔽	Name	-
		✔ Recipe_Solar →	🗸 Name

- 6. The project is then completed.
- 7. As shown above, "Mars" is selected and the corresponding items are also updated. There are 5 records so the "Count" displays "5". Try selecting different rows of the Recipe View object. Fields "Name", "Timer_1", ...will change accordingly.
- 8. Try the following operations:
- Add:

To add current data as a new record, enter "1" in "Command".

- Update:
   To update recipe database, enter "2" in "Command".
- Delete:

To delete the selected record, enter "3" in "Command".

- Sort the items.
- Click the title to change the order.



Objects

## Example 2

In this example, [RECIPE_Bit] can be used to read / write individual bits of Recipe data. Although BOOL type items cannot be added to Recipe Database, individual bit access of 16bit / 32bit data is possible.

As shown in the following figure, select [RECIPE_Bit] for the read address of Bit object and point to the target item, and then the available Bit selections will be displayed. In this manner, Recipe Database can be used to record, read, and write bit data.

Recipes 📴 🗙	Item name	Data type	Size	Display widt	h De	cimal Pt.	Alignment
1. myRecipe	A	16-bit Unsigned	1	5	0		Align left
	В	32-bit Unsigned	2	5	0		Align left
	С	32-bit Unsigned	2	5	0		Align left
Read address							
PLC name : Local HMI			~	Settings			
Address RECIPE_Bit	~	Selection-O	-				
		✓ myRecipe	Þ	<ul> <li>Select</li> </ul>	ion	•	
	nvert signal			Coun	t	•	
				Comr	nand	•	
				Result	t	+	
			- [	A		×.	0
				В		•	1
				С		+	2
							3
							4
							5
							6
							7
							8
							9
							10
							11
							12
							13
							14



15

## 13.34. Flow Block

#### 13.34.1. Overview

Flow Block object displays the flow status of the blocks in the pipe or the status of the transportation lines. Unlike Moving Shape object which requires a precise measurement between two points when drawing a straight line provided by users, the blocks flow at a fixed interval in a horizontal or vertical straight line.

The features of Flow Block:

- Each section of the Flow Block must be a horizontal or vertical straight line and the blocks flow at a fixed interval within it.
- Dynamic speed and direction adjustment (Speed and direction can be controlled by a designated register.
- Security mechanism (Interlock), which hides Flow Block when the status of designated bit is invalid.

## 13.34.2. Configuration



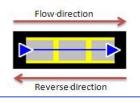
Click on the Flow Block icon on the toolbar or select [Objects] » [Flow Block] to create object.

#### **General Tab**

General	Outline	Security		
	Commen	t:		
Flow :	speed			
			Reverse direction	
			Dynamic speed	
	F	ow speed :	5	

Setting Description

The blocks flow in the direction the object is drawn (the blue arrow). If select this check box, the blocks flow in the opposite direction.





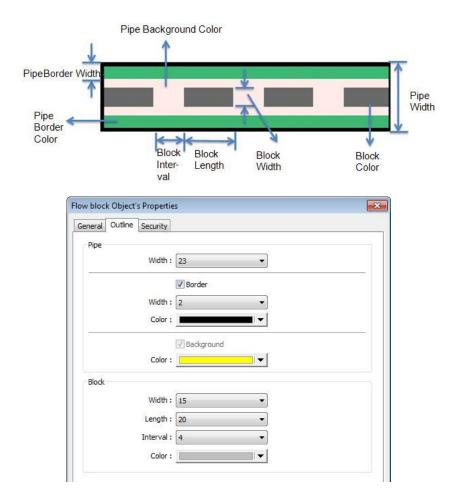
Reverse

direction

<b>D</b> ·	
Dynamic	Read address
speed	The direction and speed at which the blocks flow can be controlled
	by a designated register. The valid rage is -25 to 25. When a
	negative value is entered, the blocks flow in a reversed direction.
	Setting
	Displays the address and format of the designated register. [System
	register], [Index register], and [Tag Library] can be set here.
Flow speed	25 flow speed levels, the valid range is 0 to 25 when [Dynamic
	speed] is not selected. A larger value indicates a faster speed.

## **Outline Tab**

For setting the outline property of Flow Block. The following illustration shows each item.



Setting	Description
Pipe	Sets the properties of the pipe within which the blocks flow. The
	background color, border width and color can be set. When the
	[Border] check box is selected, the background color must be set.

Block	Sets the properties of blocks. Width, length, interval and color can
	be set here.

# Note

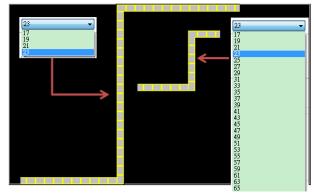
- If both [Reverse direction] and [Dynamic speed] check boxes are selected in [General] tab, when entering a negative value in the designated register of dynamic speed, the blocks flow in the direction the object is drawn.
- To avoid the pipe lines from overlapping when drawing a turn, there is a minimum width planned at each turn. As shown in Fig. 34.1, the sign on the cross cursor defines the minimum width. Fig. 34.2 demonstrates that each turn is drawn in the minimum width.



(Fig. 34.1) (Fig. 34.2)

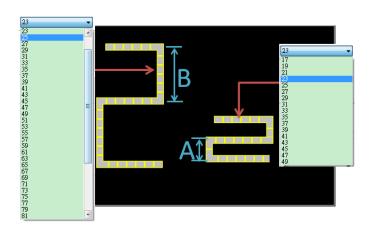
The valid range of the length, width, and height of the Flow Block can be adjusted according to the size of the object drawn and the size of the window.

As shown in the following figure, when the size of the Flow Block is larger, the valid range is restricted to prevent the flow block from exceeding the window size. When the size of the object is smaller, the adjustment range will be larger.



To prevent the flow block from overlapping itself, when the distance between two lines is shorter (Section A), the valid range is restricted. When the distance is longer (Section B), the adjustment range will be larger.





## Example 1

The demonstration below shows how to use [Dynamic speed] to control the direction and speed of Flow Block by a designated word register.

1. Create a Flow Block object and select [Dynamic speed] check box. Set [Address] to LW-0, and set the format to 16-bit Signed.

Flow block Object's Properties	23
General Outline Security	
Comment :	
Flow speed	- I
Reverse direction	
V Dynamic speed	
Read address	51
PLC name : Local HMI	
Address : LW 🔻 0 16-bit Signed	=
	- 1

2. Create a Numeric object, set [Address] to LW-0. The high limit is 25, and the low limit is -25. The format is 16-bit Signed.



3. Execute simulation or download the project to HMI. When entering a positive value in LW-0, the blocks flow in the direction the section is drawn. A larger value indicates a faster speed. When a negative value is entered, the blocks flow in a reversed direction, and the smaller value indicates a faster speed. When 0 is entered, it stops flowing.

Lick the icon to download the demo project. Please confirm your internet connection.

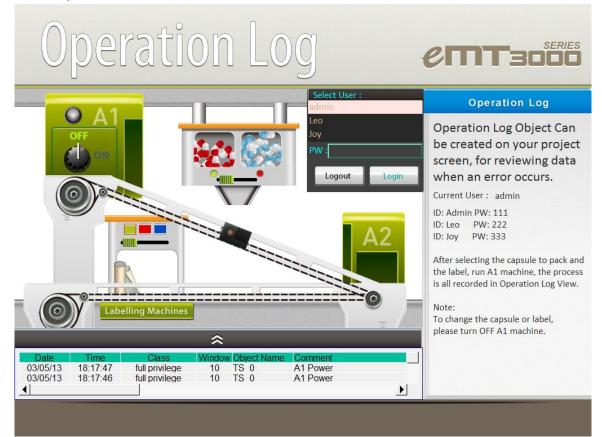


## 13.35. Operation Log

## 13.35.1. Operation Log Settings

#### 13.35.1.1. Overview

Operation Log records user's operation steps and displays the record in real-time. When an error occurs, use operation log to analyze the problem. The backup tables can be used to review the process in order to resolve the errors.



#### 13.35.1.2. Configuration



Select the objects to be recorded. Click [Objects] on the main menu, point to [Operation Log], click [Operation Log Settings], and then select [Enable operation log function] check box.



5 : PLC Response <u>■F</u> FK_0				
<u>•</u> f]FK_0				
🖃 6 : HMI Connection				
_F FK_0				
7 : Password Restriction				
_F FK_0				
8 : Storage Space Insufficient				
<pre>_F FK_0</pre>				
10 : WINDOW_010				
AE_0				
SW_0		1	Log in1	
[23] S₩_1		1	Log out	
₽OL_0		1	User ID	
≌_sB_0		1	Green Label	-
मन्त्र <mark>ता व</mark> ा ।		UN.	Dad I ahal	•
Maximum record no. in HMI men External devices for synchronizati	on / back	.000 kup ISB disk	•	
Behavior when HMI space is insufi Stop saving operation log Synchronize to external device records.		device d	oes not exist, erase the olde	est
Control address PLC name : Local HMI			▼ Settin	

Setting	Description					
Object	When Operation Log is enabled, the objects with write function are					
	listed in the setting dialog box sorted by window numbers.					
	[Filter]: By clicking 🍱 icon, the objects with write function are					
	listed. Users can filter out the objects that need not to be recorded,					
	and the log displays only the selected objects.					
Enable	The selected objects are recorded by Operation Log.					
Comment	The description of the object as shown in the following figure.					
	Set Word Object's Properties       E3         General Security Shape Label Profile       Comment : Log in         Write address       PLC name : Local HM         PLC name : Local HM       Setting         Address : UAC command       LW-100         16-bit Unsigned       Write after button is released					
	□ 10 : WINDO W_010 □ 10 : WINDO W_010 □ 12 : SW_0 □ 12 : SW_0 □ 12 : SW_1 □ 10 : W 1 □ 10 : WINDO W_010 □ 10 : W					



Select all	Selects all the listed objects. If [Filter] is used, clicking [Select all] only selects the objects in the list.
Discard all	Discards all the selected objects. If [Filter] is used, clicking [Discard
	all] only discards the objects in the list.
Storage	Sets the way the records are stored.
settings	Maximum record no. in HMI memory
U U	Sets the maximum number of records that can be stored in HMI
	memory.
	External devices for synchronization / backup
	Stores backup data to SD card or USB disk.
	Behavior when HMI space is insufficient
	When HMI memory space is insufficient, two options are provided:
	[Stop saving operation log]: Stops saving new records in order to
	keep the earlier records.
	[Synchronize to external device]: Stores the Operation Log to the
	external device. When the device does not exist, the HMI clears the
	oldest records in its memory.
Control	Entering different values in the control address sends
address	corresponding commands to Operation Log and returns the result
	of executing the command.
	If control address is LW-n (where n is an arbitrary number), the
	address that returns the result of executing the commend is
	LW-n+1.
	Control address (LW-n):
	(1): Clear all records.
	(2): Copy the records to the USB disk.
	(3): Copy the records to the SD card.
	(4): Copy the records to the USB disk and clear the records in HMI
	memory.
	(5): Copy the records to the SD card and clear the records in HMI
	memory.
	Execution result (LW-n+1):
	(0): Processing.
	(1): Execution succeeded.
	(2): The device does not exist.
	(3): The record does not exist.
	(4): Unknown error.



# Note

- Operation Log can only record the operation of the objects that are manually triggered.
   Objects that cannot be manually triggered are not recorded, such as Time Based Data Transfer object.
- When running off-line or on-line simulation, Operation Log is stored under EasyBuilder installation directory: HMI_memory\operationlog\operationlog.db
- Triggering Macro with a Set Bit object generates two records, the triggering of bit and the triggering of Macro.

## 13.35.2. Operation Log View

## 13.35.2.1. Overview

Operation Log View can be used to review the Operation Log.

## 13.35.2.2. Configuration

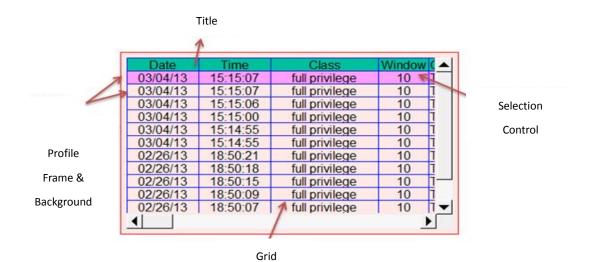


Before using Operation Log View, please follow the steps described in the preceding part to finish Operation Log Settings. Click [Objects] on the main menu, point to [Operation Log], and then click [Operation Log View].

## **General Tab**

Operation Log View Object's Properties	23
General Title Shape Profile	
Comment :	
Transparent	
Color :	
Profile Transparent	
Frame : Background :	
Grid I Enable	
Selection control	
Color :	
Font	
Color:	
Name : Arial	
Size : 12	
OK Cancel Help	





Description
Sets the color of the title row.
Transparent: If selected, the title row will be transparent. The color
selection is not available.
Sets the color of the frame and background of the object.
Transparent: Hides the frame and background. The color selection
is not available.
Sets the color of the dividing lines between the columns and rows.
Enable: If selected, displays the grid, otherwise, hides the grid.
Sets the color of the selected row.
Log View object.



#### Title Tab

	peration Log View Object's Properties	
	General Title Shape Profile	
	General       Title       Shape       Profile         Title name       Title       Date         Time       Time       Time         User name       User name       User name         User name       User name       User name         Chass       Class       Class         Window       Object Name       Object Name         Comment       Comment       Address         Address       Address       Information         Information       Information       Information         Order & Characters       Imme       Display rder         V Date       0       Vindow       Object Name         V Date       0       Vindow       Object Name         V Date       0       Vindow       Object Name         V Display items       0       Vindow       Object Name         V Class       0       Vindow       Object Name       Comment         V Mindow       0       Vindow       Object Name       Comment         V Mindow       0       Vindow       User name       Imme         V Mindow       0       Vindow       User name       Imme         V Mindow       0       Vi	
	OK Cancel Help	
Setting	Description	
Title	Sets the title displayed in Operation Log View object.	
Sort	Sorts the records in time ascending or descending order.	
Display order	Sets the order of the displayed item. If [Display chars.] is 0, all	
	characters are displayed.	

Date / TimeSets the format of date and time displayed in Operation Log View<br/>object.

### 13.35.3. Operation Log Printing

## 13.35.3.1. Overview

Operation Log Printing can generate an Operation Log sheet by printing out using a printer or by saving as JPEG file into an external device. Before using this function, please go to Operation Log Settings to finish the settings.



#### 13.35.3.2. Configuration



Select "Enable [Operation Log] printing" check box and click [Settings] button to open the Operation Log Printing dialog box.

Printing Manager	X
Enable [Operation Log] printing	<b>5-1</b> /
	Settings
	OK Cancel

Dperation Log Printing
General Layout Content
Comment :
Printer
Device : SD card
Orientation
Horizontal     O     Vertical
Font
Name : Arial 👻
Size : Middle 💌
Range Type : O Date O Record
Within : 1000 record(\$)
Trigger address
PLC name : Local HMI
Address : LB 🔻 0
Preview OK Cancel Apply Help

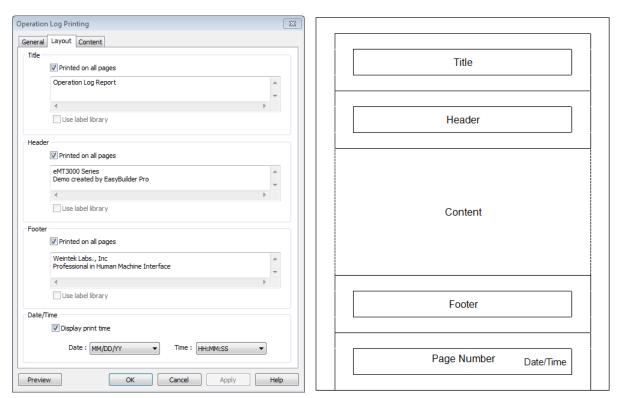


Setting	Description					
Printer	Select the device to save the Operation Log sheet. If a printer is					
	selected, the paper size should be A4. If an external device is					
	selected, the Operation Log sheet is saved as a JPEG file. The					
	system generate	s a folder nam	ed "operationlogs	sheet", and the files		
	saved in the fold	ler are named '	'print date_seque	ence number". For		
	example, the first	st JPEG file save	ed on 2013/05/08	3 is named		
	130508_0000 ar	nd so on.				
<b>Orientation</b> Sets the layout of the Operation Log sheet to be h				horizontal or		
	vertical.					
Font	Sets the font and	d the font size t	to of the Operation	on Log sheet. The		
	following table lists the corresponding size.					
	Size	Title	Content			
	Large	20 pt.	16 pt.			
	Middle	16 pt.	12 pt.			
	Small	12 pt.	8 pt.			
Range	Sets the range of the Operation Log to be included in the sheet.					
	Date					
	Sets the range by date, counted from the start day through the					
	number of days entered. The maximum available range is 30 days.					
	Record					
	Sets the range by the number of records. The maximum available					
	range is 10000 records.					
Trigger	Sets the register to control Operation Log Printing. When the					
address	register is set ON	N, it starts print	ing. When the pr	inting is done, the		
	register is set OF	F automaticall	у.			
Preview	Preview the resu	It before gene	rating the Operat	ion Log sheet.		



#### Objects

#### Layout Tab



The layout of each part is shown in the above figure.

	-
Setting	Description
Title	Sets the content of the title. The title is limited to one line.
	Printed on all pages
	If selected, the title is shown on each page; otherwise, the title is
	shown on the first page.
Header	Sets the content of the header. The header can have 5 lines in
	maximum.
	Printed on all pages
	If selected, the header is shown on each page; otherwise, the
	header is shown on the first page.
Footer	Sets the content of the footer. The footer can have 5 lines in
	maximum.
	Printed on all pages
	If selected, the footer is shown on each page; otherwise, the footer
	is shown on the last page.
Date/Time	If selected, the date/time the in the sheet is shown on the
	lower-right corner of each page; otherwise, the date/time is not
	shown.
Page number	Shown on each page.



#### **Content Tab**

Titl	e name	Title			
Dat		Date			
Tim	e	Time			
Cla	s	Class			
Wir	ıdow	Window			
Obj	ect Name	Object Na	me		
_	nment	Comment			
Act		Action			
	lress	Address			
Info	rmation	Informatic	m		
	Display item Date	15	Display chars 10		Display order Date
			10		Time
5			8	-	Class
5	Window		8		Window Object Name
8	Object Name	9	12	-	Comment
8			30		Action
8			30		Address Information
5	-		15		
	Information		30		
	1 Information		30		
	MM/DD/YY	•	Time : HH:MM	:SS	•
Date :					

Setting	Description
Title	Sets the title displayed.
Sort	Time ascending
	The latest record is placed at the bottom.
	Time descending
	The latest record is placed at the top.
Date/Time	Sets the format of date and time displayed.

#### 13.35.3.3. Demonstration

#### Example 1

The following demonstration explains how to create an Operation Log project.

- **1**. Create a Toggle Switch object and a Numeric object on window number 10.
- 2. Go to Operation Log Settings; enable the Toggle Switch object and Numeric object on window number 10.



Object	¥=	Enable	Comment	
5 : PLC Response				[
<pre>_F FK_0</pre>				
🗉 6 : HMI Connection				
<u>•</u> F FK_0				
7 : Password Restriction				
<pre>_F FK_0</pre>				
🗉 8 : Storage Space Insufficient				
<u>•</u> f FK_0				
10 : WINDOW_010				
AE_0				
0_W2 ESI		1	Login	
1_W2		1	Logout	
OL 0		1	User ID	
₩]sb_0		1	Green Label	
मम्ब (1) मार्ग (1) मिर्ग (1) मार्ग (1) मा			Pad I ahal	

- 3. Create an Operation Log View object and finish relevant settings.
- **4.** Run off-line simulation; trigger Toggle Switch and Numeric object. Operation Log is displayed by Operation Log View object.

e 5	Numeric In	put		
Time	Comment	Action	Address	Information
18:52:38	Numeric Input	Set word	Local HMI : LW-10	write 5
18:52:35	<b>Toggle Switch</b>	Toggle	Local HMI : LB-0	bit set OFF
18:52:35	<b>Toggle Switch</b>	Toggle	Local HMI : LB-0	bit set ON
18:52:34	Toggle Switch	Toggle	Local HMI : LB-0	bit set OFF
18:52:34	Toggle Switch	Toggle	Local HMI : LB-0	bit set ON
	Time 18:52:38 18:52:35 18:52:35 18:52:34	TimeComment18:52:38Numeric Input18:52:35Toggle Switch18:52:35Toggle Switch18:52:34Toggle Switch	TimeCommentAction18:52:38Numeric InputSet word18:52:35Toggle SwitchToggle18:52:35Toggle SwitchToggle18:52:34Toggle SwitchToggle	TimeCommentActionAddress18:52:38Numeric InputSet wordLocal HMI : LW-1018:52:35Toggle SwitchToggleLocal HMI : LB-018:52:34Toggle SwitchToggleLocal HMI : LB-018:52:34Toggle SwitchToggleLocal HMI : LB-0

Lick the icon to download the demo project. Please confirm your internet connection.

#### Example 2

Upload Operation Log to PC by using Utility Manager or use Backup object to send the file by email.

- Upload by Utility Manager
- 1. Open Utility Manager, click [Upload].
- 2. Select [Operation log], enter file name and HMI IP, and then click [Upload].



Jpload					<b>×</b>
	eMT3000 S	eries		•	
Project	Jointone of	01100			
E RW					
E RW_A					
Recipe database	e				
Operation log	C:\Users\use	er\Desktop\Operation	Log file.db		Browse
Data log	,				
Event log					
Extend Memory	(EM)				
Extend Memory	(EM)				
Connection	ernet C	USB cable			
4 IP Nam		000 Cable			
	IP: 192.168.1.10	0 🗾			
				г	
Upload	Stop	Settings			Exit

- Send the sheet by e-mail
- 1. Open [System Parameter Settings] » [e-Mail] tab. Set e-mail server and the address of recipient and sender.
- 2. Create a Backup object, under [Source] select [Operation log], and under [Backup position] select [e-Mail].

New Backup Object	8
General Security Shape Label e-Mail	
Comment :	
Source	
© RW © RW_A © Recipe database	
Historical event log     Historical data sampling	
Operation log	
Backup position	
SD card ○ USB disk ○ e-Mail	
Remote printer/backup server	
Note : Use LW-9032~9039 to change the backup folder name.	
Note : Use [Remote printer/backup server] to store data to a remote PC. Enable server in [System Parameter] [Printer/Backup Server] settings.	the
Trigger Mode : Touch trigger ▼	
OK Cancel H	lelp

For more information about e-Mail settings, see "5 System Parameter Settings".



#### 13.36. Combo Button

#### 13.36.1. Overview

Combo Button can execute multiple commands. The former way was to overlay multiple objects in the same position, and the commands are executed in the order of the layer of the objects. This takes time to test the order when planning the project. Combo Button allows users to easily set multiple commands with one object, and freely adjust the order of executing commands.

The following are the features of Combo Button:

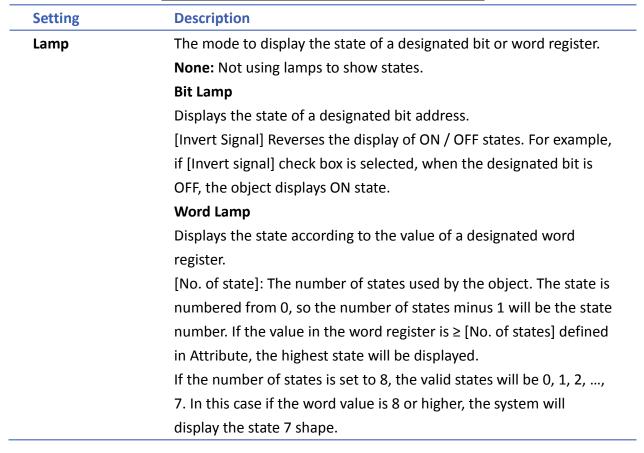
- Executes multiple commands.
- Allows adjusting the order of executing multiple commands.
- Displays the state in Bit or Word Lamp.

#### 13.36.2. Configuration

Click the Combo Button icon on the toolbar to open a Combo Button object property dialog box. Set up the properties, press OK button, and a new Combo Button object will be created.



ombo Button Object's Propertie General Security Shape Labe Lamp Mode : None None Bit Jamp Word Jam	A Profile
Actions          Set Bit (Set ON)         Set Word (Write constan         Delay (S0 ms)         Change window (11. Win         Image: Window (11. Win         Image: Window (11. Win	Set style : Set ON  Write address Local HMI  LB  Setting
Add 🗸	
	OK Cancel Help



	There are four types of actions: [Delay], [Set Bit], [Set Word], and [Change window]. A combo button can execute up to 20 actions.  Change the order of the actions.					
	ECopy 💼	Paste 🔀 Delete				
	Copy, paste, or	delete the selected actions.				
dd	Delay					
	Delays the acti	on for a few seconds. A combo button can set one				
	[Delay] action	only.				
	Set Bit					
	Sets the design	nated bit ON or OFF.				
	Set style	Description				
	Set ON	Set ON the designated bit of the device.				
	Set OFF	Set OFF the designated bit of the device.				
	Toggle	Alternates the bit state each time pressed.				
	Momentary Holds the bit ON only while button is pressed.					
	Set Word Sets the value Set style	in the designated register. Description				
	Write Constant Value	Writes a constant value to the designated register.				
	JOG+	Increases value in register by a set amount in [Inc. value] each time when the button is				
	JOG-	pressed, to the [Upper limit]. Decreases value in register by a set amount in [Dec. value] each time when the button is pressed to the [Bottom limit]				
	JOG- Dynamic limits	Decreases value in register by a set amount in				
	Dynamic	<ul> <li>Decreases value in register by a set amount in</li> <li>[Dec. value] each time when the button is</li> <li>pressed, to the [Bottom limit].</li> <li>Sets the Upper / Bottom limit by a designated</li> </ul>				

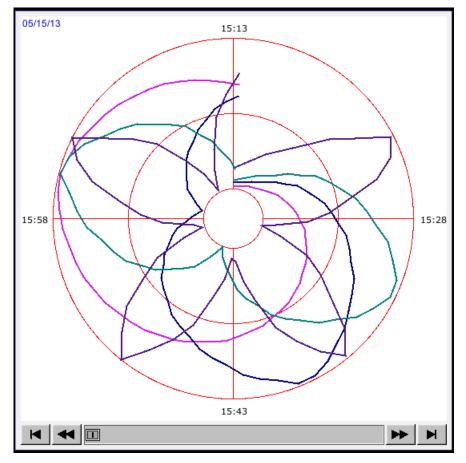
[Change Window] action, and this action is always the last one executed.



#### **13.37.** Circular Trend Display

#### 13.37.1. Overview

Circular Trend Display object draws the trend curve of Data Sampling in a polar coordinate system, where y-axis represents the radial coordinate and the x-axis represents the angular coordinate. The way to use this object is similar to using Trend Display object.



#### 13.37.2. Configureation



Click the Circular Trend Display icon on the toolbar to open the property dialog box. Set up the properties, press OK button, and a new Circular Trend Display object will be created.



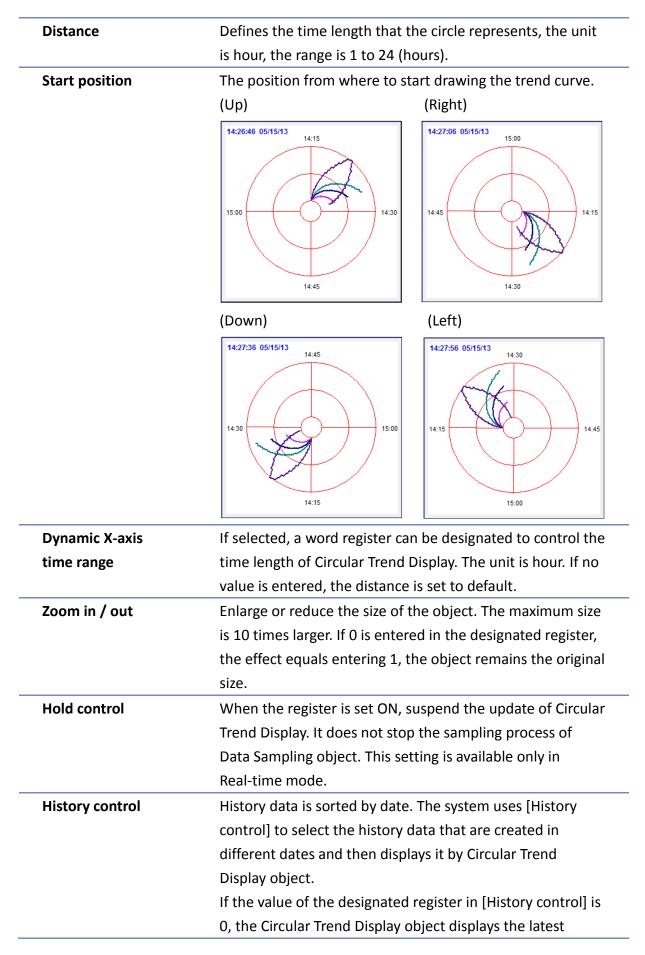
ilar Trend D	isplay Object's Properties
neral Trend	Channel Shape Profile
Com	iment :
Data San	npling : 1.
	Type : Real-time
	of channels is changed, you must reset HMI's data samplings !!
Distance :	1 v hour(s) Start position : Up v
	Vpnamic X-axis time range
PLC name :	Local HMI    Setting
Address :	LW 🔻 100 16-bit Unsigned
loom in/out	
oom injout	Enable
PLC name :	Local HMI   Setting
Address :	
- Hold control	
ioia controi -	Tenable
PLC name	
Address	
Vatch line	T Enable
PLC name :	Local HMI   Setting
Address :	
Time stamp o	utput V Enable
PLC name :	
Address :	
Auguess :	V TOO DE DIC ORINGROU

Setting	Description		
Data Sampling	Selects the data source for drawing the trend curve.		
Туре	Selects the type of the trend from [Real-time] or [History].		
	Real-time		
	In this mode, it displays a fixed number of sampling data		
	from the moment HMI starts to present. The number of		
	sampling data is determined by the [Max. data records		
	(real-time mode)] setting of Data Sampling object. If the		
	number of sampling data exceeds this number, the earlier		
	data will not be displayed. To display earlier data or the		
	data in other days, please select [History] mode.		
	[Hold control] address can be used to pause refreshing the		
	display. This only stops displaying new data in the Circular		
	Trend Display object, and the data is still being sampled by		



	Data Sampling object.
	History
	In this mode, it displays the sampled data sorted by date.
	Select the data source from [Data Sampling], and then use
	[History control] address to view the records of different
	dates.
	Note
	If [Show scroll control] check box in Trend Tab is not
	selected, the earlier data cannot be viewed when
	exceeding the specified [Distance].
	For example: Set [Distance] to 1 (hour.), then sampling data
	earlier than one hour is not displayed.
Refresh data	If enabled, the window in which the Circular Trend Display
automatically	object (in history mode) is placed will be refreshed once
	per second.
	<ul> <li>The scroll controls can be used to check the refresh</li> </ul>
	status.
	If 🔳 button is displayed, the Circular Trend Display
	will be automatically refreshed.
	If 🕨 button is displayed, the Circular Trend Display
	will stop being refreshed.
	<ul> <li>Scrolling backward and viewing earlier data will</li> </ul>
	disable [Refresh data automatically]. The button
	displayed is 🖿 at this moment.
	<ul> <li>If [Refresh data automatically] is selected, the display</li> </ul>
	is refreshed when change back to this window,
	regardless of the use of scroll controls.
	Example: If [Refresh data automatically] is selected,
	scrolling to the earlier display stops auto-refresh. At this
	moment change to another window and then change back,
	the Circular Trend Display is still refreshed.
	If [Refresh data automatically] is not enabled when
	building the project, to enable it directly on HMI, simply
	press 🕨. Please note that auto-refresh remains disabled
	after window change.







record. If the value is 1, the second latest record is displayed and so on. This setting is available only in History mode.

If use with Option List object and select data source as [Dates of historical data], the history data will be sorted by date and displayed in Option List object, see "13.29 Option List".

In the following example, when history control address is set to LW-0, and there are 4 sampling data: 20061120.dtl, 20061123.dtl, 0061127.dtl, 20061203.dtl. The

corresponding data selected by the value in history control address is as the following list.

Value in LW-0	The sampling data displayed
0	20061203.dtl
1	20061127.dtl
2	20061123.dtl
3	20061120.dtl

Watch line Displays a watch line when user touches the Circular Trend Display object, and the sampling data at the position of the watch line is output to the designated register. To display sampling data with multiple channels, the system consecutively writes the data of each channel to the designated word register and the following registers. If the data format of each channel is different, the channels are sorted by the data format of its corresponding register. In the following example, when watch address is set to LW-0, and there are 4 sampling data, the format of each data is: 16-bit Unsigned, 32-bit Unsigned, 32bit Signed, and 16-bit Signed. The corresponding watch address is as the following list. Channel Data Format Watch Address Data Length 0 16-bit Unsigned 1 Word LW-0 2 Words LW-1 1 32-bit Unsigned 2 LW-3 32-bit Signed 2 Words 3 16-bit Signed 1 Word LW-5 Time stamp output If selected, the system will start counting time from the first data sampled, and output the elapsed time counted of



the latest data sampled to the register designated in [Time stamp output + 2]. When pressing a point on the trend curve, the relative time of the nearest data sample is then output to [Time stamp output address].

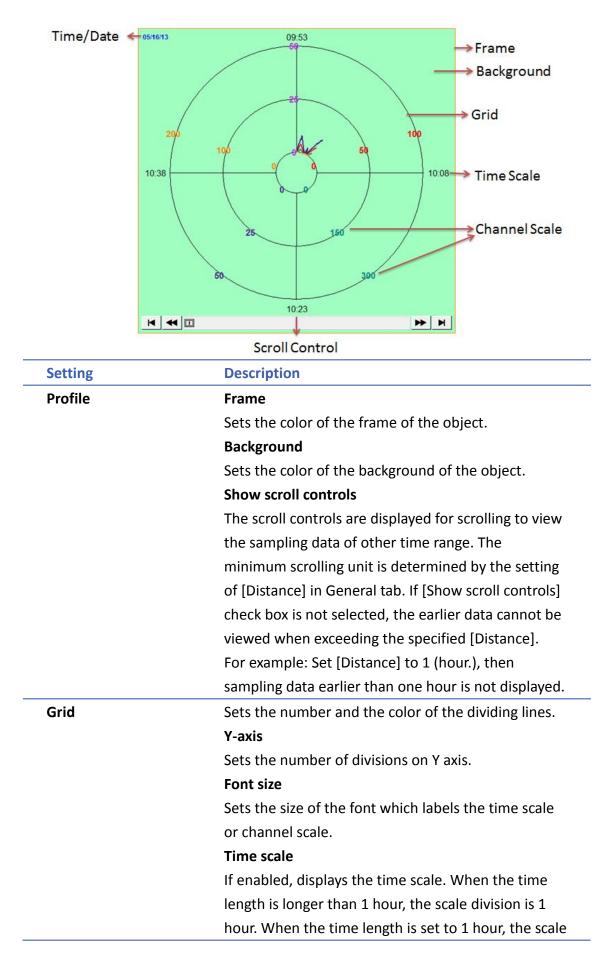
#### Note

The format of the register designated in [Time stamp output] and [Time stamp output + 2] must be 32-bit. [Time stamp output + 2] is only available for Real-time mode while [Time stamp output] is available for Real time mode and History mode.

#### **Trend Tab**

General Trend	Display Object's Properties
Profile	ame : Background : 📃 💌
Grid	🖉 Enable Color :
Y-a	axis: 4 division(s)
Font s	size : 8
Time Sedie	☑ Enable Color :
Channel sca	ale 🔄 Enable
Time/Date	
🔽 Time	● HH:MM:SS
🔽 Date	
Color :	





	division is 15 minutes.
	Channel scale
	If enabled, displays the channel scale. The color of
	the font which labels the channel scale corresponds
	to the setting of the trend curve of each channel.
Time / Date	Time
	Sets the display format of time.
	Date
	Sets the display format of date.

#### **Channel Tab**

	al Trend	Channel	Shape Profile	
ata	sampling o	- bject		
Channel Displ		Display	Description	Data type
•	0	<b>V</b>	16-bit Unsigned	16-bit Unsigned
	1	<b>V</b>	16-bit Unsigned	16-bit Unsigned
	2	<b>V</b>	16-bit Unsigned	16-bit Unsigned
	3	1	16-bit Unsigned	16-bit Unsigned
	4	<b>V</b>	16-bit Unsigned	16-bit Unsigned
	Col	or :		Width : 2 🗸 🗸
	Ze	Dy ro: 100	namic limits	Span : 1000
	nel visibilit	y control		
han		V En	able	
han				
	C name :	Local HMI		▼ Setting
PL	C name : ( Address : (		▼ 250	
PL	Address : (	LW	e corresponding bit is :	

Setting	Description
Channel	Sets the style and the color of the trend curve, and
	the upper and lower limit of data that can be drawn
	on the trend curve. Up to 8 channels are supported
	simultaneously.



	Not selec	ting [Dynami	c limits]			
	The upper and lower limits of the data are set by					
	constants.					
	Selecting [Dynamic limits]					
	The uppe	r and lower li	mits are s	set by the	designate	
	register. V	Vhen the add	lress is LV	V-n, the		
	correspor	iding address	es are as	the follov	ving list.	
	C	ata format	16-bit	32-bit		
	L	ower limit	LW-n	LW-n		
	L	Ipper limit	LW-n+1	LW-n+2	2	
Channel Visibility If [Enable] is selected, the bits of the des					gnated	
Control	word regi	ster will be us	sed to she	ow/hide e	ach	
	channel. I	irst bit (Bit-0	) controls	s the first o	channel;	
	second bi	t (Bit-1) contr	rols the se	econd cha	nnel, and	
	so on.					
	Display cl	nannel when	the corre	esponding	bit is:	
	If [ON] is selected, when the corresponding bit is					
	OFF, the channel is hidden. If [OFF] is selected, wher					
	the corresponding bit is ON, the channel is hidden.					
	In the foll	In the following example, the control address of				
	channel v	channel visibility is set to LW-0 and each channel				
	shows wh	en the corres	sponding	bit is OFF.	If there	
	are 5 char	nnels, the visi	bility of t	he channe	els is as the	
	following					
	Chann	el Control a	address	Bit state	Display	
	0	LW_bit	-000	OFF	YES	
	1	LW_bit	:-001	ON	NO	
	2	LW bit	-002	ON	NO	
	2					
	3	LW_bit	:-003	OFF	YES	



#### 13.38. Picture View

#### 13.38.1. Overview

Picture View object plays slideshow of picture files saved in an external device such as a USB drive or SD card.

#### 13.38.2. Configuration



Click the Picture View icon on the toolbar to open the property dialog box. Set up the properties, press OK button, and a new Picture View object will be created.

Picture View Object's Properties
General Security Profile
Comment : Outline
Background color : Font : Arial Display with original size when a picture size is smaller than the object size.
File position © SD card © USB disk
Directory          Image: Dynamic folder path         PLC name :       Local HMI         Address :       LW         Image: Dynamic folder path
File selection         Image: Specify file from address (hide toolbar)         PLC name : Local HMI         Address : LW         Image: Description of the selection of the selec
* Filenames must contain only ASCII characters. Unicode is not supported. OK Cancel Help



Setting	Description
Outline	Sets the toolbar position, background color, and text font
	of the Picture View object.
	Hide delete button
	If selected, the delete button will not be displayed on the
	Picture View object toolbar. The delete button is used to
	delete the picture currently viewed.
	Display with original size when a picture size is smaller
	than the object size.
	If selected, when the size of the picture is smaller than the
	Picture View object, this setting helps to prevent distortion
	caused by enlarging the picture.
File position	Select the file source of the picture files from [SD card] or
	[USB disk].
Directory	The directory where the picture files are saved.
	Dynamic folder path
	Designate folder path by a local address.
File selection	Specify file from address (hide toolbar)
	If enabled, the displayed picture is designated by a file
	name in a local address, and the toolbar will be hidden.



- The file name must be all in ASCII characters, and the Unicode characters are not supported.
- The supported picture formats are: .jpg, .bmp, .gif, .png.



#### 13.39. File Browser

#### 13.39.1. Overview

File Browser object can display files and folders saved in the SD card or USB disk. Apart from browsing for the files in the external devices, the name of the file and the file path selected in File Browser object can be written to the designated address.

#### 13.39.2. Configuration



Click the File Browser icon on the toolbar, or select Tools » File Browser to open a File Brower object property dialog box and set up the properties.

General Outline	Security Shape			
-Folder path add	ress			
	🔽 Enable			
PLC name :	Local HMI		✓ Settings	)
Address :	LW	▼]0	String(20)	
File name addr	ss			
	🔽 Enable			
PLC name :	Local HMI		✓ Settings	]
Address :	LW	▼ 0	String(20)	
PLC name :	✓ Enable Local HMI		✓ Settings	]
PLC name :	the second second		✓ Settings	]
Address :	LW	• 0	String(20)	



#### 13-205

Setting	Description
Folder path addressCurrent directory.	
File name address	The file name of the currently selected file.
Full (folder + file	The full directory and file name of the currently
name) address	selected file.

#### **Outline Tab**

eneral Outline Sec	urity Shape		
File position :	SD card 💿 USB di	sk	
File type :	All files	•	
Font			
	Arial	<u></u>	-
Color :		Size : 12	•
Background			
Color :	Transparent		
Color Grid :		Select box :	-

Setting	Description	
Folder position	Select the position of the file from SD card or USB	
	disk.	
File type	Select all files or only CSV file to be displayed.	
Font / Background / Color	Set the attributes and font of the object.	



The file name and the directory of the selected file will be written to the designated address, but changing the contents of the designated address will not change the selected file in the File Browser.



#### Objects

The system will read the folder path address and file name address when the HMI is restarted or when an external device is inserted to the unit. If valid data is can be read from the designated address, the system will then automatically navigate to the appropriate directory and highlight the file according to the data read. If [Folder path address] is not enabled, the data at Full (folder + file name) address will be read.



#### **13.40.** Recipe Import/Export

#### 13.40.1. Overview

With Recipe Import/Export object, recipe data can be imported or exported between Recipe Database and SD card /USB disk.

#### 13.40.2. Configuration

Click the Recipe Import/Export icon on the toolbar to open the Recipe Import/Export Object management dialog box. To add a Recipe Import/Export object, click [New], set up the properties, press OK button and a new Recipe Import/Export object will be created.

Recipe Import/Export	
1: File position : USB disk, Recipe database : MyRecipe	
New Delete Settings	Exit



Descript	tion :				
File posit					
The post	. 1100	🔘 SD card	💿 USB disk		
Rec	Recipe : MyRecipe				
Control address					
PLC name : [	Local	HMI		Ŧ	Settings
Address :	LW		▼ 100		16-bit Unsigned
Status : ( Result :	0: No LW-1 (0: Id LW-1	ne, 1: Import, 2 00 + 1 le, 1: Busy ) 00 + 2	: Export(no overwrit	e), 3: Export )	
( Status : ( Result : (	0: No LW-1 (0: Id LW-1 (1: Su	ne, 1: Import, 2 00 + 1 le, 1: Busy )	: Export(no overwrit	e), 3: Export )	Construction of August
( Status : ( Result :	0: No LW-1 (0: Id LW-1 (1: Su SS	ne, 1: Import, 2 00 + 1 le, 1: Busy ) 00 + 2	: Export(no overwrit	e), 3: Export )	
( Status : ( Result : (	0: No LW-1 (0: Id LW-1 (1: Su S	ne, 1: Import, 2 00 + 1 le, 1: Busy ) 00 + 2 ccess, 4 or more lude folder path	: Export(no overwrit	e), 3: Export )	Settings
( Status : ( Result : ( File name addre	0: No LW-1 (0: Id LW-1 (1: Su S Inc Local	ne, 1: Import, 2 00 + 1 le, 1: Busy ) 00 + 2 ccess, 4 or more lude folder path	: Export(no overwrit	e), 3: Export )	
( Status : ( ( Result : ( ) File name addre [ PLC name : (	0: No L W-1 (0: Id L W-1 (1: Su S Inc Local	ne, 1: Import, 2 00 + 1 le, 1: Busy ) 00 + 2 ccess, 4 or more lude folder path	:: Export(no overwrit 9: Error )	e), 3: Export )	Settings
( Status : ( Result : ( File name addre File name : [ PLC name : [ Address :	0: No L W-1 (0: Id: L W-1 (1: Su S Inc Local LW ress	ne, 1: Import, 2 00 + 1 le, 1: Busy ) 00 + 2 ccess, 4 or more lude folder path	:: Export(no overwrit 9: Error )	e), 3: Export )	Settings

Setting	Description		
File position	Select the external device from SD card or USB disk		
	for import/export.		
Recipe	Select the recipe.		
Control address	Designate the control address used for performing		
	recipe import/export, or displaying the result.		
	Control: Control Address		
	0: None		
	1: Import		
	2: Export (no overwrite)		
	3: Export		
	Status: Control Address+1		
	0: Idle		
	1: Busy		
	Result: Control Address +2		
	1: Success		
	4: The file already exists, no overwriting.		



	Other: Error
File name address	The file name of the imported/exported recipe. If
	[Include folder path] is selected, the full directory
	and file name will be included at this address.
Folder path address	The directory of the imported/exported recipe file.

#### Example 1

The following is an example on recipe export/import settings.

Catting
Setting
USB disk
Recipe_A (or other recipe)
LW-100
LW-200
LW-250

- 1. Create two ASCII Input objects. Set address to LW-200 and LW-250 respectively.
- 2. Enter the file name in LW-200: 2015_recipe.csv
- 3. Enter the folder path in LW-250: Setting
- **4.** Use a Set Word object to write value 3 to LW-100. Then, Recipe_A will be exported to the USB disk, in the "Setting/2015_recipe.csv" file.

## Note

When performing "Export (no overwrite)" command, if the target file already exists, the export operation will be canceled, and the result value will be set to "4".

The following lists the result values and the information.

-	
Result (HEX)	Information
0x1	Success
0x4	File already existed and will not overwrite
0x100	Data contains non-numeric data
0x101	Path contains invalid string ""
0x102	Communication error while updating Recipe DB
0x103	Error while reading Recipe DB information from
	project file
0x200	General exception
0x201	General status error
0x202	Import to unknown database type



0x203	Error while validating Recipe DB table definition
0x204	Error while validating Recipe DB table data
0x205	Error while writing Recipe DB table definition
0x206	Error while writing Recipe DB table data
0x300	File error: Unknown error
0x301	File error: Empty file name
0x302	File error: Insufficient file storage or no external
	devices
0x303	File error: Invalid file name (directory or special
	files)
0x304	File error: Unable to remove file
0x305	File error: Open file stream error
0x306	File error: Unhandled BOM
0x307	File error: Error while parsing CSV file (incorrect
	formats)
0x400	Database general exception
0x401	Database error: Unable to open table
0x402	Database error: Unable to get rows
0x403	Number of columns in CSV file and in Recipe DB do
	not match



#### 13.41. Pie Chart

#### 13.41.1. Overview

The Pie Chart object draws a pie chart that is divided into slices to illustrate numerical proportion, according to the value of the designated read address.

#### 13.41.2. Configuration



Click the Pie Chart icon on the toolbar to open the property dialog box. Set up the properties, press OK button, and a new Pie Chart object will be created.

#### **General Tab**

General Sect	unity Profile
	Angle : 7 Full , 0*
4	
	No. of channels : 4
	3 Border color :
– Data display	(1) 1 (m 1)
	Style: Value
	Size: 12
	Right of decimal Pt.: 0
Read address	
	ame : Local HMI 🔹 Settings
Addre	ess : LW • 0
	Channel : 0 V
Te	ext color : Background color :
Patte	em color : 🗾 💌 Pattern style : 🗾 📖

Setting

Description

Angle

Set the [Start degree] of the chart. Choose the Chart to be [Clockwise] or [Counter clockwise].

If [Full circle] isn't selected, then [End degree] must

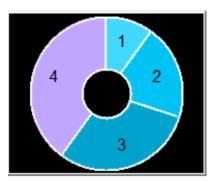


be set.

Degree			x
Start degree :	Clockwise	Counterclockw End degree :	
	Full cirde		OK Cancel

Hole

# Set the size of the hollow circle in the center of Pie Chart.



No. of channels	Set the number of channels to be displayed in the
	chart. The range is from 2 to 16.
Border color	Set the color of the border.
Data display	The [Style] can be set as [None], [Value] and
	[Percentage]. The [Font] and [Size] are for the text
	on the chart. For Value Style, the [Right of decimal
	Pt.] can set the value on the chart to be displayed
	with the decimal point. The [Right of decimal Pt.]
	option is only available for [Value] style.
Read Address	The address is for channel 1. The following
	consecutive addresses are for the rest of the
	channels. For example, if the Read Address is LW-0,
	then the Read Address for channel 2 is LW-1;
	channel 3 is LW-2and so on.
Channel	Set the [Text color], [Background color], [Pattern
	color], and [Pattern style] of the selected channel.
	The [Background color] is for the [Pattern style] that
	has background. If the [Pattern style] doesn't have a
	background, then the [Background color] doesn't
	need to be set.



#### 13.42. QR Code

#### 13.42.1. Overview

The QR Code object transfers the information from the read address into QR code.

#### 13.42.2. Configuration



Click the QR Code icon on the toolbar to open the property dialog box. Set up the properties, press OK button, and a new QR Code object will be created.

	New QR Code Object
	General
	Comment :
	Correction level : L (7%) Color :
	PLC name : Local HMI Settings
Setting	Description
Correction level	QR code has error correction capability to restore
	data if the code is dirty or damaged.
	Four correction levels are available: L, M, Q, and H.
	The data restoration rate is listed below. (The data
	restoration rate for total codewords. Codeword is a
	unit that constructs the data area.)
	Correction Level
	L 7%
	M 15% Q 25%
	H 30%
Color	Set the QR code color.
Read address	The QR Code object will display the QR code
	generated from the information entered by the read
	address. The word length limit: 1 ~ 1024.



# 14. Shape Library and Picture Library

This chapter explains how to build Shape Library and Picture Library.

14.1.	Overview	14-2
14.2.	Building Shape Library	14-2
14.3.	Building Picture Library	14-9



#### 14.1. Overview

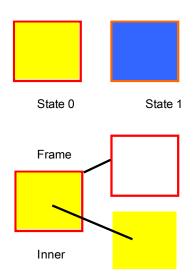
EasyBuilder Pro provides Shape Library and Picture Library for visual effects on objects. Picture Manager provides two modes: [Project] and [Library]. Pictures in [Project] mode will be stored in .emtp project file. Pictures in [Library] mode will be stored in EasyBuilder Pro libraries, or the user-defined directory.

Each Shape or Picture includes up to 256 states. This chapter explains how to build Shape Library and Picture Library.

For more information about using libraries while creating an object, see "9 Object General Properties".

#### 14.2. Building Shape Library

Shapes are vector graphics constructed by lines, curves or polygons. A Shape can have more than one state, and each state includes two parts: frame and inner, as shown in the following figure.



#### 14.2.1. Shape manager

An object can use frame, inner or both. Click [Call up Shape Library], and the [Shape manager] dialog box appears.



### 

Project Library	
button4	
Untitled	1 Untitled States : 2 Objects : 0
2 Untitled States : 2 Objects : 0	Junitided       States : 2       Objects : 0       0       Frame
f	States : 2       Objects : 0
Ktatar + 7	7     Image: Constraint of the sector of the s
Setting	Description
Project	The Shape edited here will be saved in .emtp. Up t
-,	1000 Shapes can be added.
Library	The Shape edited here will be saved to the library directory on PC and will not be saved to .emtp project file.
New library	Include existing .plb shape library files or create a new one. To create an empty library, enter a new file name and click [Open]. Up to 40 library files ca be added.
🔕 Unattach library	Exclude current library.
逐 Copy to project	Copy the selected Shape to [Project]. Only the shapes that do not belong to the System Libraries can be copied. Shapes in System Frame/System Button/System Lamp/System Pipe cannot be copied.
Background	Select and preview the background color of the Shape. The color is only displayed in [Shape manager] dialog box, and is not displayed when placing the object in the screen.
Place	Add the selected Shape to window. Only available



More options	Set the color and style of [Inner], [Frame], and
	[Pattern].
	Move the Shape to the previous / next state.
🖲 Сору	Copy the selected Shape.
间 Paste	Paste the copied Shape.
Insert transparent	Insert a blank state after the selected state.
state	
Delete	Delete the selected state of the shape.
Clean	Delete all the states of the selected shape.
ОК	Confirm to save the edited Shape.
Cancel	Cancel the editing event.
Help	Open help files.

# Note

The color of [Inner] and [Frame] can be selected in Shape Library. The selection of [Pattern Style] is only available in System Frame / System Button Library.

Bit Lamp Object's Properties	×
General Security Shape Label Profile	
Shape	-1
Shape Library	
✓ Inner ✓ Frame	
Interior pattern : Pattern Style	
Duplicate these attributes to every state	
	21

CMT-SVR Series supports using gradient patterns in [Pattern Style], as follow:

attern Style	×
○ Solid	
Left to right :	_
Top to bottom :	
Top-left to bottom-right :	
Top-right to bottom-left :	
Radial :	_
ОК	



#### 14.2.2. Steps to Build Shape Library

The following explains how to create a new Shape Library and add a Shape with two states into the library.

1. Click [New library] and enter the name of the new Shape Library.

💽 Open						×
Look in:	鷆 library		•	ø	🖻 🛤	•
Recent Places Desktop Libraries Computer Computer	Name picture shape sound arrow1.plb arrow2.plb button1.plb button2.plb button3.plb button4.plb din1.plb frame.plb pipe1.olb	*			Previ	iew not available.
Network	File <u>n</u> ame:	new_lib				▼ Open
	Files of type:	shape library (*.plb)				Cancel

- 2. Click [Open], a popup dialog appears; click [Yes] to create the file.
- A new Shape Library [new_lib] is added in [Shape manager]. This library is empty as shown in the following figure.

Shape manager		×
Project Library		
new_lib	- 🔊 🔞	
0 States : 0 Objects : 0	1 States : 0 Objects : 0	
2 States : 0 Objects : 0	3 States : 0 Objects : 0	More options
4 States : 0 Objects : 0	5 States : 0 Objects : 0	
6 Rtster + 0	7 Ktatae + 0	Delete Clean
	Background :	OK Cancel Help

**4.** Add a state to the selected Shape. First, use the drawing tools to draw a frame and inner in the window and select the frame to add to the Shape Library.



🔣 EasyBuilder Pro : EMTP2 - [:	.0 - WINDOW_010 ]	
Eile Edit View Optio	n <u>D</u> raw <u>O</u> bjects Library <u>T</u> ools <u>W</u> indow <u>H</u> elp	- 8 ×
) D 🚅 🔛 i 🐰 🖻 🛍 🖸	요   육 🤋 😢   🙀   🧕 🏢 本 ! 🛠 👳 😰 💁 🏧 🖉 🔠 😡 🖉 💆 👹 🕼 🛃 🗟 🗟	🎝 🖏 🛥 📓 🕽
N 6 N 6 N 6 O	은 🗆 🌣 🏭 🕰 🖬 🔄 📴 😒 🧠 📟 🕾 🚥 📟 🗷 🛞 날 🗄 🔛 🔶	🗱 🛍 🔿 🚧 🥅
	10-WINDOW_010 X	
Object list 🔹		<u> </u>
- 3 : Fast Selection 🔺		
- 4 : Common Windo		
5 : PLC Response		
6 : HMI Connection		
7 : Password Restrict		
8 : Storage Space In 9 : Backup		
11		
13		
14		=
15		
16		
17		
18		
19		
20		
- 24		
26		
- 27		
- 28 -		
		-
位址 視窗 網頁	۲. III	Þ
(56, 81) - (157, 149)	eMT3105 (800 x 600) Width : 102 Height : 69 X = 223 Y = 10	CAP NUM SCRL

- 5. Click [Save to Shape Library] button in the toolbar, select [new_lib], and select a number in this library. The selected number is highlighted yellow.
- 6. Save the Shape as [Frame], select [Insert], and click [Save].

Inner	
V Display	
Frame	
V Display	
Save to library	
Save to library	
Save as :	
Frame	🔘 Inner
Insert options :	
Insert	Replace
	Save

Setting	Description	
Inner	Displays the inner of the Shape.	
Frame	Displays the frame of the Shape.	
Save to library	Save as Frame	
	Saves the Shape as a frame.	
	Save as Inner	
	Saves the Shape as inner.	



#### Insert

Inserts the Shape to be a new state.

#### Replace

Replaces a state with this Shape.

Save Saves the settings above.

7. The following shows that a state of the Shape is added, and is defined as a frame.



8. Create the shape to be saved as inner. Select the shape drawn in the window.

E EasyBuilder Pro : EMTP2 - [10 - WINDOW_010]
Eile Edit View Option Draw Objects Library Tools Window Help
: D 🛎 🖬   ふ 🖻 🗈 🗠   刍 ያ 😢 👒   ∠ 🏢 本 ! 🛠 🧝 壑 🕾 🖉 🖓 🖩 😡 🌽 😳 🖬 😓 🖓 👘 🖆 🖼 🕨 🚳
IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
Object list
- 3 : Fast Selection
4 : Common Windo
- 5 : PLC Response
- 0 : mui connection - 7 : Password Restrict
- 8:Storage Space In
9: Backup
*10 : WINDOW_010
11
- 12
- 13
14
- 18
19
21
- 22
- 23
24
- 25 - 26
- 27
-28 -
· · · · · · · · · · · · · · · · · · ·
位北 視窗 網頁 ( )
(188, 85) - (286, 150) eMT3105 (800 x 600) Width : 99 Height : 66 X = 262 Y = 11 CAP NUM SCRL

 Click [Save to Shape Library] button in the toolbar, select [new_lib], and select the same number as in creating the frame in this library. The selected number is highlighted yellow.

Project Library		
	ites : 0         Image: Constraint of the second secon	States : 0 Objects : 0

**10.** Save the Shape as [Inner], select [Replace], and click [Save].

Inner	
Display	
,	
Frame	
V Display	
Save to library	
Save as :	
Frame	Inner
Insert options	
Insert	Replace
	Save

**11.** A state of a Shape can include [Inner], [Frame], or both. The state 0 of the Shape shown in the following figure includes both frame and inner. Click [OK], the state 0 of the Shape is created.



**12.** Follow the steps of creating state 0 and insert a new state set to state 1 as shown in the following figure. The Shape now has two states, click [OK] to finish setting.

Shape manager			
Project Library	• 🕅 🔞		Inner 🕼 Display
0 States : 2	States : 0		Frame I Display
2 3	Objects : 0		Save to library
States : 0 Objects : 0	States : 0 Objects : 0	Place More options	
4 <b>5</b>		0 Inner,Frame	
States : 0	States : 0 Objects : 0	1 Inner,Frame	Save as : Frame Inner Insert options :
6 7 Kitalar + 0	Stater • 0	Delete Clean	Insert     Replace     Save
	Background :	OK Cancel Help	





#### 14.3. Building Picture Library

#### 14.3.1. Picture manager

Click [Call up Picture Library] button in the toolbar and the [Picture manager] dialog box appears.

-

			l up Picture Library						
		- ICa	i op riciole Libiary						
Pictures manager									<b>x</b>
Project Library									
-0		_1	K	3 🐼	7	8	9	Clr	Esc
	States : 1	7 8 9 07 FW	States : 1		4	5	6	BS	Del
	Objects : 1	4 5 6 55 w 1 2 3 4 . 0 - Enter	, Objects : 1		1	2	3		
FI 0 0 0	States : 1		States : 1		Expor		Modify	En	ter
4	Objects : 1	5	Objects : 1		0	BM 166	IP 267x2 6482 byte		
	States : 1 Objects : 1		States : 2 Objects : 1						
	Ktatar • ?		Ktatar + 7	Ŧ	New		Delete	Cle	
More picture libraries			Background :			OK	Cano	el 🛛	Help

Setting	Description
Project	The Picture edited here will be saved in .emtp. Up
	to 1000 Pictures can be added.
Library	The Picture edited here will be saved to the library
	directory on PC and will not be saved to .emtp
	project file.
[7]	Add the existing .flb picture library files.
🕙 New library	To add a new library that does not exist, enter a
	new file name and click [Open], an empty library
	file is created. Up to 40 library files can be added.





🔕 Unattach library	Delete the current library.
🐼 Copy to project	Copy the Picture to [Project].
Background	Select the background color of the Picture. The
	color is only displayed in [Picture manager] dialog
	box, and is not displayed when placing the object in
	the screen.
More picture	Log in to Weintek Official Website to download
libraries	more libraries.
Export	Export the selected Picture.
Modify	Modify the settings of the selected Picture.
	Move the Picture to the previous / next state.
🖲 Сору	Copy the selected Picture.
1	Paste the copied Picture. The Picture copied to the
🕮 Paste	clipboard can be imported to the library by pasting.
Insert transparent state	Insert a blank state after the selected state.
New	Add a new Picture.
Delete	Delete the selected Picture.
Clean	Delete all the Pictures listed here.
ОК	Confirm to save the edited Shape.
Cancel	Cancel the editing event.
Help	Open help files.



The supported picture formats are .bmp, .jpg, .gif, .dpd, .svg and .png. When adding a gif animation file in Picture Library, the loop times of this animated Picture can be set.

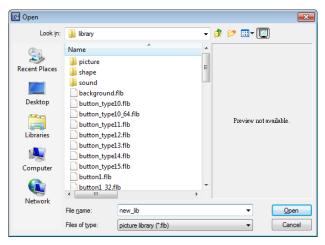




#### 14.3.2. Steps to Build Picture Library

The following example explains how to create a new Picture Library and add a Picture with two states into the library.

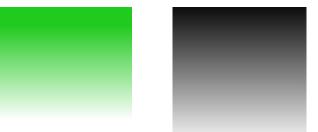
1. Click [New library] and enter the name of the new Picture Library.



- 2. Click [Open], a popup dialog appears; click [Yes] to create the file.
- 3. A new Picture Library [new_lib] is added in [Picture manager]. This library is empty as shown in the following figure.

Pictures manager							×
Project Library							
_new_lib			•	8			
	States : 0 Objects : 0	1	States : 0				
2	States : 0 Objects : 0	3	States : 0		Export	Modify	
4	States : 0 Objects : 0	5	States : 0	2			
6	Ktater + 0	7	Ktater + 0		New	Delete	Clean
More picture libraries			Background :		ОК	Cancel	Help

4. Draw the two pictures below to represent state 0 and state 1 respectively.



 Select [new_lib], and select a number in this library. The selected number is highlighted yellow.



- 6. Click [New], and select the picture for state 0.
- 7. When the following dialog box is shown, select [Enable] check box to use transparent color. Set to RGB (121, 121, 121), the corresponding color in the picture below will be transparent. Or, click on a desired area with mouse to be the transparent area, the system will show the RGB of the clicked area automatically.

💽 Open					×
Look in:	🧮 Desktop		•	G 🕸 📂 🛄 -	
Recent Places	Librari Syster	i <b>es</b> n Folder	13	<b>user</b> System Folder	
Desktop	Comp Syster	u <b>ter</b> n Folder		Network System Folder	
	EN File fo	lder		<b>TW</b> File folder	
Libraries	1.bmp Bitma 29.3 K	p image		<b>2.bmp</b> Bitmap image 29.3 KB	
Computer					
Network	File <u>n</u> ame:	1.bmp		-	Open
	Files of type:	All image files		•	Cancel
			Transparent color	0 x 100 30054 Bytes	
Bac	kground :				

- 8. To set transparent color, select [Enable] check box first, and then click an area in the picture. The RGB value of that area is shown and it will be transparent. The displayed picture is shown as in the preceding figure.
- *9.* The Picture of state 0 is created. Follow the steps of creating state 0 to create state 1 by clicking [New] as shown in the following figure.



Pictures manager						×
Project Library			- 🗑 🔞			
	States : 2 Objects : 1	1 States : 0 Objects : 0				
2	States : 0 Objects : 0	3 States : 0 Objects : 0	 	Export	Modify BMP 100x100	
4	States : 0 Objects : 0	5 States : 0 Objects : 0	······································	0	30054 bytes BMP 100x100 30054 bytes	
6 More picture libraries	Rtater • 0	7		New		ean Help

*10.* When finished, a complete Picture is created, click [OK]. In [Picture manager] dialog box it shows that the newly added Picture Number 0 is a bitmap picture with two states.

#### 14.3.3. Steps to Import Picture by Pasting

The following example explains how to import a Picture into the library by pasting the picture from the clipboard.

1. Copy the following picture to the clipboard.



2. Click the Paste icon on the right side.

Picture Manager		×
Dbjects : 1	Objects : 1	
4 States : 1 Objects : 1	5 States : 2 Objects : 1	
6 States : 2 Objects : 1	7 Fiates : 2 Objects : 1	Export Modify
B States : 1 Dbjects : 1	States : 0 Objects : 0	
More picture libraries	Background :	OK Cancel Help



3. The Pictur can be easily imported to the library.





Transparent color can only be set for .bmp, .dpd, and .jpg picture files.



14-14



# 15. Label Tag Library and Multi-Language

This chapter explains how to build and use Label Tag Library.

15.1.	Overview	. 15-2
15.2.	Label Tag Library Manager	. 15-2
15.3.	Steps to build Label Tag Library	. 15-3
15.4.	Using Label Tag Library	. 15-4
15.5.	Settings of Multi-Language	. 15-5



#### 15.1. Overview

The Label Tag Library feature enables a multi-language environment. When multiple languages are required, users can create the Label Tag Library and then select a suitable label in the project. The project will display the corresponding language in runtime based on the settings. EasyBuilder Pro supports up to 24 different languages simultaneously. This chapter will explain how to create and use the Label Tag Library.

#### 15.2. Label Tag Library Manager

Click [Library] » [Label] on the toolbar and the [Label Tag Library] dialog box appears.

Label	Tag Library											×
Labe	Font											
	Language no. : 24	•										
	State no. : 31	•	▲ ▶ 0	12	3 4 5	6 7						
٩	No. Label tag name	No. of states	Language 1	Language 2	Language 3	Language 4	Language 5	Language 6	Language 7	Language 8	Language 9	Langu
		III										•
	New	Settings		Delete	Dele	te All		S	ave Label File		Load Label File	t
	Сору	Paste						Ex	port EXCEL F	ile Ir	nport EXCEL F	ile
								ОК	C	ancel		Help

Setting	Description
Language no.	Specifies the number of languages used in a project.
State no.	Indicates the current state. Each Label has a maximum of 256 states (state no. 0 ~ 255). The state no. is determined by [Language no.]. If less than 3 languages are used, the maximum state no. is 256. If more than 4 languages are used, divide 768 by the language number to get the maximum state no For example, the number of languages is 24, then there are only 768/24 = 32 states.
New	Adds a new Label.
Settings	Sets the selected Label.
Save Label File	Saves all Labels in .lbl format.
Load Label File	Loads the existing .lbl file to the Label Library.
Export EXCEL File	Saves all Labels in .csv, .xls, or .xlsx format.
Import EXCEL file	Loads the existing .csv, .xls, or .xlsx file to the Label Library.



### Note

Unicode is not supported when importing and exporting an Excel file.

#### 15.3. Steps to create Label Tag Library

Please follow the steps to create a Label Tag Library.

1. From the Library menu, click [Label]. The Label Tag Library dialog box appears. Click [New] to specify the name of the Label and the number of states to be displayed by this Label.

La	ibel	ĺ	×
	Label name :	Label_0	
	No. of states :	1	
		OK Cancel	

 Click [OK] and a new label is added to the Label Tag Library. Select the label and click [Settings] to edit its content.

_													
Lab	el Tag	g Library											×
La	bel	Font											
	La	nguage no. : 24	-										
L.		State no. : 1	•		12	3 4 5	6 7						
	No.	Label tag name	No. of states	Language 1	Language 2	Language 3	Language 4	Language 5	Language 6	Language 7	Language 8	Language 9	Langu
	1	Label_0	1										
Ш													
Ш													
Ш													
Ш													
Ш													
Ш													
	•		III										E.
		New	Settings		Delete	Dele	te All		S	ave Label File		oad Label File	
		Сору	Paste						Ex	port EXCEL Fi	le In	nport EXCEL F	le
									ОК		ancel		Help

3. Edit the corresponding language content.

Label name :	Labe	0_1							State no. : 0			0	1	2 3 4 5 6	
anguage 1			Language 2			Language 3		Language	4		Language 5			Language 6	
now are you		* *	您好		~			~ ~		~			* *		4
€	Þ		•	Þ		< I	•	•	F.		•	Þ.		< +	
nguage 7			Language 8			Language 9		Language	10		Language 11			Language 12	
		*			*			A		*			*		-
		-			-			-		÷			÷		-
€	÷.			Þ.		۲ I	•		ł.		*	F		< →	
anguage 13			Language 14			Language 15		Language	16		Language 17			Language 18	
		^			^			A		^			^		-
		Ŧ			Ŧ			-		Ŧ			Ŧ		-
€	- F		*	+		۰ ا	Þ.	•	4		*			< >	
nguage 19			Language 20			Language 21		Language	22		Language 23			Language 24	
		^			~			A		~			*		-
		Ŧ			Ŧ			~		Ŧ			Ŧ		-
€	- F			P.		< I	Þ	*	÷.		*	P		< >	



**4.** Select [Label Tag Library] » [Font] to view each label which contains different fonts for different languages. You can also enter the font description in the Comment field.

t			
No. 🔺	Font	Comment	
▶ 1	Arial		
2	Arial Black		
3	Arial Bold		
4	Batang		
5	Browallia New		
6	Browallia New Bold		
7	Calibri		
8	Calibri Bold		=
9	Cambria		1
10	DaunPenh		
11	David		
12	David Bold		
13	Ebrima		
14	Ebrima Bold		
15	Estrangelo Edessa		
16	FrankRuehl		
17	Franklin Gothic Medium		_
18	FreesiaUPC		
19	Gabriola		
20	Gautami		
	A 1 1		_

#### 15.4. Using Label Tag Library

When there are defined labels in the Label Library, the labels can be found in the object's [Label] tab. Select [Use label library] check box, and select the label from the pull-down list [Label tag].

w Bit Lamp	-
eneral   Sec	urity Shape Label
Use la	hel
	bel library Label tag : Label_1
	Label 0
Conve	ert labels to bitmap images (Use bitmap Label_1
Lan	juage : 1 👻
	State : 0 🗸 🖌 🕨 0 1
Attribute	
	Font : Arial
	Color : Size : 16
	Align : Left  Blink : None
	Italic Underline
	Duplicate these attributes to
	Every state Every language All
Movement	
	ection : No movement 🔻
Content	
text	
lexi	
Tracking	
Tracking	Duplicate this label to every state
	OK Cancel Help



When a tag is selected, the content of the selected tag is shown in the [Content] field in its corresponding font style. Please note that from Language 2 to Language 24 can only be set the Font [Size], the others such as [Color], [Align], [Blink], etc. will follow the settings of Language 1.

#### 15.5. Settings of Multi-Language

When displaying the texts in multiple languages, the system register "[LW-9134]: language mode" should be used too.

The value of [LW-9134] is ranged from 0 to 23 (cMT Series is from 0 to 7). Different values correspond to different languages.

If not all languages are selected to compile and download, [LW-9134] will work differently. For example, user defines 5 different languages in the Label Library:

1: English, 2: Traditional Chinese, 3: Simplified Chinese, 4: French, 5: Korean

If only Language 1, Language 3, and Language 5 are selected to compile then the corresponding values of [LW-9134] are:

0: English, 1: Simplified Chinese, 2: Korean

Please follow the steps to use multiple languages.

1. Create a Text object and select [Use label library] checkbox.

New Text Object	
Text	
Use label library Label tag : Label_0	•
Convert labels to bitmap images (Use bitmap font)	bel Library
Language : 1	
Attribute	
Font : Arial	
Color : Size :	16 🔻
Align : Left  Blink :	None 🔻
Every language	
Every language	
Movement	
Direction : No movement	
Content	
text	
ioni ioni	
OK Cancel	Help
Cancer	Thep



2. Create a Numeric Input Object and use the system register [LW-9134].

New Nun	neric Input Object 🧧
General	Data Entry Numeric Format Security Shape Font
	Description :
Read	Read/Write use different addresses
PLC	name : Local HMI    Setting
Ad	ddress : LW-9134 (16bit) : language mode 🔹

3. When compiling, select the defined languages.

Compiling						×					
Project name : C:\Users\user\Desktop\EMTP1.emtp											
EXOB file name :	C:\Users\user\Desk	top\EMTP1.exob									
EXOB password :	Setting	) (used in decompiler)	) 🔲 Deco	mpilation is prohibited							
Select the languages used on the HMI Startup language after redownloading the project : Language 1											
<ul> <li>✓ Language 1</li> <li>✓ Language 7</li> <li>Language 13</li> <li>Language 19</li> <li>Note : A maximum</li> </ul>	Language 2     Language 8     Language 14     Language 20     to of 8 languages can	Language 3     Language 9     Language 15     Language 21     be selected simultane	Language 4     Language 10     Language 16     Language 22 eously.	✔ Language 5 Language 11 Language 17 Language 23	Language 6 Language 12 Language 18 Language 24						
Macro size : Address tag size : Label tag size :	14 bytes 10 bytes 40 bytes					*					
0 error(s), 0 warning(s succeeded	)					Ŧ					
Double click error mess	ages to modify the a		bjects !		Close						

**4.** The simulation is shown as followed: If the value of [LW-9134] is changed, the content of the Text object will be changed.

	English
LW9134 : language mode	0
	简体中文(SIMPLE)
LW9134 : language mode	2
	한국어 웹(KOREAN)
LW9134 : language mode	4



### Note

- For cMT Series, at most 8 different languages can be downloaded to the HMI.
- When selecting the HMI model: cMT, [LW-9134] is used to change the language mode on cMT model, while [PLW-9134] is to change the language mode on iPad.



object to switch between multiple languages. Please confirm your internet connection before downloading the demo project.



## 16. Address Tag Library

This chapter explains how to build and use Address Tag Library.

16.1.	Overview	16-2
16.2.	Building Address Tag Library	16-2
16.3.	Using Address Tag Library	16-4



EasyBuilder Pro V5.02.01

#### 16.1. Overview

Generally it is recommended to define the commonly used addresses in Address Tag Library when starting to build a project. It not only avoids accidental reuse of addresses but also improves project readability.

#### 16.2. Building Address Tag Library

Click [Library] » [Tag] on the toolbar and the [Address Tag Library] dialog box appears.

User	-defined tags	🔘 System tag	IS						
No.	Address tag name		PLC name	Addre	Address	Read/W	Data type	Comment	
1	Tag_0		Local HMI	Bit	LB-0	Read/			
			m						•
۲ Edit ؛	system_tag.xml to	customize categorie							•
	system_tag.xml to	customize categorie Delete			Settings				•

Setting	Description		
Customized	Displays user-defined address tags.		
System	Displays system registers. The registers listed cannot		
	be deleted or changed.		
New	Adds a new address tag. Please see the steps next		
	page.		
Settings	Sets the selected address tag.		
Save Tag File	Saves all current address tags as .tgl file.		
Load Tag File	Loads the existing .tgl file of address tag to the current		
	project.		
Export CSV	Saves all current address tags as .csv file.		
Import CSV	Loads the existing .csv file of address tag to the		
	current project.		
Export EXCEL	Saves all current address tags as .xls file.		
Import EXCEL	Loads the existing .xls file of address tag to the current		
	project.		
Use UTF-8 format	If selected, the .csv file will be exported in UTF-8		
to export CSV file	format. If not selected, in ANSI format.		



1. Click [New] and set the relevant properties.

Address Tags	×
Comment : test pump	
Tag name : pump	
Address	
PLC name : MODBUS RTU	•
Address type : O Bit O Word	
Device type : 3x	•
Address : 0	
Address format : DDDDD [range : 1 ~ 65535]	
Conversion (Use macro)	
Enable	
Data format : 16-bit BCD	•
Read conversion : add	•
Write conversion : None (Only data type conversion)	•
Array Size : 2	•
OK Cancel	

Setting	Description
Comment	The information of the address tag.
Tag name	The name of the address tag.
PLC name	As defined in [System Parameter Settings] » [Device list].
Address type	The tag address type; select [Bit] or [Word].
Device type	The available device types depend on [PLC name] and
	[Address type].
Address	Address of the tag.
Data format	If select [Word] in [Address type], the data format can be
	specified.
Conversion	When enabled, the data format that the address tag will
(Use Macro)	be converted into can be specified. Macro subroutines can
	be selected to do read/write conversion.
Read / Write	Select the macro subroutine to do read/write conversion.
convertion	The macro subroutine can only be selected when the data
	format is identical to the one in the macro subroutine.



2. Click [OK], a newly added tag can be found in the [User-defined tags] library.

User-defined tags	🔘 System tag	S						
No. Tag name		Conversion	Data format	Original format	Array size	PLC name	Addr	Ad
1 pump		Enable	16-bit BCD	16-bit Unsigned	2	MODBUS RTU	Word	Зх
Edit system_tag.xml t	o customize categorie:	III s of system tags	3					Þ
Edit system_tag.xml t	o customize categorie		70	ettings				F
New	Delete	s of system tags	All					+
(		s of system tags	70					t

#### 16.3. Using Address Tag Library

- **1.** Create a tag in Address Tag Library.
- 2. Create an object, select [General] » [PLC name].
- **3.** Click [Setting] to finish the settings.
- 4. Select [User-defined tag] check box.

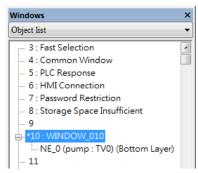
	New Nun General	neric Input Data Entry		Security	Shape	Font	l	<b>x</b>
		Data Entry Description :	Numeric Pormat	security	Shape	FOIL		
	PLC	address name : Mitsu	rite use different ad ubishi FXOS/FXON/F o (test pump)			(	Setting	
Addre		areas . [puint	(test pump)			I		×
	PLC na	me : Mitsubi	shi FX0S/FX0N/FX	IS/FX1N/F	(2			•
	Device ty	/pe : pump (	test pump)					<b>-</b>
	Addre	ess: TV-0					User-defined	tag
A	ddress forn	nat:DDD [ra	nge : 0 ~ 255]					
				Ind	ex registe	er		
	Tag Librar	у			C	OK		Cancel

- 5. In [Device type] select the defined tag.
- *6.* If [Data type] is selected when creating the address tag, the system automatically restricts the data format to the one selected.



New Numeric	: Input Object			×
General Dat	a Entry Numeric Forms	t Security	Shape Font	
Display — Data i	cormat : 16-bit Unsigned	<b>-</b>	Mask	
-Number of Left c	digits f decimal Pt. : 4	-	Right of decimal Pt. : 0	<b>*</b>
Scaling	Method : None		•	

7. When finished, the window tree will show the address tag name used by the object.





## 17. Transferring Recipe Data

This chapter explains how to transfer recipe data.

17.1.	Overview	17-2
17.2.	Steps to Update Recipe Data with Ethernet or USB Cable	17-2
17.3.	Steps to Update Recipe Data with SD Card or USB Disk	17-3
17.4.	Transferring Recipe Data	17-3
17.5.	Saving Recipe Data Automatically	17-4



#### **17.1.** Overview

Recipe Data refers to the data stored in RW and RW_A addresses. The way of reading and writing these addresses is the same as operating a word register. The difference is that recipe data is stored in flash memory, when restarting HMI, the latest data records in RW and RW_A are kept.

The size of recipe data a RW address can store is 512K words, and RW_A is 64K words. Users can update recipe data with SD card, USB disk, USB cable or Ethernet and use the data to update PLC data. Recipe Data can also be uploaded to PC; furthermore, PLC data can be saved in recipe data. The following explains the ways of transferring recipe data.

#### 17.2. Steps to Update Recipe Data with Ethernet or USB Cable

- 1. Open Utility Manager and click [Download].
- 2. Select [RW] and [RW_A] and [Browse] the source file.
- 3. After downloading, restart HMI, RW and RW_A will be updated.

When [Reboot HMI after download] is selected, users don't have to manually reboot HMI. When [Reset recipe] check box is selected, the system will clear all the data in [RW] and [RW_A] before downloading.

Download				×
Firmware				
Project	C:\EBpro\tes	t.rcp		Browse
RW	C:\EBpro\tes	t_A.rcp		Browse
E RW_A				
Recipe databas	e			
🗖 Data log				
C Startup screen				
Connection で Ett 引 IP 名額		USB cable		<u>(4</u>
	IP: 192.168.1.222	2		
Reboot HMI af	er download			
Reset recipe	Reset	recipe database	<ul> <li>Delete startup scr</li> </ul>	reen
Reset event log	Reset	data log	Reset operation l	og
Download	Stop	Settings		Exit



#### **17.3.** Steps to Update Recipe Data with SD Card or USB Disk

- 1. Open Utility Manager and click [Build Download Data for SD Card or USB Disk].
- 2. Insert a SD card or USB disk into PC.
- 3. Click [Browse] to designate the file path.
- 4. Click [Build], EasyBuilder Pro will save the data in SD card or USB disk.

Utility Manager	×
Select the folder to save download data : D:\test	Browse
Jo / Cooc	brottocini
_ Sources	
Project	
₩ Recipe (RW)	
C:\EBpro\test.rcp	Browse
	browsen
Recipe A (RW_A)	
C:\EBpro\test_A.rcp	Browse
🗖 Data log	
Build	Exit



When download data is successfully built, two folders can be found: *history* and *emt3000*. *emt3000* is for storing project file; *history* is for storing recipe data and data sampling / event log records.

#### 17.4. Transferring Recipe Data

Use [Data Transfer (Trigger-based) Object] to transfer recipe data to a specific address, or save the data of this address in [RW] and [RW_A].



New Data Transfer (Trigger-based) Object
General Security Shape Label
Comment :
Source address
PLC name : MODBUS RTU
Address : 3x 🔻 1
Destination address PLC name : Local HMI Address : RW_A 10
Attribute No. of word : 120 Mode : Touch trigger
OK Cancel Help

Setting	Description	
Source address	Sets the source of the data.	
Destination address	Sets the destination of the data to transfer to.	
Attribute Sets the number of words to transfer fro		
	to destination.	

#### 17.5. Saving Recipe Data Automatically

In order to prolong the life span of HMI flash memory, the system will automatically save the recipe data to HMI every minute. To avoid losing data when turning HMI off during the interval of saving data, system register [LB-9029: Save all recipe data to machine (set ON)] is provided. Set ON LB-9029 will make the system save recipe data once. Set ON [LB-9028: Reset all recipe data (set ON)], the system will clear all recipe data.



## 18. Macro Reference

This chapter describes the syntax, programming methods and usage of macro commands.

18.1.	Overview	
18.2.	Instructions to use the Macro Editor	
18.3.	Configuration	
18.4.	Syntax	
18.5.	Statement	
18.6.	Function Blocks	
18.7.	Built-In Function Block	
18.8.	How to Create and Execute a Macro	
18.9.	User Defined Macro Function	
18.10.	Some Notes about Using the Macro	
18.11.	Use the Free Protocol to Control a Device	
18.12.	Compiler Error Message	
18.13.	Sample Macro Code	
18.14.	Macro TRACE Function	
18.15.	Example of String Operation Functions	
18.16.	Macro Password Protection	





#### 18.1. Overview

Macros provide the additional functionality your application may need. Macros are automated sequences of commands that are executed at run-time. Macros allow you to perform tasks such as complex scaling operations, string handling, and user interactions with your projects. This chapter describes syntax, usage, and programming methods of macro commands.

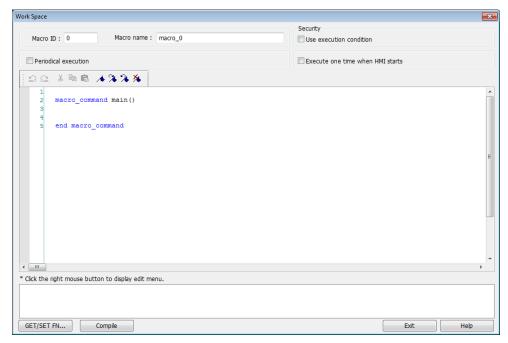
#### **18.2.** Instructions to use the Macro Editor

Macro editor provides the following functions:

- Display line number
- Undo / Redo
- Cut / Copy / Paste
- Select All
- Toggle Bookmark / Previous Bookmark / Next Bookmark / Clear All Bookmarks
- Toggle All Outlining
- Security -> Use execution condition
- Periodical execution
- Execute one time when HMI starts

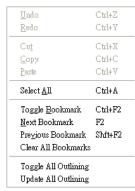
The instructions in the following part show you how to use these functions.

 Open the macro editor; you'll see the line numbers displayed on the left-hand side of the edit area.





2. Right click on the edit area to open the pop-up menu as shown in the following figure. Disabled operations are colored grey, which indicates that it is not possible to use that function in the current status of the editor. For example, you should select some text to enable the copy function, otherwise it will be disabled. Keyboard shortcuts are also shown.



3. The toolbar provides [Undo], [Redo], [Cut], [Copy], [Paste], [Toggle Bookmark], [Next Bookmark], [Previous Bookmark] and [Clear All Bookmarks] buttons.



4. Any modification will enable the [Undo] function. [Redo] function will be enabled after the undo action is used. To perform the undo/redo, right click to select the item or use the keyboard shortcuts. (Undo: Ctrl+Z, Redo: Ctrl+Y).

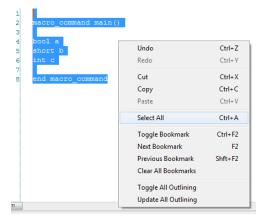
pace			Security		
Aacro ID : 0 Macro na	ame : macro_0		Use execution condition		
Periodical execution			Execute one time when H	II state	
			Execute one time when Hi	41 scarcs	
2 CL & Bo B A 74 74 74 74	•				
1 2 macro_command main() 3					
abc abc abc					
5	Undo	Ctrl+Z			
6 7 end macro_command	Redo	Ctrl+Y			
	Cut	Ctrl+X			
	Сору	Ctrl+C			
	Paste	Ctrl+V			
	Select All	Ctrl+A			
	Toggle Bookmark	Ctrl+F2			
	Next Bookmark	F2			
	Previous Bookmark	Shft+F2			
	Clear All Bookmarks				
	Toggle All Outlining				
	Update All Outlining				
k the right mouse button to display e	dit menu.				
pace		_			
here			Security		
Macro ID: 0 Macro n	ame : macro_0		Use execution condition		
Periodical execution			Execute one time when	HMI starts	
22 1 1 1 1 1 1 1 1 1 1 1	4				
1	•				
<pre>2 macro_command main() 3</pre>					
3 4 abc					
s	Undo	Ctrl+Z			
6 7 end macro command	Redo	Ctrl+Y			
	Cut	Ctrl+X			
	Сору	Ctrl+C			
	Paste	Ctrl+V			
	Select All	Ctrl+A			
	Toggle Bookmark	Ctrl+F2			
	Next Bookmark	F2			
	Previous Bookmark	Shft+F2			
	Clear All Bookmarks				
	Toggle All Outlining				
	Update All Outlining				
k the right mouse button to display e					



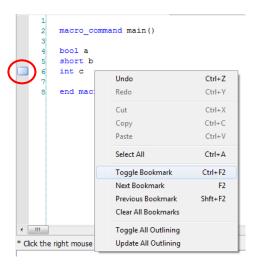
5. Select a word in the editor to enable the [Cut] and [Copy] function. After [Cut] or [Copy] is performed, [Paste] function is enabled.

abc			з	-	mand main()	
	Undo	Ctrl+Z	4	abc abc	Undo	Ctrl+Z
enc	Redo	Ctrl+Y	6		Redo	Ctrl+Y
	Cut	Ctrl+X	7	end macr	Cut	Ctrl+X
	Сору	Ctrl+C			Сору	Ctrl+C
	Paste	Ctrl+V			Paste	Ctrl+\
	Select All	Ctrl+A			Select All	Ctrl+A
	Toggle Bookmark	Ctrl+F2			Toggle Bookmark	Ctrl+F2
	Next Bookmark	F2			Next Bookmark	F2
	Previous Bookmark	Shft+F2			Previous Bookmark	Shft+F2
	Clear All Bookmarks				Clear All Bookmarks	
	Toggle All Outlining				Toggle All Outlining	
	Update All Outlining				Update All Outlining	

6. Use [Select All] to include all the content in the edit area.



- **7.** If the macro is too long, use bookmarks to manage and read the code with ease. The following illustration shows how it works.
- Move your cursor to the position in the edit area where to insert a bookmark. Right click, select [Toggle Bookmark]. There will be a blue little square that represents a bookmark on the left hand side of edit area.





- If there is already a bookmark where the cursor is placed, select [Toggle Bookmark] to close it, otherwise to open it.
- Right click and select [Next Bookmark], the cursor will move to where the next bookmark locates. Selecting [Previous Bookmark] will move the cursor to the previous bookmark.

Work Space	
Macro ID : 0 Macro name : macro_0	Security
Periodical execution	Execute one time when HMI starts
<u>□</u> □ ≤ % № ® <b>∧ % % %</b>	
<pre>1 macro_command main() 3 bool a 5 short b 6 int c 7 9 10 for b=0 to 31 11 SetData(b, "Local HMI", LW, 0, 1) 12 next 15 16 SetData(c, "Local HMI", LW, 100, 1) 17 18 end macro_command</pre>	
« m	
* Click the right mouse button to display edit menu.	
GET/SET FN Compile	Exit Help

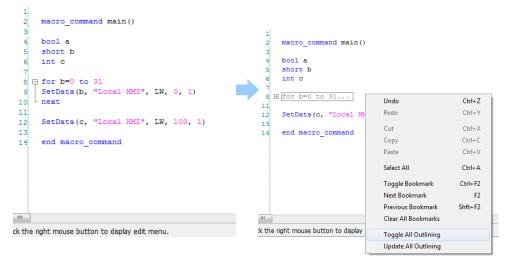
- Selecting [Clear All Bookmarks] will delete all bookmarks.

Work Space	Work Space
Macro ID: 0 Macro name : macro_0	Macro ID: 0 Macro name: macro_0
Periodical execution	Periodical execution
🗅 🗠 👌 🖻 💼 🦽 🌤 🌤 🎋	<u>♀</u> ♀∦⊫∎ <b>∧%%%</b>
<pre>* "" * Click the right mouse button to display edit menu. * "" * Click the right mouse button to display edit menu. * "" * Click the right mouse button to display edit menu. * "" * Click the right mouse button to display edit menu. * "" * Click the right mouse button to display edit menu.</pre>	<pre>     macro_command main()     d     bool a     short b     int c     for b=0 to 31     l1     SetData(c, "Local HMI", LW, 100, 1)     l3     end macro_command     * m     * Click the right mouse button to display edit menu. </pre>
GET/SET FN Compile	GET/SET FN Compile





9. Right click to select [Toggle All Outlining] to open all folded macro code blocks.



10. Sometimes the outlining might be incorrect since that the keywords are misjudged as shown in the following figure. To solve this problem, right click and select [Update All Outlining].

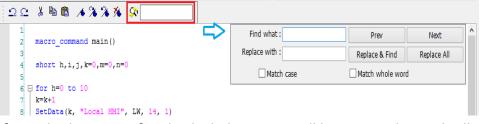
if 1 then	Undo	Ctrl+Z	3
// if 1 then	Redo	Ctrl+Y	4 □ if 1 then 5 // if 1 then
end if	Cut	Ctrl+X	6 end if
end macro_command	Сору	Ctrl+C	7 8 end macro command
	Paste	Ctrl+V	-
	Select All	Ctrl+A	
	Toggle Bookmark	Ctrl+F2	
	Next Bookmark	F2	
	Previous Bookmark	Shft+F2	
	Clear All Bookmarks		
	Toggle All Outlining		
	Update All Outlining		

- The statements enclosed in the following keywords are called a "block" of the macro code:
- Function block: sub end sub
- Iterative statements:
  - i. for next
  - ii. while wend
- Logical statements:
  - i. if end if
- Selective statements: select case end select
- **12.** The macro editor is not a monopoly window. Returning to the main screen and editing the project with the Work Space window open is allowed.

On-line Simulation Exit Help	
Jii-lille Si	Indiadon Exic Help



**13.** The macro editor provides Find and Replace features.



14. When [Periodical execution] is checked, this macro will be triggered periodically.

Periodical execution	Time interval (0 ~86400) :	10	second(s)

- 15. Select [Security] » [Use execution condition] » [Settings] to enable security settings:
- [Disable when Bit is ON]: When Bit is ON, this macro is disabled.
- [Disable when Bit is OFF]: When Bit is OFF, this macro is disabled.

Security Vse execution condition	Settings
Security	
Disable when Bit is ON	Disable when Bit is OFF
PLC name : Local HMI Address : LB	✓ Setting
	OK Cancel

**16.** Select [Execute one time when HMI starts], this macro will be executed once when HMI starts up.

#### **18.3.** Configuration

A macro contains statements. The statements contain constants, variables and operations. The statements are put in a specific order to create the desired output. A macro has the following structure:



Global Variable Declaration	Optional
Sub Function Block Declarations Local Variable Declarations End Sub	Optional
macro_command main() Local Variable Declarations [Statements]	Required
end macro_command	Required

Macro must have one and only one main function which is the execution start point of macro. The format is:

#### macro_command main()

#### end macro_command

Local variables are used within the main macro function or in a defined function block. Its value remains valid only within the specific block.

Global variables are declared before any function blocks and are valid for all functions in the macro. When local variables and global variables have the same declaration of name, only the local variables are valid.

The following example shows a simple macro which includes a variable declaration and a function call.

macro_command main()

```
short pressure = 10 // local variable declaration
SetData(pressure, "Allen-Bradley DF1", N7, 0, 1) // function calling
end macro_command
```

#### 18.4. Syntax

#### 18.4.1. Constants and Variables

#### **18.4.1.1. Constants**

Constants are fixed values and can be directly written into statements. The formats are:



Constant Type	Note	Example
Decimal integer		345, -234, 0, 23456
Hexadecimal	Must begin with 0x	0x3b, 0xffff, 0x237
ASCII	Single character must be enclosed in single	'a', "data", "name"
	quotation marks and a string (group of	
	characters) must be enclosed by double	
	quotation marks.	
Boolean		true, false

Here is an example using constants:

macro_command main()

short A, B	//	A and B are variables
A = 1234		
B = 0x12	//	1234 and 0x12 are constants
end macro_command		

#### **18.4.1.2.** Variables

Variables are names that represent information. The information can be changed as the variable is modified by statements.

#### Naming Rules for Variables

- A variable name must start with an alphabet.
- Variable names longer than 32 characters are not allowed.
- Reserved words cannot be used as variable names.

There are 8 different Variable types, 5 for signed data types and 3 for unsigned data types:

Variable Type	Description	Range
bool (boolean)	1 bit (discrete)	0, 1
char (character)	8 bits (byte)	+127 to -128
short (short integer)	16 bits (word)	+32767 to -32768
int (integer)	32 bits (double word)	+2147483647to -2147483648
float (floating point)	32 bits (double word)	
unsigned char	8 bits (byte)	0 to 255
unsigned short	16 bits (word)	0 to 65535
unsigned int	32 bits (double word)	0 to 4,294,967,295

#### **Declaring Variables**

Variables must be declared before being used. To declare a variable, specify the type before the variable name.



Example:	
int	а
short	b, switch
float	pressure
unsigned short	С

#### **Declaring Arrays**

Macros support one-dimensional arrays (zero-based index). To declare an array of variables, specify the type and the variable name followed by the number of variables in the array enclosed in brackets "[]". The length of an array could be 1 to 4096. (Macros only support at most 4096 variables per macro).

Example:

int	a[10]
short	b[20], switch[30]
float	pressure[15]

The minimum array index is 0 and the maximum is (array size -1).

Example:

char data [100] // array size is 100

In this case, the minimum of array index is 0 and maximum of array index is 99 (=100-1)

#### Variable and Array Initialization

There are two ways variables can be initialized:

```
    By statement using the assignment operator (=)
```

Example:

int a

```
float b[3]
```

```
a = 10
```

b[0] = 1

During declaration

```
char a = '5', b = 9
```

The declaration of arrays is a special case. The entire array can be initialized during declaration by enclosing comma separated values inside curly brackets "{}".

Example:

float data[4] = {11, 22, 33, 44} // now data[0] is 11, data[1] is 22....

#### 18.4.2. Operators

Operators are used to designate how data is manipulated and calculated.

### 

Macro Reference

Operator	Description	Example
=	Assignment operator	pressure = 10
Arithmetic Operators	Description	Example
+	Addition	A = B + C
<b>T</b>	Subtraction	A = B + C
*		A = B * C
	Multiplication	-
/	Division	A = B / C
%	Modulo division (return	A = B % 5
	remainder)	
Comparison Operators	Description	Example
<	Less than	if A < 10 then B = 5
<=	Less than or equal to	if A <= 10 then B = 5
>	Greater than	if A > 10 then B = 5
>=	Greater than or equal to	if A >= 10 then B = 5
==	Equal to	if A == 10 then B = 5
<>	Not equal to	if A <> 10 then B = 5
Logic Operators	Description	Example
and	Logical AND	if A < 10 and B > 5 then C = 10
or	Logical OR	if A >= 10 or B > 5 then C = 10
xor	Logical Exclusive OR	if A xor 256 then B = 5
not	Logical NOT	if not A then B = 5
		of signed/unsigned character and
•	·	left to right within the statement.
	escription	Example
<<	Shifts the bits in a bitset to	A = B << 8
	the left a specified number	
	of positions	
>>	Shifts the bits in a bitset to	A = B >> 8
	the right a specified numbe	
	-	

Bitwise Operators	Description	Example	
&	Bitwise AND	A = B & Oxf	

of positions





	Bitwise OR	A = B   C
۸	Bitwise XOR	A = B ^ C
~	One's complement	A = ~B

#### **Priority of All Operators**

The overall priority of all operations from highest to lowest is as follows:

- 1. Operations within parenthesis are carried out first
- 2. Arithmetic operations
- 3. Shift and Bitwise operations
- 4. Comparison operations
- 5. Logic operations
- 6. Assignment

#### **Reserved Keywords**

The following keywords are reserved for system. These keywords cannot be used as variable, array, or function names.

+, -, *, /, %, >=, >, <=, <, <>, ==, and, or, xor, not, <<, >>,=, &, |, ^, ~
exit, macro_command, for, to, down, step, next, return, bool, short, int, char, float, void, if, then, else, break, continue, set, sub, end, while, wend, true, false
SQRT, CUBERT, LOG, LOG10, SIN, COS, TAN, COT, SEC, CSC, ASIN, ACOS, ATAN, BIN2BCD,
BCD2BIN, DEC2ASCII, FLOAT2ASCII, HEX2ASCII, ASCII2DEC, ASCII2FLOAT, ASCII2HEX, FILL, RAND,
DELAY, SWAPB, SWAPW, LOBYTE, HIBYTE, LOWORD, HIWORD, GETBIT, SETBITON, SETBITOFF,
INVBIT, ADDSUM, XORSUM, CRC, INPORT, OUTPORT, POW, GetError, GetData, GetDataEx,
SetData, SetDataEx, SetRTS, GetCTS, Beep, SYNC_TRIG_MACRO, ASYNC_TRIG_MACRO, TRACE,
FindDataSamplingDate, FindDataSamplingIndex, FindEventLogDate, FindEventLogIndex
StringGet, StringGetEx, StringSet, StringSetEx, StringCopy, StringMid, StringDecAsc2Bin,
StringBin2DecAsc, StringLength, StringCat, StringCompare, StringCompareNoCase, StringFind,
StringReverseFind, StringFindOneOf, StringIncluding, StringExcluding, StringToUpper,
StringToLower, StringToReverse, StringTrimLeft, StringTrimRight, StringInsert



## 18.5. Statement

#### 18.5.1. Definition Statement

This covers the declaration of variables and arrays. The formal construction is as follows:

type	name	
This defines a variable with name as "name" and type as "type". Example:		
int A	// define a variable A as an integer	
type	name[constant]	

This defines an array variable called "name" with size as "constant" and type as "type". Example:

int B[10] // where define a variable B as a one-dimensional array of size 10

#### 18.5.2. Assignment Statement

Assignment statements use the assignment operator to move data from the expression on the right side of the operator to the variable on the left side. An expression is the combination of variables, constants and operators to yield a value.

VariableName	Expression
Example	

A = 2

where a variable A is assigned to 2

#### 18.5.3. Logical Statements

Logical statements perform actions depending on the condition of a boolean expression. The syntax is as follows:

#### **Single-Line Format**

f <condition> then</condition>	
[Statements]	
else	
[Statements]	
end if	





```
Example:
if a == 2 then
b = 1
else
b = 2
```

end if

## **Block Format**

If <Condition> then [Statements] else if <Condition-n> then [Statements] else [Statements] end if

## Example:

if a == 2 then b = 1 else if a == 3 then b = 2 else b = 3 end if

## Syntax description

if	Must be used to begin the statement.
<condition></condition>	Required. This is the controlling statement. It is FALSE when the <condition> evaluates to 0 and TRUE when it evaluates to non- zero.</condition>
then	Must precede the statements to execute if the <condition> evaluates to TRUE.</condition>
[Statements]	It is optional in block format but necessary in single-line format without else. The statement will be executed when the <condition> is TRUE.</condition>
else if	Optional. The else if statement will be executed when the relative <condition-n> is TRUE.</condition-n>
<condition-n></condition-n>	Optional. see <condition></condition>
else	Optional. The else statement will be executed when <condition> and <condition-n> are both FALSE.</condition-n></condition>
end if	Must be used to end an if-then statement.

18-14



#### 18.5.4. Selective Statements

The select-case construction can be used like multiple if-else statements and perform selected actions depending on the value of the given variable. When the matched value is found, all the actions below will be executed until a break statement is met. The syntax is as follows:

## Format without a Default Case

Select Case [variable]		
Case [value]		
[Statements]		
break		
end Select		
Example:		

Select Case A Case 1 b=1 break end Select

#### Format with a Default Case (Case else)

Select Case [variable]

Case [value] [Statements]

break

Case else

[Statements]

break

#### end Select

Example:

Select Case A Case 1 b=1 break Case else b=0 break end Select



Select Case [variable] Case [value1] [Statements] Case [value2] [Statements]

break

## end Select

#### Example:

Select Case A Case 1 break Case 2 b=2 break Case 3 b=3 break end Select

## Syntax description

Select Case	Must be used to begin the statement.
[variable]	Required. The value of this variable will be compared to the value of each case.
Case else	Optional. It represents the default case. If none of the cases above are matched, the statements under default case will be executed. When a default case is absent, it will skip directly to the end of the select-case statements if there is no matched case.
break	Optional. The statements under the matched case will be executed until the break command is reached. If a break command is absent, it simply keeps on executing next statement until the end command is reached.
end Select	Indicates the end of the select-case statements.

#### 18.5.5. Iterative Statements

Iterative statements control loops and repetitive tasks depending on condition. There are two types of iterative statements.



## 18.5.5.1. for-next Statements

The for-next statement runs for a fixed number of iterations. A variable is used as a counter to track the progress and test for ending conditions. Use this for fixed execution counts. The syntax is as follows:

```
for [Conunter] = <StartValue> to <EndValue> [step <StepValue>]
  [Statements]
next [Counter]
```

```
Or
```

```
for [Conunter] = <StartValue> to <EndValue> [step <StepValue>]
  [Statements]
next [Counter]
```

#### Example:

```
for a = 0 to 10 step 2
b = a
```

next a

## Syntax description

for	Must be used to begin the statement
[Counter]	Required. This is the controlling statement. The result of evaluating the variable is used as a test of comparison.
<startvalue></startvalue>	Required. The initial value of [Counter]
to/down	Required. This determines if the <step> increments or decrements the <counter>. "to" increments <counter> by <stepvalue>.</stepvalue></counter></counter></step>
	"down" decrements <counter> by <stepvalue>.</stepvalue></counter>
<endvalue></endvalue>	Required. The test point. If the <counter> is greater than this value, the macro exits the loop.</counter>
step	Optional. Specifies that a <stepvalue> other than one is to be used.</stepvalue>
[StepValue]	Optional. The increment/decrement step of <counter>. It can be omitted when the value is 1 If [step <stepvalue>] are omitted the step value defaults to 1.</stepvalue></counter>
[Statements]	Optional. Statements to execute when the evaluation is TRUE. "for-next" loops may be nested.
next	Required.
[Counter]	Optional. This is used when nesting for-next loops.
· · · · · · · · · · · · · · · · · · ·	





#### 18.5.5.2. while-wend Statements

The while-wend statement runs for an unknown number of iterations. A variable is used to test for ending conditions. When the condition is TRUE, the statements inside are executed repetitively until the condition becomes FALSE. The syntax is as follows.

[Statements] wend	
xample:	
while a < 10	
a = a + 1	10
wend	
yntax description	
while	Must be used to begin the statement.
continue	Required. This is the controlling statement. When it is TRUE, the loop begins execution. When it is FALSE, the loop terminates.
return [value]	Statements to execute when the evaluation is TRUE.
wend	Indicates the end of the while-end statements.

break	Used in for-next and while-wend. It skips immediately to the end of the
	iterative statement.
continue	Used in for-next and while-wend. It ends the current iteration of a loop and starts the next one.
return	The return command inside the main block can force the macro to stop
	anywhere. It skips immediately to the end of the main block.

## **18.6.** Function Blocks

Function blocks are useful for reducing repetitive codes. It must be defined before use and supports any variable and statement type. A function block could be called by putting its name followed by parameters in parenthesis. After the function block is executed, it returns the value to the caller function where it is used as an assignment value or as a condition. A return type is not required in function definition, which means that a function block does not have to return a value. The parameters can also be ignored in function definition while the function has no need to take any parameters from the caller. The syntax is as follows:

Function definition with return type



```
sub type <name> [(parameters)]
    Local variable declarations
    [Statements]
    [return [value]]
end sub
```

Example:

```
sub int Add(int x, int y)
int result
result = x +y
return result
end sub
```

```
macro_command main()
int a = 10, b = 20, sum
sum = Add(a, b)
end macro_command
```

#### or:

#### Function definition without return type

```
sub <name> [(parameters)]
    Local variable declarations
    [Statements]
end sub
```

Example:

sub Add(int x, int y) int result result = x +y



```
18-20
```

```
end sub
```

```
macro_command main()
int a = 10, b = 20
Add(a, b)
end macro_command
```

## or:

sub Add()

```
int result, x=10, y=20
result = x +y
end sub
```

```
macro_command main()
Add()
end macro_command
```

## Syntax description

Syntax acscription	
sub	Must be used to begin the function block
type	Optional. This is the data type of value that the function returns. A function block is not always necessary to return a value.
(parameters)	<ul> <li>Optional. The parameters hold values that are passed to the function.</li> <li>The passed parameters must have their type declared in the parameter field and assigned a variable name.</li> <li>For example: sub int MyFunction(int x, int y). x and y would be integers passed to the function. This function is called by a statement that looks similar to this: ret = MyFunction(456, pressure) where "pressure" must be integer according to the definition of function.</li> <li>Notice that the calling statement can pass hard coded values or variables to the function. After this function is executed, an integer values is return to 'ret'.</li> </ul>
Local variable	Variables that are used in the function block must be declared first.
declaration	This is in addition to passed parameters. In the above example x and y are variables that the function can used. Global variables are also available for use in function block.
[Statements]	Statements to execute
[return [value]]	Optional. Used to return a value to the calling statement. The value can be a constant or a variable. Return also ends function block execution. A function block is not always necessary to return a value, but, when the return type is defined in the beginning of the definition of function, the return command is needed.
end sub	Must be used to end a function block.
·	



# **18.7.** Built-In Function Block

EasyBuilder Pro has many built-in functions for retrieving and transferring data to the PLC, data management and mathematical functions.

## 18.7.1. Mathematical Functions

Name	SQRT	
Syntax	SQRT(source, result)	
Description	Calculate the square root of <i>source</i> and store the result into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable. <i>source</i> must be a nonnegative value.	
Example       macro_command main() float source, result         SQRT(15, result)         source = 9.0         SQRT(source, result)// result is 3.0		
	end macro_command	

Name	CUBERT
Syntax	CUBERT(source, result)
Description	Calculate the cube root of source and store the result into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable. <i>source</i> must be a nonnegative value.
Example	macro_command main() float source, result CUBERT (27, result) // result is 3.0 source = 27.0 CUBERT(source, result)// result is 3.0 end macro_command

Name	POW
Syntax	POW(source1, source2, result)
Description	-
	<i>source1</i> and <i>source2</i> can be a constant or a variable.
	<i>result</i> must be a variable.
	source1 and source2 must be a nonnegative value.



Example	macro_command main()
	float y, result
	y = 0.5
	POW (25, y, result) // result = 5
	end macro_command

Name	SIN
Syntax	SIN(source, result)
Description	Calculate the sine of <i>source</i> (degree) into <i>result</i> .
	source can be a constant or a variable. result must be a variable.
Example	macro_command main()
	float source, result
	SIN(90, result)// result is 1
	source = 30 SIN(source, result)// result is 0.5
	end macro_command

Name	COS
Syntax	COS(source, result)
Description	Calculate the cosine of <i>source</i> (degree) into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	macro_command main()
	float source, result
	COS(90, result)// result is 0
	source = 60
	GetData(source, "Local HMI", LW, 0, 1)
	COS(source, result)// result is 0.5
	end macro_command

Name	TAN
Syntax	TAN(source, result)
Description	Calculate the tangent of <i>source</i> (degree) into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	macro_command main() float source, result TAN(45, result)// result is 1



source = 60 TAN(source, result)// result is 1.732	
end macro_command	

EasyBuilder Pro V5.02.01

Name	СОТ
Syntax	COT(source, result)
Description	Calculate the cotangent of <i>source</i> (degree) into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	<pre>macro_command main() float source, result COT(45, result)// result is 1 source = 60 COT(source, result)// result is 0.5774 end macro_command</pre>

Name	SEC
Syntax	SEC(source, result)
Description	Calculate the secant of <i>source</i> (degree) into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	<pre>macro_command main() float source, result SEC(45, result)// result is 1.414 source = 60 SEC(source, result)// if source is 60, result is 2 end macro_command</pre>

Name	CSC
Syntax	CSC(source, result)
Description	Calculate the cosecant of <i>source</i> (degree) into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	<pre>macro_command main() float source, result CSC(45, result)// result is 1.414 source = 30 CSC(source, result)// result is 2 end macro_command</pre>



Name	ASIN
Syntax	ASIN(source, result)
Description	Calculate the arc sine of <i>source</i> into <i>result</i> (degree).
	<i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	macro_command main()
	float source, result
	ASIN(0.8660, result)// result is 60
	source = 0.5 ASIN(source, result)// result is 30
	end macro_command

Name	ACOS
Syntax	ACOS(source, result)
Description	Calculate the arc cosine of <i>source</i> into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	<pre>macro_command main() float source, result ACOS(0.8660, result)// result is 30 source = 0.5 ACOS(source, result)// result is 60 end macro_command</pre>

Name	ATAN
Syntax	ATAN(source, result)
Description	Calculate the arc tangent of <i>source</i> into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	<pre>macro_command main() float source, result ATAN(1, result)// result is 45 source = 1.732 ATAN(source, result)// result is 60 end macro_command</pre>



Name	LOG
Syntax	LOG (source, result)
Description	Calculates the natural logarithm of a number.
	source can be either a variable or a constant. result must be a variable.
Example	macro_command main()
	float source = 100, result
	LOG (source, result)// result is approximately 4.6052 end macro_command

Name	LOG10
Syntax	LOG10(source, result)
Description	Calculates the base-10 logarithm of a number.
	source can be either a variable or a constant. result must be a variable.
Example	macro_command main()
	float source = 100, result
	LOG10 (source, result) // result is 2
	end macro_command

Name	RAND
Syntax	RAND(result)
Description	Calculates a random integer and save into result.
	<i>result</i> must be a variable.
Example	macro_command main()
	short result
	RAND (result) //result is not a fixed value when executes macro every time end macro_command



## 18.7.2. Data Transformation

Name	BIN2BCD
Syntax	BIN2BCD(source, result)
Description	Transforms a binary-type value ( <i>source</i> ) into a BCD-type value ( <i>result</i> ).
	source can be a constant or a variable. result must be a variable.
Example	macro_command main()
	short source, result
	BIN2BCD(1234, result)// result is 0x1234
	source = 5678
	BIN2BCD(source, result)// result is 0x5678
	end macro_command

BCD2BIN
BCD2BIN(source, result)
Transforms a BCD-type value ( <i>source</i> ) into a binary-type value ( <i>result</i> ).
source can be a constant or a variable. result must be a variable.
macro_command main()
short source, result
BCD2BIN(0x1234, result)// result is 1234
source = 0x5678
BCD2BIN(source, result)// result is 5678
end macro_command

Name	DEC2ASCII
Syntax	DEC2ASCII(source, result[start], len)
Description	Transforms a decimal value ( <i>source</i> ) into an ASCII string and save it to an array ( <i>result</i> ). <i>len</i> represents the length of the string and the unit of length depends on result's type., i.e. if result's type is "char" (the size is byte), the length of the string is (byte * <i>len</i> ). If result's type is "short" (the size is word), the length of the string is (word * <i>len</i> ), and so on. The first character is put into <i>result[start]</i> , the second character is put into <i>result[start + 1]</i> , and the last character is put into <i>result[start + (len -1)]</i> . <i>source</i> and <i>len</i> can be a constant or a variable. <i>result</i> must be a variable. <i>start</i> must be a constant.
Example	macro_command main()



	short source
	char result1[4]
	short result2[4]
	char result3[6]
	source = 5678
	DEC2ASCII(source, result1[0], 4)
	<pre>// result1[0] is '5', result1[1] is '6', result1[2] is '7', result1[3] is '8'</pre>
	<pre>// the length of the string (result1) is 4 bytes( = 1 * 4)</pre>
	DEC2ASCII(source, result2[0], 4)
	// result2[0] is '5', result2[1] is '6', result2[2] is '7', result2[3] is '8'
	// the length of the string (result2) is 8 bytes( = $2 * 4$ )
	source=-123
	DEC2ASCII(source3, result3[0], 6)
	// result1[0] is '-', result1[1] is '0', result1[2] is '0', result1[3] is '1'
	// result1[4] is '2', result1[5] is '3'
	// the length of the string (result1) is 6 bytes( = 1 * 6)
	end macro command
L	

Name	HEX2ASCII
Syntax	HEX2ASCII(source, result[start], len)
Description	Transforms a hexadecimal value ( <i>source</i> ) into ASCII string saved to an array ( <i>result</i> ). <i>len</i> represents the length of the string and the unit of length depends on result's type., i.e. if result's type is "char" (the size is byte), the length of the string is (byte * <i>len</i> ). If result's type is "short" (the size is word), the length of the string is (word * <i>len</i> ), and so on. <i>source</i> and <i>len</i> can be a constant or a variable. <i>result</i> must be a variable. <i>start</i> must be a constant.
Example	macro_command main() short source char result[4] source = 0x5678 HEX2ASCII (source, result[0], 4) // result[0] is '5', result[1] is '6', result[2] is '7', result[3] is '8' end macro_command



Name	FLOAT2ASCII
Syntax	FLOAT2ASCII(source, result[start], len)
Description	Transforms a floating value ( <i>source</i> ) into ASCII string saved to an array ( <i>result</i> ). <i>len</i> represents the length of the string and the unit of length depends on result's type., i.e. if result's type is "char" (the size is byte), the length of the string is (byte * <i>len</i> ). If result's type is "short" (the size is word), the length of the string is (word * <i>len</i> ), and so on. <i>source</i> and len can be a constant or a variable. <i>result</i> must be a variable. start must be a constant.
Example	<pre>macro_command main() float source char result[4] source = 56.8 FLOAT2ASCII (source, result[0], 4) // result[0] is '5', result[1] is '6', result[2] is '.', result[3] is '8' end macro_command</pre>

Name	ASCII2DEC
Syntax	ASCII2DEC(source[start], result, len)
Description	Transforms a string ( <i>source</i> ) into a decimal value saved to a variable ( <i>result</i> ). The length of the string is <i>len</i> . The first character of the string is <i>source[start]</i> . <i>source</i> and <i>len</i> can be a constant or a variable. <i>result</i> must be a variable. <i>start</i> must be a constant.
Example	<pre>macro_command main() char source[4] short result source[0] = '5' source[1] = '6' source[2] = '7' source[3] = '8' ASCII2DEC(source[0], result, 4) // result is 5678</pre>
	end macro_command





Name	ASCII2HEX
Syntax	ASCII2HEX (source[start], result, len)
Description	Transforms a string (source) into a hexadecimal value saved to a variable
	(result).
	The length of the string is <i>len</i> . The first character of the string is <i>source[start</i> ].
	source and len can be a constant or a variable. result must be a variable. start
	must be a constant.
Example	macro_command main()
-	char source[4]
	short result
	source[0] = '5'
	source[1] = '6'
	source[2] = '7'
	source[3] = '8'
	ASCII2HEX (source[0], result, 4) // result is 0x5678
	end macro_command

Name	ASCII2FLOAT
Syntax	ASCII2FLOAT(source[start], result, len)
Description	Transforms a string ( <i>source</i> ) into a float value saved to a variable ( <i>result</i> ). The length of the string is <i>len</i> . The first character of the string is <i>source[start]</i> . <i>source</i> and <i>len</i> can be a constant or a variable. <i>result</i> must be a variable. <i>start</i> must be a constant.
Example	<pre>macro_command main() char source[4] float result source[0] = '5' source[1] = '6' source[2] = '.' source[3] = '8' ASCII2FLOAT (source[0], result, 4) // result is 56.8</pre>
	end macro_command



## **18.7.3.** Data Manipulation

Name	FILL
Syntax	FILL(source[start], preset, count)
Description	Sets the first count elements of an array ( <i>source</i> ) to a specified value ( <i>preset</i> ).
	source and start must be a variable, and preset can be a constant or variable.
Example	macro_command main()
-	char result[4]
	char preset
	FILL(result[0], 0x30, 4) // result[0] is 0x30, result[1] is 0x30, , result[2] is 0x30, , result[3] is 0x30
	preset = 0x31 FILL(result[0], preset, 2) // result[0] is 0x31, result[1] is 0x31
	end macro_command

Name	SWAPB
Syntax	SWAPB(source, result)
Description	Exchanges the high-byte and low-byte data of a 16-bit <i>source</i> into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	<pre>macro_command main() short source, result SWAPB(0x5678, result)// result is 0x7856 source = 0x123 SWAPB(source, result)// result is 0x2301 end macro_command</pre>

Name	SWAPW
Syntax	SWAPW(source, result)
Description	Exchanges the high-word and low-word data of a 32-bit <i>source</i> into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	<pre>macro_command main() int source, result SWAPW (0x12345678, result)// result is 0x56781234 source = 0x12345 SWAPW (source, result)// result is 0x23450001 end macro_command</pre>



Name	LOBYTE
Syntax	LOBYTE(source, result)
Description	Retrieves the low byte of a 16-bit <i>source</i> into <i>result</i> . <i>source</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	<pre>macro_command main() short source, result LOBYTE(0x1234, result)// result is 0x34 source = 0x123 LOBYTE(source, result)// result is 0x23 end macro_command</pre>

Name	HIBYTE
Syntax	HIBYTE(source, result)
Description	Retrieves the high byte of a 16-bit <i>source</i> into <i>result</i> .
	source can be a constant or a variable. result must be a variable.
Example	macro_command main()
	short source, result
	HIBYTE(0x1234, result)// result is 0x12
	source = 0x123 HIBYTE(source, result)// result is 0x01
	end macro_command

Name	LOWORD
Syntax	LOWORD(source, result)
Description	Retrieves the low word of a 32-bit source into result.
	source can be a constant or a variable. result must be a variable.
Example	macro_command main()
•	int source, result
	LOWORD(0x12345678, result)// result is 0x5678
	source = 0x12345
	LOWORD(source, result)// result is 0x2345
	end macro_command



18-33

Name	HIWORD
Syntax	HIWORD(source, result)
Description	Retrieves the high word of a 32-bit source into result.
	source can be a constant or a variable. result must be a variable.
Example	macro_command main()
	int source, result
	HIWORD(0x12345678, result)// result is 0x1234 source = 0x12345 HIWORD(source, result)// result is 0x0001 end macro_command
	end macro_command

## **18.7.4.** Bit Transformation

Name	GETBIT
Syntax	GETBIT(source, result, bit_pos)
Description	Gets the state of designated bit position of a data (source) into result.
	result value will be 0 or 1.
	source and bit_pos can be a constant or a variable.
	<i>result</i> must be a variable.
Example	macro_command main()
	int source, result
	short bit_pos
	GETBIT(9, result, 3)// result is 1
	source = 4
	bit pos = 2
	GETBIT(source, result, bit_pos)// result is 1
	end macro_command



Name	SETBITON
Syntax	SETBITON(source, result, bit_pos)
Description	Changes the state of designated bit position of a data (source) to 1, and put
	changed data into <i>result</i> .
	source and bit_pos can be a constant or a variable.
	<i>result</i> must be a variable.
Example	macro_command main()
	int source, result
	short bit_pos
	SETBITON(1, result, 3)// result is 9
	source = 0
	bit_pos = 2
	SETBITON (source, result, bit_pos)// result is 4
	end macro_command

Name	SETBITOFF
Syntax	SETBITOFF(source, result, bit_pos)
Description	Changes the state of designated bit position of a data (source) to 0, and put in
	changed data into <i>result</i> .
	source and bit_pos can be a constant or a variable.
	<i>result</i> must be a variable.
Example	macro_command main()
	int source, result
	short bit_pos
	SETBITOFF(9, result, 3)// result is 1
	source = 4
	bit_pos = 2
	SETBITOFF(source, result, bit_pos)// result is 0
	end macro_command



Name	INVBIT
Syntax	INVBIT(source, result, bit_pos)
Description	Inverts the state of designated bit position of a data (source), and put changed
	data into <i>result</i> .
	<i>source</i> and <i>bit_pos</i> can be a constant or a variable. <i>result</i> must be a variable.
Example	macro_command main()
	int source, result
	short bit_pos
	INVBIT(4, result, 1)// result = 6
	source = 6 bit_pos = 1 INVBIT(source, result, bit_pos)// result = 4
	end macro_command

## 18.7.5. Communication

Name	DELAY
Syntax	DELAY( <i>time</i> )
Description	Suspends the execution of the current macro for at least the specified interval ( <i>time</i> ). The unit of <i>time</i> is millisecond. <i>time</i> can be a constant or a variable.
Example	macro_command main() int time == 500 DELAY(100)// delay 100 ms DELAY(time)// delay 500 ms end macro_command



Name	ADDSUM	
Syntax	ADDSUM(source[start], result, data_count)	
Description	Adds up the elements of an array ( <i>source</i> ) from <i>source[start</i> ] to <i>source[start + data_count - 1</i> ] to generate a checksum. Puts in the checksum into <i>result</i> . <i>result</i> must be a variable. <i>data_count</i> is the amount of the accumulated elements and can be a constant or a variable.	
Example	<pre>macro_command main() char data[5] short checksum  data[0] = 0x1 data[1] = 0x2 data[2] = 0x3 data[3] = 0x4 data[4] = 0x5  ADDSUM(data[0], checksum, 5)// checksum is 0xf end macro_command</pre>	

Name	XORSUM	
Syntax	XORSUM(source[start], result, data_count)	
Description	Uses an exclusion method to calculate the checksum from <i>source</i> [ <i>start</i> ] to <i>source</i> [ <i>start</i> + <i>data_count</i> - 1]. Puts the checksum into <i>result</i> . <i>result</i> must be a variable. <i>data_count</i> is the amount of the calculated elements of the array and can be a constant or a variable.	
Example	<pre>macro_command main() char data[5] = {0x1, 0x2, 0x3, 0x4, 0x5} short checksum XORSUM(data[0], checksum, 5)// checksum is 0x1 end macro_command</pre>	

Name	BCC
Syntax	BCC(source[start], result, data_count)
Description	Uses an XOR method to calculate the checksum from <i>source</i> [ <i>start</i> ] to <i>source</i> [ <i>start</i> + <i>data_count</i> - 1]. Puts the checksum into <i>result</i> . <i>result</i> must be a variable. <i>data_count</i> is the amount of the calculated elements of the array and can be a constant or a variable.
Example	macro_command main() char data[5] = {0x1, 0x2, 0x3, 0x4, 0x5} char checksum



BCC(source[0], checksum, 5)	// checksum is 0x1
end macro_command	

Name	CRC
Syntax	CRC(source[start], result, data_count)
Description	Calculates 16-bit CRC of the variables from <i>source</i> [ <i>start</i> ] to <i>source</i> [ <i>start</i> + <i>data_count</i> - 1]. Puts in the 16-bit CRC into <i>result</i> . <i>result</i> must be a variable. <i>data_count</i> is the amount of the calculated elements of the array and can be a constant or a variable.
Example	<pre>macro_command main() char data[5] = {0x1, 0x2, 0x3, 0x4, 0x5} short 16bit_CRC CRC(data[0], 16bit_CRC, 5)// 16bit_CRC is 0xbb2a end macro_command</pre>

Syntax OU	ITPORT(source[start] device name data count)	
	OUTPORT(source[start], device_name, data_count)	
PLC dev mu	Sends out the specified data from source[ <i>start</i> ] to source[ <i>start</i> + <i>count</i> -1] to PLC via a COM port or the ethernet. <i>device_name</i> is the name of a device defined in the device table and the device must be a "Free Protocol"-type device. <i>data_count</i> is the amount of sent data and can be a constant or a variable.	
The set Bel MC ma	use an OUTPORT function, a "Free Protocol" device must be created first as lows: tem Parameter Settings Extended Memory Printer/Backup Server e-Mail Recipes Device Model General System Setting Security Font Device Ist : No. Name Location Device type Interface Local HMI Local HMI Local eMT3105 (800 Local Server MODBUS RTU Local Free Protocol COM 1 (9600,E.f) e device is named "MODBUS RTU Device". The port attribute depends on the tting of this device. (the current setting is "19200,E, 8, 1") low is an example of executing an action of writing single coil (SET ON) to a ODBUS device. acro_command main() ar command[32] ort address, checksum	



FILL(command[0], 0, 32)// command initialization
command[0] = 0x1// station no command[1] = 0x5// function code : Write Single Coil
address = 0 HIBYTE(address, command[2]) LOBYTE(address, command[3])
command[4] = 0xff// force bit on command[5] = 0
CRC(command[0], checksum, 6)
LOBYTE(checksum, command[6]) HIBYTE(checksum, command[7])
<pre>// send out a "Write Single Coil" command OUTPORT(command[0], "MODBUS RTU Device", 8)</pre>
end macro_command

Name	INPORT
Syntax	INPORT(read_data[start], device_name, read_count, return_value)
Description	Reads data from a COM port or the ethernet. These data is stored to
	read_data[start]~ read_data[start + read_count - 1].
	device_name is the name of a device defined in the device table and the device
	must be a "Free Protocol"-type device.
	<i>read_count</i> is the required amount of reading and can be a constant or a
	variable.
	If the function is used successfully to get sufficient data, return value is 1,
	otherwise is 0.
<b>Example</b> Below is an example of executing an action of reading holding registers	
	MODBUS device.
	// Read Holding Registers
	macro_command main()
	char command[32], response[32]
	short address, checksum
	short read_no, return_value, read_data[2]
	FILL(command[0], 0, 32)// command initialization
	FILL(response[0], 0, 32)



<pre>command[0] = 0x1// station no command[1] = 0x3// function code : Read Holding Registers</pre>
address = 0 HIBYTE(address, command[2]) LOBYTE(address, command[3])
read_no = 2// read 2 words (4x_1 and 4x_2) HIBYTE(read_no, command[4]) LOBYTE(read_no, command[5])
CRC(command[0], checksum, 6)
LOBYTE(checksum, command[6]) HIBYTE(checksum, command[7])
<pre>// send out a 'Read Holding Registers" command OUTPORT(command[0], "MODBUS RTU Device", 8)</pre>
<pre>// read responses for a 'Read Holding Registers" command INPORT(response[0], "MODBUS RTU Device", 9, return_value)</pre>
if return_value > 0 then read_data[0] = response[4] + (response[3] << 8)// data in 4x_1 read_data[1] = response[6] + (response[5] << 8)// data in 4x_2
SetData(read_data[0], "Local HMI", LW, 100, 2) end if
end macro_command

Name	INPORT2
Syntax	INPORT2(response[start], device_name, receive_len, wait_time)
Description	Read data from a communication port (COM Port or Ethernet Port). The data read will be saved in response. The description of <i>device_name</i> is the same as OUTPORT. <i>receive_len</i> stores the length of the data received, this must be a variable. <i>receive_len</i> total length can't exceed the size of response. <i>wait_time</i> (in millisecond) can be a constant or variable. After the data is read, if there's no upcoming data during the designated time interval, the function returns.
Example	macro_command main() short wResponse[6], receive_len, wait_time=20



INPORT2(wResponse[0], "Free Protocol", receive_len, wait_time) // wait_time unit : millisecond
if receive_len > 0 then SetData(wResponse[0], "Local HMI", LW, 0, 6) // set responses to LW0 end if
end macro_command

Name	INPORT3
Syntax	INPORT3(response[start], device_name, read_count, receive_len)
Description	Read data from a communication port (COM Port or Ethernet Port). The data read will be saved in response. The amount of data to be read can be specified. The data that is not read yet will be stored in HMI buffer memory for the next read operation, in order to prevent losing data. The description of <i>device_name</i> is the same as OUTPORT. <i>read_count</i> stores the length of the data read each time. <i>receive_len</i> stores the length of the data received, this must be a variable. <i>receive_len</i> total length can't exceed the size of response.
Example	<pre>macro_command main() short wResponse[6], receive_len INPORT3(wResponse[0], "Free Protocol", 6, receive_len) // read 6 words if receive_len &gt;= 6 then SetData(wResponse[0], "Local HMI", LW, 0, 6) // set responses to LW0 end if end macro_command</pre>

Name	GetData
Syntax	GetData(read_data[start], device_name, device_type, address_offset, data_count) or GetData(read data, device name, device type, address offset, 1)
Description	Receives data from the PLC. Data is stored into <i>read_data[start]</i> ~ <i>read_data[start + data_count - 1]</i> . <i>data_count</i> is the amount of received data. In general, <i>read_data</i> is an array, but if <i>data_count</i> is 1, <i>read_data</i> can be an array or an ordinary variable. Below are two methods to read one word data from PLC.
	macro_command main()



short read_data_1[2], read_data_2 GetData(read_data_1[0], "FATEK KB Series", RT, 5, 1) GetData(read_data_2, "FATEK KB Series", RT, 5, 1) end macro command Device name is the PLC name enclosed in the double quotation marks (") and this name has been defined in the device list of system parameters as follows (see FATEK KB Series): System Parameter Settings х Font Extended Memory Printer/Backup Server Device Model General System Setting Security Device list : No. Name Location Device type Interface Local HMI Local HMI Local MT8104iH (800 x... Free Protocol Local Server Local Free Protocol COM 1 (9600,1 Remote PLC1 FATEK FB Series Remote (IP:192.168.1.10... FATEK FB Series COM 1 (9600, *Device type* is the device type and encoding method (binary or BCD) of the PLC data. For example, if *device type* is LW BIN, it means the register is LW and the encoding method is binary. If use BIN encoding method, "BIN" can be ignored. If *device type* is LW BCD, it means the register is LW and the encoding method is BCD. Address offset is the address offset in the PLC. For example, GetData(read data 1[0], "FATEK KB Series", RT, 5, 1) represents that the address offset is 5. If address offset uses the format – "N#AAAAA", N indicates that PLC's station number is N. AAAAA represents the address offset. This format is used while multiple PLCs or controllers are connected to a single serial port. For example, GetData(read data 1[0], "FATEK KB Series", RT, 2#5, 1) represents that the PLC's station number is 2. If GetData() uses the default station number defined in the device list as follows, it is not necessary to define station number in address offset.



Image: Intro (H Sees 		Device Properties			
Image: setting image: provide iteration is setting image: provide iteration iteratiteration iteration iteration iteration iteration iteration iterati		Name : FATEK FE	3 Series		
Image: State of the second		● HMI	PLC		
with the image of the imag		Location : Remote	✓ Settings IP : 192.168.1.	10 (Port = 8000)	
PCC UF: [S22         Image: COM: [OMI]         Image: Command Image: Comman		PLC type :	FATEK FB Series		
Image: content of the second of the secon			ATEK_FB.si		
Image: state in the image: state image: st		PLC I/F : RS-232			
Image: state in the state		COM : COM1		Settings	
Image: state in the state					
Har, read-command size (words):       Image: Command size (words):         Image: Write-command size (words):       Image: Command size (words):         Image: Command main()       Image: Command main()					
Har, read-command size (words):       Image: Command size (words):         Image: Write-command size (words):       Image: Command size (words):         Image: Command main()       Image: Command main()					
Max. write-command size (words): [6]         Image: write-command main()         float f         GetData(f, "MODBUS", 6x, 2, 1)       // f will contain a floating point value end macro_command         Image: write-command main()         float f         Max: write-command main()         float f         Macro_command main()         float f         Macro_command main()         float f         Macro_command main()         float f         Macro_command main()					
The number of registers actually read from depends on both the type of the read_data variable and the value of the number of data_count.type of read_datadata_countactual number of 16-bit register readchar (8-bit)11char (8-bit)21bool (8-bit)11bool (8-bit)21short (16-bit)11short (16-bit)22int (32-bit)24float (32-bit)12int (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_command main()Examplemacro_command main()					
The number of registers actually read from depends on both the type of the read_data variable and the value of the number of data_count.type of read_datadata_countactual number of 16-bit register readchar (8-bit)11char (8-bit)21bool (8-bit)11bool (8-bit)21short (16-bit)11short (16-bit)22int (32-bit)24float (32-bit)12int (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_command main()Examplemacro_command main()					
The number of registers actually read from depends on both the type of the read_data variable and the value of the number of data_count.type of read_datadata_countactual number of 16-bit register readchar (8-bit)11char (8-bit)21bool (8-bit)11bool (8-bit)21short (16-bit)11short (16-bit)22int (32-bit)24float (32-bit)12int (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_command main()Examplemacro_command main()					
read_data variable and the value of the number of data_count.type of read_datadata_countactual number of 16-bit register readchar (8-bit)11char (8-bit)21bool (8-bit)11bool (8-bit)21short (16-bit)21short (16-bit)22int (32-bit)12int (32-bit)12float (32-bit)24float (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_commandExamplemacro_command main()			ОК	Cancel	
read_data variable and the value of the number of data_count.type of read_datadata_countactual number of 16-bit register readchar (8-bit)11char (8-bit)21bool (8-bit)11bool (8-bit)21short (16-bit)21short (16-bit)22int (32-bit)12int (32-bit)12float (32-bit)24float (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_commandExamplemacro_command main()		The number of regist	ers actually read from d	anends on both th	e type of the
type of read_datadata_countactual number of 16-bit register readchar (8-bit)11char (8-bit)2bool (8-bit)1bool (8-bit)2short (16-bit)1short (16-bit)2int (32-bit)1float (32-bit)2float (32-bit)2int (32-bit)2float (32-bit)1char (8-bit)2int (32-bit)1float (32-bit)1float (32-bit)2float (32-bit)2float (32-bit)2float (32-bit)1getData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_commandExamplemacro_command main()		_	-	-	
type of read_datadata_count16-bit register readchar (8-bit)11char (8-bit)2bool (8-bit)1bool (8-bit)2short (16-bit)1short (16-bit)2int (32-bit)2float (32-bit)2float (32-bit)2float (32-bit)2macro_command main()float fGetData(f, "MODBUS", 6x, 2, 1)macro_command main()float fmacro_command main()				_	
char (8-bit)11char (8-bit)21bool (8-bit)11bool (8-bit)21short (16-bit)11short (16-bit)22int (32-bit)12int (32-bit)24float (32-bit)12float (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1) // f will contain a floating point value end macro_command main()Examplemacro_command main()		type of read_data	data_count		
char (8-bit)21bool (8-bit)11bool (8-bit)21short (16-bit)11short (16-bit)22int (32-bit)12int (32-bit)12float (32-bit)24float (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_command main()Examplemacro_command main()		char (8-bit)	1		cricad
bool (8-bit)11bool (8-bit)21short (16-bit)11short (16-bit)22int (32-bit)12int (32-bit)24float (32-bit)12float (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_command main()Examplemacro_command main()					
bool (8-bit)21short (16-bit)11short (16-bit)2int (32-bit)1int (32-bit)2float (32-bit)1float (32-bit)2d4float (32-bit)2float (32-bit)2getData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)float f macro_command main()Examplemacro_command main()		. ,			
short (16-bit)11short (16-bit)22int (32-bit)12int (32-bit)24float (32-bit)12float (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1)// f will contain a floating point value end macro_commandExamplemacro_command main()					
short (16-bit)22int (32-bit)12int (32-bit)24float (32-bit)12float (32-bit)24When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,macro_command main() float f GetData(f, "MODBUS", 6x, 2, 1) // f will contain a floating point value end macro_commandExamplemacro_command main()		. ,	1	1	
int (32-bit)       1       2         int (32-bit)       2       4         float (32-bit)       1       2         float (32-bit)       2       4         When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,         macro_command main()       float f         GetData(f, "MODBUS", 6x, 2, 1)       // f will contain a floating point value end macro_command         Example       macro_command main()					
int (32-bit)       2       4         float (32-bit)       1       2         float (32-bit)       2       4         When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,         macro_command main()         float f         GetData(f, "MODBUS", 6x, 2, 1)       // f will contain a floating point value end macro_command         Example       macro_command main()					
float (32-bit)       1       2         float (32-bit)       2       4         When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,         macro_command main()         float f         GetData(f, "MODBUS", 6x, 2, 1)       // f will contain a floating point value end macro_command         Example       macro_command main()			2	4	
float (32-bit)       2       4         When a GetData() is executed using a 32-bit data type (int or float), the function will automatically convert the data. For example,       macro_command main()         macro_command main()       float f       GetData(f, "MODBUS", 6x, 2, 1)       // f will contain a floating point value end macro_command         Example       macro_command main()       Imacro_command main()       Imacro_command main()			1	2	
function will automatically convert the data. For example,         macro_command main()         float f         GetData(f, "MODBUS", 6x, 2, 1)         end macro_command         Example         macro_command main()			2	4	
macro_command main()         float f         GetData(f, "MODBUS", 6x, 2, 1)       // f will contain a floating point value         end macro_command         Example       macro_command main()		When a GetData() is o	executed using a 32-bit of	data type (int or flo	pat), the
float f         GetData(f, "MODBUS", 6x, 2, 1)       // f will contain a floating point value end macro_command         Example       macro_command main()		function will automat	ically convert the data.	For example,	
end macro_command       Example       macro_command main()			in()		
Example macro_command main()				ntain a floating po	int value
•	Fxample	—			
bool a	LYQUINALE	bool a			

18-42



bool b[30]
short c
short d[50]
int e
int [10]
double g[10]
// get the state of LB2 to the variable a
GetData(a, "Local HMI", LB, 2, 1)
<pre>// get 30 states of LB0 ~ LB29 to the variables b[0] ~ b[29]</pre>
GetData(b[0], "Local HMI", LB, 0, 30)
<pre>// get one word from LW-2 to the variable c</pre>
GetData(c, "Local HMI", LW, 2, 1)
// get 50 words from LW-0 ~ LW-49 to the variables $d[0] \sim d[49]$
GetData(d[0], "Local HMI", LW, 0, 50)
// get 2 words from LW-6 ~ LW-7 to the variable e
// note that the type of e is int
GetData(e, "Local HMI", LW, 6, 1)
// get 20 words (10 integer values) from LW-0 ~ LW-19 to variables f[0] ~ f[9]
// since each integer value occupies 2 words
GetData(f[0], "Local HMI", LW, 0, 10)
<pre>// get 2 words from LW-2 ~ LW-3 to the variable f</pre>
GetData(f, "Local HMI", LW, 2, 1)
end macro_command

Name	GetDataEx
Syntax	GetDataEx(read_data[start], device_name, device_type, address_offset,
	data_count)
	or Calification of the second state
	GetDataEx(read_data, device_name, device_type, address_offset, 1)
Description	Receives data from the PLC and continue executing next command even if no
	response from this device.
	Descriptions of read_data, device_name, device_type, address_offset and
	data_count are the same as GetData.
Example	macro_command main()
	bool a
	bool b[30]



short c short d[50] int e int f[10] double g[10]
// get the state of LB2 to the variable a GetDataEx (a, "Local HMI", LB, 2, 1)
<pre>// get 30 states of LB0 ~ LB29 to the variables b[0] ~ b[29] GetDataEx (b[0], "Local HMI", LB, 0, 30)</pre>
<pre>// get one word from LW-2 to the variable c GetDataEx (c, "Local HMI", LW, 2, 1)</pre>
<pre>// get 50 words from LW-0 ~ LW-49 to the variables d[0] ~ d[49] GetDataEx (d[0], "Local HMI", LW, 0, 50)</pre>
<pre>// get 2 words from LW-6 ~ LW-7 to the variable e // note that he type of e is int GetDataEx (e, "Local HMI", LW, 6, 1)</pre>
<pre>// get 20 words (10 integer values) from LW-0 ~ LW-19 to f[0] ~ f[9] // since each integer value occupies 2 words GetDataEx (f[0], "Local HMI", LW, 0, 10)</pre>
<pre>// get 2 words from LW-2 ~ LW-3 to the variable f GetDataEx (f, "Local HMI", LW, 2, 1)</pre>
end macro_command

Name	SetData
Syntax	SetData(send_data[start], device_name, device_type, address_offset, data_count) or SetData(send_data, device_name, device_type, address_offset, 1)
Description	Send data to the PLC. Data is defined in <i>send_data[start]~ send_data[start + data_count - 1</i> ]. <i>data_count - 1</i> ]. <i>data_count</i> is the amount of sent data. In general, <i>send_data</i> is an array, but if <i>data_count</i> is 1, <i>send_data</i> can be an array or an ordinary variable. Below are two methods to send one word data.
	macro_command main() short send_data_1[2] = { 5, 6}, send_data_2 = 5 SetData(send_data_1[0], "FATEK KB Series", RT, 5, 1)



	• ·	FATEK KB Series	5", RT, 5, 1)	
	end macro_command			
	device_name is the PLC nar		•	
	this name has been defined			
	device_type is the device ty		• • •	
	data. For example, if device		· ·	
	encoding method is binary.			-
	If device_type is LW_BCD, it	t means the reg	ister is LW and the enc	oding method
	is BCD.			
	address_offset is the addre			
	For example, SetData(read	_data_1[0], "FA	TEK KB Series", RT, 5, 1)	represents
	that the address offset is 5.			
	If address_offset uses the f	ormat – "N#AA	AAA", N indicates that F	PLC's station
	number is N. AAAAA repres	sents the addre	ss offset. This format is	used while
	multiple PLCs or controllers	are connected	to a single serial port.	For example,
	SetData(read_data_1[0], "F	ATEK KB Series	", RT, 2#5, 1) represents	s that the PLC's
	station number is 2. If SetD	ata () uses the d	default station number	defined in the
	device list, it is not necessa	ry to define sta	tion number in <i>address</i>	_offset.
	The number of registers act	tually sends to o	depends on both the ty	pe of the
	send_data variable and the	value of the nu	umber of <i>data_count</i> .	_
			actual number of	
	type of <i>read_data</i>	data_count	16-bit register send	
	ahar (0 hit)	1	1	_
	char (8-bit)	1	1	_
	char (8-bit)	2	1	-
	bool (8-bit)	1	1	_
	bool (8-bit)	2	1	_
	short (16-bit)	1	1	_
	short (16-bit)	2	2	_
	int (32-bit)	1	2	4
	int (32-bit)	2	4	4
	float (32-bit)	1	2	
	float (32-bit)	2	4	
	When a SetData() is execute	ed using a 32-b	it data type (int or float	;), the function
	will automatically send int-	format or float-	format data to the dev	ice. For
	example,			
	macro_command main()			
	float f = 2.6			
	SetData(f, "MODBUS", 6x, 2	2, 1) // will se	end a floating point val	ue to the
	device			
	end macro_command			
Example				
	macro_command main()			



bool a = true	
bool b[30]	
short c = false	
short d[50]	
int e = 5	
int f[10]	
for i = 0 to 29	
b[i] = true	
next i	
for i = 0 to 49	
d[i] = i * 2	
next i	
for i = 0 to 9	
f [i] = i * 3	
next i	
// set the state of LB2	
SetData(a, "Local HMI", LB, 2, 1)	
// set the states of LB0 ~ LB29	
SetData(b[0], "Local HMI", LB, 0, 30)	
// set the value of LW-2	
SetData(c, "Local HMI", LW, 2, 1)	
// set the values of LW-0 ~ LW-49	
SetData(d[0], "Local HMI", LW, 0, 50)	
// set the values of LW-6 ~ LW-7, note that the type of e is int	
SetData(e, "Local HMI", LW, 6, 1)	
// set the values of LW-0 ~ LW-19	
// 10 integers equal to 20 words, since each integer value occupies 2 words.	
SetData(f[0], "Local HMI", LW, 0, 10)	
end macro_command	

Name	SetDataEx
Syntax	SetDataEx (send_data[start], device_name, device_type, address_offset, data_count) or
	SetDataEx (send_data, device_name, device_type, address_offset, 1)



Description	Send data to the PLC and continue executing next command even if no response from this device. Descriptions of <i>send_data, device_name, device_type, address_offset</i> and <i>data_count</i> are the same as SetData.
Example	macro_command main() int i bool a = true bool b[30] short c = false short d[50] int e = 5 int f[10]
	for i = 0 to 29 b[i] = true next i
	for i = 0 to 49 d[i] = i * 2 next i
	for i = 0 to 9 f [i] = i * 3 next i
	// set the state of LB2 SetDataEx (a, "Local HMI", LB, 2, 1)
	// set the states of LB0 ~ LB29 SetDataEx (b[0], "Local HMI", LB, 0, 30)
	<pre>// set the value of LW-2 SetDataEx (c, "Local HMI", LW, 2, 1)</pre>
	// set the values of LW-0 ~ LW-49 SetDataEx (d[0], "Local HMI", LW, 0, 50)
	<pre>// set the values of LW-6 ~ LW-7, note that the type of e is int SetDataEx (e, "Local HMI", LW, 6, 1)</pre>
	<pre>// set the values of LW-0 ~ LW-19 // 10 integers equal to 20 words, since each integer value occupies 2 words. SetDataEx (f[0], "Local HMI", LW, 0, 10)</pre>
	end macro_command

Name	GetError
Syntax	GetError ( <i>err</i> )
Description	Get an error code.
Example	macro_command main()
	short err
	char byData[10]
	GetDataEx(byData[0], "MODBUS RTU", 4x, 1, 10)// read 10 bytes // if err is equal to 0, it is successful to execute GetDataEx() GetErr(err)// save an error code to err end macro_command

Name	PURGE
Syntax	PURGE (com_port)
Description	<i>com_port</i> refers to the COM port number which ranges from 1 to 3. It can be
	either a variable or a constant. This function is used to clear the input and
	output buffers associated with the COM port.
Example	macro_command main()
	int com_port=3
	PURGE (com_port)
	PURGE (1)
	end macro_command

Name	SetRTS
Syntax	SetRTS(com_port, source)
Description	Set RTS state for RS232. <i>com_port</i> refers to the COM port number. It can be either a variable or a
	constant. <i>source</i> can be either a variable or a constant.
	This command raise RTS signal while the value of source is greater than 0 and
	lower RTS signal while the value of <i>source</i> equals to 0.
Example	macro_command main()
	char com_port=1
	char value=1
	SetRTS(com_port, value) // raise RTS signal of COM1 while value>0
	SetRTS(1, 0) // lower RTS signal of COM1
	end macro_command



Name	GetCTS
Syntax	GetCTS(com_port, result)
Description	Get CTS state for RS232. <i>com_port</i> refers to the COM port number. It can be either a variable or a constant. <i>result</i> is used for receiving the CTS signal. It must be a variable. This command receives CTS signal and stores the received data in the <i>result</i> variable. When the CTS signal is pulled high, it writes 1 to <i>result</i> , otherwise, it writes 0.
Example	<pre>macro_command main() char com_port=1 char result GetCTS(com_port, result) // get CTS signal of COM1 GetCTS (1, result) // get CTS signal of COM1 end macro_command</pre>

## 18.7.6. String Operation Functions

Name	String	Get						
Syntax	StringGet(read_data[start], device_name, device_type, address_offset, data_count)							
Description	Receives data from the PLC. The String data is stored into <i>read_data[start]</i> ~ <i>read_data[start + data_count - 1]. read_data</i> must be a one-dimensional char array. <i>Data_count</i> is the number of received characters, it can be either a constant or a variable. <i>Device_name</i> is the PLC name enclosed in the double quotation marks (") and this name has been defined in the device list of system parameters as follows (see FATEK KB Series):							
	Syste	m Parameter	Settings				<b>-</b> ×	
		Font	E	xtended Memory		Printer/Backu	p Server	]
		Device	Model	General	S	system Setting	Security	
	De	vice list :						
	N	o.	Name	Location		Device type	Interface	
	Lo	ocal HMI	Local HMI	Local		MT8104iH (800 x	-	
	Lo	ocal Server	Free Protocol	Local		Free Protocol	COM 1 (9600,I	
	Remote PLC 1 FATEK FB Series Remote (IP:192.168.1.10 FATEK FB Series COM 1 (9600,1							
	Device_type is the device type and encoding method (binary or BCD) of the PLC							
	data. F encodi	or exampl ng metho ce_type is	le, if device_ d is binary. If	type is LW_BIN f use BIN enco	I, it r ding	neans the regis method, "_BIN is LW and the e	ster is LW ar I" can be ign	nd the nored.





Address_offset is the address offset in the PLC. For example, StringGet(read_data_1[0], "FATEK KB Series", RT, 5, 1) represents that the address offset is 5. If address_offset uses the format – "N#AAAAA", N indicates that PLC's station number is N. AAAAA represents the address offset. This format is used while multiple PLCs or controllers are connected to a single serial port. For example, StringGet(read_data_1[0], "FATEK KB Series", RT, 2#5, 1) represents that the PLC's station number is 2. If StringGet() uses the default station number defined in the device list as follows, it is not necessary to define station number in address_offset.

	IN address_offset.	•			1		
		TEK FB Series					
		HMI () PL	r				
	Location : Re		Settings ] IP : 192.168.1.10 (I	Port = 8000)			
	PLC type :		FATEK FB Series				
	V.1 PLC I/F : RS	1.80, FATEK_FB.si					
		-232	<b></b>				
	сом : со	M1		Settings			
		C default station n					
		Default station no.	. use station no. variable				
		of block pack (wo					
	Max. write-c	command size (wo	rds) : 64 🔍				
			ОК	Cancel			
	The number of registers actually read from depends on the value of the						
		-	e that the <i>read_data</i>				
	type of read_c		data count	actual numb			
			—	16-bit regist	er read		
	char (8-bit)		1	1			
	char (8-bit)		2	1			
	· · ·	16-bit) eq	uals to the size of 2	ASCII characters	. According to		
		eading 2	ASCII characters is ac		-		
Example	macro_command char str1[20]						
	// read 10 word	s from LW	/-0~LW-9 to the varia	ables str1[0] to s	str1[19]		



<pre>// since that 1 word can store 2 ASCII characters, reading 20 ASCII // characters is actually reading 10 words of register StringGet(str1[0], "Local HMI", LW, 0, 20)</pre>
end macro_command

Name	StringGetEx
Syntax	<pre>StringGetEx (read_data[start], device_name, device_type, address_offset, data_count)</pre>
Description	Receives data from the PLC and continue executing next command even if no response from this device. Descriptions of <i>read_data, device_name, device_type, address_offset</i> and <i>data_count</i> are the same as GetData.
Example	<pre>macro_command main() char str1[20] short test=0 // macro will continue executing test = 1 even if the MODBUS device is // not responding StringGetEx(str1[0], "MODBUS RTU", 4x, 0, 20) test = 1 // macro will not continue executing test = 2 until MODBUS device responds StringGet(str1[0], "MODBUS RTU", 4x, 0, 20) test = 2 end macro_command</pre>



Name	StringSet				
Syntax	StringSet(send_data[sta	rt], device_name, de	vice_type, address_offset,		
	data_count)				
Description	data_count)Send data to the PLC. Data is defined in send_data[start]~ send_data[start + data_count - 1]. send_data must be a one-dimensional char array. data_count is the number of sent characters, it can be either a constant or a variable.device_name is the PLC name enclosed in the double quotation marks (") and this name has been defined in the device list of system parameters. device_type is the device type and encoding method (binary or BCD) of the PLC data. For example, if device_type is LW_BIN, it means the register is LW and the encoding method is binary. If use BIN encoding method, "_BIN" can be ignored. If device_type is LW_BCD, it means the register is LW and the encoding method is BCD. address_offset is the address offset in the PLC. For example, StringSet(read_data_1[0], "FATEK KB Series", RT, 5, 1) represents that the address offset is 5. If address_offset uses the format – "N#AAAAA", N indicates that PLC's station number is N. AAAAA represents the address offset. This format is used while multiple PLCs or controllers are connected to a single serial port. For example, StringSet(read_data_1[0], "FATEK KB Series", RT, 5, 1) represents that the address_offset uses the format – "N#AAAAA", N indicates that PLC's station number is N. AAAAA represents the address offset. This format is used while multiple PLCs or controllers are connected to a single serial port. For example, StringSet(read_data_1[0], "FATEK KB Series", RT, 2#5, 1) represents that the PLC's station number is 2. If SetData () uses the default station number defined in the device list, it is not necessary to define station number in address_offset.				
	of <i>data_count</i> , since tha			_	
	type of	data_count	actual number of		
	read data	_	16-bit register send		
	char (8-bit)	1	1	_	
	char (8-bit)	2	1		
	1 WORD register(16-bit) equals to the size of 2 ASCII characters. According to the above table, sending 2 ASCII characters is actually writing to one 16-bit register. The ASCII characters are stored into the WORD register from low byte to high byte. While using the ASCII Display object to display the string data stored in the registers, <i>data_count</i> must be a multiple of 2 in order to display full string content. For example: macro_command main() char src1[10]="abcde" StringSet(src1[0], "Local HMI", LW, 0, 5) end macro_command				



	The ASCII Display object shows:
	abcd
	If <i>data_count</i> is an even number that is greater than or equal to the length of the string, the content of string can be completely shown:
	macro_command main() char src1[10]="abcde" StringSet(src1[0], "Local HMI", LW, 0, 6) end macro_command
	abcde
Example	macro_command main()
	char str1[10]="abcde"
	<ul> <li>// Send 3 words to LW-0~LW-2</li> <li>// Data are being sent until the end of string is reached.</li> <li>// Even though the value of data_count is larger than the length of string</li> <li>// , the function will automatically stop.</li> <li>StringSet(str1[0], "Local HMI", LW, 0, 10)</li> </ul>
	end macro_command

Name	StringSetEx			
Syntax	<pre>StringSetEx (send_data[start], device_name, device_type, address_offset, data_count)</pre>			
Description	Send data to the PLC and continue executing next command even if no response from this device. Descriptions of <i>send_data, device_name, device_type, address_offset</i> and <i>data_count</i> are the same as StringSet.			
Example	<pre>macro_command main() char str1[20]="abcde" short test=0 // macro will continue executing test = 1 even if the MODBUS device is // not responding StringSetEx(str1[0], "MODBUS RTU", 4x, 0, 20) test = 1 // macro will not continue executing test = 2 until MODBUS device responds StringSet(str1[0], "MODBUS RTU", 4x, 0, 20) test = 2</pre>			



Г

	end macro_command

Name	StringCopy			
Syntax	<pre>success = StringCopy ("source", destination[start])</pre>			
	or			
	<pre>success = StringCopy (source[start], destination[start])</pre>			
Description	Copy one string to another. This function copies a static string (which is enclosed in quotes) or a string that is stored in an array to the destination buffer. The source string parameter accepts both static string (in the form: "source")			
	and char array (in the form: source[start]).			
	destination[start] must be an one-dimensional char array.			
	This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the length of source string exceeds the max. size of destination buffer, it returns false and the content of destination remains the same. The success field is optional.			
Example	macro_command main()			
•	char src1[5]="abcde"			
	char dest1[5]			
	bool success1			
	<pre>success1 = StringCopy(src1[0], dest1[0])</pre>			
	// success1=true, dest1="abcde"			
	char dest2[5]			
	bool success2			
	<pre>success2 = StringCopy("12345", dest2[0]) // success2=true, dest2="12345"</pre>			
	char src3[10]="abcdefghij" char dest3[5]			
	bool success3			
	<pre>success3 = StringCopy(src3[0], dest3[0]) // success3=false, dest3 remains the same.</pre>			
	char src4[10]="abcdefghij" char dest4[5] bool success4			
	success4 = StringCopy(src4[5], dest4[0]) // success4=true, dest4="fghij"			
	end macro_command			



Name	StringDecAsc2Bin
Syntax	<pre>success = StringDecAsc2Bin(source[start], destination)</pre>
	or
	success = StringDecAsc2Bin("source", destination)
Description	This function converts a decimal string to an integer. It converts the decimal
	string in source parameter into an integer, and stores it in the destination
	variable.
	The source string parameter accepts both static string (in the form: "source")
	and char array (in the form: source[start]).
	Destination must be a variable, to store the result of conversion.
	This function returns a Boolean indicating whether the process is successfully
	done or not. If successful, it returns true, otherwise it returns false. If the
	source string contains characters other than '0' to '9', it returns false.
	The success field is optional. macro command main()
Example	char src1[5]="12345"
	int result1
	bool success1
	success1 = StringDecAsc2Bin(src1[0], result1)
	// success1=true, result1 is 12345
	char result2
	bool success2
	success2 = StringDecAsc2Bin("32768", result2)
	<pre>// success2=true, but the result exceeds the data range of result2</pre>
	char src3[2]="4b"
	char result3
	bool success3
	success3 = StringDecAsc2Bin (src3[0], result3)
	<pre>// success3=false, because src3 contains characters other than '0' to '9'</pre>
	end macro_command



Name	StringBin2DecAsc
Syntax	<pre>success = StringBin2DecAsc (source, destination[start])</pre>
Description	This function converts an integer to a decimal string. It converts the integer in source parameter into a decimal string, and stores it in the destination buffer. Source can be either a constant or a variable. Destination must be an one-dimensional char array, to store the result of conversion. This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the length of decimal string after conversion exceeds the size of destination buffer, it returns false. The success field is optional.
Example	<pre>macro_command main() int src1 = 2147483647 char dest1[20] bool success1 success1 = StringBin2DecAsc(src1, dest1[0]) // success1=true, dest1="2147483647" short src2 = 0x3c char dest2[20] bool success2 success2 = StringBin2DecAsc(src2, dest2[0]) // success2=true, dest2="60" int src3 = 2147483647 char dest3[5] bool success3 success3 = StringBin2DecAsc(src3, dest3[0]) // success3=false, dest3 remains the same. end macro_command</pre>



Name	StringDecAsc2Float
Syntax	success = StringDecAsc2Float (source[start], destination)
	or
	success = StringDecAsc2Float ("source", destination)
Description	This function converts a decimal string to floats. It converts the decimal string in source parameter into float, and stores it in the destination variable. The source string parameter accepts both static string (in the form: "source") and char array (in the form: source[start]). Destination must be a variable, to store the result of conversion. This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the source string contains characters other than '0' to '9' or '', it returns false. The success field is optional.
Example	<pre>macro_command main() char src1[10]="12.345" float result1 bool success1 success1 = StringDecAsc2Float(src1[0], result1) // success1=true, result1 is 12.345</pre>
	float result2 bool success2 success2 = StringDecAsc2Float("1.234567890", result2) // success2=true, but the result exceeds the data range of result2, which // might result in loss of precision
	<pre>char src3[2]="4b" float result3 bool success3 success3 = StringDecAsc2Float(src3[0], result3) // success3=false, because src3 contains characters other than '0' to '9' or // '.' end macro_command</pre>



Name	StringFloat2DecAsc
Syntax	<pre>success = StringFloat2DecAsc(source, destination[start])</pre>
Description	This function converts a float to a decimal string. It converts the float in source parameter into a decimal string, and stores it in the destination buffer. Source can be either a constant or a variable. Destination must be an one-dimensional char array, to store the result of conversion. This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the length of decimal string after conversion exceeds the size of destination buffer, it returns false. The success field is optional.
Example	<pre>macro_command main() float src1 = 1.2345 char dest1[20] bool success1 success1 = StringFloat2DecAsc(src1, dest1[0]) // success1=true, dest1="1.2345" float src2 = 1.23456789 char dest2 [20] bool success2 success2 = StringFloat2DecAsc(src2, dest2 [0]) // success2=true, but it might lose precision float src3 = 1.2345 char dest3[5] bool success3 success3 = StringFloat2DecAsc(src3, dest3 [0]) // success3=false, dest3 remains the same.</pre>
	end macro_command



Name	StringHexAsc2Bin
Syntax	success = StringHexAsc2Bin (source[start], destination)
-	or
	success = StringHexAsc2Bin ("source", destination)
Description	This function converts a hexadecimal string to binary data. It converts the
	hexadecimal string in source parameter into binary data, and stores it in the
	destination variable.
	The source string parameter accepts both static string (in the form: "source")
	and char array (in the form: source[start]).
	Destination must be a variable, to store the result of conversion.
	This function returns a Boolean indicating whether the process is successfully
	done or not. If successful, it returns true, otherwise it returns false. If the
	source string contains characters other than '0' to '9', 'a' to 'f' or 'A' to 'F', it
	returns false.
	The success field is optional.
Example	macro_command main()
	char src1[5]="0x3c"
	int result1
	bool success1
	<pre>success1 = StringHexAsc2Bin(src1[0], result1) // success1=true_result1 is 2s</pre>
	// success1=true, result1 is 3c
	short result2
	bool success2
	success2 = StringDecAsc2Bin("1a2b3c4d", result2)
	// success2=true, result2=3c4d.The result exceeds the data range of
	// result2
	char src3[2]="4g"
	char result3
	bool success3
	success3 = StringDecAsc2Bin (src3[0], result3)
	// success3=false, because src3 contains characters other than '0' to '9'
	// , 'a' to 'f' or 'A' to 'F'
	end macro_command



Name	StringBin2HexAsc
Syntax	success = StringBin2HexAsc (source, destination[start])
Description	This function converts binary data to a hexadecimal string. It converts the binary data in source parameter into a hexadecimal string, and stores it in the destination buffer. Source can be either a constant or a variable. Destination must be an one-dimensional char array, to store the result of conversion. This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the length of hexadecimal string after conversion exceeds the size of destination buffer, it returns false. The success field is optional.
Example	macro_command main() int src1 = 20 char dest1[20] bool success1 success1 = StringBin2HexAsc(src1, dest1[0]) // success1=true, dest1="14" short src2 = 0x3c char dest2[20] bool success2 success2 = StringBin2HexAsc(src2, dest2[0]) // success2=true, dest2="3c" int src3 = 0x1a2b3c4d char dest3[6] bool success3 success3 = StringBin2HexAsc(src3, dest3[0]) // success3=false, dest3 remains the same. end macro_command



Name	StringMid
Syntax	success = StringMid (source[start], count, destination[start])
	or
	<pre>success = StringMid ("string", start, count, destination[start])</pre>
Description	Retrieve a character sequence from the specified offset of the source string and store it in the destination buffer. The source string parameter accepts both static string (in the form: "source") and char array (in the form: source[start]). For source[start], the start offset of
	the substring is specified by the index value. For static source string("source"), the second parameter(start) specifies the start offset of the substring.
	The count parameter specifies the length of substring being retrieved. Destination must be an one-dimensional char array, to store the retrieved substring.
	This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the
	length of retrieved substring exceeds the size of destination buffer, it returns false.
	The success field is optional.
Example	macro_command main() char src1[20]="abcdefghijklmnopqrst" char dest1[20] bool success1
	<pre>success1 = StringMid(src1[5], 6, dest1[0]) // success1=true, dest1="fghijk"</pre>
	char src2[20]="abcdefghijklmnopqrst" char dest2[5]
	bool success2
	<pre>success2 = StringMid(src2[5], 6, dest2[0]) // success2=false, dest2 remains the same.</pre>
	char dest3[20]="12345678901234567890" bool success3
	success3 = StringMid("abcdefghijklmnopqrst", 5, 5, dest3[15]) // success3= true, dest3="123456789012345fghij"
	end macro_command



Name	StringLength
Syntax	length = StringLength (source[start])
	or
	length = StringLength ("source")
Description	Obtain the length of a string. It returns the length of source string and stores it in the length field on the left-hand side of '=' operator.
	The source string parameter accepts both static string (in the form: "source")
	and char array (in the form: source[start]).
	The return value of this function indicates the length of the source string.
Example	macro_command main()
	char src1[20]="abcde"
	int length1
	length1= StringLength(src1[0])
	// length1=5
	char src2[20]={'a', 'b', 'c', 'd', 'e'}
	int length2
	length2= StringLength(src2[0])
	// length2=20
	char src3[20]="abcdefghij"
	int length3
	length3= StringLength(src3 [2])
	// length3=8
	end macro_command



Name	StringCat
Syntax	success = StringCat (source[start], destination[start])
-	or
	<pre>success = StringCat ("source", destination[start])</pre>
Description	This function appends source string to destination string. It adds the contents of source string to the last of the contents of destination string. The source string parameter accepts both static string (in the form: "source") and char array (in the form: source[start]). Destination must be an one-dimensional char array.
	This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the length of result string after concatenation exceeds the max. size of destination buffer, it returns false.
	The success field is optional.
Example	<pre>macro_command main() char src1[20]="abcdefghij" char dest1[20]="1234567890" bool success1 success1= StringCat(src1[0], dest1[0]) // success1=true, dest1="123456790abcdefghij"</pre>
	char dest2 [10]="1234567890" bool success2 success2= StringCat("abcde", dest2 [0]) // success2=false, dest2 remains the same.
	char src3[20]="abcdefghij" char dest3[20] bool success3 success3= StringCat(src3[0], dest3[15]) // success3=false, dest3 remains the same.
	end macro_command



Name	StringCompare
Syntax	<pre>ret = StringCompare (str1[start], str2[start]) ret = StringCompare ("string1", str2[start]) ret = StringCompare (str1[start], "string2") ret = StringCompare ("string1", "string2")</pre>
Description	Do a case-sensitive comparison of two strings. The two string parameters accept both static string (in the form: "string1") and char array (in the form: str1[start]). This function returns a Boolean indicating the result of comparison. If two strings are identical, it returns true. Otherwise it returns false. The ret field is optional.
Example	macro_command main() char a1[20]="abcde" char b1[20]="ABCDE" bool ret1 ret1= StringCompare(a1[0], b1[0]) // ret1=false char a2[20]="abcde" char b2[20]="abcde" bool ret2 ret2= StringCompare(a2[0], b2[0]) // ret2=true char a3 [20]="abcde" char b3[20]="abcdefg" bool ret3 ret3= StringCompare(a3[0], b3[0]) // ret3=false
	end macro_command



Name	StringCompareNoCase
Syntax	ret = StringCompareNoCase(str1[start], str2[start])
-	ret = StringCompareNoCase("string1", str2[start])
	ret = StringCompareNoCase(str1[start], "string2")
	ret = StringCompareNoCase("string1", "string2")
Description	Do a case-insensitive comparison of two strings.
	The two string parameters accept both static string (in the form: "string1") and
	char array (in the form: str1[start]).
	This function returns a Boolean indicating the result of comparison. If two
	strings are identical, it returns true. Otherwise it returns false.
	The ret field is optional.
Example	macro_command main()
	char a1[20]="abcde"
	char b1[20]="ABCDE"
	bool ret1
	ret1= StringCompareNoCase(a1[0], b1[0])
	// ret1=true
	char a2[20]="abcde"
	char b2[20]="abcde"
	bool ret2
	ret2= StringCompareNoCase(a2[0], b2[0])
	// ret2=true
	char a3 [20]="abcde"
	char b3[20]="abcdefg"
	bool ret3
	ret3= StringCompareNoCase(a3[0], b3[0])
	// ret3=false
	end macro_command



Name	StringFind
Syntax	position = StringFind (source[start], target[start]) position = StringFind ("source", target[start])
	position = StringFind (source[start], "target")
Description	position = StringFind ("source", "target") Return the position of the first occurrence of target string in the source string. The two string parameters accept both static string (in the form: "source") and char array (in the form: source[start]). This function returns the zero-based index of the first character of substring in the source string that matches the target string. Notice that the entire sequence of characters to find must be matched. If there is no matched substring, it returns -1.
Example	<pre>macro_command main() char src1[20]="abcde" char target1[20]="cd" bool pos1 pos1= StringFind(src1[0], target1[0]) // pos1=2</pre>
	char target2[20]="ce" bool pos2 pos2= StringFind("abcde", target2[0]) // pos2=-1
	char src3[20]="abcde" bool pos3 pos3= StringFind(src3[3], "cd") // pos3=-1
	end macro_command



Name	StringReverseFind
Syntax	position = StringReverseFind (source[start], target[start])
	position = StringReverseFind ("source", target[start])
	position = StringReverseFind (source[start], "target")
	position = StringReverseFind ("source", "target")
Description	Return the position of the last occurrence of target string in the source string.
	The two string parameters accept both static string (in the form: "source") and
	char array (in the form: source[start]).
	This function returns the zero-based index of the first character of substring in
	the source string that matches the target string. Notice that the entire
	sequence of characters to find must be matched. If there exists multiple
	substrings that matches the target string, function will return the position of
	the last matched substring. If there is no matched substring, it returns -1.
Example	macro_command main()
-	char src1[20]="abcdeabcde"
	char target1[20]="cd"
	bool pos1
	pos1= StringReverseFind(src1[0], target1[0])
	// pos1=7
	// P /
	char target2[20]="ce"
	bool pos2
	pos2= StringReverseFind("abcdeabcde", target2[0])
	// pos2=-1
	// posz=-1
	char src3[20]="abcdeabcde"
	bool pos3
	•
	pos3= StringReverseFind(src3[6], "ab")
	// pos3=-1
	end macro_command



Name	StringFindOneOf
Syntax	<pre>position = StringFindOneOf (source[start], target[start]) position = StringFindOneOf ("source", target[start]) position = StringFindOneOf (source[start], "target") position = StringFindOneOf ("source", "target")</pre>
Description	Return the position of the first character in the source string that matches any character contained in the target string. The two string parameters accept both static string (in the form: "source") and char array (in the form: source[start]). This function returns the zero-based index of the first character in the source string that is also in the target string. If there is no match, it returns -1.
Example	<pre>macro_command main() char src1[20]="abcdeabcde" char target1[20]="sdf" bool pos1 pos1= StringFindOneOf(src1[0], target1[0]) // pos1=3 char src2[20]="abcdeabcde" bool pos2 pos2= StringFindOneOf(src2[1], "agi") // pos2=4 char target3 [20]="bus" bool pos3 pos3= StringFindOneOf("abcdeabcde", target3[1]) // pos3=-1 end macro_command</pre>





Name	StringIncluding
Syntax	<pre>success = StringIncluding (source[start], set[start], destination[start]) success = StringIncluding ("source", set[start], destination[start]) success = StringIncluding (source[start], "set", destination[start]) success = StringIncluding ("source", "set", destination[start])</pre>
Description	Retrieve a substring of the source string that contains characters in the set string, beginning with the first character in the source string and ending when a character is found in the source string that is not in the target string. The source string and set string parameters accept both static string (in the form: "source") and char array (in the form: source[start]). This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the length of retrieved substring exceeds the size of destination buffer, it returns false.
Example	<pre>macro_command main() char src1[20]="cabbageabc" char set1[20]="abc" char dest1[20] bool success1 success1 = StringIncluding(src1[0], set1[0], dest1[0]) // success1=true, dest1="cabba" char src2[20]="gecabba" char dest2[20] bool success2 success2 = StringIncluding(src2[0], "abc", dest2[0]) // success2=true, dest2="" char set3[20]="abc" char dest3[4] bool success3 success3 = StringIncluding("cabbage", set3[0], dest3[0]) // success3=false, dest3 remains the same.</pre>
	end macro_command



Name	StringExcluding
Syntax	<pre>success = StringExcluding (source[start], set[start], destination[start])</pre>
	<pre>success = StringExcluding ("source", set[start], destination[start])</pre>
	<pre>success = StringExcluding (source[start], "set", destination[start])</pre>
	<pre>success = StringExcluding ("source", "set", destination[start])</pre>
Description	Retrieve a substring of the source string that contains characters that are not in the set string, beginning with the first character in the source string and ending when a character is found in the source string that is also in the target string. The source string and set string parameters accept both static string (in the form: "source") and char array (in the form: source[start]). This function returns a Boolean indicating whether the process is successfully done or not. If successful, it returns true, otherwise it returns false. If the length of retrieved substring exceeds the size of destination buffer, it returns
	false.
Example	<pre>macro_command main() char src1[20]="cabbageabc" char set1[20]="ge" char dest1[20] bool success1 success1 = StringExcluding(src1[0], set1[0], dest1[0]) // success1=true, dest1="cabba"</pre>
	char src2[20]="cabbage" char dest2[20] bool success2 success2 = StringExcluding(src2[0], "abc", dest2[0]) // success2=true, dest2=""
	char set3[20]="ge" char dest3[4] bool success3 success3 = StringExcluding("cabbage", set3[0], dest3[0]) // success3=false, dest3 remains the same.
	end macro_command



Name	StringToUpper
Syntax	<pre>success = StringToUpper (source[start], destination[start])</pre>
	<pre>success = StringToUpper ("source", destination[start])</pre>
Description	Convert all the characters in the source string to uppercase characters and
	store the result in the destination buffer.
	The source string parameter accepts both static string (in the form: "source")
	and char array (in the form: source[start]).
	This function returns a Boolean indicating whether the process is successfully
	done or not. If successful, it returns true, otherwise it returns false. If the
	length of result string after conversion exceeds the size of destination buffer, it
	returns false.
Example	macro_command main()
	char src1[20]="aBcDe"
	char dest1[20]
	bool success1
	<pre>success1 = StringToUpper(src1[0], dest1[0])</pre>
	<pre>// success1=true, dest1="ABCDE"</pre>
	char dest2[4]
	bool success2
	success2 = StringToUpper("aBcDe", dest2[0])
	// success2=false, dest2 remains the same.
	end macro_command

Name	StringToLower
Syntax	<pre>success = StringToLower (source[start], destination[start])</pre>
	<pre>success = StringToLower ("source", destination[start])</pre>
Description	Convert all the characters in the source string to lowercase characters and store
	the result in the destination buffer.
	The source string parameter accepts both static string (in the form: "source")
	and char array (in the form: source[start]).
	This function returns a Boolean indicating whether the process is successfully
	done or not. If successful, it returns true, otherwise it returns false. If the
	length of result string after conversion exceeds the size of destination buffer, it
	returns false.



Example	<pre>macro_command main() char src1[20]="aBcDe" char dest1[20] bool success1 success1 = StringToUpper(src1[0], dest1[0]) // success1=true, dest1="abcde"</pre>
	char dest2[4] bool success2 success2 = StringToUpper("aBcDe", dest2[0]) // success2=false, dest2 remains the same. end macro_command

Name	StringToReverse
Syntax	<pre>success = StringToReverse (source[start], destination[start])</pre>
	<pre>success = StringToReverse ("source", destination[start])</pre>
Description	Reverse the characters in the source string and store it in the destination buffer.
	The source string parameter accepts both static string (in the form: "source")
	and char array (in the form: source[start]).
	This function returns a Boolean indicating whether the process is successfully
	done or not. If successful, it returns true, otherwise it returns false. If the
	length of reversed string exceeds the size of destination buffer, it returns false.
Example	macro_command main()
	char src1[20]="abcde"
	char dest1[20]
	bool success1
	success1 = StringToUpper(src1[0], dest1[0])
	<pre>// success1=true, dest1="edcba"</pre>
	char dest2[4]
	bool success2
	success2 = StringToUpper("abcde", dest2[0])
	// success2=false, dest2 remains the same.
	end macro_command



Name	StringTrimLeft
Syntax	<pre>success = StringTrimLeft (source[start], set[start], destination[start])</pre>
	success = StringTrimLeft ("source", set[start], destination[start])
	<pre>success = StringTrimLeft (source[start], "set", destination[start])</pre>
	<pre>success = StringTrimLeft ("source", "set", destination[start])</pre>
Description	Trim the leading specified characters in the set buffer from the source string.
	The source string and set string parameters accept both static string (in the
	form: "source") and char array (in the form: source[start]).
	This function returns a Boolean indicating whether the process is successfully
	done or not. If successful, it returns true, otherwise it returns false. If the
	length of trimmed string exceeds the size of destination buffer, it returns false.
Example	macro_command main()
	char src1[20]= "# *a*#bc"
	char set1[20]="# *"
	char dest1[20]
	bool success1
	<pre>success1 = StringTrimLeft (src1[0], set1[0], dest1[0])</pre>
	<pre>// success1=true, dest1="a*#bc"</pre>
	char set2[20]={'#', ' ', '*'}
	char dest2[4]
	success2 = StringTrimLeft ("# *a*#bc", set2[0], dest2[0])
	<pre>// success2=false, dest2 remains the same.</pre>
	char src3[20]="abc *#"
	char dest3[20]
	bool success3
	success3 = StringTrimLeft (src3[0], "# *", dest3[0])
	<pre>// success3=true, dest3="abc *#"</pre>
	end macro_command



Name	StringTrimRight
Syntax	success = StringTrimRight (source[start], set[start], destination[start])
	success = StringTrimRight ("source", set[start], destination[start])
	success = StringTrimRight (source[start], "set", destination[start])
	<pre>success = StringTrimRight ("source", "set", destination[start])</pre>
Description	Trim the trailing specified characters in the set buffer from the source string.
_	The source string and set string parameters accept both static string (in the
	form: "source") and char array (in the form: source[start]).
	This function returns a Boolean indicating whether the process is successfully
	done or not. If successful, it returns true, otherwise it returns false. If the
	length of trimmed string exceeds the size of destination buffer, it returns false.
Example	macro_command main()
	char src1[20]= "# *a*#bc# * "
	char set1[20]="# *"
	char dest1[20]
	bool success1
	<pre>success1 = StringTrimRight(src1[0], set1[0], dest1[0])</pre>
	<pre>// success1=true, dest1="# *a*#bc"</pre>
	char set2[20]={'#', ' ', '*'}
	char dest2[20]
	success2 = StringTrimRight("# *a*#bc", set2[0], dest2[0])
	<pre>// success2=true, dest2="# *a*#bc"</pre>
	char src3[20]="ab**c *#"
	char dest3[4]
	bool success3
	<pre>success3 = StringTrimRight(src3[0], "# *", dest3[0])</pre>
	<pre>// success3=false, dest3 remains the same.</pre>
	end macro_command



Name	StringInsert
Syntax	success = StringInsert (pos, insert[start], destination[start])
-,	success = StringInsert (pos, "insert", destination[start])
	success = StringInsert (pos, insert[start], length, destination[start])
	success = StringInsert (pos, "insert", length, destination[start])
Description	Insert a string in a specific location within the destination string content. The
_	insert location is specified by the pos parameter.
	The insert string parameter accepts both static string (in the form: "source")
	and char array (in the form: source[start]).
	The number of characters to insert can be specified by the length parameter.
	This function returns a Boolean indicating whether the process is successfully
	done or not. If successful, it returns true, otherwise it returns false. If the
	length of string after insertion exceeds the size of destination buffer, it returns
	false.
Example	macro_command main()
	char str1[20]="but the question is"
	char str2[10]=", that is"
	char dest[40]="to be or not to be"
	bool success
	success = StringInsert(18, str1[3], 13, dest[0])
	<pre>// success=true, dest="to be or not to be the question"</pre>
	<pre>success = StringInsert(18, str2[0], dest[0])</pre>
	<pre>// success=true, dest="to be or not to be, that is the question"</pre>
	success = StringInsert(0, "Hamlet:", dest[0])
	// success=false, dest remains the same.
	end macro_command



## **18.7.7.** Recipe Query Function

Name	RecipeGetData
Syntax	RecipeGetData(destination, recipe_address, record_ID)
Description	Get Recipe Data. The gained data will be stored in <i>destination</i> , and must be a variable. <i>recipe_address</i> consists of recipe name and item name: "recipe_name.item_name". record_ID specifies the ID number of the record in recipe being gained.
Example	<pre>macro_command main() int data=0 char str[20] int recordID bool result recordID = 0 result = RecipeGetData(data, "TypeA.item_weight", recordID) // From recipe "TypeA" get the data of the item "item_weight" in record 0. recordID = 1 result = RecipeGetData(str[0], "TypeB.item_name", recordID) // From recipe "TypeB" get the data of the item "item_name" in record 1.</pre>
	end macro_command

Name	RecipeQuery
Syntax	RecipeQuery (SQL_command, destination)
Description	Use SQL statement to query recipe data. The number of records of query result will be stored in the <i>destination</i> . This must be a variable. SQL command can be static string or char array. Example: RecipeQuery("SELECT * FROM TypeA", destination) or RecipeQuery(sql[0], destination) SQL statement must start with "SELECT * FROM" followed by recipe name and query condition.
Example	<pre>macro_command main() int total_row=0 char sql[100]="SELECT * FROM TypeB" bool result result = RecipeQuery("SELECT * FROM TypeA", total_row) // Query Recipe "TypeA". Store the number of records of query result in total_row. result = RecipeQuery(sql[0], total_row) // Query Recipe "TypeB". Store the number of records of query result in total_row.</pre>



end macro_command

Name	RecipeQueryGetData
Syntax	RecipeQueryGetData (destination, recipe_address, result_row_no)
Description	Get the data in the query result obtained by RecipeQuery. This function must be called after calling RecipeQuery, and specify the same recipe name in <i>recipe_address</i> as RecipeQuery. <i>result_row_no</i> specifies the sequence row number in query result
Example	<pre>macro_command main() int data=0 int total_row=0 int row_number=0 bool result_query bool result_data result_query = RecipeQuery("SELECT * FROM TypeA", total_row) // Query Recipe "TypeA". Store the number of records of query result in total_row. if (result_query) then for row_number=0 to total_row-1 result_data = RecipeQueryGetData(data, "TypeA.item_weight", row_number) next row_number end if end macro_command</pre>

Name	RecipeQueryGetRecordID
Syntax	RecipeQueryGetRecordID (destination, result_row_no)
Description	Get the record ID numbers of those records gained by RecipeQuery. This function must be called after calling RecipeQuery. <i>result_row_no</i> specifies the sequence row number in query result, and write the obtained record ID to destination.
Example	<pre>macro_command main() int recordID=0 int total_row=0 int row_number=0 bool result_query bool result_id  result_query = RecipeQuery("SELECT * FROM TypeA", total_row) // Query Recipe "TypeA". Store the number of records of query result in total_row. if (result_query) then for row_number=0 to total_row-1 result_id = RecipeQueryGetRecordID(recordID, row_number)</pre>



next row_number end if
end macro_command

Name	RecipeSetData
Syntax	RecipeSetData(source, recipe address, record_ID)
Description	Write data to recipe. If success, returns true, else, returns false. recipe_address consists of recipe name and item name: "recipe_name.item_name". record_ID specifies the ID number of the record in recipe being modified.
Example	<pre>macro_command main() int data=99 char str[20]="abc" int recordID bool result recordID = 0 result = RecipeSetData(data, "TypeA.item_weight", recordID) // set data to recipe "TypeA", where item name is "item_weight" and the record ID is 0. recordID = 1 result = RecipeSetData(str[0], "TypeB.item_name", recordID) // set data to recipe "TypeB", where item name is "item_name" and the record ID is 1. end macro_command</pre>

## 18.7.8. Miscellaneous

Name	Веер
Syntax	Beep ()
Description	Plays beep sound.
-	This command plays a beep sound with frequency of 800 hertz and duration of
	30 milliseconds.
Example	macro_command main()
	Beep() end macro command



Name	Buzzer
Syntax	Buzzer ()
Description	Turn ON / OFF the buzzer.
Example	char on = 1, off = 0
	Buzzer(on) // turn on the buzzer
	DELAY(1000) // delay 1 second
	Buzzer(off) // turn off the buzzer
	DELAY(500) // delay 500ms
	DELAT(500) // delay 500ms
	$P_{\text{upper}}(1)$ // turn on the hurzer
	Buzzer(1) // turn on the buzzer
	DELAY(1000) // delay 1 second
	Buzzer(0) // turn off the buzzer
	DELAY(1000) // delay 1 second Buzzer(0) // turn off the buzzer

Name	SYNC_TRIG_MACRO
Syntax	SYNC_TRIG_MACRO(macro_id or name)
Description	Trigger the execution of a macro synchronously (use <i>macro_id</i> or macro name to designate this macro) in a running macro. The current macro will pause until the end of execution of this called macro. <i>macro_id</i> can be a constant or a variable.
Example	<pre>macro_command main() char ON = 1, OFF = 0 SetData(ON, "Local HMI", LB, 0, 1) SYNC_TRIG_MACRO(5)// call a macro (its ID is 5) SYNC_TRIG_MACRO( "macro_1" ) // call a macro (its name is macro_1) SetData(OFF, "Local HMI", LB, 0, 1) and macro_command</pre>
	end macro_command

Name	ASYNC_TRIG_MACRO
Syntax	ASYNC_TRIG_MACRO (macro_id or name)
Description	Trigger the execution of a macro asynchronously (use <i>macro_id or macro name</i> to designate this macro) in a running macro. The current macro will continue executing the following instructions after triggering the designated macro; in other words, the two macros will be active simultaneously. <i>macro_id</i> can be a constant or a variable.
Example	macro_command main() char ON = 1, OFF = 0 SetData(ON, "Local HMI", LB, 0, 1) ASYNC_TRIG_MACRO(5)// call a macro (its ID is 5) ASYNC_TRIG_MACRO( "macro_1" ) // call a macro (its name is macro_1) SetData(OFF, "Local HMI", LB, 0, 1) end macro command

Name	TRACE
Syntax	TRACE(format, argument)
Description	Use this function to send specified string to the EasyDiagnoser. Users can print out the current value of variables during run-time of macro for debugging. When TRACE encounters the first format specification (if any), it converts the value of the first argument after format and outputs it accordingly. format refers to the format control of output string. A format specification, which consists of optional (in []) and required fields (in bold), has the following form: %[flags] [width] [.precision] <b>type</b> Each field of the format specification is described as below: <i>flags</i> (optional): - +
	<ul> <li>width (optional):         <ul> <li>A nonnegative decimal integer controlling the minimum number of characters printed.</li> <li>precision (optional):</li></ul></li></ul>
	type:C or c: specifies a single-byte character.d: signed decimal integer.i: signed decimal integer.o: unsigned octal integer.u: unsigned decimal integer.X or x: unsigned hexadecimal integer.



	<ul> <li>E or e : Signed value having the form. [ – ]d.dddd e [sign]ddd where d is a single decimal digit, dddd is one or more decimal digits, ddd is exactly three decimal digits, and sign is + or –.</li> <li>f : Signed value having the form [ – ]dddd.dddd, v</li> </ul> The length of output string is limited to 256 characters. The extra characters will be ignored. The <i>argument</i> part is optional. One format specification converts exactly one argument.
Example	macro_command main() char c1 = 'a' short s1 = 32767 float f1 = 1.234567 TRACE("The results are") // output: The results are TRACE("c1 = %c, s1 = %d, f1 = %f", c1, s1, f1) // output: c1 = a, s1 = 32767, f1 = 1.234567 end macro_command

Name	FindDataSamplingDate
Syntax	return_value = FindDataSamplingDate (data_log_number, index, year, month, day) or FindDataSamplingDate (data_log_number, index, year, month, day)
Description	A query function for finding the date of specified data sampling file according to the data sampling no. and the file index. The date is stored into year, month and day respectively in the format of YYYY, MM and DD.
	No.         Description         Read address         Sample mode         Trigger address         Clear address         Hold address         Auto. stop           1         Local HMI:LW-0         Periodical         Disable         Disable         Disable         Enable
	2 Local HMI : LW-100 Periodical Disable Local HMI : LB0 Local HMI : LB0 Enable Data sampling no.
	The directory of saved data: [Storage location]\[filename]\yyyymmdd.dtl. The data sampling files under the same directory are sorted according to the file name and are indexed starting from 0. The most recently saved file has the smallest file index number. For example, if there are four data sampling files as follows:
	20101210.dtl
	20101230.dtl
	20110110.dtl
	20110111.dtl
	The file index are:
	20101210.dtl -> index is 3
	20101230.dtl -> index is 2
	20110110.dtl -> index is 1
	20110111.dtl -> index is 0



	<i>return_value</i> equals to 1 if referred data sampling file is successfully found, otherwise it equals to 0. <i>data_log_number</i> and <i>index</i> can be constant or variable. <i>year, month, day</i> and <i>return_value</i> must be variable. <i>return_value</i> is optional.
Example	<pre>macro_command main() short data_log_number = 1, index = 2, year, month, day short success // if there exists a data sampling file named 20101230.dtl, with data sampling // number 1 and file index 2. // the result after execution: success == 1, year == 2010, month == 12 and //day == 30 success = FindDataSamplingDate(data_log_number, index, year, month, day) end macro_command</pre>

Name	FindDataSamplingIndex
Syntax	return_value = FindDataSamplingIndex ( <i>data_log_number, year, month, day, index</i> ) or FindDataSamplingIndex ( <i>data_log_number, year, month, day, index</i> )
Description	•••
	20101210.dtl -> index is 3 20101230.dtl -> index is 2 20110110.dtl -> index is 1 20110111.dtl -> index is 0
	<i>return_value</i> equals to 1 if referred data sampling file is successfully found, otherwise it equals to 0. <i>data_log_number, year, month</i> and <i>day</i> can be constant or variable. <i>index</i> and <i>return value</i> must be variable. <i>return value</i> is optional.



Example	macro_command main() short data_log_number = 1, year = 2010, month = 12, day = 10, index short success
	<pre>// if there exists a data sampling file named 20101210.dtl, with data sampling // number 1 and file index 2. // the result after execution: success == 1 and index == 2 success = FindDataSamplingIndex (data_log_number, year, month, day, index) end macro_command</pre>



EasyBuilder Pro V5.02.01

Name	FindEventLogDate
Syntax	return value = FindEventLogDate ( <i>index, year, month, day</i> )
-,	or
	FindEventLogDate ( <i>index, year, month, day</i> )
Description	A query function for finding the date of specified event log file according to file index. The date is stored into year, month and day respectively in the format of YYYY, MM and DD.
	The event log files stored in the designated position (such as HMI memory storage or external memory device) are sorted according to the file name and are indexed starting from 0. The most recently saved file has the smallest file index number. For example, if there are four event log files as follows:
	EL_20101210.evt EL_20101230.evt EL_20110110.evt
	EL_20110111.evt The file index are: EL_20101210.evt -> index is 3
	EL_20101230.evt -> index is 2 EL_20110110.evt -> index is 1
	EL_20110111.evt -> index is 0 return_value equals to 1 if referred data sampling file is successfully found, otherwise it equals to 0.
	<i>index</i> can be constant or variable. <i>year, month, day</i> and <i>return_value</i> must be variable. <i>return_value</i> is optional.
Example	macro_command main()

Example	macro_command main()
-	short index = 1, year, month, day
	short success
	<pre>// if there exists an event log file named EL_20101230.evt , with index 1 // the result after execution: success == 1, year == 2010, month == 12, day //== 30 success = FindEventLogDate (index, year, month, day)</pre>
	end macro_command



Name	FindEventLogIndex	
Syntax	return_value = FindEventLogIndex ( <i>year, month, day, index</i> )	
	or	
	FindEventLogIndex (year, month, day, index)	
Description	A query function for finding the file index of specified event log file according to date. The file index is stored into index. year, month and day are in the format of YYYY, MM and DD respectively. The event log files stored in the designated position (such as HMI memory storage or external memory device) are sorted according to the file name and are indexed starting from 0. The most recently saved file has the smallest file index number. For example, if there are four event log files as follows: EL_20101210.evt EL_20101230.evt EL_20110110.evt EL_20110111.evt	
	The file index are:	
	EL 20101210.evt -> index is 3	
	EL_20110110.evt -> index is 1	
	EL_20110111.evt -> index is 0	
	return_value equals to 1 if referred data sampling file is successfully found,	
	otherwise it equals to 0.	
	<i>index</i> can be constant or variable. <i>year, month, day</i> and <i>return_value</i> must be variable. <i>return_value</i> is optional.	
Example	macro_command main() short year = 2010, month = 12, day = 10, index short success	
	<pre>// if there exists an event log file named EL_20101210.evt, with index 2 // the result after execution: success == 1, index == 2 success = FindEventLogIndex (year, month, day, index)</pre>	
	end macro_command	

## **18.8.** How to Create and Execute a Macro

## 18.8.1. How to Create a Macro

Please follow the steps below to create a macro.

1. Click on 📓 [Macro Manager] icon on the tool bar in EasyBuilder Pro to open Macro Manager dialog box as follows.



Macro	<b>—</b> ———————————————————————————————————
Macro list	
	New
	Delete
	Edit
	Сору
	Paste
	ОК
	Cancel
	Library
*I : Execute one time when HMI starts *P : Periodical execution *S : Use execution condition	
Macro under development	
	Help
Password protect	
*Decompilation cannot recover MACROs when checks [Password prote-	ct].
Address variables use [DDDDdd] address format to access [DDDDh] p address format in Macro functions (i.e. SetData, GetData,)	oartial-hexadecimal

In Macro Manager, all macros compiled successfully are displayed in "Macro list", and all macros under development or cannot be compiled are displayed in "Macro under development". The following is a description of the various buttons.

Setting Description	
New	Opens a blank "WorkSpace" editor for creating a
	new macro.
Delete	Deletes the selected macro.
Edit	Opens the "WorkSpace" editor, and loads the
	selected macro.
<b>Copy</b> Copies the selected macro into the clipboard.	
Paste	Pastes the macro in the clipboard into the list, and
	creates a new name for the macro.
ОК	Confirm all the edited Macros and click this button
	to save the new contents before leaving this dialog .
Cancel	Cancel the editing and leave Macro editing dialog.
Library	Open Macro Function Library managing dialog.





 Press the [New] button to create an empty macro and open the macro editor. Every macro has a unique number defined at [Macro ID], and must have a macro name, otherwise an error will appear while compiling.

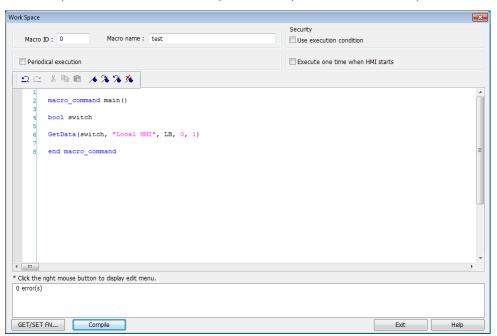
Work Space			×
	Security		
Macro ID: 0 Macro name: test	Use execution condition		
Periodical execution	Execute one time when HM	1I starts	
129 × 10 18 / % % %			
1 2 macro_command main() 3 4 end macro_command			<u> </u>
			=
			-
Click the right mouse button to display edit menu.			•
0 error(s)			
GET/SET FN Compile		Exit	Help

Design your macro. To use built-in functions (like SetData() or Getdata()), press [Get/Set FN...] button to open API dialog box and select the function and set essential parameters.

API	
	Build-in Library
Function name :	GetData 👻
	GetData(switch, "Local HMI", LB, 0, 1)
[Description] Read data from a	device.
[Usage] GetData(desti, PLO	E name, device type, address, data count)
[Example] char byData[10]	
•	4
Variable 1	
Variable type :	bool
Variable :	switch 👻
Read address	
PLC name :	Local HMI 🔹
Device type :	LB 🔹
Address :	0 System tag
	User-defined tag
Address format :	DDDDD [range : 0 ~ 12399]
	BIN    Data count : 1
	OK Cancel



4. After the completion of a new macro, press [Compile] button to compile the macro.



 If there is no error, press [Exit] button and a new macro "macro_test" will be in "Macro list".

Macro	
Macro list	
[ID:000] test	New
	Delete
	Edit
	Сору
	Paste
	ОК
	Cancel
	Library
*I : Execute one time when HMI starts	
*P : Periodical execution *S : Use execution condition	
Macro under development	
	Help
Password protect	
*Decompilation cannot recover MACROs when checks [Password prote	ct].
Address variables use [DDDDdd] address format to access [DDDDh] address format in Macro functions (i.e. SetData, GetData,)	partial-hexadecimal



## 18.8.2. Execute a Macro

There are several ways to execute a macro.

- Use a PLC Control object
- 1. Open [PLC Control] and add one PLC Control object with the [Type of control] as [Execute macro program].
- 2. Select the macro in [Macro name]. Choose a bit and select a trigger condition to trigger the macro. In order to guarantee that the macro will run only once, consider latching the trigger bit, and then resetting the trigger condition within the macro.
- 3. Use a [Set Bit] or Toggle Switch object to change the bit to activate the macro.
- Use a [Set Bit] or Toggle Switch object
- 1. On the [General] tab of the [Set Bit] or [Toggle Switch] dialog box, select the [Execute Macro] option.
- Select the macro to execute. The macro will be executed one time when the button is activated.



- If [Set Bit] uses [Periodic Toggle], the macro will be executed every time [Set Bit] toggles.
- Use a Function Key object
- **1.** On the [General] tab of the [Function Key] dialog, select the [Execute Macro] option.
- 2. Select the macro to execute. The macro will execute one time when the button is activated.
- In macro editor, use
- 1. [Periodical Execution]: Macro will be triggered periodically.
- 2. [Execute one time when HMI starts]: Macro will be executed once HMI starts.

## **18.9.** User Defined Macro Function

When editing Macro, to save time of defining functions, user may search for the needed from built-in Macro Function Library. However, certain functions, though frequently used, may not be found there. In this case, user may define the needed function and save it for future use. Next time when the same function is required, the saved functions can be called from [Macro Function Library] for easier editing. Additionally, [Macro Function Library] greatly enhances the portability of user-defined functions. Before building a function please check the built-in functions or online function library to see if it exists.



	F	unction Editor	
		122 / BB / 7 7 7 7	
		1 = sub int add(short x, short y)	
		2	
		3 int result 4 result = x + y	
		5 return result	
		6 7 end sub	
		8	
		* Click the right mouse button to display edit menu.	
		Edit description here :	
		GET/SET FN Comple Save Cancel Help	
Macro Fu	unction Library	y la	×
No.	Function Na	ume	<u>^</u>
1	short add	eft3 (char, short)	
2		y (short, short)	
3	int add ( sh		E
4	add2 (shor		
5	short multip		
6	add3 ( )		
7		tiply (short, int )	
8	int add_multiply (short, int ) unsigned int add4 (unsigned char, unsigned short, unsigned int )		
9		nort operation ( int )	
10		( unsigned int )	
11		, char, short, int, float, unsigned char, unsigned short, unsigned int )	
1	8003 ( 800)	III	•
			A
			~
			*
New	/ De	lete Edit	
Exp	ort Im	port	OK

#### 18.9.1. Import Function Library File

Open a project in HMI programming software, the default Function Library File will be read automatically and the function information will be loaded in. At this moment if a user-defined function is called, the relevant .mlb file must be imported first.

- 1. Default Function Library File Name: MacroLibrary (without filename extension)
- 2. Function Library Directory: HMI programming software installation directory\library (folder)
- \library (folder) contains two types of function library files:
   Without filename extension: MacroLibrary, the Default Function Library for HMI programming software to read at the beginning.

With filename extension (.mlb): Such as "math.mlb". The files to be read / written when users import / export. These files are portable and can be called from the folder when needed.



**4.** When opening HMI programming software, only the functions in Default Function Library will be loaded in, to use functions in .mlb files, please import them first.

Organize   New fold	der	
🔆 Favorites	Name	Date modified
🧮 Desktop	picture	2011/10/13 上午1.
🚺 Downloads	🍶 shape	2011/10/12 上午 0.
Recent Places	🌙 sound	2011/10/12 上午 0.
	length 0926.mlb	2008/7/16下午02:
🥱 Libraries	MacroLibrary	2007/8/5 上午 01:3
Documents	map1.flb	2007/8/5 上午 01:3
J Music	math.mlb	2007/8/5 上午 01:3
Pictures	menu01.flb	2007/8/5 上午 01:3

#### 18.9.2. How to Use Macro Function Library

**1.** Select the function directly from Macro Function Library.

Aacro Fu	unction Library	×
No.	Function Name	
1	short add_left3 ( char, short )	
2	short right_y (short, short)	
3	int add (short, short)	E
4	add2 ( short, int )	
5	short multiply ( )	
6	add3 ( )	
7	int add_multiply ( short, int )	
8	unsigned int add4 ( unsigned char, unsigned short, unsigned int )	
9	unsigned short operation ( int )	
10	int return_x ( unsigned int )	
11	add5 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	-
•	III	4
		*
		-
		Þ
New	/ Delete Edit	
INCH		
Exp	ort Import	ок

2. In WorkSpace click [GET/SET FN...] to open API dialog box.

Work Space	
Macro ID : 20 Macro name : macro_20	Security Use execution condition
Periodical execution	Execute one time when HMI starts
29 × B B / * * * *	
<pre>1 macro_command main() 3 4 5 end macro_command</pre>	E
< m	
* Click the right mouse button to display edit menu.	
GET/SET FN Compile	Exit Help



API

3. At least check one from [Library] or [Build-in] and select the function to be used.

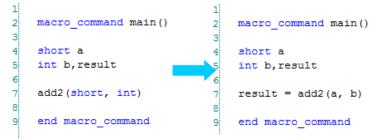
API			×	
	🔽 Build-in	Library		
Function name :	ACOS	•		
	ACOS(, )			
[Description] The result is equa	I to the arcosine of the so	urce.	<u>^</u>	
[Usage] ACOS(source, res	[Usage] ACOS(source, result)			
[Example] float source = 0.5	[Example] float source = 0.5, result			
•			Þ	
Variable 1				
Variable type :				
Variable :		Array index : 0	)	
Variable 2				
Variable type :				
Variable :	<b></b>	Array index : 0	)	

4. The description displayed in API dialog box is the same as written in Function Editor.

×

🖾 Build-in 🔍 Library	
Function name : add2 🗸	Function Editor
add2(short, int)	シロ X 階 電 🔺 🛪 🛪 🛪
parameter 1 : short  parameter 2 : int	
reutrn : none	3 int result
	S SetData (result, "Local HMI", LW, 60, 1) 6 E
	7 L end sub
(	
	•
	Click the right mouse button to display edit menu.
	Edit description here :
	parameter 1: short
	parameter 2 : int reutri : none
OK Cancel	GET/SET FN Compile Save Cancel Help

 Select the function to be used, fill in the corresponding variables according to the data type.



6. Upon completion of the steps above, user-defined functions can be used freely without defining the same functions repeatedly.



### **18.9.3.** Function Library Management Interface

**1.** Open macro management dialog, click [Library] to open [Macro Function Library] dialog box.

Macro				<b>—</b>
Macro list				
[ID:000] [ID:001]	sub - 1 sub - 2		-	New
[ID:002] [ID:003]	sub - 3 sub - 4			Delete
[ID:004] [ID:005]	sub - 5 sub - 6			Edit
[ID:006] [ID:007]	sub - 7 sub - 8			Сору
[ID:008] [ID:009] [ID:010]	sub - 9 sub - 10 sub - 11		Ξ	Paste
[ID : 011] [ID : 012]	sub - 12 sub - 13			ОК
[ID:013] [ID:014]	sub - 14 sub - 15			Cancel
[ID:015] [ID:016]	sub - 16 sub - 17			Library
[ID:017] [ID:018]	sub - 18 sub - 19		-	
*I : Execute on	e time when HMI starts			
*P : Periodical e	xecution *S	: Use execution conditio	n	
Macro under de	velopment			
				Help
Password pro	tect			
*Decompilation	cannot recover MACRO	when checks [Password	prote	ct].
		ess format to access [DD a. SetData, GetData,)	DDh] (	partial-hexadecimal

2. A list of functions is shown. When the project is opened, the software will load all the functions in the Macro Function Library.

/lacro Fu	nction Library	×
No.	Function Name	
1	short add_left3 ( char, short )	
2	short right_y ( short, short )	
3	int add (short, short)	=
4	add2 ( short, int )	
5	short multiply ( )	
6	add3 ( )	-
7	int add_multiply ( short, int )	-
8	unsigned int add4 ( unsigned char, unsigned short, unsigned int )	
9	unsigned short operation ( int )	
10	int return_x ( unsigned int )	
11	add5 ( bool, char, short, int, float, unsianed char, unsianed short, unsianed int )	Ψ.
<	III	
	ter 1 : short ter 2 : int none	*
New		
Expo	OK	

**3.** Each listed function has the following format:

return_type function_name ( parameter_type1, ..., parameter_typeN)



*return_type* indicates the type of the return value. If this value does not exist, this column will be omitted. function_name indicates the name of the function. "N" in *parameter_typeN* stands for the number of parameter types. If this function does not need any parameter, this column will be omitted.

```
1 □ sub int ADD(int a, int b)
2 | int ret
3 | ret = a+b
4 | return ret
5 | end sub
6
```

4. Macro function can be embedded in the project file. Select the function and then click [Copy To Project], then you can find this function in [Project] tab. When opening the project on another computer, this function can still be used. When compiling the project, the .exob file will included the functions that are used. Please note that decompiling the project will only produce the macro commands that are used.

	Macro Function Library	×
Project	Global Library	
No.	Function Name	^
1	short add_left3 ( char, short )	
2	short right y (short, short)	
3	int add (short, short)	
4	add2 (short, int )	
5	short multiply ( )	
6	add3()	
7	int add multiply (short, int )	
8	unsigned int add4 ( unsigned char, unsigned short, unsigned int )	
9	unsigned short operation ( int )	
10	int return_x ( unsigned int )	
11	add5 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
12	float add6 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
13	unsigned char return_255 ( )	
14	unsigned short return_65535 ( )	
15	unsigned int return_int ( )	~
<	>	
		$\sim$
<	>	
The subr	putine can be invoked by Conversion Tags or the Numeric object's Scaling feature under the following conditions:	
The subri	builte can be invoked by Conversion rags or the Numeric objects scaling reactire under the following condutors:	
	laration of a subroutine is: nc_name(type_b name), where type_b must be the same data format as the numeric data, for example, both data types are 16-bit	
onsigneu.		
2. NOT rea	ad/write the non-HMI local address.	
	I the following functions or commands : SYNC_TRIG_MACRO(), SYNC_TRIG_MACRO(), DELAY(), FindDataSamplingDate(), amplingIndex(), FindEventLogDate(), FindEventLogIndex(), INPORT(), INPORT2(), INPORT3(), OUTPORT(), PURGE(), for, while.	
New	. Delete Edit Export Import Copy To Project OK	

### 18.9.3.1. Create a Function

1. Click [New] to enter Function Editor.



oject	Global Library	
0.	Function Name	
	short add left3 (char, short )	
	shortright y (short, short)	
	int add (short, short)	
	add2(short,int)	
	short multiply ( )	
	add3 ( )	
	int add multiply (short, int)	
	unsigned int add4 ( unsigned char, unsigned short, unsigned int )	
	unsigned short operation ( int )	
)	int return x (unsigned int)	
L	add5 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
2	float add6 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
8	unsigned char return 255 ( )	
ł	unsigned short return_65535 ( )	
5	unsigned intreturn int ( )	
		Þ
		ŀ
e subr	outine can be invoked by Conversion Tags or the Numeric object's Scaling feature under the following conditions:	<u>۲</u>
he de e_a fu igned. IOT re	outine can be invoked by Conversion Tags or the Numeric object's Scaling feature under the following conditions: daration of a subroutine is: n.c.name(type. Jname), where type_b must be the same data format as the numeric data, for example, both data types are 16-bit	•

**2.** Edit function in Function Editor.

Function Editor				
으르 🎄 🖻 🔺	% % %			
Fu	unction Ec	liting Fiel	d	E
				-
<				•
* Click the right mouse button to disp Edit description here :	play edit menu.			
F	unction D	escriptior	ı Field	
1		2		

- 3. Edit the function description to describe what the specification is, how to use ... etc.
- **4.** After editing, click [Compile] and [Save] to save this function to the Library. Otherwise, a warning is shown.



5. Successfully add a function into Macro Function Library.



cro Fu	unction Library	-
No.	Function Name	
11	add5 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
12	float add6 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
3	unsigned char return_255 ( )	
4	unsigned short return_65535 ( )	
5	unsigned int return_int ( )	
6	short sub_case ( short )	
7	int sub_for (short)	
В	float return_float ( short )	-
)	char return_unsigned_char ( unsigned char )	
0	short return_short ( short , short )	
_		
-		+
		Þ
		4
New	w Delete Edit	+
New	v Delete Edit	Þ

# Note

- The total size of data type can be declared in a function is 4096 bytes.
- Function name must only contain alphanumeric characters, and cannot start with a number.

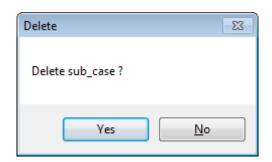
## 18.9.3.2. Delete a Function

**1.** In function list select the function to be deleted and click [Delete].

	Macro Function Library	×
Project	Global Library	
No.	Function Name	^
1	short add left3 (char, short)	
2	short right y (short, short)	-
3	int add ( short, short )	
4	add2 (short, int )	
5	short multiply ( )	
6	add3()	
7	int add multiply (short, int )	
8	unsigned int add4 ( unsigned char, unsigned short, unsigned int )	
9	unsigned short operation ( int )	
10	int return x (unsigned int)	- 11
11	add5 (bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
12	float add6 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
13	unsigned char return 255 ( )	
14	unsigned short return 65535 ( )	
15	unsigned int return int ( )	~
<	د د	>
<		~
*The subr	- outine can be invoked by Conversion Tags or the Numeric object's Scaling feature under the following conditions:	
1. The de	daration of a subroutine is: nc_name(type_b name), where type_b must be the same data format as the numeric data, for example, both data types are 16-bit	
2. NOT re	ad/write the non-HMI local address.	
	II the following functions or commands : SYNC_TRIG_MACRO(), SYNC_TRIG_MACRO(), DELAY(), FindDataSamplingDate(), amplingIndex(), FindEventLogDate(), FindEventLogIndex(), INPORT(), INPORT2(), INPORT3(), OUTPORT(), PURGE(), for, while.	
New.	. Delete Edit Export Import Copy To Project OK	

2. Click [Yes] to confirm, [No] to cancel the deletion. Click [Yes] to delete MAX_SHORT function.





## 18.9.3.3. Modify a Function

- 1. Users can modify the functions exist in the Library.
- 2. Select a function to modify by clicking [Edit] to enter Function Editor.

	Macro Function Library	×
Project	Global Library	
No.	Function Name	٦.
1	short add left3 ( char, short )	
2	short right_y (short, short)	
3	int add ( short, short )	
4	add2(short, int)	
5	short multiply ( )	
6	add3()	
7	int add_multiply (short, int )	
8	unsigned int add4 ( unsigned char, unsigned short, unsigned int )	
9	unsigned short operation ( int )	
10	intreturn x (unsigned int)	
11	add5 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
12	float add6 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
13	unsigned char return 255 ( )	
14	unsigned short return 65535 ( )	
15	unsigned int return int ( )	
<	>	
		5
		`
	~	
<	>	
*The subr	outine can be invoked by Conversion Tags or the Numeric object's Scaling feature under the following conditions:	-
The bab	outric carroc arrocke by contrelation rugs of the Hameric objects scaling restaric ander the following contaitons.	
	daration of a subroutine is: nc_name(type_b name), where type_b must be the same data format as the numeric data, for example, both data types are 16-bit	
2. NOT rea	ad/write the non-HMI local address.	
	ll the following functions or commands : SYNC_TRIG_MACRO(), SYNC_TRIG_MACRO(), DELAY(), FindDataSamplingDate(), amplingIndex(), FindEventLogDate(), FindEventLogIndex(), INPORT(), INPORT2(), INPORT3(), OUTPORT(), PURGE(), for, while.	
New	Delete Edit Export Import Copy To Project OK	

3. Double click the function to be modified can also enter Function Editor.





Function Editor	<b>X</b>
Subconcentration Sub short ADD (short a, short b) Sub short c cratb return c end sub Modify Function Name and Definition Here	E
< * * Click the right mouse button to display edit menu. Edit description here :	,
This is the function of A + B Modify Function Description	
1 Compile 2 Save	
GET/SET FN Compile Save Cancel	Help

4. After modifying, [Compile] then [Save] before leaving.

## 18.9.3.4. Import a Function

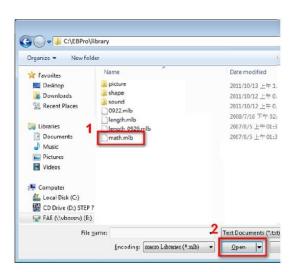
**1.** Functions can be imported using an external .mlb file.

	Macro Function Library	×
Project	Global Library	
No	Function Name	^
1	short add left3 (char, short)	
2	short right y (short, short)	-
3	int add (short, short)	
4	add2 (short, int )	
5	short multiply ( )	
6	add3()	
7	int add multiply (short, int )	
8	unsigned int add4 ( unsigned char, unsigned short, unsigned int )	
9	unsigned short operation ( int )	
10	int return x (unsigned int)	
11	add5 (bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
12	float add6 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )	
13	unsigned char return 255 ( )	
14	unsigned short return 65535 ( )	
15	unsigned int return int ( )	~
<		>
		~
< *The subr	outine can be invoked by Conversion Tags or the Numeric object's Scaling feature under the following conditions:	>
	daration of a subroutine is: nc_name(type_b name), where type_b must be the same data format as the numeric data, for example, both data types are 16-bit	
2. NOT re	ad/write the non-HMI local address.	
3. NOT ca FindDataS	II the following functions or commands : SYNC_TRIG_MACRO(), SYNC_TRIG_MACRO(), DELAY(), FindDataSamplingDate(), amplingIndex(), FindEventLogDate(), FindEventLogIndex(), INPORT(), INPORT2(), INPORT3(), OUTPORT(), PURGE(), for, while.	
New.	Delete Edit Export Import Copy To Project OK	

 For example, import a function library "math.mlb" which contains a function "test1". Click [Open].







3. When importing a function which already exists in the Library, a confirmation pop-up will be shown. The buttons are:

No.	Function Name	
1	int ADD (int, int )	
2	int SUBS (int, int )	
3	int MUL (int, int )	
4	int DIV (int, int )	
5	short test1 (short)	
this is a	macro abouts OK No Yes to all No to all	

[OK]: Overwrite the existing function with the imported one.

[NO]: Cancel the importing of the function with the same name.

[Yes to all]: Overwrite using all the imported functions with the same name.

[No to all]: Cancel the importing of all the functions with the same name.

**4.** The imported functions will be saved in Default Function Library, so if "math.mlb" file is deleted, "test1" will still exist in the Library, even restarting EasyBuilder Pro.

No.	Function Name	
1	int ADD (int, int )	
2	int SUBS ( int, int )	
3	int MUL (int, int )	
1	int DTV (int, int)	
5	short test1 (short )	
_		
	New Eurotion Successfully Added	
	New Function Successfully Added	
	-	
Net		,
Nev	w Delete Edit	÷
New	w Delete Edt	, ,
Nes	w Delete Edit	, OK



### 18.9.3.5. Export a Function

**1.** Export the function from Function Library and save as .mlb file. Click [Export].

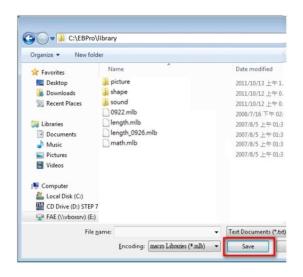
Macro Function Library			
Project	Global Library		
No.	Function Name	1	
	short add left3 (char, short)		
1			
2	short right_y (short, short)		
3	int add (short, short)		
4	add2 (short, int )		
5	short multiply ( )		
6	add3 ( )		
7	int add_multiply ( short, int )		
8	unsigned int add4 ( unsigned char, unsigned short, unsigned int )		
9	unsigned short operation (int)		
10	int return_x (unsigned int )	1	
11	add5 (bool, char, short, int, float, unsigned char, unsigned short, unsigned int )		
12	float add6 ( bool, char, short, int, float, unsigned char, unsigned short, unsigned int )		
13	unsigned char return_255 ( )		
14	unsigned short return_65535()		
15	unsigned int return_int ( )		
<	>		
	^	1	
1°			
	✓		
<	>		
STILL AND A	outine can be invoked by Conversion Tags or the Numeric object's Scaling feature under the following conditions:	1	
*The subro	outre can be invoked by Conversion rags or the numeric objects scaling reature under the following conditions:		
<ol> <li>The declaration of a subroutine is: type_a func_name(type_b name), where type_b must be the same data format as the numeric data, for example, both data types are 16-bit Unsigned.</li> </ol>			
2. NOT read/write the non-HMI local address.			
<ol> <li>NOT call the following functions or commands : SYNC_TRIG_MACRO(), SYNC_TRIG_MACRO(), DELAY(), FindDataSamplingDate(), FindDataSamplingIndex(), FindEventLogDate(), FindEventLogIndex(), INPORT(), INPORT2(), INPORT3(), OUTPORT(), PURGE(), for, while.</li> </ol>			
New	. Delete Edit Export Import Copy To Project OK		

2. Select the function to be exported, and click [Export].

Select fun	ctions to export		×
No.	Function Name	4	
1	short test1 (short)		
<b>V</b> 2	int ADD (int, int)		
<b>V</b> 3	int SUBS (int, int)		
<b>V</b> 4	int MUL (int, int )		
5	int DIV (int, int)		
6	length (short)		
			2
Select	All		4
			Export
			Cancel

- *3.* A "math.mlb" file can be found under export directory. This file contains 4 functions: ADD, SUBS, MUL, and DIV.
- **4.** The exported .mlb file can be imported on another PC. Open HMI programming software, import, then the functions in this file can be used.





## 18.10. Some Notes about Using the Macro

**1.** The maximum storage space of local variables in a macro is 4K bytes. So the maximum array size of different variable types are as follows:

char	a[4096]
bool	b[4096]
short	c[2048]
int	d[1024]
float	e[1024]

- 2. A maximum of 255 macros are allowed in an EasyBuilder Pro project.
- 3. A macro may cause the HMI unresponsive. Possible reasons are:
- A macro contains an infinite loop with no PLC communication.
- The size of an array exceeds the storage space in a macro.
- **4.** The PLC communication speed affects the running time for the macro to execute. Also, too many macros may slow down the communication between HMI and PLC.

# **18.11.** Use the Free Protocol to Control a Device

If EasyBuilder Pro does not provide a driver for a specific device, users can use OUTPORT and INPORT built-in functions to control the device. The data sent by OUTPORT and INPORT must follow the communication protocol of the device. The following example explains how to use these two functions to control a MODBUS RTU device.

1. First, create a new device in the device table. The device type of the new device is set to "Free Protocol" and named with "MODBUS RTU device" as follows:



Device Properties		System Parameter Settings
Name : Location : PLC type :	Modbus RTU Device       HMI     PLC       Local     Settings       Free Protocol     >       V.1.00, FREE_PROTOCOL.e30     >	Extended Memory         Printer/Backup Server         e-Mail         Recipes           Device         Model         General         System Setting         Security         Font           Device list :         No.         Name         Location         Device type         Interface         L           Local HMI         Local         eMIT3105 (800
PLC I/F : COM :	RS-232 • COM1 (9600,E,8,1) Settings	<pre>     ""     New Delete Settings Project description :     ^ </pre>
	OK Cancel	< , , , , , , , , , , , , , , , , , , ,

 The interface of the device (PLC I/F) uses [RS-232]. If a MODBUS TCP/IP device is connected, the interface should be [Ethernet] with correct IP and port number as follows:

Device Properties	
Name :	Modbus RTU Device
	○ HMI
Location :	Local   Settings
PLC type :	Free Protocol
	Ethernet
IP :	192.168.1.100, Port=502
	Use UDP (User Datagram Protocol )

Suppose that the HMI will read the data of  $4x_1$  and  $4x_2$  on the device. First, utilize OUTPORT to send out a read request to the device. The format of OUTPORT is:

OUTPORT(command[start], device_name, cmd_count)

Since "MODBUS RTU device" is a MODBUS RTU device, the read request must follow MODBUS RTU protocol. The request uses" Reading Holding Registers (0x03)" command to read data. The following picture displays the content of the command. (The items of the station number (byte 0) and the last two bytes (CRC) are ignored).



Request				
	Function code	1 Byte	0x03	
	Starting Address	2 Bytes	0x0000 to 0xFFFF	
	Quantity of Registers	2 Bytes	1 to 125 (0x7D)	
Response				
	Function code	1 Byte	0x03	
	Byte count	1 Byte	2 x N*	
	Register value	N* x 2 Bytes		
*N = Quantity of Registers				
Error				
	Error code	1 Byte	0x83	
	Exception code	1 Byte	01 or 02 or 03 or 04	

Depending on the protocol, the content of a read command as follows (The total is 8 bytes):

command[0]: station number	(BYTE 0)
command[1]: function code	(BYTE 1)
command[2]: high byte of starting address	(BYTE 2)
command[3]: low byte of starting address	(BYTE 3)
command[4]: high byte of quantity of registers	(BYTE 4)
command[5]: low byte of quantity of registers	(BYTE 5)
command[6]: low byte of 16-bit CRC	(BYTE 6)
command[7]: high byte of 16-bit CRC	(BYTE 7)
So a read request is designed as follows:	

```
So a read request is designed as follows:

char command[32]

short address, checksum

FILL(command[0], 0, 32) // initialize command[0]~command[31] to 0

command[0] = 0x1 // station number

command[1] = 0x3 // read holding registers (function code is 0x3)

address = // starting address (4x_1) is 0

HIBYTE(address, command[2])

LOBYTE(address, command[2])

read_no = 2 // the total words of rading is 2 words

HIBYTE(read_no, command[4])

LOBYTE(read_no, command[5])

CRC(command[0], checksum, 6) // calculate 16-bit CRC

LOBYTE(checksum, command[6])
```

HIBYTE(checksum, command[7])

Lastly, use OUPORT to send out this read request to PLC.

OUTPORT(command[0], "MODBUS RTU Device", 8) // send read request

After sending out the request, use INPORT to get the response from PLC. Depending on the protocol, the content of the response is as follows (the total byte is 9):

18-103



command[0]: station number	(BYTE 0)
command[1]: function code	(BYTE 1)
command[2]: byte count	(BYTE 2)
command[3]: high byte of 4x_1	(BYTE 3)
command[4]: low byte of 4x_1	(BYTE 4)
command[5]: high byte of 4x_2	(BYTE 5)
command[6]: high byte of 4x_2	(BYTE 6)
command[7]: low byte of 16-bit CRC	(BYTE 7)
command[8]: high byte of 16-bit CRC	(BYTE 8)
The format of INPORT is:	

```
INPORT(response[0], "MODBUS RTU Device", 9, return_value) // read reponse
```

Where the real read count is restored to the variable return_value (unit is byte). If return_value is 0, it means reading fails in executing INPORT.

According to the MODBUS RTU protocol specification, the correct response[1] must be equal to 0x03. After getting correct response, calculate the data of 4x_1 and 4x_2 and put in the data into LW-100 and LW-101 of HMI.

```
If (return_value) >0 and response[1] == 0x3) then
  read_data[0] = response[4] + (response[3] << 8) // 4x_1
  read_data[1] = response[6] + (response[5] << 8) // 4x_2
  SetData(read_data[0], "Local HMI", LW, 100, 2)
endif</pre>
```

The complete macro is as follows:



```
// Read Holding Registers
macro command main()
  char command[32], response[32]
  short address, checksum
  short read no, return value, read data[2], i
  FILL(command[0], 0, 32)// initialize command[0]~command[31] to 0
  FILL(response[0], 0, 32)
  command[0] = 0x1// station number
  command[1] = 0x3// read holding registers (function code is 0x3)
  address = 0
  address = 0// starting address (4x_1) is 0
  HIBYTE(address, command[2])
  LOBYTE(address, command[3])
  read_no = 2/ the total words of reading is 2 words
  HIBYTE(read no, command[4])
  LOBYTE(read_no, command[5])
  CRC(command[0], checksum, 6)// calculate 16-bit CRC
  LOBYTE(checksum, command[6])
  HIBYTE(checksum, command[7])
  OUTPORT(command[0], "MODBUS RTU Device", 8 )// send request
  INPORT(response[0], "MODBUS RTU Device", 9, return_value)// read response
  if (return value > 0 and response[1] == 0x3) then
    read data[0] = response[4] + (response[3] \langle 8 \rangle// 4x 1
    read_data[1] = response[6] + (response[5] << 8)// 4x_2</pre>
    SetData(read data[0], "Local HMI", LW, 100, 2)
  end if
  end macro command
```

The following example explains how to design a request to set the status of 0x_1. The request uses "Write Single Coil(0x5)" command.



Request					
•	Function code	1 Byte	0x05		
	Output Address	2 Bytes	0x0000 to 0xFFFF		
	Output Value	2 Bytes	0x0000 or 0xFF00		
Respo	Response				
	Function code	1 Byte	0x05		
	Output Address	2 Bytes	0x0000 to 0xFFFF		
	Output Value	2 Bytes	0x0000 or 0xFF00		
Error					
	Error code	1 Byte	0x85		
	Exception code	1 Byte	01 or 02 or 03 or 04		

The complete macro is as follows:

```
// Write Single Coil (ON)
macro_command main()
char command[32], response[32]
short address, checksum
short i, return value
FILL(command[0], 0, 32)// initialize command[0]~ command[31] to 0
FILL(response[0], 0, 32)
command[0] = 0x1// station number
command[1] = 0x5// function code : write single coil
address = 0
HIBYTE(address, command[2])
LOBYTE(address, command[3])
command[4] = 0xff// force 0x_1 on
command[5] = 0
CRC(command[0], checksum, 6)
LOBYTE(checksum, command[6])
HIBYTE(checksum, command[7])
OUTPORT(command[0], "MODBUS RTU Device", 8)// send request
INPORT(response[0], "MODBUS RTU Device", 8, return_value)// read response
```

## end macro_command

## 18.12. Compiler Error Message

Error Message Format
 error C# : error description
 (# is the error message number)



Example: error C37 : undeclared identifier : i When there are compile errors, the description of the error can be found by the compiler error message number.

Error Description
 (C1) syntax error : 'identifier'
 There are many possibilities to cause compiler error.

For example: macro_command main() char i, 123xyz // this is an unsupported variable name end macro_command

(C2) 'identifier' used without having been initializedMacro must define the size of an array during declaration.

For example: macro_command main() char i int g[i] // i must be a numeric constant end macro_command

(C3) redefinition error : 'identifier'The name of variable and function within its scope must be unique.

For example: macro_command main() int g[10] , g // error end macro_command

(C4) function name error : 'identifier'Reserved keywords and constant cannot be the name of a function

For example : sub int if() // error

(C5) parentheses have not come in pairs Statement missing "(" or ")"



EasyBuilder Pro V5.02.01

#### Macro Reference

For example : macro_command main ) // missing "("

(C6) illegal expression without matching 'if' Missing expression in "if" statement

(C7) illegal expression (no 'then') without matching 'if' Missing "then" in "if" statement

(C8) illegal expression (no 'end if')Missing "end if"

(C9) illegal 'end if' without matching 'if' Unfinished "If' statement before "End If"

(C10) illegal 'else'The format of "if" statement is :if [logic expression] then[ else [if [logic expression] then ] ]

end if

Any format other than this format will cause a compile error.

(C17) illegal expression (no 'for') without matching 'next' "for" statement error : missing "for" before "next"

(C18) illegal variable type (not integer or char) Should be integer or char variable

(C19) variable type error Missing assign statement

(C20) must be keyword 'to' or 'down' Missing keyword "to" or "down"

(C21) illegal expression (no 'next') The format of "for" statement is:



for [variable] = [initial value] to [end value] [step]

next [variable]

Any format other than this format will cause a compile error.

(C22) 'wend' statement contains no 'while'"While" statement error : missing "while" before "Wend"

(C23) illegal expression without matching 'wend' The format of "While" statement is :

while [logic expression]

wend

Any format other than this format will cause a compile error.

(C24) syntax error : 'break'"break" statement can only be used in "for", "while" statement.

(C25) syntax error : 'continue' "continue" statement can only be used in "for" statement, or "while" statement.

(C26) syntax error Error in expression.

(C27) syntax error The mismatch of an operation object in expression can cause a compile error.

For example : macro_command main() int a, b for a = 0 to 2 b = 4 + xyz // illegal : xyz is undefined next a end macro command



(C28) must be 'macro_command' There must be 'macro_command'

(C29) must be key word 'sub' The format of function declaration is:

```
sub [data type] function_name(...)
```

.....

end sub

For example::
sub int pow(int exp)
end sub

format other than this format will cause a compile error.

(C30) number of parameters is incorrect Mismatch of the number of parameters

(C31) parameter type is incorrect

Mismatch of data type of parameter. When a function is called, the data type and the number of parameters should match the declaration of function, otherwise it will cause a compile error.

(C32) variable is incorrect The parameters of a function must be equivalent to the arguments passing to a function to avoid compile error.

(C33) function name : undeclared function

(C34) expected constant expression Illegal array index format.

(C35) invalid array declaration

(C36) array index error

(C37) undeclared identifier : i 'identifier'



Any variable or function should be declared before use.

(C38) un-supported PLC data address The parameter of GetData( ... ) , SetData( ... ) should be legal PLC address. If the address is illegal, this error message will be shown.

(C39) 'idenifier' must be integer, char or constantThe format of array is:Declaration: array_name[constant] (constant is the size of the array)Usage: array_name[integer, character or constant]Any format other than this format will cause a compile error.

(C40) execution syntax should not exist before variable declaration or constant definition

```
For example :
macro_command main( )
int a, b
for a = 0 To 2
    b = 4 + a
int h , k // illegal – definitions must occur before any statements or expressions
    // for example, b = 4 + a
next a
end macro_command
```

(C41) float variables cannot be contained in shift calculation

(C42) function must return a value

(C43) function should not return a value

(C44) float variables cannot be contained in calculation

```
(C45) PLC address error
```

- (C46) array size overflow (max. 4k)
- (C47) macro command entry function is not only one



EasyBuilder Pro V5.02.01

(C49) an extended addressee's station number must be between 0 and 255

For example : SetData(bits[0], "PLC 1", LB, 300#123, 100) // illegal : 300#123 means the station number is 300, but the maximum is 255

(C50) an invalid PLC name PLC name is not defined in the device list of system parameters.

(C51) macro command do not control a remote device A macro can only control a local machine.

For example : SetData(bits[0], "PLC 1", LB, 300#123, 100) "PLC 1" is connected with the remote HMI, so it cannot work.

# 18.13. Sample Macro Code

"for" statement and other expressions (arithmetic, bitwise shift, logic and comparison)
 macro_command main()
 int a[10], b[10], i

```
b[0] = (400 + 400 << 2) / 401

b[1] = 22 *2 - 30 % 7

b[2] = 111 >> 2

b[3] = 403 > 9 + 3 >= 9 + 3 < 4 + 3 <= 8 + 8 == 8

b[4] = not 8 + 1 and 2 + 1 or 0 + 1 xor 2

b[5] = 405 and 3 and not 0

b[6] = 8 & 4 + 4 & 4 + 8 | 4 + 8 ^ 4

b[7] = 6 - (\sim 4)

b[8] = 0x11

b[9] = 409
```



```
for i = 0 to 4 step 1
  if (a[0] == 400) then
       GetData(a[0],"Device 1", 4x, 0,9)
       GetData(b[0],"Device 1", 4x, 11,10)
  end If
  next i
  end macro_command
     "while", "if" and "break" statements
macro_command main()
  int b[10], i
  i = 5
  while i == 5 - 20 % 3
       GetData(b[1], "Device 1", 4x, 11, 1)
      if b[1] == 100 then
            break
      end if
  wend
  end macro_command
    Global variables and function call
char g
  sub int fun(int j, int k)
      int y
       SetData(j, "Local HMI", LB, 14, 1)
       GetData(y, "Local HMI", LB, 15, 1)
      g = y
      return y
  end Sub
  macro_command main()
      int a, b, i
  a = 2
  b = 3
```

EasyBuilder Pro V5.02.01

```
i = fun(a, b)
     SetData(i, "Local HMI", LB, 16, 1)
end macro_command
   "if" statement
macro_command main()
     int k[10], j
     for j = 0 to 10
          k[j] = j
     next j
     if k[0] == 0 then
     SetData(k[1], "Device 1", 4x, 0, 1)
     end if
if k[0] == 0 then
          SetData(k[1], "Device 1", 4x, 0, 1)
     else
     SetData(k[2], "Device 1", 4x, 0, 1)
end if
     if k[0] == 0 then
          SetData(k[1], "Device 1", 4x, 1, 1)
     else if k[2] == 1 then
     SetData(k[3], "Device 1", 4x, 2, 1)
end If
     if k[0] == 0 then
     SetData(k[1], "Device 1", 4x, 3, 1)
else if k[2] == 2 then
     SetData(k[3], "Device 1", 4x, 4, 1)
else
     SetData(k[4], "Device 1", 4x, 5, 1)
end If
end macro_command
```



"while" and "wend" statements

EasyBuilder Pro V5.02.01

wend

```
macro_command main()
char i = 0
int a[13], b[14], c = 4848
b[0] = 13
while b[0]
          a[i] = 20 + i * 10
    if a[i] == 120 then
         c =200
              break
    end if
    i = i + 1
SetData(c, "Device 1", 4x, 2, 1)
end macro_command
   "break" and "continue" statements
macro_command main()
char i = 0
int a[13], b[14], c = 4848
b[0] = 13
while b[0]
         a[i] = 20 + i * 10
```

```
if a[i] == 120 then
c =200
i = i + 1
     continue
end if
```

i=i+1



```
if c == 200 then
        SetData(c, "Device 1", 4x, 2, 1)
        break
        end if
wend
end macro_command
Array
macro_command main()
int a[25], b[25], i
b[0] = 13
for i = 0 to b[0] step 1
        a[i] = 20 + i * 10
```

next i

SetData(a[0], "Device 1", 4x, 0, 13) end macro_command



## 18.14. Macro TRACE Function

TRACE function can be used with EasyDiagnoser to show the current content of the variables. The following example illustrates how TRACE function could be used in macro.

First of all, add a new macro "macro_0" in the project, and in "macro_0" add TRACE ("LW = %d", a). "%d" indicates display current value of LW in decimal format. The content of "macro_0" is as follows:

```
2
    macro_command main()
 3
 4
    short a
 5
    GetData(a, "Local HMI", LW, 0, 1)
 6
    a=a+1
    SetData(a, "Local HMI", LW, 0, 1)
 7
 8
    TRACE ("LWO = d" , a)
 9
10
    end macro command
```

 Secondly, add a Numeric Display object and a Function Key object in window no. 10 of the project. The Function Key object is used to execute macro_0.

10 - WINDOW_010 X				
Object Display				
LW-0 [■] ####				
Function Key	ro_0			

- 3. Lastly, compile the project and execute [Off-line simulation] or [On-line simulation].
- When processing simulation on PC, right click and select "Run EasyDiagnoser" in the pop-up menu.

Object Display		
LW-0		Exit simulation
		Run EasyDiagnoser
Function Key	macro_0	Screenshot



5. Afterwards, EasyDiagnoser will be started. [Logger] window displays whether EasyDiagnoser is able to connect with the HMI to be watched or not. [Output] window displays the output of the TRACE function. The illustration below shows that EasyDiagnoser succeeds in connecting with HMI.

Co <u>m</u> mand: F	lead + Wi	rite	•	Device: Al	1		▼ <u>Sta</u>	tion: 0	
ddress Type:	All			<u>R</u> ange:	0	~ 99999		<u><u> </u></u>	apture
No	Cmd.	PID	Device		St.	Index	Address / Length	Time	Error
ager									n
i <mark>gger</mark> 12:59:11] Loc 12:59:11] Col	king for t	the target establishe	HMI ed with the ta	arget HMI.				_	ņ

When EasyDiagnoser is not able to connect with HMI, [Logger] window displays content as shown in the following figure:

Logger	
[03:01:08] Looking for the target HMI	
📄 Logger 🔚 Devices 🧭 Output 😻 Polling Packages	
Logger more la output a Polling Packages	

6. The possible reason of not being able to get connection with HMI can be failure in executing simulation on PC. Another reason is that the Port No. used in project for simulation on PC is incorrect (or occupied by system). Please change Port No. as shown, compile project then do simulation again.

Extended Memory		Printer/Backup Server		e-Mail	Recipes
Device Model		General	System Setting	Security	Font
ымт	model : Lawran	of (000 (00)			
HMI	model : eMT310	05 (800 x 600)			•
	model : eMT310	05 (800 x 600) 👻			•
HMI stat		05 (800 x 600) ▼			•

7. In EasyDiagnoser, the Port No. should be set the same as the Port No. in the project.



Select HMI			×
1 IP Name			Þ
<u>H</u> MI Name:	eMT3105 <u>Search</u> Sgarch All	192.168.1.118 (Default HMI) 192.168.1.12 (Default HMI) 192.168.1.131 (eMT3105) 192.168.1.136 (Default HMI) 192.168.1.162 (Default HMI) 192.168.1.221 (mt8 104/h_susan) 192.168.1.236 (Default HMI)	
Project Port:	8005 💌	ОК	Exit

The three consecutive ports of the project port no. are preserved for HMI communication. In the setting above as an example, Port No. is set as 8005. Port 8005, 8006 and 8007 should be reserved. In this case when executing simulation on PC, please make sure that these ports are not occupied by other programs.

#### TRACE Syntax List

Name	TRACE				
Syntax	TRACE(format, argument)				
Description	Use this function to send specified string to the EasyDiagnoser. Users can print out the current value of variables during run-time of macro for debugging. When TRACE encounters the first format specification (if any), it converts the value of the first argument after format and outputs it accordingly. format refers to the format control of output string. A format specification, which consists of optional (in [ ]) and required fields (in bold), has the following form: %[flags] [width] [.precision] type Each field of the format specification is described as below: flags (optional): - +				
	width (optional): A nonnegative decimal integer controlling the minimum number of characters printed. precision (optional): A nonnegative decimal integer which specifies the precision and the				
	<pre>number of characters to be printed. type: C or c : specifies a single-byte character. d : signed decimal integer. i : signed decimal integer. o : unsigned octal integer. u : unsigned decimal integer. X or x : unsigned hexadecimal integer. E or e : Signed value having the form. [ - ]d.dddd e [sign]ddd where d is a single decimal digit, dddd is one or more decimal digits, ddd i exactly three decimal digits, and sign is + or f : Signed value having the form [ - ]dddd.dddd, where dddd is one or more decimal digits.</pre>				





	The argument part is optional.
Example	macro_command main() char c1 = ' a' short s1 = 32767 float f1 = 1.234567 TRACE( "The results are" ) // output: The results are TRACE( "c1 = %c, s1 = %d, f1 = %f", c1, s1, f1)
	// output: c1 = a, s1 = 32767, f1 = 1.234567 end macro_command

- 8. Use LB-9059 to disable MACRO TRACE function (when ON). When set ON, the output message of TRACE won't be sent to EasyDiagnoser.
- 9. Users can directly execute EasyDiagnoser.exe from Utility Manager. In Utility Manager, current HMI on line will be listed; users can simply select the HMI to be watched. Please note that Project Port should be the same as Port No. used in project file.

🦪 Utility Manager	- • •					
- HMI IP, Password						
Type : eMT3000 Seri	ies 💌					
Settings	Reboot HMI					
Connection						
Ethernet     OUSE	3 cable					
HMI IP :	T					
Data/Event Log	File Information					
Utility						
EasyBui	ilder Pro					
EasyConverter	EasyAddressViewer					
EasyPrinter	EasyDiagnoser					
Recipe/Extende	d Memory Editor					
Build Download Dat	a for SD/USB Disk					
		Select H	IMI			<b>×</b>
Download	Upload	4	IP Name			4
On-line Simulation	Off-line Simulation		HMI Name:	eMT3105 🔹	192.168.1.118 (Default HMI) 192.168.1.12 (Default HMI)	E
Pass-th	rough			Search	192.168.1.131 (eMT3105)	E
	-			S <u>e</u> arch All	192.168.1.136 (Default HMI) 192.168.1.162 (Default HMI)	
					192.168.1.221 (mt8104ih_susan) 192.168.1.236 (Default HMI)	-
Help	Exit		Project Port:	8005 💌	ОК	Exit

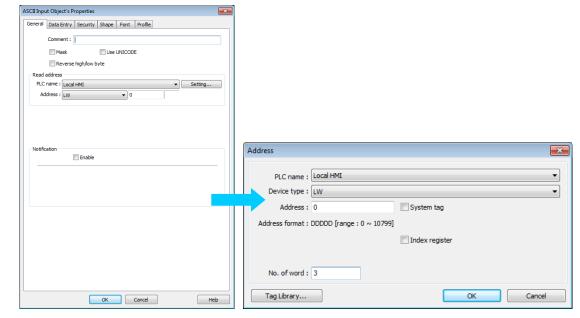
- 10. Download the project to HMI and start the project. If EasyDiagnoser is unable to get connection with the HMI to be watched, it is possible that HMI power is not ON, or Port No. is incorrect. This may cause EasyDiagnoser to connect then disconnect with HMI continuously. Please check the Port No. in EasyDiagnoser settings.
- **11.** When EasyDiagnoser succeeds in connecting with HMI, simply execute macro_0, [Output] window will then display the output of the TRACE function.



Object Display LW-0 5 Function Key macro_0	Weintek HMI Diagnostic T         File       View         Output       #         (ID 0, Ln 10)       LW0 = 2         (ID 0, Ln 10)       LW0 = 3         (ID 0, Ln 10)       LW0 = 4         (ID 0, Ln 10)       LW0 = 5
<pre>1 2 macro_command main() 3 4 short a 5 6 GetData(a, "Local HMI", LW, 0, 1) 7 a=a+1 8 SetData(a, "Local HMI", LW, 0, 1) 9 10 TRACE("LWO = %d" , a) 11 12 end macro_command</pre>	▲ ► ► Logger MDevices 2 Output S Pollin

#### 18.15. Example of String Operation Functions

String operation functions are added to macro to provide a convenient way to operate strings. The term "string" means a sequence of ASCII characters, and each of them occupies 1 byte. The sequence of characters can be stored into 16-bit registers with least significant byte first. For example, create an ASCII Input object and setup as follows:



Run simulation and input "abcdef":





The string "abcdef" is stored in LW-0~LW-2 as follows (LB represents low byte and HB represents high byte):

	HB	LB
LW0 LW1 LW2	'B' 'D' 'F'	'A' 'C' 'E'
LW3 LW4 LW5		

The ASCII Input object reads 1 word (2 bytes) at a time as described in the previous chapter. Suppose an ASCII Input object is set to read 3 words as shown in the above example, it can actually read at most 6 ASCII characters since that one ASCII character occupies 1 byte. The functionality of each string operation function is described in the following table:

Function name	Description
StringGet	Read string data from a device.
StringGetEx	Read string data from a device and continue executing next
	command even if no response from that device.
StringSet	Write string data to a device.
StringSetEx	Write string data to a device and continue executing next
StingSetEx	command even if no response from that device.
StringCopy	Copy one string to another.
StringMid	Retrieve a substring.
StringDecAsc2Bin	Convert a decimal string to an integer.
StringBin2DecAsc	Convert an integer to a decimal string.
StringDecAsc2Float	Convert a decimal string to floats.
StringFloat2DecAsc	Convert a float to a decimal string.
StringHexAsc2Bin	Convert a hexadecimal string to binary data.
StringBin2HexAsc	Convert binary data into a hexadecimal string.
StringLength	Obtain the length of a string.
StringCat	Append source string to destination string.
StringCompare	Do a case-sensitive comparison of two strings.
StringCompareNoCase	Do a case-insensitive comparison of two strings.
StringFind	Find a substring inside a larger string.
StringReverseFind	Find a substring inside a larger string; starts from the end.
StringFindOneOf	Find the first matching character from a set.
StringIncluding	Extracts a substring that contains only the characters in a set.



StringExcluding	Extracts a substring that contains only the characters not in a
StringExcluding	set.
StringToUpper	Convert the characters of a string to uppercase.
StringToLower	Convert the characters of a string to lowercase.
StringToReverse Reverse the characters of a string.	
Chuin a Tuine Laft	Trim the leading specified characters in a set from the source
StringTrimLeft	string.
StringTrimPight	Trim the trailing specified characters in a set from the source
StringTrimRight	string.
StringInsert	Insert a string in a specific location within another string.

For more detailed information of the above string operation functions, please check out the "Built-In Function Block" section. In order to demonstrate the powerful usage of string operation functions, the following examples will show you step by step how to create executable project files using the new functions; starts from creating a macro, ends in executing simulation.

**1.** To read (or write) a string from a device:

Create a new macro:

Macro	
Macro list	
	New

Edit the content:

1	
2	macro_command main()
3	
4	char str[20]
5	
6	<pre>StringGet(str[0], "Local HMI", LW, 0, 20)</pre>
7	<pre>StringSet(str[0], "Local HMI", LW, 50, 20)</pre>
8	
9	end macro_command
-	Cird maero_command

The first function "StringGet" is used to read a string from LW-0~LW-19, and store it into the str array. The second function "StringSet" is used to output the content of str array. Add one ASCII Input object and one II Function Key object in window 10 of the project. The settings of these objects are shown as below. Function Key object is used to execute macro_0.





	ASCII	Input	object
--	-------	-------	--------

Address				<b>—</b> ×
PLC name :	Local HMI			
Device type :	LW			•
Address :	0	System tag	9	
Address format :	DDDDD [range : 0 ~ 1	10799]		
		🕅 Index regi	ster	
No. of word :	10			
Tag Library			ОК	Cancel

**I** Function Key object:

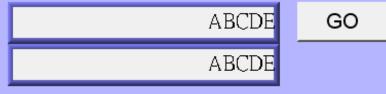
	Local HMI		
Device type :	LW	· · · · · · · · · · · · · · · · · · ·	
Address :	50	System tag	
Address format :	DDDDD [range : 0 ~ 10799]	Index register	
No. of word :	10		
Tag Library		OK Cancel	

Lastly, use  $\Re$  [Compile] to compile the project and execute  $\cancel{2}$  [Off-line simulation] or  $\cancel{2}$  [On-line simulation]. Follow the steps below to operate the executing project:

Step 1. Input string.

Step 2.	Press	"GO"	button.
Step Z.	LIC22	00	bullon.

Test	t 1:		
		ABCDE	GO
Step 3.	Output string.		
Test	t 1:		
			~~~



2. Initialization of a string.

Create a new macro and edit the content:



```
1
2 macro_command main()
3
4 char str1[20]="abcde"
5 char str2[20]={'a','b','c','d','e'}
6
7 StringSet(str1[0], "Local HMI", LW, 0, 20)
8 StringSet(str2[0], "Local HMI", LW, 50, 20)
9
10 end macro_command
```

The data enclosed in double quotation mark ("") is viewed as a string. str1 is initialized as a string while str2 is initialized as a char array. The following snapshot of simulation shows the difference between str1 and str2 using two ASCII Input objects.



Macro compiler will add a terminating null character ($\langle 0' \rangle$) at the end of a string. The function "StringSet" will send each character of str1 to registers until a null character is reached. The extra characters following the null character will be ignored even if the data count is set to a larger value than the length of string.

On the contrary, macro compiler will not add a terminating null character ('0') at the end of a char array. The actual number of characters of str2 being sent to registers depends on the value of data count that is passed to the "StringSet" function.

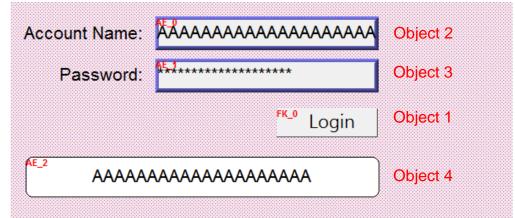
3. A simple login page.

Create a new macro and edit the content, for example, Macro [ID:001] macro_1.



```
1
    macro_command main()
2
3
    char name[20]="admin"
 4
    char password[20]="123456"
 5
    char name_input[20], password_input[20]
 6
 7
    char message_success[40]="Success! Access Accepted."
    char message_fail[40]="Fail! Access Denied."
8
 9
    char message_clear[40]
    bool name_match=false, password_match=false
10
11
    StringGet(name_input[0], "Local HMI", LW, 0, 20)
12
    StringGet(password_input[0], "Local HMI", LW, 50, 20)
13
14
    name_match = StringCompare(name_input[0], name[0])
15
   password_match = StringCompare(password_input[0], password[0])
16
17
    FILL(message_clear[0], 0x20, 40) //FILL with white space
18
    StringSet(message_clear[0], "Local HMI", LW, 100, 40)
19
20
21 🗇 if (name match==true and password match==true) then
        StringSet(message success[0], "Local HMI", LW, 100, 40)
22
    else
23
24
        StringSet(message fail[0], "Local HMI", LW, 100, 40)
25
   L end if
26
    end macro command
27
```

The first two "StringGet" functions will read the strings input by users and store them into arrays named name_input and password_input separately. Use the function "StringCompare" to check if the input account name and password are matched. If the account name is matched, name_match is set true; if the password is matched, password_match is set true. If both name_match and password_match are true, output the string "Success! Access Accepted.". Otherwise, output the string "Fail! Access Denied.". Add ASCII Input and Function Key I objects in window 10 of the project. The settings of these objects are shown as below. Function Key object is used to execute macro_1.



Object 1: Function Key **1**

Select [Execute macro] and Macro: [ID:000] macro_1.



Object 2: ASCII Inpu	it 🛄
	Address
	PLC name : Local HMI
	Device type : LW
	Address : 0 System tag
	Address format : DDDDD [range : $0 \sim 10799$]
	Index register
	No. of word: 10
	Tag Library OK Cancel
Object 3: ASCII Inpu	it internet in the second seco
	ASCII Input Object's Properties
	General Data Entry Security Shape Font Profile
	Comment :
	Mask USE UNICODE
	Reverse high/low byte
	Read address
	PLC name : Local HMI Setting
	Address : LW v 50
	Address
	PLC name : Local HMI
	Device type : LW
	Address : 50 System tag
	Address format : DDDDD [range : 0 ~ 10799]
	Index register
	No. of word : 10
	Tag Library OK Cancel
Object 4: ASCII Disp	lay 🛄
Add	iress 💽
	PLC name : Local HMI
	Device type : LW
	Address format : DDDDD [range : 0 ~ 10799]
	Index register
	No. of word : 20
	Tag Library OK Cancel

Lastly, use 🛠 [Compile] to compile the project and execute 星 [Off-line simulation] or 🖳 [On-line simulation]. Follow the steps below to operate the executing project:



	• 💌
Account Name: admin	
Account Name.	
Password:	
Login	
! @ # \$ % ^ & * () BS	
~qwertyuiop{}	
Capszxcvbnm $<$ $>$?	
Clear SPACE + =	
Fast Sel	

Step 2. Enter password and press [Login] button.

Account Name: admin	
Password: ******	
1 assword.	
Login	
Login	
MAX: 0 MIN: 0	
111111	
7 8 9 Cir Esc	
4 5 6 BS Del	
. 0 - Enter	
Fast Sel	

Step 3. Login succeeded or failed.

Account Name:	admin		Account Name:	admin	
Password:	*****		Password:	*****	
	L	ogin			Login
Succes	s! Access Acce		Fail! Ac	ccess Denied.	





18.16. Macro Password Protection

Password protect	Set password
*Decompilation cannot recover MACROs w	hen checks [Password protect].

On MACRO editing window there's the [Password protect] selection, tick it and click [Set password...] to set a password less than or equals to 10 characters (support ASCII character only, ex. "a\$#*hFds").

After setting MACRO password, users will have to input correct password when opening MACRO editing window. EasyBuilder Pro should be rebooted for typing the password again after 3 incorrect attempts.

EasyBuilde	er Pro	×
<u> </u>	Password error three times ! Please restart this application !	
	ОК	

Note

When MACRO is password protected, de-compilation of EXOB file will not be able to restore MACRO contents.



19. Configure HMI as a MODBUS Server

This chapter explains how to configure HMI as a MODBUS Server.

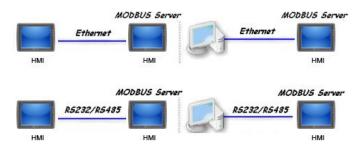
Overview	. 19-2
Steps to Create a MODBUS Server	19-2
Steps to Access a MODBUS Server	19-4
Changing MODBUS Server Station Number Online	19-6
MODBUS Address Type	19-6
	Steps to Create a MODBUS Server Steps to Access a MODBUS Server Changing MODBUS Server Station Number Online



19.1. Overview

Once the HMI is configured as a MODBUS device, the data of HMI can be read or written via MODBUS protocol.

As shown in the following figure, the HMI is configured as a MODBUS device (also called MODBUS Server). The HMI, PC or other devices can use MODBUS protocol to read or write HMI data via Ethernet or RS-232 / RS-485 interface.



19.2. Steps to Create a MODBUS Server

 To configure the HMI as a MODBUS device, add a new device in the device list first. Click [PLC type] drop-down box and select "MODBUS Server" driver. There are six selections for [PLC I/F]: RS-232 / RS-485 2W / RS-485 4W / Ethernet / USB / CAN. Choose the PLC interface appropriate for your PLC model.

Device Properties	
Name :	MODBUS Server
	○ HMI ● PLC
Location :	Local Settings
PLC type :	MODBUS Server
	V.1.00, MODBUS_SERVER.e30
PLC I/F :	RS-232 RS-232 RS-485 2W RS-485 4W Ethernet
COM :	USB CAN (Controller Area Network) Bus Comp (South, Controller Area Network) Bus Settings
	Station no. : 1
	Use broadcast command
	OK Cancel

 If [PLC I/F] is set to [RS-232] or [RS-485], please select [COM] (COM 1 ~ COM 3) and set correct communication parameters as shown in the following figure. MODBUS Server



19-2



[Station no.] is set to 1. Click [Settings], the maximum LW address range read / written by Modbus Client can be set. When the object in the project uses a LW register, the Modbus Client will not be able to read or write an address that is not within the specified range.

PLC type :	MODBUS Server
V.1.00, MODBUS_SERV	/ER.e30
PLC I/F : RS-232	~
COM : COM1 (9600,E,8,1)	Settings
Station no.	
Station no.	: 1
COM Port Settings	
	Timeout (sec) : 1.0
COM : COM 1 -	
Baud rate : 9600 -	Turn around delay (ms): 0
Data bits : 8 Bits 🔹	
Parity : Even 👻	
Stop bits : 1 Bit 👻	Limit LW maximum read/write address
	Max. LW address (0~9999): 5000
* OS version 20120920 or later support 1	14400 baud rate OK Cancel

If [PLC I/F] is set to [Ethernet], please set [Port no.].

PLC type :	MODBUS Server	
	V.1.00, MODBUS_SERVER.e30 Ethernet	
	Local,Port=8000(=HMI Port)	Settings
	Station no. : 1	

The [Port no.] of MODBUS Server and HMI must be the same. To change the port number, please set in the [Model] tab.

stem Parame	tem Parameter Settings						
Extended	d Memory	Printer/Back	kup Server	e-Mail	Recipes		
Device	Model	General System Setting Security Font					
	model : eMT31	05 (800 x 600) •			▼		
P	ort no. : 8000	(use	ed as MODBUS ser	ver's port no.)			



3. When finished, MODBUS Server is listed in [Device] tab. The configuration of MODBUS device is completed. Compile the .emtp file and download the compiled .exob file to the HMI, then HMI data can be read or written by using MODBUS protocol.

Extended Memory Printer/Backup Server e-Mail Recipes								
Device	Model	General		System Sett				Font
Device list :								
Device list .								
No.	Name	Location	Devi	ice type	Interfa	се	I/F P	Station no.
				ice type 3105 (800		се	I/F P	Station no

Note

For cMT-SVR, if [Ethernet] PLC interface is chosen, then enter the port number.

IP Address Settings	
Port no. : 502	
	Turn around delay (ms): 0
	OK Cancel

19.3. Steps to Access a MODBUS Server

Two HMIs can be configured as one MODBUS client and one MODBUS server to communicate and exchange data.

 Add a new device in client's device list. If the client chooses [Ethernet] PLC interface, set [PLC type] to "MODBUS TCP/IP" and fill in the correct [IP address] (the IP of MODBUS Server), [Port no.], and [Station no.].



Device Properties	
Name : MODBUS TCP/IP	
MMI O PLC	
Local Settings	
PLC type : MODBUS TCP/IP	
V.1.90, MODBUS_TCPIP.e30	
PLC I/F : Ethernet	
IP Address Settings	
IP address: 192 . 168 . 1 . 100	
Port no.: 502	Settings
Timeout (sec) : 1.0 Turn around delay (ms) : 0	
Send ACK delay (ms): 0	
	imit
The number of resending commands : 0	
OK Cancel	
ОК	Cancel

If the client chooses [RS-232] or [RS-485] PLC interface, the [PLC type] must be set to "MODBUS RTU", and its communication parameters also must be configured correctly.

evice Properties	
Name : M	IODBUS RTU
O	HMI PLC
Location :	ocal 🔹 Settings
PLC type :	MODBUS RIU
COM Port Settings	
COM : Baud rate : Data bits : Parity : Stop bits :	9600 Turn around delay (ms): 0 8 Bits Send ACK delay (ms): 0 Even
* OS version 201209	The number of resending commands : 0
Max. read-	l of block pack (words) : 5 Address Range Limit command size (words) : 120 command size (words) : 120
	OK Cancel



19-5



2. When finished, click [OK], then a new device "MODBUS RTU" is listed in the [Device] tab.

Extended	Memory	Printer	r/Back	up Server	e	-Mail		Recipes
Device Model		General	General System Settir			Security		Font
No.	Name	Location	De	/ice type	Interfac	_	I/F Pr	Station no

3. In the setting page of each object, select "MODBUS RTU" in [PLC name], and set the address of MODBUS RTU.

-Read address	1		
PLC name :	MODBUS RTU	•	Setting
Address :	0x 🔻	1	
	0x		
	1x		
	3x_Bit		
	4x_Bit		
	6x_Bit		
	6x_Bit 0x_multi_coils		

Since the server is an HMI, the corresponding read and write addresses are listed below :

0x/1x (1 ~ 12096)	LB (0 ~ 12095)
3x/4x/5x (1 ~ 9999)	LW (0 ~ 9998)
3x/4x/5x (10000 ~ 65535)	RW (0 ~ 55535)

19.4. Changing MODBUS Server Station Number Online

EasyBuilder Pro provides the following system registers to change MODBUS Server station number online.

LW-9541	MODBUS/ASCII server station no. (COM 1)
LW-9542	MODBUS/ASCII server station no. (COM 2)
LW-9543	MODBUS/ASCII server station no. (COM 3)
LW-9544	MODBUS/ASCII server station no. (Ethernet)

19.5. MODBUS Address Type

In the EasyBuilder Pro, the address types of MODBUS protocol are 0x, 1x, 3x, 4x, 5x, 6x, 3x_bit and 4x_bit. MODBUS RTU function codes are listed below:

0x:	A read and write device type. When reading a bit
Coils	with this device type, the function code is 01H.
	When writing a bit, the function code is 05H. When
	writing multiple bits, the function code is 0fH.



A read only device type. When reading a bit the function code is 02H.
A read only device type. When reading data, the function code is 04H.
A read and write device type. When reading data, the function code is 03H. When writing data, the function code is 10H.
The function code is the same as 4x. The difference is that 5x makes double word swap when the format is 32-bit unsigned. If the data read by 4x is 0x1234, the data read by 5x is 0x3412.
A read and write device type. When reading data, the function code is 03H. The difference from 4x is that when writing data, the function code is 06H, meaning to write a single register.
The function code is the same as 3x. The difference is that 3x_bit reads a single bit in the data.
The function code is the same as 4x. The difference is that 4x bit reads a single bit in the data.

For more information, see "37 MODBUS TCP/IP Gateway".





20. How to Connect a Barcode Reader

This chapter explains how to connect a Barcode reader and the relevant settings.

20.1.	Overview	20-2
20.2.	Steps to Connect a Barcode Reader	20-2



EasyBuilder Pro V5.02.01

20.1. Overview

HMI can connect with barcode reader via the following interfaces:

- USB
- COM port

To connect a barcode reader, please add a new device by the following steps.

20.2. Steps to Connect a Barcode Reader

1. In EasyBuilder Pro » [Edit] » [System Parameter Settings] » [Device list], add a new device.

	S	System Parameter Settings								
		Font Device M		Extended Memory			Pri			
				odel	General	ral System Setting Securit			Security	
		Device list :								
		No.	Name	Location	Device type		Interface	I/F Protoc	ol Station no	
		Local HMI	Local HMI	Local	MT6070iH/MT807	70		-	0	
Device P	Prop	perties								
		Name :	Barcode/K	eyboard (l	JSB/COM)					
			© HMI	PLC	C					
		Location :	Local	•	Settings					
		PLC type :		Barr	ode/Keyboard (USB/C	юм	0	•		
			V.1.30, BA	RCODE.si						
		PLC I/F :	RS-232		•					
		COM :	COM1 (960	00,N,8,1)				(Settings	

2. Click [Settings] and finish [Barcode Device / Keyboard Settings].

Barcode Device / Keyboard Settings	
Barcode device	© Keyboard
Timeout : 1.0 v sec COM : COM 1 v Baud rate : 9600 v Data bits : 8 Bits v Parity : None v	 Read byte limit 10 Use a start code Start code : 0
Stop bits : 1 Bit 🔹	Terminator © CR/LF
	OK Cancel



Setting	Description
Timeout	When select [Barcode device], if the device reads
	slowly, a longer timeout is suggested to read data
	completely.
	When select [Keyboard], a time range can be set for
	keyboard entries. The system starts counting time from
	the first entry.
СОМ	When using COM port, please set the communication
Baud rate Data bits	parameters correctly.
Parity	When using USB, there is no need to set the
Stop bits	parameters.
Read byte limit	If this check box is selected, the number of bytes a
-	barcode reader reads is restricted in order to prevent
	overloading. The range is 10 to 512.
	Please note that the data cannot be read if it exceeds
	the limit.
Use a start code	If this check box is selected, the data is only valid when
	the first data is identical to the start code, otherwise
	the data will be ignored. The start code will not be
	stored in the address of barcode reader.
	For example: if the start code is 255 (0xff), and the data
	read is:
	0xff 0x34 0x39 0x31 0x32 0x30 0x30 0x34 0x37
	The data saved in the designated barcode reader
	address will be:
	0x34 0x39 0x31 0x32 0x30 0x30 0x34 0x37
Terminator	Terminator means the end of data. When a terminator
	is detected, it stands for the end of data stream.
CR/LF	0x0a or 0x0d stands for the end of data stream.
STX/ETX	0x02 or 0x03 stands for the end of data stream.
Othor	Users can set the terminator.
Other	
None	If this check box is selected, HMI will save all the data to

When finish setting, a new device is added to the [Device list]. Now the barcode reader can be selected in [PLC type] when creating an object. The address types are listed in the following table.

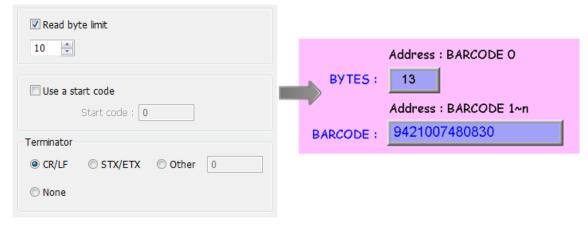




Address Type	Address Name	Description					
Bit	FLAG	FLAG 0 indicates the status of data					
		reading. When reading data, the status of					
		FLAG 0 is set OFF and will return ON after					
		reading successfully.					
	RESET	RESET 0 clears the data of BARCODE and					
		RESULT when set ON.					
	CONNECT_STAT	CONNECT_STATUS 0 indicates whether					
	US	the barcode reader (USB interface) is					
		connected. When the status is ON, the					
		barcode reader is connected.					
Word	BARCODE	barcode reader is connected. BARCODE 0: Number of bytes currently					
word		read.					
		BARCODE 1 ~ n: Stores the data read.					
	RESULT	RESULT 0 indicates the result of data					
		reading. The following codes indicate:					
		0x00 Waiting to read BARCODE.					
		0x01 BARCODE successfully read.					
		0x02 Invalid BARCODE format.					
		0x03 The number of bytes specified in					
		[Read byte limit] exceeded.					
		0x04 The Start Code of the data read					
		does not match the setting.					
		0x05 The Terminator of the data read					
		does not match the setting.					

Example 1

The following is a setting example, the barcode is 9421007480830. BARCODE 0 is the address of Numeric Display Object (BYTES) and BARCODE 1 \sim n is the address of ASCII Display object (BARCODE).



Barcode Reader Address	Data
	13 bytes (decimal)
	However, the data saved is 14 bytes = 7 words.
BARCODE 0	It is because when the number of bytes is an
	odd number, the system adds a byte (0x00) to
	make it an even number.
BARCODE 1	3439 (HEX)
BARCODE 2	3132 (HEX)
BARCODE 3	3030 (HEX)
BARCODE 4	3437 (HEX)
BARCODE 5	3038 (HEX)
BARCODE 6	3338 (HEX)
BARCODE 7	0030 (HEX)

In the example the data stored in the barcode reader address is listed in the following table:



HMI can only connect with one USB barcode reader. When the device list in the project includes this kind of device, the system register LB-9064: [enable USB barcode device (disable keyboard) (when ON)] is set ON. To enable USB keyboard again and stop using USB barcode reader, please set LB-9064 OFF.

Lick the icon to download the demo project. Please confirm your internet connection.





21. Ethernet Communication and Multi-HMI Connection

This chapter explains how to connect multiple devices via Ethernet.

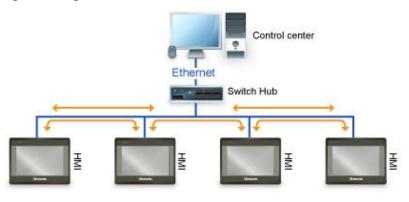
21.1.	Overview	21-2
21.2.	HMI to HMI Communication	21-2
21.3.	PC to HMI Communication	21-3
21.4.	Operate the PLC Connected with Other HMI	21-4



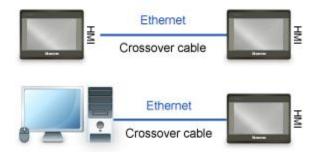
21.1. Overview

There are two ways of Ethernet communication:

• Use RJ45 straight through cable and hub.



• Use RJ45 crossover cable and without hub, but this is limited to point-to-point connection (HMI to HMI or PC to HMI).



Through Ethernet network, the system provides the following methods for data transmission:

- HMI to HMI communication.
- PC to HMI communication.
- Operating the PLC connected to another HMI.

21.2. HMI to HMI Communication

To exchange data between one HMI and another HMI, add a new remote HMI device in [System Parameter Settings]. If there are 2 HMIs (HMI A and HMI B), in order to use a Set Bit object on HMI A to control [LB-0] on HMI B, the setting of the project of HMI A is explained in the following part.



1. Set the IP address of the two HMIs, for example, HMI A: 192.168.1.1, HMI B: 192.168.1.2.



2. In [System Parameter Settings] » [Device list], add a remote HMI B (IP: 192.168.1.2).

Device Properties	
Name :	HMIB
	● HMI
Location :	Remote Settings IP : 192.168.1.2 (Port = 8000)
	ress Settings hernet IP address : 192 . 168 . 1 . 2 Port no. : 8000 OK Cancel

 Create a Set Bit Object, select "HMI B" in [PLC name] to control the address of the remote HMI.

New Set Bit Object	×
General Security Shape Label	
Comment :	
Write address	
PLC name : HMI B 🔹 Setting	
Address : LB 🔹 0	
Write after button is released	
Attribute Set style : Set ON	•
Macro	

Note

- One HMI can handle requests from a maximum of 64 HMIs simultaneously.
- One cMT-SVR can handle requests from a maximum of 32 HMIs simultaneously.

21.3. PC to HMI Communication

With On-line Simulation, PC can collect data from HMI through Ethernet network and save the data files to PC. To connect PC with two HMIs (HMI A and HMI B), the setting of the project on PC is explained in the following part.



- 1. Set the IP address of the two HMIs, for example, HMI A: 192.168.1.1, HMI B: 192.168.1.2.
- 2. In [System Parameter Settings] » [Device list], add a remote HMI A (IP: 192.168.1.1) & HMI



B (IP: 192.168.1.2).

Extended I	Memory	Printer/Ba	ackup Serve	r	e-Mail		Recipes	
Device	evice Model		General System			Security	Font	
No.	Name Local					Interface	L.	
Local HM	Local	HMI Lo	ocal	eMT310	5 (800	-	-	
Remote H	IMI1 HMIA	R	emote (I	eMT/MT	SERIES	Ethernet	Т	
		R				Ethernet	-	

 Create a Set Bit Object, select "HMI A" in [PLC name] to control the address of the remote HMI A. Same for the HMI B.

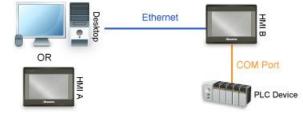
New Set Bit Object
General Security Shape Label
Comment :
Write address
PLC name : HMI A Setting
Address : LB 🔹 0
Write after button is released
Attribute
Set style : Set ON

Note

- A PC can control at most 64 HMIs simultaneously.
- As shown above, HMI can also control PC. PC can be seen as another HMI, that is, adding a remote HMI in the project of HMI A / HMI B, and the IP of the remote HMI is set to the IP of PC.

21.4. Operating the PLC Connected with Other HMI

Through Ethernet network, PC or HMI can operate the PLC that is connected to another HMI. If PLC is connected to COM 1of HMI B, when using PC or HMI A to read PLC data, the setting of the project of PC or HMI A is explained in the following part.





21.4.1. Settings of eMT / mTV Series

- **1.** Set the IP address of HMI B, for example, 192.168.1.2.
- In [System Parameter Settings] » [Device list], add a remote PLC, and set [Name] to "PLC on HMI B". Set correct parameters. Since this PLC is connected to remote HMI B, set the IP address to HMI B (IP: 192.168.1.2).

Device Properties	
Name :	PLC on HMI B
(⊖ HMI
Location : (Remote Settings IP : 192.168.1.2 (Port = 8000)
PLC type :	Mitsubishi FX03/FX0N/FX1S/FX1N/FX2
	RS-485 4W
сом :	COM1 Settings
	PLC default station no. : 0
	al of block pack (words) : 5
	d-command size (words): 32 * e-command size (words): 32 *
	OK Cancel

3. Create a Set Bit Object, select "PLC on HMI B" in [PLC name] to control the PLC connected with the remote HMI B.

New Set Bit Object
General Security Shape Label
Comment : Write address PLC name : PLC on HMI B Address : X 0
Write after button is released
Attribute Set style : Set ON
Macro



21.4.2. Settings of cMT-SVR Series

- 1. Set the IP address of HMI B, for example, 192.168.1.2.
- In [System Parameter Settings] » [Device list], click [New HMI]. Set the IP address to HMI B (IP: 192.168.1.2).

Device Properties
Name : HMI B
• HMI
Location : Remote Settings IP : 192.168.1.2 (Port = 8000)
IP Address Settings
IP address: 192 . 168 . 1 . 2 Port no.: 8000
OK Cancel

 In the project of HMI B, go to [System Parameter Settings] » [Device list], click [New PLC], set [Name] to "PLC on HMI B". Set correct parameters.

System Parameter Settings						e	×			
Extended Memor	y		e-Mail			Recipes				
Device Model	General	Syste	em Setting	Secur	rity	Font Mapping				
Device list :										
No. Na	me	Location	n Device ty	pe	Inter	face I.				
		Local	cMT-SVR		-	-				
Remote HMI 1 HM	В	Remote	(I CMT-SVR	1	Ethe	rnet T				
		Î	Device Properties	s						
				Name : [PLC on	HMI B				
						PLC				
			Lo	cation : (Remot	e 🔻 Sett	ings .	IP : 192.	168.1.2 (Port	t = 8000)
			PLC	C type :		Mitsubishi FXOs/	FX0n/I	FX1s/FX1n/FX2		
					V.1.40,	MITSUBISHI_FX0N.e3	30			
2	2		P	LC I/F : [RS-485	4W		•		
•										
New HMI	New PLC									
				COM :	COM1					Settings
				F	PLC def	ault station no. : 0				
				[Defa	ult station no. use sta	ation r	no. variable		
				Interv	al of bl	ock pack (words) : 5		•		
						nand size (words): 3 nand size (words): 3		v		
									ОК	Cancel

4. When finished, a remote PLC can be found under Remote HMI 1. Local HMI 1 stands for HMI A, Remote HMI 1 stands for HMI B, and Remote PLC 1 is connected with HMI B.

	emory		e-M	lail	_		Recipes	
Model	General		System S	etting	Securi	ty	Font Map	oping
	Name		Location	Device ty	pe	Interf	ace	Ŀ
MI	Local HMI	L	Local	cMT-SVR	(1024	-		-
HMI 1	HMI B	F	Remote (I	CMT-SVF	t	Ether	net	Т
iote P	PLC on HMI B	F	Remote (I	Mitsubish	ni FX0	COM	1 (9600,E	. F
	HMI 1	MI Local HMI	MI Local HMI HMI B	MI Local HMI Local HMI 1 HMI B Remote (I	MI Local HMI Local CMT-SVR HMI 1 HMI B Remote (I CMT-SVR	Local HMI Local CMT-SVR (1024 HMI 1 HMI B Remote (I CMT-SVR	Local HMI Local CMT-SVR (1024 HMI 1 HMI B Remote (I CMT-SVR Ether	Local HMI Local cMT-SVR (1024 - HMI 1 HMI B Remote (I CMT-SVR Ethernet

5. Create a Set Bit Object, select "PLC on HMI B" in [PLC name] to control the PLC connected with the remote HMI B.

New Set Bit Object	X
General Security Shape Label	
Comment :	
Write address	5
PLC name : PLC on HMI B Setting	
Address : X 🗸	
Attribute Set style : Set ON	•
Macro	

Note

Remote HMI in a cMT-SVR project must be a cMT-SVR machine. Thus, a cMT-SVR cannot communicate with PLCs connected with other series, such as eMT, mTV-series.



22. System Registers

This chapter introduces different types of registers.

22. S	System F	Registers	22-1
22.	1. Ove	erview	22-3
22.	2. The	Address Ranges of Local HMI	
	22.2.1.	Bits	22-4
	22.2.2.	Words	22-4
22.	3. Sys	tem Registers	22-5
	22.3.1.	HMI Time	22-5
	22.3.2.	HMI Operation	22-6
	22.3.3.	Touch Position	22-7
	22.3.4.	Local HMI Network Information	22-7
	22.3.5.	Project File Information	22-9
	22.3.6.	Storage Space Management	22-10
	22.3.7.	Recipe and Extended Memory	22-10
	22.3.8.	Data Sampling	22-12
	22.3.9.	Event Log	22-12
	22.3.10	. Station Number Variables	22-13
	22.3.11	. Index Registers	22-14
	22.3.12	. MODBUS Server Communication	22-15
	22.3.13	. Communication Parameter Settings	22-16
	22.3.14	. Communication Status and Control with PLC (COM)	22-20
	22.3.15	. Communication Status and Control with PLC (Ethernet)	22-21
	22.3.16	. Communication Status and Control with PLC (USB)	22-23
	22.3.17	. Communication Status and Control with PLC (CAN Bus)	22-24
	22.3.18	. Communication Status and Control with Remote HMI	22-24
	22.3.19	. Communication Status and Control with Remote PLC	22-28
	22.3.20	. Local/Remote Operation Restrictions	22-30
	22.3.21	. Communication Error Codes	22-30
	22.3.22	. Driver ID	22-32
	22.3.23	. DLT645 Controller	22-32
	22.3.24	. [PLC No Response] Window Control	22-32
	22.3.25	. [Fast Selection] Window Control	22-33
	22.3.26	. EasyAccess	22-33



22-1

22-33
22-34
22-34
22-35
22-36
22-36
22-37
22-38
22-39
22-39
22-41



22.1. Overview

Some Word and Bit addresses are reserved in EasyBuilder Pro. These registers are reserved for different functions. This chapter introduces different types of registers.

The "C" letter in the register tables stands for "Control", which means that this register not only allows write operation, but also can be controlled by Macro or a remote HMI. When using a cMT Series model, the PLW and PLB registers can be selected. LW/LB are local registers whereas PLW/PLB are client registers. The client device can be: cMT-iV5, iPad, Android device...etc. When connecting a cMT Series model with one or multiple client devices, the PLW/PLB registers can be set on the client devices.

٩o.	Address tag name		PLC name	Addre	Address	Read/W	Comment	
1	LB-9000 : initia	lized as ON	Local HMI	Bit	LB-9000	Read/		-0
2	LB-9001 : initia	lized as ON	Local HMI	Bit	LB-9001	Read/		
3	LB-9002 : initia	lized as ON	Local HMI	Bit	LB-9002	Read/		
1	LB-9003 : initia	lized as ON	Local HMI	Bit	LB-9003	Read/		
5	LB-9004 : initia	lized as ON	Local HMI	Bit	LB-9004	Read/		
5	LB-9005 : initia	lized as ON	Local HMI	Bit	LB-9005	Read/		
7	LB-9006 : initia	lized as ON	Local HMI	Bit	LB-9006	Read/		
3	LB-9007 : initia	lized as ON	Local HMI	Bit	LB-9007	Read/		
9	LB-9008 : initia	lized as ON	Local HMI	Bit	LB-9008	Read/		
10	LB-9009 : initia	lized as ON	Local HMI	Bit	LB-9009	Read/		
1	LB-9010 : data	download indicator	Local HMI	Bit	LB-9010	Read		
12	LB-9011 : data	upload indicator	Local HMI	Bit	LB-9011	Read		
13	LB-9012 : data	download/uploa	Local HMI	Bit	LB-9012	Read		
14	LB-9013 : FS w	rindow control[hi	Local HMI	Bit	LB-9013	Read/		
15	LB-9014 : FS b	utton control[hid	Local HMI	Bit	LB-9014	Read/		
16	LB-9015 : FS w	rindow/button c	Local HMI	Bit	LB-9015	Read/		
17	LB-9016 : statu	is on when a cl	Local HMI	Bit	LB-9016	Read/		
18	LB-9017 : disab	le write-back in	Local HMI	Bit	LB-9017	Read/		
19	LB-9018 : disab	le mouse cursor	Local HMI	Bit	LB-9018	Read/		
20	LD 0010 + disph	la lanabla buttar	Local UMT	D#-	10.0010	Dood/		F.
	New	Delete	Delete All		Settings			
S	ave Tag File	Load Tag File						



22.2. The Address Ranges of Local HMI

22.2.1. Bits

Register	Device Type	Range	Format
Local Bits	LB	0~12399	DDDDD
Client Bits	PLB	0~12399	DDDDD
Local Word	LW_Bit	0 ~	DDDDDdd
Bits		1130015	DDDDD: address
			dd: bit no. (00 ~ 15)
Client Word	PLW_Bit	0 ~	DDDDDdd
Bits		1079915	DDDDD: address
			dd: bit no. (00 ~ 15)
Retentive Bit	RBI	0~65535f	DDDDDh
Index			DDDDD: address
			h: bit no. (0 ~ f)
			Use LW-9000 as Index
			Register, and
			correspond to RW_Bit
Retentive	RW_Bit	0 ~	DDDDDh
Word		524287f	DDDDD: address
Bits			h: bit no. (0 ~ f)
Retentive A	RW_A_Bit	0~65535f	DDDDDh
Word			DDDDD: address
Bits			h: bit no. (0 ~ f)

22.2.2. Words

Register	Device Type	Range	Format
Local Words	LW	0~11300	DDDDD
Client Words	PLW	0~10799	DDDDD
Retentive	RW	0~524287	DDDDDD
Words			
Retentive A	RW_A	0~65535	DDDDD
Words			
Retentive	RWI	0~65535	DDDDD
Word			Use LW-9000 as Index
Index			Register, and
			correspond to RW
Extended	EM0 ~ EM9	0 ~	DDDDDDDDD
Memory		1073741823	
Words			



22-4

22.3. System Registers

22.3.1. HMI Time

		Read(R)/	/Write(W)/Contro	
Address	Description	Local HMI	Macro	Remote HMI
LB-11958	time setting error (when ON) *Note 3	R	R	R
LW-9010	(16bit-BCD) : local second	R/W	R/C	R/C
LW-9011	(16bit-BCD) : local minute	R/W	R/C	R/C
LW-9012	(16bit-BCD) : local hour	R/W	R/C	R/C
LW-9013	(16bit-BCD) : local day	R/W	R/C	R/C
LW-9014	(16bit-BCD) : local month	R/W	R/C	R/C
LW-9015	(16bit-BCD) : local year	R/W	R/C	R/C
LW-9016	(16bit-BCD) : local week	R	R	R
LW-9017	(16bit) : local second	R/W	R/C	R/C
LW-9018	(16bit) : local minute	R/W	R/C	R/C
LW-9019	(16bit) : local hour	R/W	R/C	R/C
LW-9020	(16bit) : local day	R/W	R/C	R/C
LW-9021	(16bit) : local month	R/W	R/C	R/C
LW-9022	(16bit) : local year *Note 1	R/W	R/C	R/C
LW-9023	(16bit) : local week *Note 2	R	R	R
LW-9030	(32bit) : system time (unit : 0.1 second)	R	R	R
LW-9048	(16bit) : time (0 : AM, 1 : PM)	R/W	R/C	R/C
LW-9049	(16bit) : local hour (12-hour format)	R/W	R/C	R/C

Note

- **1.** Value range: 2000 ~ 2037.
- 2. Value range: 0 ~ 6, stand for Sunday ~ Saturday.
- 3. When use LW-9010 to LW-9023 to update RTC time, the system will check if RTC time is successfully updated. If the system still fails to update RTC time, the system register [LB-11958: time setting error] will be set ON, and restore to the time before update. Updating time on PC during simulation by using LW-9010 to LW-9023 is ineffective.





22.3.2. HMI Operation

		Read(R)/	Write(W)/	Control(C)
Address	Description	Local HMI	Macro	Remote HMI
LB-9018	disable mouse cursor (set ON)	R/W	R/C	R/C
LB-9019	disable/enable buzzer	R/W	R/C	R/C
LB-9020	show (set ON)/ hide (set OFF) system setting bar	R/W	R/C	R/C
LB-9033	disable(when on)/enable (when off) HMI upload function *Note 1	R/W	R/C	R
LB-9040	backlight up (set ON) *Note 2	w	С	С
LB-9041	backlight down (set ON) *Note 2	w	С	С
LB-9047	reboot HMI (set ON when LB-9048 is on)	w	С	С
LB-9048	reboot-HMI protection	R/W	R/C	R/C
LB-9062	open hardware setting dialog (set ON)	w	С	С
LB-9063	disable(set ON)/enable(set OFF) popuping information	D / M		D/C
	dialog while finding an USB disk	R/W	R/C	R/C
LB-9064	enable USB barcode device (disable keyboard) (when ON)	R/W	R/C	R
	*Note 5	.,		
LB-11959	LED indicator control *Note 4	R/W	R/C	R/C
LB-12042	open/close [System information] dialog (set ON/set OFF)	R/W	R/C	R/C
LB-12051	buzzer status (active when ON)	R/W	R/C	R/C
LW-9007	(16bit) : hardware index	R	R	R
LW-9008	(32bit-float) : battery voltage *Note 3	R	R	R
LW-9025	(16bit) : CPU loading (x 100%)	R	R	R
LW-9026	(16bit) : OS version (year)	R	R	R
LW-9027	(16bit) : OS version (month)	R	R	R
LW-9028	(16bit) : OS version (day)	R	R	R
LW-9040	(16bit) : backlight index *Note 2	R	R	R
LW-9051	(16bit) : audio volume (0 ~100)	R/W	R/C	R/C
LW-9080	(16bit) : backlight saver time (unit : minute)	R/W	R/C	R/C
LW-9081	(16bit) : screen saver time (unit : minute)	R/W	R/C	R/C
LW-9199	(16bit) : external keyboard layout : 0 (QWERTY), 1 (AZERTY)	R/W	R/C	R/C
LW-9350	(16bit) : pending command no. in local HMI	R	R	R
LW-10884	(16 words) : HMI name	R/W	R/C	R/C
LW-11155	(32bit) : the total size of HMI memory (unit : KB)	R	R	R
LW-11157	(32bit) : the free size of HMI memory (unit : KB)	R	R	R
LW-11159	(16bit) : memory loading (x 100%)	R	R	R



Note

- 1. After changing the settings, please reboot HMI for the updates to take effect.
- Use LW-9040 together with LB-9040 ~ LB-9041 to adjust the backlight brightness, range: 0 2. ~ 31.
- 3. Only supported by eMT Series. When the battery voltage level, indicated by LW-9008, drops below 2.8V, battery replacement is recommended.
- 4. When multiple mTV or cMT-SVR devices are used, this register can be triggered to make the LED indicator blink for identifying the device.
- 5. LB-9064: Enable USB barcode device (disable keyboard).
- Lick the icon to download the demo project. Please confirm your internet connection.

22.3.3. Touch Position

			Control(C)	
Address	Description		Remote HMI	
LW-9041	(16bit) : touch status word(bit 0 on = user is touching the screen)	R	R	R
LW-9042	(16bit) : touch x position	R	R	R
LW-9043	(16bit) : touch y position	R	R	R
LW-9044	(16bit) : leave x position	R	R	R
LW-9045	(16bit) : leave y position	R	R	R

Lick the icon to download the demo project that explains how to how to trigger relevant registers to change page with finger slide. Please confirm your internet connection.

22.3.4. Local HMI Network Information

		Read(R)/	Read(R)/Write(W)/Control(C		
Address	Description			Remote HMI	
LB-12041	refresh HMI IP (LW-9129~9132) (set ON)	R/W	R/C	R/C	
LB-12094	update ethernet 1 setting (IP, netmask, gateway) (set ON)	R/W	R/C	R/C	
LB-12095	update ethernet 2 setting (IP, netmask) (set ON)	R/W	R/C	R/C	
LW-9125	(16bit) : HMI ethernet 1 gateway 0 (machine used only)	R/W	R/C	R/C	
LW-9126	(16bit) : HMI ethernet 1 gateway 1 (machine used only)	R/W	R/C	R/C	
LW-9127	(16bit) : HMI ethernet 1 gateway 2 (machine used only)	R/W	R/C	R/C	
LW-9128	(16bit) : HMI ethernet 1 gateway 3 (machine used only)	R/W	R/C	R/C	
LW-9129	(16bit) : HMI ethernet 1 IP 0 (machine used only)	R/W	R/C	R/C	
LW-9130	(16bit) : HMI ethernet 1 IP 1 (machine used only)	R/W	R/C	R/C	



LW-9131	(16bit) : HMI ethernet 1 IP 2 (machine used only)	R/W	R/C	R/C
LW-9132	(16bit) : HMI ethernet 1 IP 3 (machine used only)	R/W	R/C	R/C
LW-9133	(16bit) : ethernet port no.	R	R	R
LW-9135	(16bit) : ethernet 1 media access control (MAC) address 0	R	R	R
LW-9136	(16bit) : ethernet 1 media access control (MAC) address 1	R	R	R
LW-9137	(16bit) : ethernet 1 media access control (MAC) address 2	R	R	R
LW-9138	(16bit) : ethernet 1 media access control (MAC) address 3	R	R	R
LW-9139	(16bit) : ethernet 1 media access control (MAC) address 4	R	R	R
LW-9140	(16bit) : ethernet 1 media access control (MAC) address 5	R	R	R
LW-9141	(16bit) : HMI station no.	R/W	R/C	R/C
LW-10750	(16bit) : HMI ethernet 1 Mask 0 (machine used only)	R/W	R/C	R/C
LW-10751	(16bit) : HMI ethernet 1 Mask 1 (machine used only)	R/W	R/C	R/C
LW-10752	(16bit) : HMI ethernet 1 Mask 2 (machine used only)	R/W	R/C	R/C
LW-10753	(16bit) : HMI ethernet 1 Mask 3 (machine used only)	R/W	R/C	R/C
LW-10786	(16bit) : HMI ethernet 2 IP 0 (machine used only)	R/W	R/C	R/C
LW-10787	(16bit) : HMI ethernet 2 IP 1 (machine used only)	R/W	R/C	R/C
LW-10788	(16bit) : HMI ethernet 2 IP 2 (machine used only)	R/W	R/C	R/C
LW-10789	(16bit) : HMI ethernet 2 IP 3 (machine used only)	R/W	R/C	R/C
LW-10790	(16bit) : HMI ethernet 2 netmask 0 (machine used only)	R/W	R/C	R/C
LW-10791	(16bit) : HMI ethernet 2 netmask 1 (machine used only)	R/W	R/C	R/C
LW-10792	(16bit) : HMI ethernet 2 netmask 2 (machine used only)	R/W	R/C	R/C
LW-10793	(16bit) : HMI ethernet 2 netmask 3 (machine used only)	R/W	R/C	R/C
LW-10794	(16bit) : HMI ethernet 2 gateway 0 (machine used only)	R/W	R/C	R/C
LW-10795	(16bit) : HMI ethernet 2 gateway 1 (machine used only)	R/W	R/C	R/C
LW-10796	(16bit) : HMI ethernet 2 gateway 2 (machine used only)	R/W	R/C	R/C
LW-10797	(16bit) : HMI ethernet 2 gateway 3 (machine used only)	R/W	R/C	R/C
LW-10798	(16bit) : ethernet 2 media access control (MAC) address 0	R	R	R
LW-10799	(16bit) : ethernet 2 media access control (MAC) address 1	R	R	R
LW-10800	(16bit) : ethernet 2 media access control (MAC) address 2	R	R	R
LW-10801	(16bit) : ethernet 2 media access control (MAC) address 3	R	R	R
LW-10802	(16bit) : ethernet 2 media access control (MAC) address 4	R	R	R
LW-10803	(16bit) : ethernet 2 media access control (MAC) address 5	R	R	R
LW-10804	(16bit) : HMI ethernet 1 domain name system (DNS) server IP0	R	R	R
LW-10805	(16bit) : HMI Ethernet 1 domain name system (DNS) server IP1	R	R	R
LW-10806	(16bit) : HMI Ethernet 1 domain name system (DNS) server	R	R	R



System Registers

	IP2			
LW-10807	(16bit) : HMI Ethernet 1 domain name system (DNS) server IP3	R	R	R
LW-10808	(16bit) : HMI ethernet 2 domain name system (DNS) server IPO	R	R	R
LW-10809	(16bit) : HMI ethernet 2 domain name system (DNS) server IP1	R	R	R
LW-10810	(16bit) : HMI ethernet 2 domain name system (DNS) server IP2	R	R	R
LW-10811	(16bit) : HMI ethernet 2 domain name system (DNS) server IP3	R	R	R
LW-10812	(16bit) : obtain an IP address automatically (DHCP => 0 : off,1 : on)	R/W	R/C	R/C
LW-10813	(16bit) : obtain an ethernet 2 IP address automatically (DHCP => 0 : off, 1 : on)	R/W	R/C	R/C
LW-10815	(16bit) : link speed of ethernet 1 (0:failure, 10 (10M), 100 (100M), 1000 (1G))	R	R	R
LW-10816	(16bit) : link speed of ethernet 2 (0:failure, 10 (10M), 100 (100M), 1000 (1G))	R	R	R

Note

1. Registers relevant to Ethernet 2 are only available for cMT-SVR model.

22.3.5. Project File Information

		Read(R)/	Write(W)/0	Control(C)
Address	Description	Local HMI	Macro	Remote HMI
LW-9100	(16bit) : project name (16 words)	R	R	R
LW-9116	(32bit) : project size in bytes	R	R	R
LW-9118	(32bit) : project size in K bytes	R	R	R
LW-9120	(32bit) : compiler version	R	R	R
LW-9122	(16bit) : project compiled date [year]	R	R	R
LW-9123	(16bit) : project compiled date [month]	R	R	R
LW-9124	(16bit) : project compiled date [day]	R	R	R



22.3.6. Storage Space Management

		Read(R)/	Read(R)/Write(W)/Control(C)		
Address	Description			Remote HMI	
LB-9035	HMI free space insufficiency alarm (when ON)	R	R	R	
LB-9036	SD card free space insufficiency alarm (when ON)	R	R	R	
LB-9037	USB disk free space insufficiency alarm (when ON)	R	R	R	
LB-12048	USB disk status (exists when ON)	R	R	R	
LB-12050	SD card status (exists when ON)	R	R	R	
LW-9070	(16bit) : free space insufficiency warning (Mega bytes)	R	R	R	
LW-9071	(16bit) : reserved free space size (Mega bytes)	R	R	R	
LW-9072	(32bit) : HMI current free space (K bytes)	R	R	R	
LW-9074	(32bit) : SD current free space (K bytes)	R	R	R	
LW-9076	(32bit) : USB disk current free space (K bytes)	R	R	R	

Click the icon to download the demo project that explains how to use LW-9072 ~ LW-9076 with Backup Object. Please confirm your internet connection.

22.3.7. Recipe and Extended Memory

		Read(R)/	Read(R)/Write(W)/Control(
Address	Description	Local HMI	Macro	Remote HMI	
LB-9028	reset all recipe data (set ON)	W	С	С	
LB-9029	save all recipe data to machine (set ON)	W	С	С	
LB-9460	EM0's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9461	EM1's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9462	EM2's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9463	EM3's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9464	EM4's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9465	EM5's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9466	EM6's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9467	EM7's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9468	EM8's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9469	EM9's storage device (SD card) does not exist (when ON)	R	R	R	
LB-9470	EM0's storage device (USB disk) does not exist (when ON)	R	R	R	
LB-9471	EM1's storage device (USB disk) does not exist (when ON)	R	R	R	
LB-9472	EM2's storage device (USB disk) does not exist (when ON)	R	R	R	
LB-9473	EM3's storage device (USB disk) does not exist (when ON)	R	R	R	



System Registers

LB-9474	EM4's storage device (USB disk) does not exist (when ON)	R	R	R
LB-9475	EM5's storage device (USB disk) does not exist (when ON)	R	R	R
LB-9476	EM6's storage device (USB disk) does not exist (when ON)	R	R	R
LB-9477	EM7's storage device (USB disk) does not exist (when ON)	R	R	R
LB-9478	EM8's storage device (USB disk) does not exist (when ON)	R	R	R
LB-9479	EM9's storage device (USB disk) does not exist (when ON)	R	R	R



22.3.8. Data Sampling

		Read(R)/	Read(R)/Write(W)/Control(
Address	Description	Local HMI	Macro	Remote HMI	
LB-9025	delete the earliest data sampling file on HMI memory (set ON)	w	с	С	
LB-9026	delete all data sampling files on HMI memory (set ON)	w	С	С	
LB-9027	refresh data sampling information on HMI memory (set ON)	w	С	С	
LB-9034	save event/data sampling to HMI, USB disk, SD card (set ON)	w	с	с	
LB-11949	delete the earliest data sampling file on SD card (set ON)	w	С	С	
LB-11950	delete all data sampling files on SD card (set ON)	w	С	С	
LB-11951	refresh data sampling information on SD card (set ON)	w	С	С	
LB-11952	delete the earliest data sampling file on USB disk (set ON)	w	С	С	
LB-11953	delete all data sampling files on USB disk (set ON)	w	С	С	
LB-11954	refresh data sampling information on USB disk (set ON)	w	С	С	
LW-9063	(16bit) : no. of data sampling files on HMI memory	R	R	R	
LW-9064	(32bit) : size of data sampling files on HMI memory	R	R	R	
LW-10489	(16bit) : no. of data sampling files on SD card	R	R	R	
LW-10490	(32bit) : size of data sampling files on SD card	R	R	R	
LW-10492	(16bit) : no. of data sampling files on USB disk	R	R	R	
LW-10493	(32bit) : size of data sampling files on USB disk	R	R	R	

Note

1. The registers for deleting or updating data samplings do not work during simulation on PC.

22.3.9. Event Log

		Read(R)/	Write(W)/0	Control(C)
Address	Description			Remote HMI
LB-9021	reset current event log (OFF->ON)	W	С	С
LB-9022	delete the earliest event log file on HMI memory (set ON)	W	С	С
LB-9023	delete all event log files on HMI memory (set ON)	W	С	С
LB-9024	refresh event log information on HMI memory (set ON)	W	С	С
LB-9034	save event/data sampling to HMI, USB disk, SD card (set ON)	w	С	с
LB-9042	acknowledge all alarm events (set ON)	W	С	С



LB-9043	unacknowledged events exist (when ON)	R	R	R
LB-11940	delete the earliest event log file on SD card (set ON)	W	С	С
LB-11941	delete all event log files on SD card (set ON)	W	С	С
LB-11942	refresh event log information on SD card (set ON)	W	С	С
LB-11943	delete the earliest event log file on USB disk (set ON)	W	С	С
LB-11944	delete all event log files on USB disk (set ON)	W	С	С
LB-11945	refresh event log information on USB disk (set ON)	W	С	С
LW-9060	(16bit) : no. of event log files on HMI memory	R	R	R
LW-9061	(32bit) : size of event log files on HMI memory	R	R	R
LW-9450	(16bit) : time tag of event log – second *Note 1	R/W	R/C	R/C
LW-9451	(16bit) : time tag of event log – minute *Note 1	R/W	R/C	R/C
LW-9452	(16bit) : time tag of event log – hour *Note 1	R/W	R/C	R/C
LW-9453	(16bit) : time tag of event log – day *Note 1	R/W	R/C	R/C
LW-9454	(16bit) : time tag of event log – month *Note 1	R/W	R/C	R/C
LW-9455	(16bit) : time tag of event log – year *Note 1	R/W	R/C	R/C
LW-10480	(16bit) : no. of event log files on SD card	R	R	R
LW-10481	(32bit) : size of event log files on SD card	R	R	R
LW-10483	(16bit) : no. of event log files on USB disk	R	R	R
LW-10484	(32bit) : size of event log files on USB disk	R	R	R

Note

- If LW-9450 ~ LW-9455 are used to get Event Log time, please enable in [system parameters] » [General].
- 2. The registers for deleting or updating event logs do not work during simulation on PC.

Lick the icon to download the demo project that explains how to use the system registers

LW-9450 to LW-9455 to be the time tag of event log. Please confirm your internet connection.

22.3.10. Station Number Variables

			Read(R)/Write(W)/Control(C)		
Address	Iress Description		Local HMI	Macro	Remote HMI
LW-10000	(16bit) : var0 - station no variable	(usage : var0#address)	R/W	R/C	R/C
LW-10001	(16bit) : var1 - station no variable	(usage : var1#address)	R/W	R/C	R/C
LW-10002	(16bit) : var2 - station no variable	(usage : var2#address)	R/W	R/C	R/C
LW-10003	(16bit) : var3 - station no variable	(usage : var3#address)	R/W	R/C	R/C
LW-10004	(16bit) : var4 - station no variable	(usage : var4#address)	R/W	R/C	R/C
LW-10005	(16bit) : var5 - station no variable	(usage : var5#address)	R/W	R/C	R/C





LW-10006	(16bit) : var6 - station no variable (usage : var6#address)	R/W	R/C	R/C
LW-10007	(16bit) : var7 - station no variable (usage : var7#address)	R/W	R/C	R/C
LW-10008	(16bit) : var8 - station no variable (usage : var8#address)	R/W	R/C	R/C
LW-10009	(16bit) : var9 - station no variable (usage : var9#address)	R/W	R/C	R/C
LW-10010	(16bit) : var10 - station no variable (usage : var10#address)	R/W	R/C	R/C
LW-10011	(16bit) : var11 - station no variable (usage : var11#address)	R/W	R/C	R/C
LW-10012	(16bit) : var12 - station no variable (usage : var12#address)	R/W	R/C	R/C
LW-10013	(16bit) : var13 - station no variable (usage : var13#address)	R/W	R/C	R/C
LW-10014	(16bit) : var14 - station no variable (usage : var14#address)	R/W	R/C	R/C
LW-10015	(16bit) : var15 - station no variable (usage : var15#address)	R/W	R/C	R/C

Lick the icon to download the demo project. Please confirm your internet connection.

22.3.11. Index Registers

		Read(R)/Write(W)/Control(C)		
Address	Description	Local HMI	Macro	Remote HMI
LW-9200	(16bit) : address index 0	R/W	R/C	R/C
LW-9201	(16bit) : address index 1	R/W	R/C	R/C
LW-9202	(16bit) : address index 2	R/W	R/C	R/C
LW-9203	(16bit) : address index 3	R/W	R/C	R/C
LW-9204	(16bit) : address index 4	R/W	R/C	R/C
LW-9205	(16bit) : address index 5	R/W	R/C	R/C
LW-9206	(16bit) : address index 6	R/W	R/C	R/C
LW-9207	(16bit) : address index 7	R/W	R/C	R/C
LW-9208	(16bit) : address index 8	R/W	R/C	R/C
LW-9209	(16bit) : address index 9	R/W	R/C	R/C
LW-9210	(16bit) : address index 10	R/W	R/C	R/C
LW-9211	(16bit) : address index 11	R/W	R/C	R/C
LW-9212	(16bit) : address index 12	R/W	R/C	R/C
LW-9213	(16bit) : address index 13	R/W	R/C	R/C
LW-9214	(16bit) : address index 14	R/W	R/C	R/C



LW-9215	(16bit) : address index 15	R/W	R/C	R/C
LW-9230	(32bit) : address index 16	R/W	R/C	R/C
LW-9232	(32bit) : address index 17	R/W	R/C	R/C
LW-9234	(32bit) : address index 18	R/W	R/C	R/C
LW-9236	(32bit) : address index 19	R/W	R/C	R/C
LW-9238	(32bit) : address index 20	R/W	R/C	R/C
LW-9240	(32bit) : address index 21	R/W	R/C	R/C
LW-9242	(32bit) : address index 22	R/W	R/C	R/C
LW-9244	(32bit) : address index 23	R/W	R/C	R/C
LW-9246	(32bit) : address index 24	R/W	R/C	R/C
LW-9248	(32bit) : address index 25	R/W	R/C	R/C
LW-9250	(32bit) : address index 26	R/W	R/C	R/C
LW-9252	(32bit) : address index 27	R/W	R/C	R/C
LW-9254	(32bit) : address index 28	R/W	R/C	R/C
LW-9256	(32bit) : address index 29	R/W	R/C	R/C
LW-9258	(32bit) : address index 30	R/W	R/C	R/C
LW-9260	(32bit) : address index 31	R/W	R/C	R/C

Lick the icon to download the demo project. Please confirm your internet connection.

22.3.12. MODBUS Server Communication

		Read(R)/	Write(W)/	Control(C)
Address	Description	Local HMI	Macro	Remote HMI
LB-9055	MODBUS server (COM 1) receives a request (when ON)	R	R	R
LB-9056	MODBUS server (COM 2) receives a request (when ON)	R	R	R
LB-9057	MODBUS server (COM 3) receives a request (when ON)	R	R	R
LB-9058	MODBUS server (ethernet) receives a request (when ON)	R	R	R
LB-12052	MODBUS server status (disabled when ON)	R/W	R/C	R/C
LW-9270	(16bit) : request's function code - MODBUS server (COM 1)	R	R	R
LW-9271	(16bit) : request's starting address - MODBUS server (COM 1)	R	R	R
LW-9272	(16bit) : request's quantity of registers - MODBUS server (COM 1)	R	R	R
LW-9275	(16bit) : request's function code - MODBUS server (COM 2)	R	R	R
LW-9276	(16bit) : request's starting address - MODBUS server (COM 2)	R	R	R
LW-9277	(16bit) : request's quantity of registers - MODBUS server	R	R	R





	(COM 2)			
LW-9280	(16bit) : request's function code - MODBUS server (COM 3)	R	R	R
LW-9281	(16bit) : request's starting address - MODBUS server (COM 3)	R	R	R
LW-9282	(16bit) : request's quantity of registers - MODBUS server (COM 3)	R	R	R
LW-9285	(16bit) : request's function code - MODBUS server (ethernet)	R	R	R
LW-9286	(16bit) : request's starting address - MODBUS server (ethernet)	R	R	R
LW-9287	(16bit) : request's quantity of registers - MODBUS server (ethernet)	R	R	R
LW-9288	(16bit) : last error code - MODBUS server (ethernet)	R	R	R
LW-9541	(16bit) : MODBUS/ASCII server station no. (COM 1)	R/W	R/C	R/C
LW-9542	(16bit) : MODBUS/ASCII server station no. (COM 2)	R/W	R/C	R/C
LW-9543	(16bit) : MODBUS/ASCII server station no. (COM 3)	R/W	R/C	R/C
LW-9544	(16bit) : MODBUS/ASCII server station no. (ethernet)	R/W	R/C	R/C
LW-9570	(32bit) : received data count (bytes) (COM 1 MODBUS server)	R	R	R
LW-9572	(32bit) : received data count (bytes) (COM 2 MODBUS server)	R	R	R
LW-9574	(32bit) : received data count (bytes) (COM 3 MODBUS server)	R	R	R
LW-9576	(32bit) : received data count (bytes) (Ethernet MODBUS server)	R	R	R

22.3.13. Communication Parameter Settings

Address	Description	Read(R)/Write(W)/Control(C)			
		Local HMI	Macro	Remote HMI	
LB-9030	update COM 1 communication parameters (set ON)	R/W	R/C	R/C	
LB-9031	update COM 2 communication parameters (set ON)	R/W	R/C	R/C	
LB-9032	update COM 3 communication parameters (set ON)	R/W	R/C	R/C	
LB-9065	disable/enable COM 1 broadcast station no.	R/W	R/C	R/C	
LB-9066	disable/enable COM 2 broadcast station no.	R/W	R/C	R/C	
LB-9067	disable/enable COM 3 broadcast station no.	R/W	R/C	R/C	
LW-9550	(16bit) : COM 1 mode(0:RS232,1:RS485 2W,2:RS485 4W)	R/W	R/C	R/C	





LW-9551	(16bit) : COM 1 baud rate			
	(7:1200,8:2400,0:4800,1:9600,10:14400,	R/W	R/C	R/C
	2:19200,11:28800,3:38400,4:57600,)			
LW-9552	(16bit) : COM 1 databits (7 : 7 bits, 8 : 8 bits)	R/W	R/C	R/C
LW-9553	(16bit) : COM 1 parity (0:none, 1:even, 2:odd, 3:mark, 4:space)	R/W	R/C	R/C
LW-9554	(16bit) : COM 1 stop bits (1 : 1 bit, 2 : 2 bits)	R/W	R/C	R/C
LW-9555	(16bit) : COM 2 mode(0:RS232,1:RS485 2W,2:RS485 4W)	R/W	R/C	R/C
LW-9556	(16bit) : COM 2 baud rate			
	(7:1200,8:2400,0:4800,1:9600,10:14400,	R/W	R/C	R/C
	2:19200,11:28800,3:38400,4:57600,)			
LW-9557	(16bit) : COM 2 databits (7 : 7 bits, 8 : 8 bits)	R/W	R/C	R/C
LW-9558	(16bit) : COM 2 parity (0:none, 1:even, 2:odd, 3:mark, 4:space)	R/W	R/C	R/C
LW-9559	(16bit) : COM 2 stop bits (1 : 1 bit, 2 : 2 bits)	R/W	R/C	R/C
LW-9560	(16bit) : COM 3 mode(0:RS232,1:RS485 2W)	R/W	R/C	R/C
LW-9561	(16bit) : COM 3 baud rate			
	(7:1200,8:2400,0:4800,1:9600,10:14400,	R/W	R/C	R/C
	2:19200,11:28800,3:38400,4:57600,)			
LW-9562	(16bit) : COM 3 databits (7 : 7 bits, 8 : 8 bits)	R/W	R/C	R/C
LW-9563	(16bit) : COM 3 parity (0:none, 1:even, 2:odd, 3:mark,	R/W	R/C	R/C
	4:space)	r, vv	куC	R/C
LW-9564	(16bit) : COM 3 stop bits (1 : 1 bit, 2 : 2 bits)	R/W	R/C	R/C
LW-9565	(16bit) : COM 1 broadcast station no.	R/W	R/C	R/C
LW-9566	(16bit) : COM 2 broadcast station no.	R/W	R/C	R/C
LW-9567	(16bit) : COM 3 broadcast station no.	R/W	R/C	R/C
LW-10500	(16bit) : PLC 1 timeout (unit : 100ms)	R/W	R/C	R/C
LW-10501	(16bit) : PLC 1 turn around delay (unit : ms)	R/W	R/C	R/C
LW-10502	(16bit) : PLC 1 send ACK delay (unit : ms)	R/W	R/C	R/C
LW-10503	(16bit) : PLC 1 parameter 1	R/W	R/C	R/C
LW-10504	(16bit) : PLC 1 parameter 2	R/W	R/C	R/C
LW-10505	(16bit) : PLC 2 timeout (unit : 100ms)	R/W	R/C	R/C
LW-10506	(16bit) : PLC 2 turn around delay (unit : ms)	R/W	R/C	R/C
LW-10507	(16bit) : PLC 2 send ACK delay (unit : ms)	R/W	R/C	R/C
LW-10508	(16bit) : PLC 2 parameter 1	R/W	R/C	R/C
LW-10509	(16bit) : PLC 2 parameter 2	R/W	R/C	R/C
LW-10510	(16bit) : PLC 3 timeout (unit : 100ms)	R/W	R/C	R/C



LW-10511	(16bit) : PLC 3 turn around delay (unit : ms)	R/W	R/C	R/C
LW-10512	(16bit) : PLC 3 send ACK delay (unit : ms)	R/W	R/C	R/C
LW-10513	(16bit) : PLC 3 parameter 1	R/W	R/C	R/C
LW-10514	(16bit) : PLC 3 parameter 2	R/W	R/C	R/C
LW-10515	(16bit) : PLC 4 timeout (unit : 100ms)	R/W	R/C	R/C
LW-10516	(16bit) : PLC 4 turn around delay (unit : ms)	R/W	R/C	R/C
LW-10517	(16bit) : PLC 4 send ACK delay (unit : ms) (SIEMENS S7/400 Link type)	R/W	R/C	R/C
LW-10518	(16bit) : PLC 4 parameter 1 (SIEMENS S7/400 rack)	R/W	R/C	R/C
LW-10519	(16bit) : PLC 4 parameter 2 (SIEMENS S7/400 CPU slot)	R/W	R/C	R/C
LW-10520	(16bit) : PLC 5 timeout (unit : 100ms)	R/W	R/C	R/C
LW-10521	(16bit) : PLC 5 turn around delay (unit : ms)	R/W	R/C	R/C
LW-10522	(16bit) : PLC 5 send ACK delay (unit : ms) (SIEMENS S7/400 Link type)	R/W	R/C	R/C
LW-10523	(16bit) : PLC 5 parameter 1 (SIEMENS S7/400 rack)	R/W	R/C	R/C
LW-10524	(16bit) : PLC 5 parameter 2 (SIEMENS S7/400 CPU slot)	R/W	R/C	R/C
LW-10525	(16bit) : PLC 6 timeout (unit : 100ms)	R/W	R/C	R/C
LW-10526	(16bit) : PLC 6 turn around delay (unit : ms)	R/W	R/C	R/C
LW-10527	(16bit) : PLC 6 send ACK delay (unit : ms) (SIEMENS S7/400 Link type)	R/W	R/C	R/C
LW-10528	(16bit) : PLC 6 parameter 1 (SIEMENS S7/400 rack)	R/W	R/C	R/C
LW-10529	(16bit) : PLC 6 parameter 2 (SIEMENS S7/400 CPU slot)	R/W	R/C	R/C
LW-10530	(16bit) : PLC 7 timeout (unit : 100ms)	R/W	R/C	R/C
LW-10531	(16bit) : PLC 7 turn around delay (unit : ms)	R/W	R/C	R/C
LW-10532	(16bit) : PLC 7 send ACK delay (unit : ms) (SIEMENS S7/400 Link type)	R/W	R/C	R/C
	(16bit) : PLC 7 parameter 1 (SIEMENS S7/400 rack)	R/W	R/C	R/C
LW-10533				
	(16bit) : PLC 7 parameter 2 (SIEMENS S7/400 CPU slot)	R/W	R/C	R/C
LW-10534	(16bit) : PLC 7 parameter 2 (SIEMENS S7/400 CPU slot) (16bit) : PLC 8 timeout (unit : 100ms)	R/W R/W	R/C R/C	-
LW-10534 LW-10535				R/C
LW-10534 LW-10535 LW-10536	 (16bit) : PLC 8 timeout (unit : 100ms) (16bit) : PLC 8 turn around delay (unit : ms) (16bit) : PLC 8 send ACK delay (unit : ms) (SIEMENS S7/400 	R/W	R/C	R/C R/C
LW-10534 LW-10535 LW-10536 LW-10537	 (16bit) : PLC 8 timeout (unit : 100ms) (16bit) : PLC 8 turn around delay (unit : ms) (16bit) : PLC 8 send ACK delay (unit : ms) (SIEMENS S7/400 Link type) 	R/W R/W R/W	R/C R/C R/C	R/C R/C R/C R/C
LW-10534 LW-10535 LW-10537 LW-10538	 (16bit) : PLC 8 timeout (unit : 100ms) (16bit) : PLC 8 turn around delay (unit : ms) (16bit) : PLC 8 send ACK delay (unit : ms) (SIEMENS S7/400 Link type) (16bit) : PLC 8 parameter 1 (SIEMENS S7/400 rack) 	R/W R/W R/W	R/C R/C R/C R/C	R/C R/C R/C R/C R/C
LW-10533 LW-10535 LW-10536 LW-10537 LW-10538 LW-10539	 (16bit) : PLC 8 timeout (unit : 100ms) (16bit) : PLC 8 turn around delay (unit : ms) (16bit) : PLC 8 send ACK delay (unit : ms) (SIEMENS S7/400 Link type) (16bit) : PLC 8 parameter 1 (SIEMENS S7/400 rack) (16bit) : PLC 8 parameter 2 (SIEMENS S7/400 CPU slot) 	R/W R/W R/W R/W	R/C R/C R/C R/C R/C	R/C R/C R/C R/C R/C R/C
LW-10534 LW-10535 LW-10537 LW-10538	 (16bit) : PLC 8 timeout (unit : 100ms) (16bit) : PLC 8 turn around delay (unit : ms) (16bit) : PLC 8 send ACK delay (unit : ms) (SIEMENS S7/400 Link type) (16bit) : PLC 8 parameter 1 (SIEMENS S7/400 rack) 	R/W R/W R/W	R/C R/C R/C R/C	R/C R/C R/C R/C R/C



System Registers

22-2	19
------	----

LW-10658	(16bit) : PLC 32 parameter 1	R/W	R/C	R/C
LW-10659	(16bit) : PLC 32 parameter 2	R/W	R/C	R/C



		Read(R)/	Write(W)/	Control(C)
Address	Description	Local HMI	Macro	Remote HMI
LB-9150	auto. connection for PLC 1 (COM 1) (when ON)	R/W	R/C	R/C
LB-9151	auto. connection for PLC 2 (COM 2) (when ON)	R/W	R/C	R/C
LB-9152	auto. connection for PLC 3 (COM 3) (when ON)	R/W	R/C	R/C
LB-9200	PLC 1 status (SN0, COM 1), set on to retry connection.	R/W	R/C	R/C
LB-9201	PLC 1 status (SN1, COM 1), set on to retry connection	R/W	R/C	R/C
LB-9202	PLC 1 status (SN2, COM 1), set on to retry connection	R/W	R/C	R/C
LB-9203	PLC 1 status (SN3, COM 1), set on to retry connection	R/W	R/C	R/C
LB-9204	PLC 1 status (SN4, COM 1), set on to retry connection	R/W	R/C	R/C
LB-9205	PLC 1 status (SN5, COM 1), set on to retry connection	R/W	R/C	R/C
LB-9206	PLC 1 status (SN6, COM 1), set on to retry connection	R/W	R/C	R/C
LB-9207	PLC 1 status (SN7, COM 1), set on to retry connection	R/W	R/C	R/C
LB-9455	PLC 1 status (SN255, COM 1), set on to retry connection	R/W	R/C	R/C
LB-9500	PLC 2 status (SN0, COM 2), set on to retry connection.	R/W	R/C	R/C
LB-9501	PLC 2 status (SN1, COM 2), set on to retry connection	R/W	R/C	R/C
LB-9502	PLC 2 status (SN2, COM 2), set on to retry connection	R/W	R/C	R/C
LB-9503	PLC 2 status (SN3, COM 2), set on to retry connection	R/W	R/C	R/C
LB-9504	PLC 2 status (SN4, COM 2), set on to retry connection	R/W	R/C	R/C
LB-9505	PLC 2 status (SN5, COM 2), set on to retry connection	R/W	R/C	R/C
LB-9506	PLC 2 status (SN6, COM 2), set on to retry connection	R/W	R/C	R/C
LB-9507	PLC 2 status (SN7, COM 2), set on to retry connection	R/W	R/C	R/C
LB-9755	PLC 2 status (SN255, COM 2), set on to retry connection	R/W	R/C	R/C
LB-9800	PLC 3 status (SN0, COM 3), set on to retry connection	R/W	R/C	R/C
LB-9801	PLC 3 status (SN1, COM 3), set on to retry connection	R/W	R/C	R/C
LB-9802	PLC 3 status (SN2, COM 3), set on to retry connection	R/W	R/C	R/C
LB-9803	PLC 3 status (SN3, COM 3), set on to retry connection	R/W	R/C	R/C
LB-9804	PLC 3 status (SN4, COM 3), set on to retry connection	R/W	R/C	R/C
LB-9805	PLC 3 status (SN5, COM 3), set on to retry connection	R/W	R/C	R/C
LB-9806	PLC 3 status (SN6, COM 3), set on to retry connection	R/W	R/C	R/C
LB-9807	PLC 3 status (SN7, COM 3), set on to retry connection	R/W	R/C	R/C
LB-10055	PLC 3 status (SN255, COM 3), set on to retry connection	R/W	R/C	R/C
LB-12030	COM 1 status (OFF : normal, ON : open failed) *Note 1	R	R	R
LB-12031	COM 2 status (OFF : normal, ON : open failed)	R	R	R
LB-12032	COM 3 status (OFF : normal, ON : open failed)	R	R	R





LB-12033	COM 4 status (OFF : normal, ON : open failed)	R	R	R
LB-12034	COM 5 status (OFF : normal, ON : open failed)	R	R	R
LB-12035	COM 6 status (OFF : normal, ON : open failed)	R	R	R
LB-12036	COM 7 status (OFF : normal, ON : open failed)	R	R	R
LB-12037	COM 8 status (OFF : normal, ON : open failed)	R	R	R
LB-12038	COM 9 status (OFF : normal, ON : open failed)	R	R	R
LW-9351	(16bit) : pending command no. in PLC 1 (COM 1)	R	R	R
LW-9352	(16bit) : pending command no. in PLC 2 (COM 2)	R	R	R
LW-9353	(16bit) : pending command no. in PLC 3 (COM 3)	R	R	R

Note

1. The ON state of COM is for checking if COM is occupied by other program during simulation on PC.

22.3.15. Communication Status and Control with PLC (Ethernet)

		Read(R)/	Read(R)/Write(W)/Control(C)			
Address	Description	Local HMI	Macro	Remote HMI		
LB-9153	auto. connection for PLC 4 (ethernet) (when ON)	R/W	R/C	R/C		
LB-9154	auto. connection for PLC 5 (ethernet) (when ON)	R/W	R/C	R/C		
LB-9155	auto. connection for PLC 6 (ethernet) (when ON)	R/W	R/C	R/C		
LB-9156	auto. connection for PLC 7 (ethernet) (when ON)	R/W	R/C	R/C		
LB-9157	auto. connection for PLC 8 (ethernet) (when ON)	R/W	R/C	R/C		
LB-9158	auto. connection for PLC 9 (ethernet) (when ON)	R/W	R/C	R/C		
LB-9189	auto. connection for PLC 40 (ethernet) (when ON)	R/W	R/C	R/C		
LB-10070	forced to reconnect PLC 4 (ethernet) when IP or system parameters changed on-line (set ON)	R/W	R/C	R/C		
LB-10071	forced to reconnect PLC 5 (ethernet) when IP or system parameters changed on-line (set ON)	R/W	R/C	R/C		
LB-10072	forced to reconnect PLC 6 (ethernet) when IP or system parameters changed on-line (set ON)	R/W	R/C	R/C		
LB-10073	forced to reconnect PLC 7 (ethernet) when IP or system parameters changed on-line (set ON)	R/W	R/C	R/C		
LB-10074	forced to reconnect PLC 8 (ethernet) when IP or system parameters changed on-line (set ON)	R/W	R/C	R/C		
LB-10075	forced to reconnect PLC 9 (ethernet) when IP or system parameters changed on-line (set ON)	R/W	R/C	R/C		
LB-10099	forced to reconnect PLC 33 (ethernet) when IP or system	R/W	R/C	R/C		



	parameters changed on-line (set ON)			
LB-10100	PLC 4 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-10400	PLC 5 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-10700	PLC 6 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11000	PLC 7 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11300	PLC 8 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11600	PLC 9 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11900	PLC 10 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11901	PLC 11 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11902	PLC 12 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11903	PLC 13 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11904	PLC 14 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11905	PLC 15 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11906	PLC 16 status (ethernet), set on to retry connection	R/W	R/C	R/C
LB-11939	PLC 49 status (ethernet), set on to retry connection	R/W	R/C	R/C
LW-9354	(16bit) : pending command no. in PLC 4 (ethernet)	R	R	R
LW-9355	(16bit) : pending command no. in PLC 5 (ethernet)	R	R	R
LW-9356	(16bit) : pending command no. in PLC 6 (ethernet)	R	R	R
LW-9357	(16bit) : pending command no. in PLC 7 (ethernet)	R	R	R
LW-9389	(16bit) : pending command no. in PLC 39 (ethernet)	R	R	R
LW-9600	(16bit) : PLC 4's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9601	(16bit) : PLC 4's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9602	(16bit) : PLC 4's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9603	(16bit) : PLC 4's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9604	(16bit) : PLC 4's port no.	R/W	R/C	R/C
LW-9605	(16bit) : PLC 5's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9606	(16bit) : PLC 5's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9607	(16bit) : PLC 5's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9608	(16bit) : PLC 5's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9609	(16bit) : PLC 5's port no.	R/W	R/C	R/C
LW-9610	(16bit) : PLC 6's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9611	(16bit) : PLC 6's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9612	(16bit) : PLC 6's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9613	(16bit) : PLC 6's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9614	(16bit) : PLC 6's port no.	R/W	R/C	R/C



LW-9615	(16bit) : PLC 7's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9616	(16bit) : PLC 7's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9617	(16bit) : PLC 7's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9618	(16bit) : PLC 7's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9619	(16bit) : PLC 7's port no.	R/W	R/C	R/C
LW-9620	(16bit) : PLC 8's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9621	(16bit) : PLC 8's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9622	(16bit) : PLC 8's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9623	(16bit) : PLC 8's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9624	(16bit) : PLC 8's port no.	R/W	R/C	R/C
LW-9625	(16bit) : PLC 9's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9626	(16bit) : PLC 9's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9627	(16bit) : PLC 9's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9628	(16bit) : PLC 9's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9629	(16bit) : PLC 9's port no.	R/W	R/C	R/C
LW-9765	(16bit) : PLC 37's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9766	(16bit) : PLC 37's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9767	(16bit) : PLC 37's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9768	(16bit) : PLC 37's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9769	(16bit) : PLC 37's port no.	R/W	R/C	R/C

22.3.16. Communication Status and Control with PLC (USB)

Address		Read(R)/	Write(W)/0	Control(C)
	Description	Local HMI	Macro	Remote HMI
LB-9190	auto. connection for PLC (USB) (when ON)	R/W	R/C	R/C
LB-9191	PLC status (USB), set on to retry connection	R/W	R/C	R/C
LW-9390	(16bit) : pending command no. in PLC (USB)	R	R	R



		Read(R)/	Write(W)/	Control(C)
Address	Description	Local HMI	Macro	Remote HMI
LB-12080	auto. connection for PLC (CAN Bus) (when ON)	R/W	R/C	R/C
LB-12081	PLC status (CAN Bus) set on to retry connection	R/W	R/C	R/C
LB-12100	pause CAN Bus device 1 communication (when ON)	R/W	R/C	R/C
LB-12101	pause CAN Bus device 2 communication (when ON)	R/W	R/C	R/C
LB-12102	pause CAN Bus device 3 communication (when ON)	R/W	R/C	R/C
LB-12103	pause CAN Bus device 4 communication (when ON)	R/W	R/C	R/C
LB-12104	pause CAN Bus device 5 communication (when ON)	R/W	R/C	R/C
LB-12105	pause CAN Bus device 6 communication (when ON)	R/W	R/C	R/C
LB-12106	pause CAN Bus device 7 communication (when ON)	R/W	R/C	R/C
LB-12107	pause CAN Bus device 8 communication (when ON)	R/W	R/C	R/C
LB-12108	pause CAN Bus device 9 communication (when ON)	R/W	R/C	R/C
LB-12109	pause CAN Bus device 10 communication (when ON)	R/W	R/C	R/C
LB-12354	pause CAN Bus device 255 communication (when ON)	R/W	R/C	R/C
LW-9392	(16bit) : pending command no. in PLC (CAN Bus)	R	R	R

22.3.17. Communication Status and Control with PLC (CAN Bus)

22.3.18. Communication Status and Control with Remote HMI

		Read(R)/	Write(W)/	Control(C)
Address	Description	Local HMI	HMI R/C R/C R/C R/C	Remote HMI
LB-9068	auto. connection for remote HMI 1 (when ON)	R/W	R/C	R/C
LB-9069	auto. connection for remote HMI 2 (when ON)	R/W	R/C	R/C
LB-9070	auto. connection for remote HMI 3 (when ON)	R/W	R/C	R/C
LB-9071	auto. connection for remote HMI 4 (when ON)	R/W	R/C	R/C
LB-9072	auto. connection for remote HMI 5 (when ON)	R/W	R/C	R/C
LB-9073	auto. connection for remote HMI 6 (when ON)	R/W	R/C	R/C
LB-9074	auto. connection for remote HMI 7 (when ON)	R/W	R/C	R/C
LB-9075	auto. connection for remote HMI 8 (when ON)	R/W	R/C	R/C
LB-9099	auto. connection for remote HMI 32 (when ON)	R/W	R/C	R/C
LB-9100	remote HMI 1 status (set on to retry connection)	R/W	R/C	R/C
LB-9101	remote HMI 2 status (set on to retry connection)	R/W	R/C	R/C
LB-9102	remote HMI 3 status (set on to retry connection)	R/W	R/C	R/C
LB-9103	remote HMI 4 status (set on to retry connection)	R/W	R/C	R/C
LB-9104	remote HMI 5 status (set on to retry connection)	R/W	R/C	R/C



22-24



LB-9105	remote HMI 6 status (set on to retry connection)	R/W	R/C	R/C
LB-9106	remote HMI 7 status (set on to retry connection)	R/W	R/C	R/C
LB-9107	remote HMI 8 status (set on to retry connection)	R/W	R/C	R/C
LB-9148	remote HMI 49 status (set on to retry connection)	R/W	R/C	R/C
LB-9149	forced to reconnect remote HMI when IP changed on-line (set ON)	R/W	R/C	R/C
LW-9800	(16bit) : remote HMI 1's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9801	(16bit) : remote HMI 1's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9802	(16bit) : remote HMI 1's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9803	(16bit) : remote HMI 1's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9804	(16bit) : remote HMI 1's port no.	R/W	R/C	R/C
LW-9805	(16bit) : remote HMI 2's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9806	(16bit) : remote HMI 2's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9807	(16bit) : remote HMI 2's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9808	(16bit) : remote HMI 2's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9809	(16bit) : remote HMI 2's port no.	R/W	R/C	R/C
LW-9810	(16bit) : remote HMI 3's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9811	(16bit) : remote HMI 3's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9812	(16bit) : remote HMI 3's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9813	(16bit) : remote HMI 3's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9814	(16bit) : remote HMI 3's port no.	R/W	R/C	R/C
LW-9815	(16bit) : remote HMI 4's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9816	(16bit) : remote HMI 4's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9817	(16bit) : remote HMI 4's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9818	(16bit) : remote HMI 4's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9819	(16bit) : remote HMI 4's port no.	R/W	R/C	R/C
LW-9820	(16bit) : remote HMI 5's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9821	(16bit) : remote HMI 5's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9822	(16bit) : remote HMI 5's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9823	(16bit) : remote HMI 5's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9824	(16bit) : remote HMI 5's port no.	R/W	R/C	R/C
LW-9825	(16bit) : remote HMI 6's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9826	(16bit) : remote HMI 6's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9827	(16bit) : remote HMI 6's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9828	(16bit) : remote HMI 6's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9829	(16bit) : remote HMI 6's port no.	R/W	R/C	R/C
LW-9830	(16bit) : remote HMI 7's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C



LW-9831	(16bit) : remote HMI 7's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9832	(16bit) : remote HMI 7's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9833	(16bit) : remote HMI 7's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9834	(16bit) : remote HMI 7's port no.	R/W	R/C	R/C
LW-9835	(16bit) : remote HMI 8's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9836	(16bit) : remote HMI 8's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9837	(16bit) : remote HMI 8's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9838	(16bit) : remote HMI 8's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9839	(16bit) : remote HMI 8's port no.	R/W	R/C	R/C
LW-9895	(16bit) : remote HMI 20's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9896	(16bit) : remote HMI 20's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9897	(16bit) : remote HMI 20's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9898	(16bit) : remote HMI 20's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9899	(16bit) : remote HMI 20's port no.	R/W	R/C	R/C
LW-9905	(16bit) : remote HMI 21's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9906	(16bit) : remote HMI 21's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9907	(16bit) : remote HMI 21's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9908	(16bit) : remote HMI 21's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9909	(16bit) : remote HMI 21's port no.	R/W	R/C	R/C
LW-9910	(16bit) : remote HMI 22's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9911	(16bit) : remote HMI 22's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9912	(16bit) : remote HMI 22's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9913	(16bit) : remote HMI 22's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9914	(16bit) : remote HMI 22's port no.	R/W	R/C	R/C
LW-9915	(16bit) : remote HMI 23's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9916	(16bit) : remote HMI 23's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9917	(16bit) : remote HMI 23's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9918	(16bit) : remote HMI 23's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9919	(16bit) : remote HMI 23's port no.	R/W	R/C	R/C
LW-9920	(16bit) : remote HMI 24's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9921	(16bit) : remote HMI 24's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9922	(16bit) : remote HMI 24's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9923	(16bit) : remote HMI 24's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9924	(16bit) : remote HMI 24's port no.	R/W	R/C	R/C
LW-9925	(16bit) : remote HMI 25's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9926	(16bit) : remote HMI 25's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9927	(16bit) : remote HMI 25's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C



LW-9928	(16bit) : remote HMI 25's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9929	(16bit) : remote HMI 25's port no.	R/W	R/C	R/C
LW-9930	(16bit) : remote HMI 26's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9931	(16bit) : remote HMI 26's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9932	(16bit) : remote HMI 26's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9933	(16bit) : remote HMI 26's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9934	(16bit) : remote HMI 26's port no.	R/W	R/C	R/C
LW-9935	(16bit) : remote HMI 27's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9936	(16bit) : remote HMI 27's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9937	(16bit) : remote HMI 27's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9938	(16bit) : remote HMI 27's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9939	(16bit) : remote HMI 27's port no.	R/W	R/C	R/C
LW-9940	(16bit) : remote HMI 28's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9941	(16bit) : remote HMI 28's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9942	(16bit) : remote HMI 28's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9943	(16bit) : remote HMI 28's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9944	(16bit) : remote HMI 28's port no.	R/W	R/C	R/C
LW-9945	(16bit) : remote HMI 29's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9946	(16bit) : remote HMI 29's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9947	(16bit) : remote HMI 29's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9948	(16bit) : remote HMI 29's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9949	(16bit) : remote HMI 29's port no.	R/W	R/C	R/C
LW-9950	(16bit) : remote HMI 30's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9951	(16bit) : remote HMI 30's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9952	(16bit) : remote HMI 30's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9953	(16bit) : remote HMI 30's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9954	(16bit) : remote HMI 30's port no.	R/W	R/C	R/C
LW-9955	(16bit) : remote HMI 31's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9956	(16bit) : remote HMI 31's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9957	(16bit) : remote HMI 31's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9958	(16bit) : remote HMI 31's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9959	(16bit) : remote HMI 31's port no.	R/W	R/C	R/C
LW-9960	(16bit) : remote HMI 32's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9961	(16bit) : remote HMI 32's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9962	(16bit) : remote HMI 32's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9963	(16bit) : remote HMI 32's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9964	(16bit) : remote HMI 32's port no.	R/W	R/C	R/C



LW-9995	(16bit) : remote HMI 39's IP0 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9996	(16bit) : remote HMI 39's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9997	(16bit) : remote HMI 39's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9998	(16bit) : remote HMI 39's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-9999	(16bit) : remote HMI 39's port no.	R/W	R/C	R/C

22.3.19. Communication Status and Control with Remote PLC

		Read(R)/	Write(W)/	Control(C)
Address	Description	Local HMI	Macro	Remote HMI
LW-10050	(16bit) : IPO of the HMI connecting to remote PLC 1 (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10051	(16bit) : IP1 of the HMI connecting to remote PLC 1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10052	(16bit) : IP2 of the HMI connecting to remote PLC 1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10053	(16bit) : IP3 of the HMI connecting to remote PLC 1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10054	(16bit) : port no. of the HMI connecting to remote PLC 1	R/W	R/C	R/C
LW-10055	(16bit) : IPO of the HMI connecting to remote PLC 2 (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10056	(16bit) : IP1 of the HMI connecting to remote PLC 2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10057	(16bit) : IP2 of the HMI connecting to remote PLC 2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10058	(16bit) : IP3 of the HMI connecting to remote PLC 2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10059	(16bit) : port no. of the HMI connecting to remote PLC 2	R/W	R/C	R/C
LW-10060	(16bit) : IPO of the HMI connecting to remote PLC 3 (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10061	(16bit) : IP1 of the HMI connecting to remote PLC 3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10062	(16bit) : IP2 of the HMI connecting to remote PLC 3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10063	(16bit) : IP3 of the HMI connecting to remote PLC 3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10064	(16bit) : port no. of the HMI connecting to remote PLC 3	R/W	R/C	R/C



LW-10065	(16bit) : IPO of the HMI connecting to remote PLC 4 (IP			
	address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10066	(16bit) : IP1 of the HMI connecting to remote PLC 4 (IP	- 4	- 1-	- /-
	address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10067	(16bit) : IP2 of the HMI connecting to remote PLC 4 (IP	D () M	D/0	D / C
	address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10068	(16bit) : IP3 of the HMI connecting to remote PLC 4 (IP	D (M)	D/C	D/C
	address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10069	(16bit) : port no. of the HMI connecting to remote PLC 4	R/W	R/C	R/C
LW-10205	(16bit) : IPO of the HMI connecting to remote PLC 32 (IP	D (M)	D/C	D/C
	address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10206	(16bit) : IP1 of the HMI connecting to remote PLC 32 (IP	D (M)	D/C	D/C
	address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10207	(16bit) : IP2 of the HMI connecting to remote PLC 32 (IP	R/W	R/C	R/C
	address = IP0:IP1:IP2:IP3)	K/ VV	K/C	K/C
LW-10208	(16bit) : IP3 of the HMI connecting to remote PLC 32 (IP	D (M)	D/C	D/C
	address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10209	(16bit) : port no. of the HMI connecting to remote PLC 32	R/W	R/C	R/C
LW-10300	(16bit) : remote PLC 1's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10301	(16bit) : remote PLC 1's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10302	(16bit) : remote PLC 1's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10303	(16bit) : remote PLC 1's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10304	(16bit) : remote PLC 1's port no.	R/W	R/C	R/C
LW-10305	(16bit) : remote PLC 2's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10306	(16bit) : remote PLC 2's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10307	(16bit) : remote PLC 2's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10308	(16bit) : remote PLC 2's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10309	(16bit) : remote PLC 2's port no.	R/W	R/C	R/C
LW-10310	(16bit) : remote PLC 3's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10311	(16bit) : remote PLC 3's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10312	(16bit) : remote PLC 3's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10313	(16bit) : remote PLC 3's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10314	(16bit) : remote PLC 3's port no.	R/W	R/C	R/C
LW-10315	(16bit) : remote PLC 4's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10316	(16bit) : remote PLC 4's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10317	(16bit) : remote PLC 4's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10318	(16bit) : remote PLC 4's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C



LW-10319	(16bit) : remote PLC 4's port no.	R/W	R/C	R/C
LW-10455	(16bit) : remote PLC 32's IPO (IP address = IPO:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10456	(16bit) : remote PLC 32's IP1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10457	(16bit) : remote PLC 32's IP2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10458	(16bit) : remote PLC 32's IP3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-10459	(16bit) : remote PLC 32's port no.	R/W	R/C	R/C

22.3.20. Local/Remote Operation Restrictions

Address		Read(R)/	Write(W)/0	Control(C)
	Description	Local HMI	Write(W)/Control(C) Macro Remote HMI R/C R/C R/C R/C R/C R/C R/C R/C R/C R/C R/C R/C	
LB-9044	disable remote control (when ON)	R/W	R/C	R/C
LB-9053	prohibit password remote-read operation (when ON)	R/W	R/C	R/C
LB-9054	prohibit password remote-write operation (when ON)	R/W	R/C	R/C
LB-9196	local HMI supports monitor function only (when ON)	R/W	R/C	R/C
LB-9197	support monitor function only for remote HMIs (when ON)	R/W	R/C	R/C
LB-9198	disable local HMI to trigger a MACRO (when ON)	R/W	R/C	R/C
LB-9199	disable remote HMI to trigger a MACRO (when ON)	R/W	R/C	R/C

22.3.21. Communication Error Codes

		Read(R)/	Write(W)/0	Control(C)
Address	Description	Local HMI	R F R F R F R F R F R F R F R F R F R F	Remote HMI
LW-9400	(16bit) : error code for PLC 1	R	R	R
LW-9401	(16bit) : error code for PLC 2	R	R	R
LW-9402	(16bit) : error code for PLC 3	R	R	R
LW-9403	(16bit) : error code for PLC 4	R	R	R
LW-9404	(16bit) : error code for PLC 5	R	R	R
LW-9405	(16bit) : error code for PLC 6	R	R	R
LW-9406	(16bit) : error code for PLC 7	R	R	R
LW-9407	(16bit) : error code for PLC 8	R	R	R
LW-9449	(16bit) : error code for PLC 50	R	R	R
LW-9490	(16bit) : error code for USB PLC	R	R	R
LW-9491	(16bit) : error code for CAN-Bus PLC	R	R	R

Note

1. A list of the explaination of PLC communication error codes:



Error Code	Cause of Communication Error
0	Normal
1	The device is busy and not yet ready to process a command.
2	Communication error due to unexpected reason.
3	The device does not exist.
4	The device using the specified station number does not exist.
5	Incorrect address format.
6	Read/Write unsupported address.
7	The driver of the device does not exist.
8	The COM port does not exist.
9	Incorrect IP address or unable to connect the device.
10	Checksum error.
11	Unidentified command.
12	Ignore
20	The USB device is improperly connected.
21	The CAN Bus device is improperly connected.
22	No reply from the device.
23	Insufficient data read from the device during timeout.
24	The Conversion Tag used by the object does not exist or the content is incorrect.
25	HMI is not accepting any commands from a remote HMI.
251	Read/Write exceeding number of words from/to the register of the MODBUS device.
252	The MODBUS device replies incorrect data format.
253	MODBUS device checksum error.



Address	Description	Read(R)/Write(W)/Control(C)			
		Local HMI	Macro	Remote HMI	
LW-9300	(16bit) : driver ID of local PLC 1	R	R	R	
LW-9301	(16bit) : driver ID of local PLC 2	R	R	R	
LW-9302	(16bit) : driver ID of local PLC 3	R	R	R	
LW-9303	(16bit) : driver ID of local PLC 4	R	R	R	
LW-9331	(16bit) : driver ID of local PLC 32	R	R	R	

22.3.23. DLT645 Controller

		Read(R)/Write(W)/Control(C)		
Address	Description	Local HMI	Macro	Remote HMI
LW-10700	(4 words) : DLT_645 operator (COM 1)	R/W	R/C	R/C
LW-10704	(4 words) : DLT_645 password (COM 1)	R/W	R/C	R/C
LW-10708	(6 words) : DLT_645 address (COM 1)	R/W	R/C	R/C
LW-10715	(4 words) : DLT_645 operator (COM 2)	R/W	R/C	R/C
LW-10719	(4 words) : DLT_645 password (COM 2)	R/W	R/C	R/C
LW-10723	(6 words) : DLT_645 address (COM 2)	R/W	R/C	R/C
LW-10730	(4 words) : DLT_645 operator (COM 3)	R/W	R/C	R/C
LW-10734	(4 words) : DLT_645 password (COM 3)	R/W	R/C	R/C
LW-10738	(6 words) : DLT_645 address (COM 3)	R/W	R/C	R/C

22.3.24. [PLC No Response] Window Control

		Read(R)/	Write(W)/0	Control(C)
Address	Description	Local HMI	Macro	Remote HMI
LB-9192	disable USB PLC's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-11960	disable PLC 1's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-11961	disable PLC 2's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-11962	disable PLC 3's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-11963	disable PLC 4's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-11964	disable PLC 5's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-11965	disable PLC 6's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-11966	disable PLC 7's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-11967	disable PLC 8's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-12023	disable PLC 64's "PLC No Response" dialog (when ON)	R/W	R/C	R/C
LB-12082	disable CAN Bus device's "PLC No Response" dialog (when ON)	R/W	R/C	R/C





Address		Read(R)/Write(W)/Control(C)			
	Description	Local HMI	Macro	Remote HMI	
LB-9013	FS window control[hide(ON)/show(OFF)]	R/W	R/C	R/C	
LB-9014	FS button control[hide(ON)/show(OFF)]	R/W	R/C	R/C	
LB-9015	FS window/button control[hide(ON)/show(OFF)]	R/W	R/C	R/C	

22.3.26. EasyAccess

Address	Description	Read(R)/Write(W)/Control(C)			
		Local HMI	Macro	Remote HMI	
LB-9051	disconnect (set OFF)/connect (set ON) EasyAccess server	R/W	R/C	R/C	
LB-9052	status of connecting to EasyAccess server	R	R	R	
0					

For more information about EasyAccess, please visit <u>http://www.ihmi.net/</u>.

Lick the icon to download the demo project. Please confirm your internet connection.

22.3.27. EasyAccess 2.0

	Description	Read(R)/	Read(R)/Write(W)/Control(C)			
Address		Local HMI	Macro	Remote HMI		
LW-10820	(16bit) : disable (set 0)/enable (set 1) (EasyAccess 2.0)	R/W	R/C	R/C		
LW-10821	(5 words) : session ID (EasyAccess 2.0)	R/W	R/C	R/C		
LW-10826	(2 words) : password (EasyAccess 2.0)	R/W	R/C	R/C		
LW-10828	(16bit) : execution status (EasyAccess 2.0)	R	R	R		
LW-10829	(16bit) : the last error code (EasyAccess 2.0)	R	R	R		
LW-11170	(16bit) : Proxy Disable/Enable (0:disable, 1:enable) (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11171	(16bit) : Proxy Type (0:HTTP, 1:SOCKSv4, 2:SOCKSv5) (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11172	(16bit) : Proxy Server IPO (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11173	(16bit) : Proxy Server IP1 (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11174	(16bit) : Proxy Server IP2 (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11175	(16bit) : Proxy Server IP3 (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11176	(16bit) : Proxy Server Port (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11177	(16bit) : Proxy authentication (0:disable, 1:enable) (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11178	(16 words) : Proxy username (EasyAccess 2.0)	R/W	R/C	R/C		
LW-11194	(16 words) : Proxy password (EasyAccess 2.0)	R/W	R/C	R/C		



LW-11210	(20 words) : Hardware key (EasyAccess 2.0)	R	R	R
LW-11296	(16bit) : Location of EasyAccess 2.0 server (0 : Global, 1 : China)	R	R	R

22.3.28. Remote Print/Backup Server

		Read(R)/Write(W)/Control(C)			
Address	Description	Local HMI	Macro	Remote HMI	
LB-10069	forced to reconnect remote printer/backup server when IP changed on-line (set ON)	R/W	R/C	R/C	
LB-12040	remote printer/backup server disconnection alarm (when ON)	R	R	R	
LW-9770	(16bit) : remote printer/backup server IP0 (IP0:IP1:IP2:IP3)	R/W	R/C	R/C	
LW-9771	(16bit) : remote printer/backup server IP1 (IP0:IP1:IP2:IP3)	R/W	R/C	R/C	
LW-9772	(16bit) : remote printer/backup server IP2 (IP0:IP1:IP2:IP3)	R/W	R/C	R/C	
LW-9773	(16bit) : remote printer/backup server IP3 (IP0:IP1:IP2:IP3)	R/W	R/C	R/C	
LW-9774	(6 words) : remote printer/backup server user name *Note 1	R/W	R/C	R/C	
LW-9780	(6 words) : remote printer/backup server password *Note 1	R/W	R/C	R/C	

Note

1. When change settings using LW-9774 and LW-9780, please reboot HMI for the new settings to take effect.

Lick the icon to download the demo project. Please confirm your internet connection.

22.3.29. Pass-Through Settings

		Read(R)/Write(W)/Control(C)			
Address	Description	Local HMI	Macro	Remote HMI	
LW-9901	(16bit) : pass-through source COM port (1~3 : COM 1~COM 3)	R/W	R/C	R/C	
LW-9902	(16bit) : pass-through destination COM port (1~3 : COM 1~COM 3)	R/W	R/C	R/C	
LW-9903	(16bit) : pass-through control (0 : normal, 1 : pause, 2 : stop communications between HMI and PLC when executing pass-through)	R/W	R/C	R/C	
LW-9904	(16bit) : pass-through server port no. (2000~2100)	R/W	R/C	R/C	



System Registers

LW-10850	(16bit) : disable/enable (0 : disable, 1 : normal, 2 : IP	R/W	R/C	R/C	
	limited) (siemens pass-through)	,	,		
LW-10851	(16bit) : destination COM port (siemens pass-through)	R/W	R/C	R/C	
LW-10852	(16bit) : destination PLC station no. (siemens pass-through)	R/W	R/C	R/C	
LW-10853	(16bit) : communication protocol (0 : invalid, 1 : PPI, 2 :	5.444	D/0	D/0	
	MPI) (siemens pass-through)	R/W	R/C	R/C	
LW-10854	(16bit) : IPO of connecting client (IP address =	5 /h./	5/6	D / 0	
	IP0:IP1:IP2:IP3) (siemens pass-through)	R/W	R/C	R/C	
LW-10855	(16bit) : IP1 of connecting client (IP address =	- 4	- 1-	- /-	
	IPO:IP1:IP2:IP3) (siemens pass-through)	R/W	R/C	R/C	
LW-10856	(16bit) : IP2 of connecting client (IP address =	R/W		- /-	
	IPO:IP1:IP2:IP3) (siemens pass-through)		R/C	R/C	
LW-10857	(16bit) : IP3 of connecting client (IP address =	R/W	R/W R/C	R/C	
	IP0:IP1:IP2:IP3) (siemens pass-through)				
LW-10858	(16bit) : IPO of designated client (IP address =	R/W R/C			- /-
	IP0:IP1:IP2:IP3) (siemens pass-through)		R/C	R/C	
LW-10859	(16bit) : IP1 of designated client (IP address =	- 4	5/6	5/2	
	IP0:IP1:IP2:IP3) (siemens pass-through)	R/W	R/C	R/C	
LW-10860	(16bit) : IP2 of designated client (IP address =	5 /h./	5/6	5/2	
	IP0:IP1:IP2:IP3) (siemens pass-through)	R/W	R/C	R/C	
LW-10861	(16bit) : IP3 of designated client (IP address =	- 4	- /-	- /-	
	IP0:IP1:IP2:IP3) (siemens pass-through)	R/W	R/W R/C	R/C	
LW-10862	(16bit) : connection status (0 : ready, 1 : client connecting)		2	_	
	(siemens pass-through)	R	R	R	
LW-10863	(16bit) : execution status (0 : normal, 1 : error) (siemens	-	5	-	
	pass-through)	R	R	R	
LW-10864	(16bit) : the last error (siemens pass-through)	R	R	R	

For more information about Siemens pass-through feature, see "29 Pass-through".

Lick the icon to download the demo project. Please confirm your internet connection.

22.3.30. VNC Control

	Address		Read(R)/Write(W)/Control(C)			
Ac		Description	Local HMI	Macro	Remote HMI	
LB-	-12088	enable VNC monitor mode (when ON) *Note 1	R/W	R/C	R/C	
LB-	-12089	VNC pass word free (when ON) *Note 1	R/W	R/C	R/C	

WE!NTEK

LB-12090	a VNC client connecting to HMI (when ON)(OS version 20120621 or later supports only)	R	R	R
LB-12091	disable auto-logout function when a VNC client connecting to HMI (when ON)(OS version 20120621 or later supports only)	R/W	R/C	R/C
LB-12092	enable VNC (set ON), disable VNC (set OFF)	NC (set ON), disable VNC (set OFF) R/W R/C		R/C
LB-12093	VNC connection mode (OFF: single connection, ON: multi connection) (OS version 2013.05.09 or later support)*Note1	R/W	R/C	R/C
LW-9530	(8 words) : VNC server password			R/C

Note

1. To change VNC mode, use LB-12092 to stop and then restart VNC to update the setting.

22.3.31. Project Key and HMI Key

		Read(R)/Write(W)/Control(C)			
Address	Description	Local HMI			
LB-9046	046 project key is different from HMI key (when ON)		R	R	
LW-9046	7-9046 (32bit) : HMI key *Note 1 R/W		R/C	R	

Note

- 1. When change HMI Key using LW-9046, please reboot HMI for the new settings to take effect.
- Lick the icon to download the demo project. Please confirm your internet connection.

22.3.32. USB Security Key

		Read(R)/Write(W)/Control(C)		
Address	Address Description		Macro	Remote HMI
LW-11160	(16bit) : start time of USB security key - year	R	R	R
LW-11161	(16bit) : start time of USB security key - month	R	R	R
LW-11162	162 (16bit) : start time of USB security key - day		R	R
LW-11163	(16bit) : start time of USB security key - hour	R	R	R
LW-11164	(16bit) : start time of USB security key - minute	R	R	R
LW-11165	(16bit) : expiration time of USB security key - year R		R	R
LW-11166	(16bit) : expiration time of USB security key - month	R	R	R
LW-11167	(16bit) : expiration time of USB security key - day	R	R	R
LW-11168	(16bit) : expiration time of USB security key - hour	R	R	R
LW-11169	(16bit) : expiration time of USB security key - minute	R	R	R





22.3.33. User Name and Password

				Control(C)
Address	Description			Remote HMI
LB-9050	.B-9050 user logout		с	С
LB-9060	password error	R	R	R
LB-9061	update password (set ON)	W	С	с
LB-12056	the user touches an unauthorized object (when ON)	R/W	R/C	R/C
LW-9082	(16bit) : auto logout time (unit : minute, 0 : disable the function)	R/W	R/C	R/C
LW-9219	(16bit) : user no. (1~12)	R/W	R/C	R/C
LW-9220	(32bit) : password	R/W	R/C	R/C
LW-9222	(16bit) : classes can be operated for current user (bit 0:A, bit 1:B,bit 2:C,)	R	R	R
PLW-9222	(16bit) : classes can be operated for current user (bit 0:A, bit 1:B,bit 2:C,)	R	R	R
LW-9500	(32bit) : user 1's password	R/W	R/C	R/C
LW-9502	(32bit) : user 2's password	R/W	R/C	R/C
LW-9504	(32bit) : user 3's password	R/W	R/C	R/C
LW-9506	(32bit) : user 4's password	R/W	R/C	R/C
LW-9508	(32bit) : user 5's password	R/W	R/C	R/C
LW-9510	(32bit) : user 6's password	R/W	R/C	R/C
LW-9512	(32bit) : user 7's password	R/W	R/C	R/C
LW-9514	(32bit) : user 8's password	R/W	R/C	R/C
LW-9516	(32bit) : user 9's password	R/W	R/C	R/C
LW-9518	(32bit) : user 10's password	R/W	R/C	R/C
LW-9520	(32bit) : user 11's password	R/W	R/C	R/C
LW-9522	(32bit) : user 12's password	R/W	R/C	R/C
LW-10754	(8 words) : current user name *Note 1	R	R	R
PLW-10754	(8 words) : current user name *Note 1	R	R	R

Note

1. Only for [Security] » [Enhanced security mode].

Lick the icon to download the demo project. Please confirm your internet connection.



22.3.34. Macro

		Read(R)/Write(W)/C Local Macro HMI		Control(C)	
Address	Description			Remote HMI	
LB-9059 disable macro TRACE function (when ON)*Note1		R/W	R/C	R/C	
LW-10900	(16bit) : macro 0 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R	R	R	
LW-10901					
LW-10902	(16bit) : macro 2 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R	R	R	
LW-10903	(16bit) : macro 3 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R	R	R	
LW-10904	(16bit) : macro 4 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R	R	R	
LW-10905	(16bit) : macro 5 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal endR(exceed array size))(exceed array size)		R	R	
LW-10906	(16bit) : macro 6 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R	R	R	
LW-10907	(16bit) : macro 7 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R	R	R	
LW-10908	(16bit) : macro 8 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R R R		R	
LW-10909	(16bit) : macro 9 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R R R		R	
LW-11154	(16bit) : macro 254 status (0:ready, 3:executing, 5:waiting response, 9:waiting sync, 17:delay, 32:abnormal end (exceed array size))	R	R	R	

Note

1. LB-9059: Disable macro trace function.

Lick the icon to download the demo project. Please confirm your internet connection.



22.3.35. Input Object Function

		Read(R)/Write(W)/Control(C		
Address	ddress Description		Macro	Remote HMI
LW-9002	(32bit-float) : input high limit	R	R	R
LW-9004	(32bit-float) : input low limit	R	R	R
LW-9052	(32bit-float) : the previous input value of the numeric input object	R	R	R
PLW-9052	(32bit-float) : the previous input value of the numeric input object	R	R	R
LW-9150	(32 words) : keyboard's input data (ASCII)	R	R	R
LW-9540	(16bit) : reserved for caps lock	R/W	R/C	R/C

22.3.36. Time Sync./Daylight Saving Time

		Read(R)/	ad(R)/Write(W)/Control		
Address	Description	Local HMI	Macro	Remote HMI	
LB-12055	failed to execute time synchronization (when ON)	R	R	R	
LB-12355	daylight saving time period (when ON)	R	R	R	
LW-11260	(16bit) : enable/disable the daylight saving time (DST)	R/W	R/C	R/C	
	(0:disable, 1:enable)	K/ VV	R/C	K/C	
LW-11261	(16bit) : hour of the DST bias	R/W	R/C	R/C	
LW-11262	(16bit) : minute of the DST bias	R/W	R/C	R/C	
LW-11263	(16bit) : month of the year when DST starts	R/W	R/C	R/C	
LW-11264	(16bit) : week of the month when DST starts (1~5)	R/W	R/C	R/C	
LW-11265	(16bit) : day of the week when DST starts (0 $^{\sim}$ 6)	R/W	R/C	R/C	
LW-11266	(16bit) : hour of local time when DST starts	R/W	R/C	R/C	
LW-11267	(16bit) : minute of local time when DST starts	R/W	R/C	R/C	
LW-11268	(16bit) : month of the year when DST ends	R/W	R/C	R/C	
LW-11269	(16bit) : week of the month when DST ends (1~5)	R/W	R/C	R/C	
LW-11270	(16bit) : day of the week when DST ends (0~6)	R/W	R/C	R/C	
LW-11271	(16bit) : hour of local time when DST ends	R/W	R/C	R/C	
LW-11272	(16bit) : minute of local time when DST ends	R/W	R/C	R/C	
LW-11273	(16bit) : enable/disable time synchronization via NTP		D/C	D/C	
	(Network Time Protocol) server (0:disable, 1:enable)	R/W	R/C	R/C	
LW-11274	(16bit) : execute time synchronization when HMI starts	DAA	D/C	D/C	
	(0:disable, 1:enable)	R/W	R/C	R/C	





System Registers

LW-11275	(16bit) : server response time has been adjusted in	R/W	R/C	R/C
	accordance with DST (0:disable, 1:enable)			
LW-11276	(16bit) : HMI time zone (unit : minute)	R/W	R/C	R/C
LW-11277	(16bit) : server response time (server time zone) (unit : minute)	R/W	R/C	R/C
LW-11278	(16bit) : IP 0 of network time server 1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11279	(16bit) : IP 1 of network time server 1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11280	(16bit) : IP 2 of network time server 1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11281	(16bit) : IP 3 of network time server 1 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11282	(16bit) : IP 0 of network time server 2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11283	(16bit) : IP 1 of network time server 2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11284	(16bit) : IP 2 of network time server 2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11285	(16bit) : IP 3 of network time server 2 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11286	(16bit) : IP 0 of network time server 3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11287	(16bit) : IP 1 of network time server 3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11288	(16bit) : IP 2 of network time server 3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11289	(16bit) : IP 3 of network time server 3 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11290	(16bit) : IP 0 of network time server 4 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11291	(16bit) : IP 1 of network time server 4 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11292	(16bit) : IP 2 of network time server 4 (IP address = IP0:IP1:IP2:IP3)	R/W	R/C	R/C
LW-11293	(16bit) : IP 3 of network time server 4 (IP address =	R/W	R/C	R/C



LW-11294(32bit) : update interval (time synchronization interval ~ 86400, unit : second)) (10 R/W	R/C	R/C
---	--------------	-----	-----

22.3.37. Miscellaneous

		Read(R)/	Read(R)/Write(W)/ControlLocalMacroHMIHN	
Address	Description			
LB-9000~	initialized as ON	R/W	R/C	R/C
LB-9009	9009		N/C	ŊС
LB-9010	data download indicator	R	R	R
LB-9011	data upload indicator	R	R	R
LB-9012	data download/upload indicator	R	R	R
LB-9016	status is on when a client connects to this HMI	R	R	R
LB-9017	disable write-back in PLC control's [change window]	R/W	R/C	R/C
LB-9039	status of file backup activity (backup in process if ON)	R	R	R
LB-9045	memory-map communication fails (when ON)	R	R	R
LB-9049	enable (set ON)/disable (set OFF) watch dog *Note 1	R/W	R/C	R/C
LB-12053	failed to send an [Event Log] e-Mail (when ON)	R	R	R
LB-12054	failed to send an [Backup Object] e-Mail (when ON)	R	R	R
LW-9006	(16bit) : connected client no.	R	R	R
LW-9024	(16bit) : memory link system register	R/W	R/C	R/C
LW-9032	(8 words) : folder name of backup history files to SD, USB memory *Note 3		R/C	R/C
LW-9050	(16bit) : current base window ID	R	R	R
PLW-9050	(16bit) : current base window ID	R	R	R
LW-9134	(16bit) : language mode *Note 2	R/W	R/C	R/C
PLW-9134	(16bit) : language mode *Note 2	R/W	R/C	R/C
LW-9216			R	R
LW-9900	(16bit) : HMI run mode (0 : normal mode, 1-3 : test mode	5.4	D (2	E / 5
	(COM 1-COM 3)	R/W	R/C	R/C
LW-10814	(16bit) : connecting to a Weintek HMI (0:none,	_	_	_
	1:connecting) *Note 5	R	R	R

Note

- When LB-9049 watch dog function is enabled, if there's a failure in the communication, HMI will reboot 10 seconds later.
- To display texts on objects in multiple languages, except for using Label Library, the system reserved register [LW-9134: language mode] is needed. The value range in LW-9134 is 0 ~ 23 (PLW-9134 is 0~7). The values in LW-9134 relates to the languages downloaded to HMI. LW-9134 value and language correspondence vary according to the languages selected during project compilation and download.

For example: If 5 languages are defined by user in Label Library as Language 1 (Traditional Chinese), Language 2 (Simplified Chinese), Language 3 (English), Language 4 (French), and Language 5 (Japanese). If only Language 1, 3, 5 are downloaded, the corresponding language of the value in LW-9134 will be $0 \rightarrow$ Language 1 (Traditional Chinese), $1 \rightarrow$ Language 3 (English), $2 \rightarrow$ Language 5 (Japanese). The following demo project explains how to switch languages using Option List Object and LW-9134.

Lick the icon to download the demo project. Please confirm your internet connection.

- 3. The default name of the backup data folder will be the HMI name.
- 4. 1: import succeded, 2: import failed (file doesn't exist)
- **5.** When the USB Host is connected to a Weintek HMI, the address is set to 1, for testing if the USB Client of another Weintek HMI works normally.



23. HMI Supported Printers

This chapter describes the printers supported by HMI and the setup steps.

23.1.	The Supported Printer Types 2	.3-2
23.2.	Steps to Add a New Printer and Start Printing 2	3-5



23.1. The Supported Printer Types

HMI supported printer drivers include the following types:

Printer type	Description
• SP-M, D, E, F	Serial printers, please configure communication
	parameters to match the printer. [Pixels of width]
	must be correctly set and can't exceed printer
	default setting:
	100 pixels for 1610 series printers.
	220 pixels for 2407, 4004 series printers.
	The driver uses EPSON ESC Protocol for Serial
	Micro Printer.
EPSON ESC/P2 Series	Serial printers, please configure communication
	parameters to match the printer.
	The ESPON ESC/P2 printer protocol is used.
	Impact Printer: LQ-300, LQ-300+, LQ-300K+
	(RS-232), LQ-300+II (RS-232)
-6	Inkjet Printer: Stylus Photo 750
	Laser Printer: EPL-5800
y	
 HP PCL Series (USB) 	HP compatible USB printers that support HP PCL5
	protocol or PostScript3 Printer Control Language.
-	The printers that support PCL5 or later versions will
00	support PCL5 protocol due to the downward
	compatibility of PCL.





• EPSON TM-L90



EPSON TM-T70



BRIGHTEK WH-A19



Micro printer from France connects via serial port; please configure communication parameters to match the printer.

Serial printers, please configure communication parameters to match the printer. [Pixels of width] must be correctly set and can't exceed printer default setting "100".

Serial printers, please configure communication parameters to match the printer. [Pixels of width] must be correctly set and can't exceed printer default setting "576".

Serial printers, please configure communication parameters to match the printer. [Pixels of width] must be correctly set and can't exceed printer default setting "576".

The paper cutting mode can be selected: [No cut] / [Partial cut].

Supported models: A92R10-00E72A 72 in model number represents hexadecimal printer, and A represents wide voltage 5~9V. This is the same as the A6 16 impact printer.



BRIGHTEK WH-E19



BRIGHTEK WH-E22



BRIGHTEK WH-C1/C2



Serial printers, please configure communication parameters to match the printer. The paper cutting mode can be selected: [No cut] / [Half cut] / Full cut].

E22R10-00E725: Same as A7 16 impact printer.

E221R90-00E11740GA: Serial printer, connects through RS-485 port, please use a RS232-to-RS485

Serial printers, please configure the same

communication parameters as the printer.

Supported models:

converter.

A7 represents A72R90-31E72A.

Remote Printer Server



Use EasyPrinter to start printing by the printers connected with PC via Ethernet. This works under MS Windows so most printers on the market are supported.



23.2. Steps to Add a New Printer and Start Printing

- **1.** Add printer type.
- In [System Parameter Settings] » [Model] select the printer type and set the relevant parameters.

	emory	Printer/Bacl	kup Server	e-Mail	Recipes
Device Model		General	System Sett	ing Secu	ity Font
HMI station	odel : eMT307 a no : 0 t no. : 8000	•	ed as MODBUS se	erver's port no.)	•
limer					
rinter				_	
Т	уре : SP-М, Г), E, F	-		
C	OM : None				
C Baud	EPSON	ÉSC/P2 Series		t bits : 8 Bits	•
Baud	rate : EPSON HP PCL Axiohm	ESC/P2 Series Series (USB) A630	. D6 A DN T)	bits : 8 Bits bits : 1 Bit	•
Baud	rate : EPSON arity : Axiohm sity : SPRT (S idth : EPSON BRIGH	ESC/P2 Series Series (USB) A630 P-DIII, DIV, D5 TM-L90	, D6, A, DN, T)		•
Baud Ps Pixels of w Scroll bar	rate : EPSON arity : Axiohm sity : SPRT (S idth : EPSON BRIGH	ESC/P2 Series Series (USB) A630 P-DIII, DIV, D5, TM-L90 TM-T70 IEK WH-E19	, D6, A, DN, T)	bits : 1 Bit scale : 100%	•

• To connect Remote Printer Server, set the parameters in [System Parameter Settings] » [Printer/Backup Server].

Device M	odel	General	System Setti	ng	Security	Font
Extended Memor	У	Printer/Back	up Server	e	-Mail	Recipes
✓ Use Remote Prin Note: Use EasyPrin tutput settings Orientation Printer size Margin	ter to config :	gure PC for printi contal	ng screen hardco Vertical [Fit to printer mm 0 mm	4	\$	lata.
Communication setti	ngs					
IP address	: 192	. 168 . 1	. 20			
Port	: 8005					
	: admin					
U ser name						



- 2. Start printing.
- Start printing with Function Key.

ew Fund	tion Key Obj	ect			×
General	Security Shap	pe Label			
	Comment :				
		Activate after butb	on is released		
-	Change full-scre Display popup w		Change comm	non window	
	Return to previo		Close window	W	
	/UNICODE mo:		a [C]]	@ T1	
		[Backspace]	[Clear]	🔘 [Ex]	
	ASCII] / [UNIC Execute macro	-			
01	Window title bar				
	opy screen to U Screen hard copy	SB disk, SD card c /	r printer Printer : SP-M, D, E	, F	•
F	Rotate image 90	degrees	Mode : grayscale		-
- Notific		Enable			
		ОК	Cancel		Help

• Or, use PLC Control [Screen hardcopy] to start printing with a designated bit address.

PLC Control
Comment :
PLC name : Local HMI
Attribute
Type of control : Screen hardcopy
C Active only when designated window opened
Rotate image 90 degrees
Trigger address PLC name : Local HMI V Setting
Address : LB 🗸
Trigger mode : OFF->ON
Source window for print
○ Current base window
PLC name : Local HMI Setting
Address : Lw v v 0 16-bit Unsigned
Printer : SP-M, D, E, F Mode : black and white
Houe . Dialk and white
OK Cancel



24. Recipe Editor

This chapter explains how to use Recipe Editor.

24.1.	Overview	24-2
24.2.	Recipe / Extended Memory Editor Setting	24-2
24.3.	Recipe Records Setting	24-4



24.1. Overview

Recipe Editor is used to create, view, and edit recipe data.

EasyBuilder Pro also provides another tool for editing recipe: Recipe Records. To use this tool, first define a recipe in EasyBuilder Pro [System Parameter Settings] » [Recipe] tab, and then use [Recipe View Object] to display the content. The following introduces the usage of these two editing tools.

24.2. Recipe / Extended Memory Editor Setting

- 1. Open Utility Manager and click [Recipe/Extended Memory Editor].
- 2. To add new .rcp or .emi files, click [File] » [New].
- 3. Set address range and select data format.

Data Format			— ×
Address range From 0	(unit : word)	Select your data format Format 0 Save Format	▼ Delete Format
Data format Size 1 WORD 10 WORDs 1 WORD 1 WORD	Type 16-bit Unsigned String [ASCII] 16-bit Unsigned 16-bit Unsigned	Description Product No. Name Store No. Category	Add Delete Clear All Modify
٩ [m		OK Cancel

Setting	Description
Address range	Fill in address range, the unit of which is word.
Select your data	Save the specified data format for loading next time.
format	The saved file name is "dataEX.fmt" under
	EasyBuilder Pro's installation directory.
Data format	Edit new data format in this field.



4. Click [Add] to enter a description of the data type, and select data format. When selecting [String], please enter the length (words) and select [ASCII] or [Unicode].

Data Type	×
Description : Data 4	
C 16-bit BCD	C 32-bit BCD
C 16-bit HEX	C 32-bit HEX
• 16-bit Unsigned	C 16-bit Signed
C 32-bit Unsigned	C 32-bit Signed
C Float	
C String	WORD(s)
ASCII	C Unicode
ОК	Cancel

5. After setup, click [OK] to start editing recipe data.

øN	🖄 New document*								
ID	ADDRESS	Product No.	Name	Store No.	Category				
0	0	0	shampoo	9	4				
1	13	1	knife	1	5				
2	26	2	chair	3	2				
3	39	3	coffee	3	3				
4	52	4	pencil	6	5				
5	65	5	muffin	6	3				
6	78	6	donut	5	3				
7	91	7	DVD	9	6				
8	104	8	postcard	4	5				
9	117	9	maps	5	5				
10	130	10	camera	2	1				

In this example, the total length of data format is 13 words. Each 13 words will be one set of recipe data.

The first set: "product no." = address 0, "Name" = address 1 ~ 10, "Store No." = address 11, "Category" = address 12;

The second set: "product no." = address 13, "Name" = address 14 ~ 23, "Store No." = address 24, "Category" = address 25;...and so on.





After editing recipe data, it can be saved as .rcp, .emi, or .csv files. The .rcp files can be downloaded to HMI using Utility Manager or external devices (USB drive or SD card). The .emi files can be saved directly to the external device which is inserted to HMI as extended memory (EM).

24.3. Recipe Records Setting

Before using Recipe Records, first enable it in EasyBuilder Pro [System Parameter Settings]
 » [Recipes]. Please see "5 System Parameter Settings" for more detail.

Device	Mode		Gen	eral	Syst	em Settin	g	Security	y F	ont
Extended Memory			Printer/Backup Server e			e-Mail Recipes		pes		
ecipes List :										
Recipes 🛛	<u>•</u> ×	Ite	m name	Data type	Size	Display	wi	Decimal Pt.	Alignm	
1. MyRecipe	1		wItem	16-bit B	-	5		0	Align left	
2. MyRecipe			wItem1	16-bit B	-	5		0	Align left	
		Ne	wItem2	ASCII	1	5		0	Align left	
		N	ew	Setting	35	Delet	e]		
Export definit	ion file of :			Setting	JS	Delet	e]		

2. When finished, Recipe Records can be opened in main menu » [Library] » [Recipe Records]. In the example shown below, there are Recipe1 and Recipe2. Three items are shown on the right hand side. The names of recipe come from System Parameter Settings.



Recipe Records					×
Recipes :	Add Dele	ete			
MyRecipe1 (0) MyRecipe2 (0)	NewItem	NewItem1	NewItem2		
Export Import Setting	Descrip	tion		ОК	Cancel
Recipes:		-	in System Para		-
	The nur	mber enclos	ed in brackets s	shows the to	tal
	numbe	r of records i	in the correspo	nding recipe	•
Add	Inserts	records into	the recipe acco	ording to the	item
	format.				
Delete	Deletes	the edited o	content.		
Up / Down Arrows	Moves	the selected	record upward	d / downward	d.

3. To define recipes according to the specified format, click [Add] button above the record list to insert a new record and start editing each item. When click on the item, the item format will be shown under the record list. This helps users to fill in each item with legal value. Click [OK] to confirm and save the records.

Recipe Records						X
Recipes : MyRecipe 1 (6)	A	dd Dele	te			
MyRecipe2 (0)		NewItem	NewItem1	NewItem2		
	1	0	12	AA		
	2	11	2	BB		
	3	22	2	CC		
	▶ 4	33	5	DD		
	5	44	3	EE		
	6	55	0	FF		
Export Import	Data ty Decima	ype = '16-bit Unsig I Pt. = '0'	ined'		ОК	Cancel







- Each recipe can hold a maximum of 10000 records.
- The recipe records will be stored in the .exob file after compilation and will be downloaded to HMI. These recipes cannot be shared with other project files. If users need to modify the recipe contents and download them to the HMI, make sure that [Reset recipe database] check box is selected during download. If not, the recipe database in the HMI will not be updated.

Download (х
Ethernet O USB cable Password : Setting	
4 IP Name D	-
IP: 192.168.1.222 ▼	
✓ Firmware ✓ Font files	
* Necessary if update firmware or execute download first time.	
Use user-defined startup screen	
Reset recipe Reset event log Reset data sampling Reset recipe database Reset operation log	
Reboot HMI after download Automatically using current settings to download after compiling	
Download Stop	



25. EasyConverter

This Chapter explains how to use EasyConverter.

Overview	25-2
Converting Data Sampling File to Excel File	25-2
Converting Event Log File to Excel File	25-4
Converting Operation Log File to Excel File	25-5
Converting Multiple Files	25-6
Scaling Function	25-7
	Converting Data Sampling File to Excel File Converting Event Log File to Excel File Converting Operation Log File to Excel File Converting Multiple Files



25.1. Overview

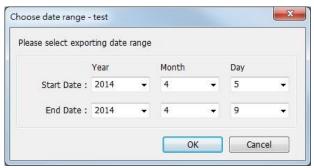
EasyConverter reads the Data Sampling file, Event Log file, and Operation Log file in HMI and convert the files to Excel format.

- From Utility Manager click [EasyConverter].
- From EasyBuilder Pro menu select [Tool] » [Data/Event Log Converter].



25.2. Converting Data Sampling File to Excel File

 If the Data Sampling file format is .db, and the file includes data of more than one day, the data to be viewed can be specified by selecting a date range. (If the file format is .dtl, please skip this step.)



2. The following is the setting dialog box, please set based on actual needs.



No	Name	Туре	Word Size	Digits	Scali	ng
1	temperature	16-bit Unsigned	1	0	No	•
2	humidity	16-bit Unsigned	1	0	No	•
Sca N/	aling & Offset A	_				

3. Click [OK], the Data Sampling layout is shown in the following figure. Click [Export to Excel]. The file will be converted to Excel format.

"Date"		econd","tempe	rature" "h	umidity"	
		30, 46, 0, 0			
2014/	4/5","04:03:	33, 459, 0,	·0·		
2014/	4/5","04:03:	36", 456", 0",	-0-		
2014/	4/5","04:03:	39, 456, 0,	·0·		
2014/	4/5","04:03:	42,457,0,	'0'		
2014/	4/5","04:03:	45,457,0,	'0'		
2014/	4/5","04:03:	48, 457, 10	, 0		
2014/	4/5","04:03:	51, 458, 10	, 23		
2014/	4/5","04:03:	54, 457, 10	, 23		
2014/	4/5","04:03:	57, 458, 42	, 23		
2014/	4/5","04:04:	00", 456", 14	, 67		
2014/	4/5", 04:04:	03, 457, 14	, 55		
2014/	4/5","04:04:	06, 459, 14	, 55		
2014/	4/7","04:04:	09, 264, 14	, 55		
		12, 209, 23	10 March 10		
2020330023		15, 21, 23,			
2014/	4/7", 04:04:	18,209,23	, 96		

4. The Excel layout is shown in the following figure.

and the second second	Date	There			E	F
and the second second		Time	Millisecond	temperature	humidity	
2 0	2014/4/5	4:03:30	46	0	0	
3 2	2014/4/5	4:03:33	459	0	0	
4 2	2014/4/5	4:03:36	456	0	0	
5 2	2014/4/5	4:03:39	456	0	0	
6 2	2014/4/5	4:03:42	457	0	0	
7 2	2014/4/5	4:03:45	457	0	0	
8 2	2014/4/5	4:03:48	457	10	0	
9 2	2014/4/5	4:03:51	458	10	23	
10 2	2014/4/5	4:03:54	457	10	23	
11 2	2014/4/5	4:03:57	458	42	23	
12 2	2014/4/5	4:04:00	456	14	67	
13 2	2014/4/5	4:04:03	457	14	55	
14 2	2014/4/5	4:04:06	459	14	55	
15 2	2014/4/7	4:04:09	264	14	55	
16 2	2014/4/7	4:04:12	209	23	55	
17 2	2014/4/7	4:04:15	21	23	96	



25.3. Converting Event Log File to Excel File

If the Event Log file format is .db, and the file includes data of more than one day, the data to be viewed can be specified by selecting a date range.

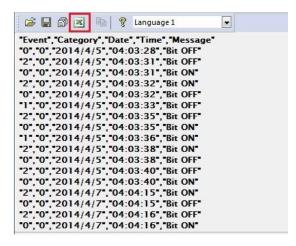
(If the file format is .evt, please skip this step.)

١	/ear		Month		Day	
Start Date :	2014	•	4	•	5	•
End Date :	2014	•	4	•	9	•

2. If the .db file of Event Log contains multiple languages, the language to be viewed can be specified. (If the file format is .evt, please skip this step.)

elect language - event		
Select your event log lar	nguage	
Language 1		
🗖 Don't ask me again		
	ОК	Cancel

 Click [OK], the Event Log layout is shown in the following figure. Click [Export to Excel]. The file will be converted to Excel format.



4. The Excel layout is shown in the following figure.



14	A	В	С	D	E	F
1	Event	Category	Date	Time	Message	
2	0	0	2014/4/5	4:03:28	Bit OFF	
3	2	0	2014/4/5	4:03:31	Bit OFF	
4	0	0	2014/4/5	4:03:31	Bit ON	
5	2	0	2014/4/5	4:03:32	Bit ON	
6	0	0	2014/4/5	4:03:32	Bit OFF	
7	1	0	2014/4/5	4:03:33	Bit OFF	
8	2	0	2014/4/5	4:03:35	Bit OFF	
9	0	0	2014/4/5	4:03:35	Bit ON	
10	1	0	2014/4/5	4:03:36	Bit ON	
11	2	0	2014/4/5	4:03:38	Bit ON	
12	0	0	2014/4/5	4:03:38	Bit OFF	
13	2	0	2014/4/5	4:03:40	Bit OFF	
14	0	0	2014/4/5	4:03:40	Bit ON	
15	2	0	2014/4/7	4:04:15	Bit ON	

Note

The "Event" column can be found. 0-> Event triggered; 1-> Event acknowledged; 2-> Event returns to normal.

25.4. Converting Operation Log File to Excel File

1. If the Operation Log file includes data of more than one day, the data to be viewed can be specified by selecting a date range.

ease select expo	orting date	range				
	Year		Month		Day	
Start Date :	2014	•	4	•	28	•
End Date :	2014	•	4	•	29	•

 Click [OK], the Operation Log layout is shown in the following figure. Click [Export to Excel]. The file will be converted to Excel format.

	28","06:47:57","","","10","NE_9","month","Set word","LW-9220 (32bit) : password","write 111"
	28", "06:47:59", ":noname:", "ADEF", "10", "NE_4", "day", "Set word", "LW-9020 (16bit) : local day", write 29"
	9" 106.48.02", noname. ADEF, 10", NE.5", hour, Set word, 1W-9019 (16bit) : local hour, write 9"
1. 2014/4/2	29", "09:48:10", ":noname:", "ADEF", "10", "NE_2", "year", "Set word", "LW-9022 (16bit) : local year", "write 2014"
5","2014/4/2	29", "09:48:13", ":noname:", "ADEF", "10", "NE_2", 'year", "Set word", "LW-9022 (16bit) : local year", "write 2014"
5","2014/4/2	29", "09:48:16", ":noname:", "ADEF", "10", "NE_6", "minute", "Set word", "LW-9018 (16bit) : local minute", "write 50"
7, 2014/4/2	29", "09:50:20", ":noname:", "ADEF", "10", "NE_8", "month", "Set word", "LW-9219 (16bit) : user no. (1~12)", write 2"
3","2014/4/2	29", "09:50:22", ":noname:", "ADEF", "10", "NE_9", "month", "Set word", "LW-9220 (32bit) : password", "write 222"
9","2014/4/2	29","09:50:26",":noname:","B","10","NE_3","month","Set word","LW-9021 (16bit) : local month","write 6"

3. The Excel layout is shown in the following figure.



	Α	В	С	D	E	F	G	Н	1	J	K	L
1	ID	Date	Time	User Name	Class	Window	Object Name	Comment	Action	Address	Information	
2	1	2014/4/28	6:47:57			10	NE_9	month	Set word	LW-9220 (32bit) : password	write 111	
3	2	2014/4/28	6:47:59	:noname:	ADEF	10	NE_4	day	Set word	LW-9020 (16bit) : local day	write 29	
4	3	2014/4/29	6:48:02	:noname:	ADEF	10	NE_5	hour	Set word	LW-9019 (16bit) : local hour	write 9	
5	4	2014/4/29	9:48:10	:noname:	ADEF	10	NE_2	year	Set word	LW-9022 (16bit) : local year	write 2014	
6	5	2014/4/29	9:48:13	:noname:	ADEF	10	NE 2	year	Set word	LW-9022 (16bit) : local year	write 2014	
7	6	2014/4/29	9:48:16	:noname:	ADEF	10	NE_6	minute	Set word	LW-9018 (16bit) : local minute	write 50	
8	7	2014/4/29	9:50:20	:noname:	ADEF	10	NE_8	month	Set word	LW-9219 (16bit) : user no. (1~12)	write 2	
9	8	2014/4/29	9:50:22	:noname:	ADEF	10	NE 9	month	Set word	LW-9220 (32bit) : password	write 222	
10	9	2014/4/29	9:50:26	:noname:	В	10	NE 3	month	Set word	LW-9021 (16bit) : local month	write 6	
11												
12												

25.5. Converting Multiple Files

1. Click [Multi-File] to open the following dialog box.

j 😂 🖬 🗗	Language 1		
1			*

2. Click [Add File] to add the files to be converted. If click [OK] without selecting [Combine to a file], the files will be exported to separate Excel files.

ulti-File Management	
Convert file list :	
H:\EasyConverter\emt3070\datalog\test\20 H:\EasyConverter\emt3070\eventlog\EL_20 H:\EasyConverter\emt3070\operation.db H:\EasyConverter\history\datalog\test.db H:\EasyConverter\history\eventlog\event.db	140322.evt
	Add File Delete File
Enable setting file	
Combine to a file	
Merge data into a single sheet	
C:\Users\nicolas\Desktop\all files.xls	
	OK Cancel
	Cancer

3. If [Combine to a file] is selected, the files will be separated into different sheets of one Excel file as shown in the following figure.



2 2014 3 2014 4 2014 5 2014 6 2014	1/3/22 1/3/22 1/3/22	Time 6:36:52 6:36:55 6:36:58 6:37:01	Millisecond 260 250 250	temperature 2 6	humidity 1 3	
3 2014 4 2014 5 2014 6 2014	1/3/22 1/3/22 1/3/22	6:36:55 6:36:58	250	6	1	
4 2014 5 2014 6 2014	1/3/22 1/3/22	6:36:58			3	
5 2014 6 2014	1/3/22		250	10		
6 2014		6.37.01		10	6	
	1000		300	13	8	
7 2014	13/22	6:37:04	280	17	10	
	/3/22	6:37:07	250	21	13	
8						
9						
10						
11						
12						
13						
14						
15	-					

25.6. Scaling Function

The equation of scaling new value = $[(value + A) \times B] + C$, and users can set the values of A, B, and C.

A -> lower limit of the value ; B -> [(scaled max) - (scaled min) / (upper limit) - (lower limit)] ; C -> scaled min.

For example, here is a voltage data with a format of 16-bit unsigned (range: 0 ~ 4096).

To convert the data to volt, range form -5V to +5V, the new value = [(value + 0) x 0.0024] + (-5).

1 sample 16-bit Unsigned 1 3	Yes 🗖	3	1	16-bit Unsigned	sample	
					sumpre	1
a 1 a off 1				set	aling <mark>&</mark> Of	Sca
Scaling & Offset						
A B C		С		B	A	Г
		C -5	24			E
A B C		C -5		0.00	0	ne
A B C 0 0.0024 -5		C -5		0.00 ((value + A) x B	0 w value =	



After scaling: Before scaling: + 😂 🔚 🗗 💌 🖬 % Language 1 🖼 🔒 🗗 💌 🖻 % Language 1 Ŧ "Date", "Time", "Millisecond", "sample" "2014/06/30", "23:02:50", "80", "-5.000" "2014/06/30", "23:02:54", "30", "-5.000" "2014/06/30", "23:02:57", "990", "-4.868" "2014/06/30", "23:03:02", "70", "-4.868" "Date", "Time", "Millisecond", "sample" Date", Time", Millisecond", sample 2014/06/30", 23:02:50", 80", 0" 2014/06/30", 23:02:54", 30", 0" 2014/06/30", 23:02:57", 990", 55" 2014/06/30", 23:03:02", 70", 55" 2014/06/30", 23:03:10", 20", 159" 2014/06/30", 22:03:14", 20", 159" 2014/06/30, 23:03:02, 70, -4.808 2014/06/30, 23:03:06, 20, -4.786 2014/06/30, 23:03:10, 20, -4.618 2014/06/30, 23:03:14, 30, -3.728 2014/06/30, 23:03:14, 30, 530 "2014/06/30","23:03:18","20","898" "2014/06/30","23:03:22","40","1024" "2014/06/30","23:03:26","0","2055" 2014/06/30", 23:03:18", 20", -2.845" 2014/06/30, 23:03:10, 20, -2.542 2014/06/30, 23:03:22, 40, -2.542 2014/06/30, 23:03:26, 0, -0.068 2014/06/30, 23:03:30, 30, -0.068 2014/06/30, 23:03:30, 30, 2055

The settings described earlier can be saved and loaded next time. The extension name of the setting file is *.lgs.



26. EasyPrinter

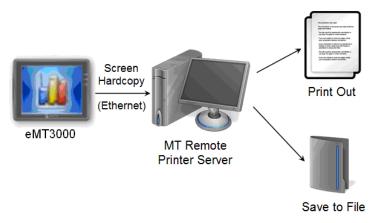
This chapter explains the setup steps of EasyPrinter.

26.1.	Overview	
26.2.	Using EasyPrinter as a Printer Server	
26.3.	Using EasyPrinter as a Backup Sever	
26.4.	EasyPrinter Operation Guide	
26.5.	Convert Batch File	



26.1. Overview

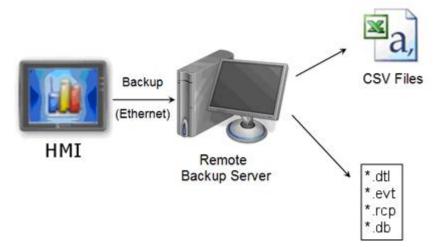
EasyPrinter is a Win32 application and can only run on MS Windows 2000 / XP / Vista / 7 / 8. It enables HMI to output screen hardcopies to a remote PC via Ethernet. The following explains how to use EasyPrinter.



Here are some advantages of using EasyPrinter:

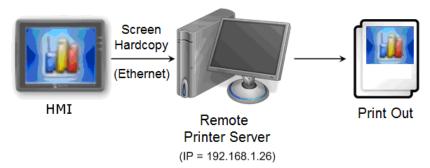
- EasyPrinter provides two modes of hardcopy output: [Print Out] and [Save to File]. Users can use either or both modes.
- Since EasyPrinter runs on MS Windows system, it supports most of the printers available on the market.
- Multiple HMIs can share one printer so users don't have to prepare printers for each HMI.

Additionally, EasyPrinter can also be a backup server. Users can use Backup objects on HMI to copy history files such as Data Sampling records and Event Log to a remote PC via Ethernet. Please see the following illustration:





26.2. Using EasyPrinter as a Printer Server



Users can make screen hardcopies with a Function Key object. The hardcopies will be transferred to the Remote Printer Server via Ethernet and then printed out.

26.2.1. Setup Procedure in EasyPrinter

In EasyPrinter's main menu, select [Options] » [Settings] and the following dialog box appears:

MT8xxx Remote P	rinter Server Settings	x
General	Server	
Hardcopy	Port number of the server socket: 8005	
Backup	User name: [Max. length = 12 characters] admin	
	Password: [Max. length = 12 characters] 111111	
	Naming Convention for HMI Folder (when writing files) Use IP address Use HMI name (assign HMI name by L W9032~L W9039) Prefix: IP	
	OK Cance	I

- **1.** Select [General] on the left hand side.
- In [Server], set [Port number of the server socket] to "8005", [User name] to "admin" and [Password] to "111111". (These are default values.)
- In [Naming Convention for HMI Folder], select [Use IP address] and enter "IP_" in the [Prefix] field.
- 4. In [Properties], select [Minimize to system tray] check box.



Set the print out location.

MT8xxx Remote I	Printer Server Settings
General	Output
Hardcopy	Print out to:
Backup	Microsoft XPS Document Writer
1	OK Cancel

- **1.** Select [Hardcopy] on the left hand side.
- 2. Under [Output] select [Print out to] and choose a printer as the output device for screen hardcopies. (The printer shown in the image above is an example; please select an actual printer located in your network environment.)
- **3.** Click [OK] to confirm the settings.
- In EasyPrinter main menu select [File] » [Enable Output] to output any incoming print request.

26.2.2. Setup Procedure in EasyBuilder Pro

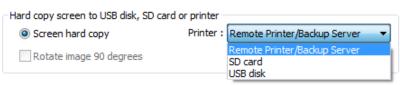
The setting procedure of EasyPrinter in EasyBuilder Pro:

- **1.** Open a new project or an existing project in EasyBuilder Pro.
- In EasyBuilder Pro main menu select [Edit] » [System Parameter Settings] »
 [Printer/Backup Server] and select [Use Remote Printer/Backup Server] check box.



Device	Model	General	System Setting	Security	Font
Extended	Memory	Printer/Bad		e-Mail	Recipes
Note: Use E Dutput settir	ngs	nfigure PC for prin	-	y and storing backup	o data.
	itation :) Hor er size :) Orio		○ Vertical ○ Fit to printer m		
FILLO	er size . 🕑 Orig	jindi size	Ht to printer in	argins	
) 📩 mm 0	0	mm	
Communicati	on settings				
IP ac	ddress : 192	. 168 . 1	. 26		
	Port: 8005				
Use	r name : admin				
Pa	ssword : 11111	1			

- 3. Under [Output settings] set appropriate values for left / top / right / bottom margins. (The margins are all set to 15mm in the example.)
- 4. Under [Communication settings] fill in the [IP address] of the printer server according to the settings in EasyPrinter. Set [Port] to "8005", [User name] to "admin" and [Password] to "111111".
- **5.** Click [OK].
- In EasyBuilder Pro main menu select [Objects] » [Button], select [Function Key], select [Screen hardcopy] and set [Printer] to [MT Remote Printer/Backup Server].



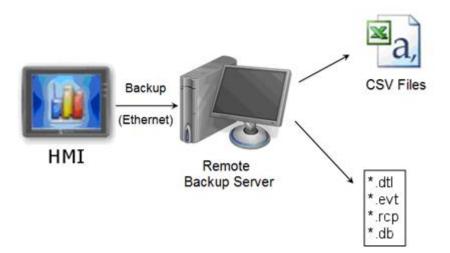
- **7.** Place the Function Key object in the common window (window no. 4) so that screen hardcopies can be captured anytime when needed.
- Compile and download the project to HMI. Press the Function Key object on the screen to make a screen hardcopy.



- A PLC Control object can also be used to make screen hardcopies.
- Alarm information cannot be printed via EasyPrinter.
- EasyPrinter can only communicate with HMI via Ethernet. Please check that the HMI has appropriate network settings.



26.3. Using EasyPrinter as a Backup Sever



Backup objects can upload historical data and Operation Log to remote backup server.

26.3.1. Setup Procedure in EasyPrinter

In EasyPrinter's main menu, select [Objects] » [Settings] and the following dialog box will appear:

MT8xxx Remote Pi	inter Server Settings
General	Server
Hardcopy	Port number of the server socket: 8005
Backup	User name: [Max. length = 12 characters] admin
	Password: [Max. length = 12 characters] 111111
	Naming Convention for HMI Folder (when writing files)
	OK Cancel

- **1.** Select [General] on the left hand side.
- Under [Server] set [Port number of the server socket] to "8005", [User name] to "admin" and [Password] to "111111". (These are default values.)
- Under [Naming Convention for HMI Folder] select [Use IP address] and enter "IP_" in the [Prefix] field.



4. Under [Properties] select [Minimize to system tray].

Set the backup location.

MT8xxx Remote	Printer Server Settings
General	Output
Hardcopy	Backup files in:
Backup	C:\EBpro
	When target file has existed: Qverwrite it. (The content will be destroyed) Append .BAK to the file name.
	Convert Batch File Enable C: \EBpro\convert2csv.def
	OK Cancel

- **1.** Select [Backup] on the left.
- 2. Under [Output] click the incoming history files.
- 3. Click [OK] to confirm the settings.
- 4. In the main menu, select [File] » [Enable Output] to backup data in the selected directory.

26.3.2. Setup Procedure in EasyBuilder Pro

The setup procedure of EasyPrinter in EasyBuilder Pro:

- **1.** Open a new project or an existing project in EasyBuilder Pro.
- In EasyBuilder Pro's main menu, select [Edit] » [System Parameter Settings] » [Printer/Backup Server] and select the [Use Remote Printer/Backup Server] check box.



Device	Model	General	System Settin	g Security	Font
Extended M	emory	Printer/Back	up Server	e-Mail	Recipes
✓ Use Remote Note: Use Eas Dutput settings	yPrinter to co		ting screen hardco	py and storing bac	kup data.
Orienta	tion : 💿 Ho	rizontal 🔺			
Printer	size : 💿 Ori	ginal size	Fit to printer n	nargins	
Ma	argin :	0	🚖 mm		
		0 💮 mm	0	mm	
Communication	settings				
IP addr	ress : 192	. 168 . 1	. 26		
I	Port : 8005				
User n	iame : admin				
Passv	word : 11111	11			

- 3. Under [Communication settings] fill in the [IP address] of the printer server according to the settings in EasyPrinter. Set the [Port] to "8005", [User name] to "admin" and [Password] to "111111". (Note: These are default values.)
- **4.** Click [OK].

Create a Backup object.

 In EasyBuilder Pro's main menu, select [Objects] » [Backup] and the following dialog box appears:



lew Backup Object 🗾
General Security Shape Label
Comment :
Source
RW RW_A Recipe database
Historical event log Historical data sampling
Operation log
Backup position
Remote printer/backup server
Note : Use LW-9032~9039 to change the backup folder name.
Note : Use [Remote printer/backup server] to store data to a remote PC. Enable the server in [System Parameter][Printer/Backup Server] settings.
Range Start : Today Yesterday
Within : All (max. 90 days)
Trigger
Mode : Touch trigger 🔻
*LB-9039 indicates the status of file backup activity (backup in process if status is ON)
OK Cancel Help

- 2. Under [Source] select [Historical event log] (or [RW], [RW_A] if needed.)
- 3. Under [Backup position] select [Remote printer/backup server].
- 4. Under [Range] select [Today] and [All] (or other options if needed.)
- 5. Under [Trigger] select [Touch trigger].
- 6. Click [OK].
- Place the [Backup] object in the common window (window no. 4), and users will be able to make backups anytime when needed.
- *8.* Compile and download the project to HMI. Press the Backup object on the screen to make a backup of the history data.

Note

- The Backup object can also be triggered by a bit address.
- Users can arrange a Scheduler object, which turns a bit ON at the end of a week, to trigger the Backup object to automatically back up all history data.



26.4. EasyPrinter Operation Guide

The following introduces the interface and operation of EasyPrinter.

26.4.1. EasyPrinter Managing Window

EasyPrinter main menu is divided into 5 parts as shown in the following figure:

法 Weintek MT Remote Printer Ser	ver - EasyPrinter					
	<u>H</u> elp					
🖨 🙆 😭 😭						
Job ID Address	Download Progress	д 🖬	Preview	Ф 🔛	Properties	д 🗵
	From	Progress				
1	2		3		4	
Logger						д
[10:16:55] Printer server is activ	e and ready for reque	ests. 5				
Ready					CAP	NUM SCRL

Area	Name	Description
1	Job List	Lists all incoming tasks, such as screen
		hardcopy and backup requests.
2	Download Progress	Shows the download progress of incoming
		requests.
3	Preview	Shows the preview image of the screen
		hardcopy task selected from [Job List].
4	Properties	Shows the information about the task
		selected from [Job List].
5	Logger	Shows the time and message information of
		events such as incoming request, incorrect
		password, etc.



26.4.2. Operation Guide

The following describes the function of EasyPrinter menu items.

Menu	Description
File	Enable Output
	If selected, EasyPrinter processes the tasks one by one,
	otherwise, EasyPrinter stores the tasks in memory.
Edit	Edit
	Edits screen hardcopy by setting [Orientation], [Scaling] and
	[Margins].
	Delete
	Deletes the selected tasks permanently.
	Select All
	Selects all tasks from [Job List].
View	Properties Bar
	Shows or hide the Property Window.
	Preview Bar
	Shows or hide the Preview Window.
	Download Bar
	In [Download Progress] Window, the mode to display download
	progress can be set by clicking the header of the [progress]
	column as shown in the following figure:
	Download Progress 📮 📧
	From Process
	Data Length Display
	Logger Bar
	EasyPrinter can reserve up to 10,000 messages in Message
	Window. If a new message comes in, the oldest message will be
	deleted.
Options	Please see the following page.

Note

EasyPrinter can only reserve up to 128 MB of task data in memory. If the memory is full, any request coming in afterwards will be rejected. Users must either operate [Enable Output] or delete some tasks to make room for new tasks.



- The backup task is not editable.
- Edit] is available only when a task is selected.
- [Delete] is available when at least one task is selected.

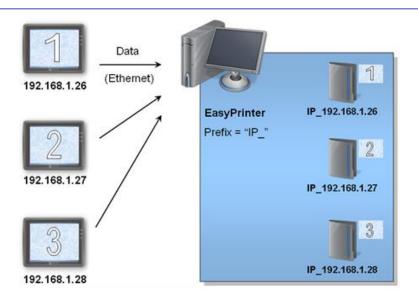
The following is the detail for [Options] » [Settings]

• In General Tab:

MT8xxx Remote P	rinter Server Settings		
General	Server		
Hardcopy Backup	Port number of the server socket: 8005 User name: [Max. length = 12 characters] admin Password: [Max. length = 12 characters] 111111 Naming Convention for HMI Folder (when writing files) Image: Convention for HMI Folder (when writing files) Image: Use IP address Image: Use HMI name (assign HMI name by L W9032~L W9039)		
	Prefix: IP_ (Ex: IP_192. 168. 1.25) Properties Migimize to system tray Detailed message		
	OK Cancel		

Setting	Description	
Server	Port number of the server socket	
	Sets the Ethernet port number to connect the HMI. Range: 1 $^{\sim}$	
	65535. Default: 8005.	
	User name / Password	
	Sets the user name and password to let only authorized HMIs send	
	requests to EasyPrinter.	
Naming	EasyPrinter uses different folders to store files (e.g. hardcopy	
Convention	bitmap files, backup files) from different HMI. There are two ways	
for HMI	to name the folders:	
Folder	Use IP address	
	EasyPrinter names the folder as [Prefix] + [IP address] after the	
	HMI at this IP address sends request.	





Use HMI name

EasyPrinter names the folder in [Prefix] + [HMI name] after the HMI this name indicates sends request.

PropertiesMinimize to system trayIf this check box is selected, the EasyPrinter shortcut icon will be
placed in the system tray in PC. Double click the on icon in system
tray to open EasyPrinter.Detailed message
Select this check box to display more detailed messages about

events in the message window.

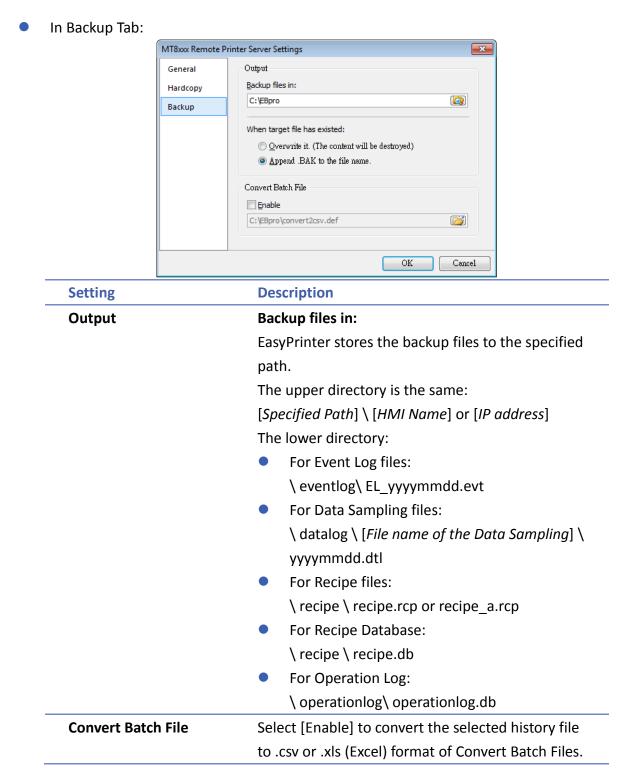
In Hardcopy Tab:

General	Output
Hardcopy	Print out to:
Backup	Microsoft XPS Document Writer 🔹
	Save to files in:
	C:\EBpro
	OK Cance

SettingDescriptionOutputPrint out toEasyPrinter prints out the hardcopy result with the specified
printers.Save to files in
EasyPrinter converts the hardcopy result into a bitmap file and



saves it in the specified directory. The bitmap files are found at: [Specified Path] \ [HMI Folder] \ yymmdd_hhmm.bmp For example, when a hardcopy request is given at 17:35:00, 12/Jan/2009, the bitmap file will be named "090112_1735.bmp". And if there is another bitmap file generated within the same minute, it will be named "090112_1735_01.bmp" and so on.





Note

System registers LW-9032 to LW-9039 can be used to specify HMI name.

26.5. Convert Batch File

EasyPrinter provides a conversion tool to convert the uploaded Data Sampling and Event Log history files to .csv files automatically. To do so, please select [Enable] under [Convert Batch File] to make EasyPrinter convert the history files.

In the following illustration, the conversion is actually executed by EasyConverter. EasyPrinter simply follows the criteria in Convert Batch File and activates EasyConverter with proper arguments to achieve the conversion.



Note

- EasyConverter is another Win32 application that converts history data into .csv or MS Excel .xls files. Users can find it in the EasyBuilder Pro installation directory.
- Users requesting this function must ensure EasyPrinter and EasyConverter are placed in the same directory.

26.5.1. The Default Value of Convert Batch File

The following is the default Convert Batch File: convert2csv.def

Listing 1. Default Convert Batch File

1: "dtl", "EasyConverter /c \$(Pathname)" 2: "evt", "EasyConverter /c \$(Pathname)"

There are two lines in the file. Each line has two arguments separated by a comma and forms a criterion of how to process a specific type of files. The first argument stands for the extension name of the file type to be processed. The second argument stands for the command to be executed in console mode. Please note that "\$(Pathname)" is a key word to inform EasyPrinter to replace it with the real name of the converted backup file. For example, if a Data Sampling history file named 20090112.dtl is uploaded and stored, EasyPrinter will send out the following command to a console window:



EasyPrinter

1: EasyConverter /c 20090112.dtl

A file named 20090112.csv is created.

The criteria of the default Convert Batch File:

- 1. Convert all Data Sampling history files (.dtl) into .csv files.
- 2. Convert all Event Log history files (.evt) into .csv files.

Note

"\$(Pathname)" in the second argument stands for the full path name of the file. In the previous case, EasyPrinter replaces it with:

[Specified Path] \ [HMI Folder] \ [datalog] \ [Folder name of the Data-Sampling object] \ 20090112.dtl

- EasyPrinter interprets the Convert Batch File in line basis, that is, each line forms a criterion.
- Any two arguments should be separated by a comma.
- Every argument should be put in double quotes.
- Do not put any comma inside an argument.

For more information, see "25 Easy Converter".

26.5.2. Specialized Criteria

The specialized criterion are needed when:

- Upload file to a specific HMI, see listing 2.
- Identify the HMI by HMI name, see listing 3.
- Process differently to different Data Sampling, see listing 4.

(This can only be used for Data Sampling file with the file name "voltage".)

The 3rd argument ("*") indicates this criterion accepts the Data Sampling files that meet the criterion from any HMI. Users can also change the 3rd argument to "192.168.1.26",

"192.168.1.", or HMI name, etc. for narrowing the range of the target HMI.

Listing 2. Specialized Criterion for the HMI IP: 192.168.1.26

1: "dtl", "EasyConverter /c \$(Pathname)", "192.168.1.26"

Listing 3. Specialized Criterion for HMI name: Weintek_01

1: "dtl", "EasyConverter /c \$(Pathname)", "Weintek_01"

Listing 4. Specialized Creterion for Data Sampling file name: Voltage

1: "dtl", "EasyConverter /s Voltage.lgs \$(Pathname)", "*", "Voltage"



EasyBuilder Pro V5.02.01

26.5.3. The Format of a Convert Batch File

The following explains the arguments in a criterion.

File Type Command (line) HMI IP / Name Condition 1 Condition 2

- File Type
 This argument specifies the extension name of the uploaded file in this criterion.
 (e.g. ".dtl" for Data Sampling history files, ".evt" for Event Log history files)
- Command (line)
 The command EasyPrinter sends to a console window if the uploaded file meets the criterion.
- HMI IP / Name

This argument specifies the HMI that meets the criterion.

Condition 1

This argument specifies the folder name of the Data Sampling files that meet the criterion. This is not effective to other format of files.

• Condition 2

Not used (Reserved for future use).

26.5.4. The Order of Examining Criterion

EasyPrinter examines criterion in descending order every time a file is uploaded. Once the file meets a criterion, it stops the examination and starts over for the next file. Therefore, users should place the criterion with a wider range downward in the Convert Batch File and place the more specific criteria upward. For example:

"evt", "EasyConverter /c \$(Pathname)"

"dtl", "EasyConverter /c \$(Pathname)"

- "dtl", "EasyConverter /c \$(Pathname)", "192.168.1.26"
- "dtl", "EasyConverter /c \$(Pathname)", "my_HMI_01"
- "dtl", "EasyConverter /c \$(Pathname)", "my_HMI_02"
- "dtl", "EasyConverter /s Voltage.lgs \$(Pathname)", "*", "Voltage"

The correct order of examination would be: (from bottom to top)

"dtl", "EasyConverter /s Voltage.lgs \$(Pathname)", "*", "Voltage"

"dtl", "EasyConverter /c \$(Pathname)", "my_HMI_02"

"dtl", "EasyConverter /c \$(Pathname)", "my_HMI_01"

"dtl", "EasyConverter /c \$(Pathname)", "192.168.1.26"

- "dtl", "EasyConverter /c \$(Pathname)"
- "evt", "EasyConverter /c \$(Pathname)"



27. EasySimulator

This chapter explains how to use EasySimulator.

27.1.	Overview	2
27.2.	Steps to setup EasySimulator	2



27.1. Overview

EasySimulator allows a project to be run in the On- or Off-line simulator without having to start the simulator from EasyBuilder Pro. To do this, please prepare the required files and follow the steps to setup EasySimulator.

27.2. Steps to setup EasySimulator

- 1. Prepare the following required files.
- [driver] \rightarrow [win32]
- com_e30.exe
- EasySimulator.exe
- gui_e30.exe
- sqlite3.dll
- xob_pos.def
- 2. Open **xob_pos.def** by using a text editing tool (e.g. Notepad) and edit the contents.

🗍 xob_pos.def - Notepad	×
File Edit Format View Help	
"2" // operation mode 2: off-line 3: on-line (eMT3000 SERIES) "C:\EasySimulation" // define the directory of com_e30.exe and gui_e30.exe "C:\EasySimulation\EMTP1.exob" // define the directory of exob file	111 >
	-
	h at

Line number	Description
1	"2" run an Off-line Simulation; "3" run an On-line Simulation.
2	The directories of the relevant files.
	(e.g. com_e30.exe, gui_e30.exe, EasySimulator.exeetc.)
3	The full path of the .exob file.

- 3. Double click on EasySimulator.exe to start a simulation.
- 4. On-line / Off-line Simulation is displayed on the screen.

Note

- The required files can be found in the EasyBuilder Pro installation directory. Please install EasyBuilder Pro first then copy the required files to your PC.
- If EasySimulator.exe is not activated, please check if the installation directory is correct.
- If the "Failed to open project file: No such file or directory" dialog box appears, this indicates that there is an error of the .exob file path, please check again.



28. Multi-HMI Communication (Master Slave Mode)

This chapter explains how to connect multiple HMIs.

28.1.	Overview	28-2
28.2.	Steps to Create a Project of Master HMI	28-2
28.3.	Steps to Create a Project of Slave HMI	28-3
28.4.	Steps to Connect with MT500 Slave HMI	28-5



28.1. Overview

Multi-HMI Communication means that a HMI connects with a remote HMI via COM port, and read the data in the PLC connected to the remote HMI as shown in the following figure.



The PLC is connected with HMI 1, and HMI 1 is connected with HMI 2 via COM port, so that HMI 2 can read the data in PLC through HMI 1.

The following parts explain how to create the projects used in HMI 1 and HMI 2 by using EasyBuilder Pro.

28.2. Steps to Create a Project of Master HMI

The following is the settings of HMI 1 in [System Parameter Settings] » [Device List].

Device list :

	No.	Name	Location	Device type	Interface
	Local HMI	Local HMI	Local	eMT/MT SERIES	-
	Local PLC 1	FATEK FB Series	Local	FATEK FB Series	COM 1 (9600,E,
۲	Local Server	Master-Slave S	Local	Master-Slave S	COM 3 (115200,

- 1. Since COM 1 of HMI 1 connects to PLC; the device list must include [Local PLC 1], and set the correct parameters. In this example the connected PLC is "FATEK FB Series".
- 2. COM 3 of HMI 1 is used to receive commands from HMI 2; a new device must be added– [Master-Slave Server] for setting communication properties of COM 3. The parameters of COM 3 in the example are set to "115200, E, 8, 1", and uses RS232. These parameters are not required to be the same as PLC settings, but the [Data bits] must be set to 8. In general, a higher baud rate is recommended for HMI 2 to efficiently read PLC data.



28.3. Steps to Create a Project of Slave HMI

The following is the settings of HMI 2 in [System Parameter Settings] » [Device List].

Dev	ice list :				
	No.	Name	Location	Device type	Interface
	Local HMI	Local HMI	Local	eMT/MT SERIES	-
►	*Remote PL	FATEK FB Series	COM 1 (FATEK FB Series	COM 1 (115200,

Since the PLC that HMI 2 reads is connected with HMI 1, thus for HMI 2, PLC is a remote device. Therefore, it is necessary to add a [Remote PLC] into the device list. In this example the connected PLC is "FATEK FB Series". The way to create [*Remote PLC 1] is described in the following steps.

1. Add a new device. Set [PLC type] to [FATEK FB Series] and [PLC default station no.] must be set in accordance with the connected PLC.

N	
Narr	e: FATEK FB Series
	⊘ HMI
Locatio	n : Local
PLC typ	e : FATEK FB Series
	V.1.80, FATEK_FB.e30
PLC I	F: RS-232 •
* Support comm	unications between HMI and PLC in pass-through mode
* Set LW-9903 t	o 2 to enhance the speed of download/upload PLC program in pass-through mode
co	M : COM1 (9600,E,7,1) Settings
cc	M : COM1 (9600,E,7,1) Settings PLC default station no. : 1
co	
со	PLC default station no. : 1
	PLC default station no. : 1
I	PLC default station no. : 1 Default station no. use station no. variable Use broadcast command
Iı Max	PLC default station no. : 1 Default station no. use station no. variable Use broadcast command iterval of block pack (words) : 5

2. Correctly set the parameters. Since COM 1 of HMI 2 connects with COM 3 of HMI 1 instead of directly connect with PLC, the settings of PLC will be ignored. HMI 2 COM 1 and HMI 1 COM 3 must set to the same communication parameters and interfaces. As shown in the following figure, use RS232, and set parameters to [115200, E, 8, 1].



COM Port Settings		
COM : COM 1 -	Timeout (sec) :	1.0 -
Baud rate : 115200 🔹	Turn around delay (ms) :	0
Data bits : 8 Bits 🔹	Send ACK delay (ms) :	0
Parity : Even 🔻	Parameter 1 :	0
Stop bits : 1 Bit 🔹	Parameter 2 :	0
	Parameter 3 :	0
* OS version 20120920 or later support 1-	4400 baud rate OK	Cancel

3. For HMI 2, PLC is a remote device, change [Location] to [Remote], and select [COM port] to connect remote HMI (HMI 1).

Nam	FATEK FB Series		
	○ HMI		
Locatio	: Remote	gs COM 1 (master-slave	mode)
PLC typ	FATEK	FB Series	•
	V.1.80, FATEK_FB.e30		
PLC I/	: RS-232	•	
	IP Address Settings		
	© Ethernet	(use master-slave protocol)	
со			ettings
	_		
	L	OK Cancel	
	Default station no. use station	on no. variable	
I	erval of block pack (words) : 5	•	
Max	read-command size (words) : 64	-	
Max.	vrite-command size (words) : 64		
		ОК	Cancel
		OK	Callee
list :			

No. Name Location Device type Interface Local HMI Local eMT/MT SERIES *Remote PL... FATEK FB Series COM 1 (... FATEK FB Series COM 1 (115200,E...)

4. When finished, a new device [Remote PLC] can be found in the [Device List]. This device has a "*" symbol, which means, even if it contains "Remote" in the name, it actually gives commands and gets replies through a local COM port, and therefore the connection with PLC can be checked from a local system register. [*Remote PLC 1], [*Remote PLC 2], [*Remote PLC 3] and [Local PLC 1], [Local PLC 2], [Local PLC 3] use the same system registers from the listed below.



Register	Description
LB-9150	When ON, automatically connects with PLC (COM 1) when disconnected.
	When OFF, ignores disconnection with PLC.
LB-9151	When ON, automatically connects with PLC (COM 2) when disconnected.
	When OFF, ignores disconnection with PLC.
LB-9152	When ON, automatically connects with PLC (COM 3) when disconnected.
LD-9192	When OFF, ignores disconnection with PLC.
	These local registers indicate the connection states with PLC (through
	COM1).
10.0000	LB9200 indicates the connection state with PLC (station no. 0), and
LB-9200~	LB9201 indicates the connection state with PLC (station no. 1) and so on.
LB-9455	When ON, indicates the connection state is normal.
	When OFF, indicates disconnection with PLC.
	Set ON again, the system will then try to connect with PLC.
LB-9500~	These local registers indicate the connection states with PLC (through
LB-9755	COM2).
	LB9500 indicates the connection state with PLC (station no. 0), and
	LB9501 indicates the connection state with PLC (station no. 1) and so on.
	When ON, indicates the connection state is normal.
	When OFF, indicates disconnection with PLC.
	Set ON again, the system will then try to connect with PLC.
LB-9800~	These local registers indicate the connection states with PLC (through
LB-10055	COM3).
	LB9800 indicates the connection state with PLC (station no. 0), and
	LB9801 indicates the connection state with PLC (station no. 1) and so on.
	When ON, indicates the connection state is normal.
	When OFF, indicates disconnection with PLC.

28.4. Steps to Connect with MT500 Slave HMI

EasyBuilder Master-Slave Protocol enables MT500 to exchange data with eMT3000 local data via the connected PLC.

28.4.1. Settings in EasyBuilder Pro

1. Select [Master-Slave Server] and click [Settings]. If a PLC is connected, follow the original settings.



Extende	ed Memory		Printer/	Backup Server		Recipes
Device	Model	General	S	ystem Setting	Securit	y Font
evice list : No.	Name	9	Location	Device type	Interfa	ace I/F
evice list : No. Local HMI		-	Location Local	Device type eMT/MT SERIE		ace I/F -

2. Select [RS-232], click [Settings].

Device Properties	
Name :	Master-Slave Server
	○ HMI
Location :	Local Settings
PLC type :	Master-Slave Server
	V.1.00, MASTER_SLAVE.e30
PLC I/F :	RS-232 🔹
COM :	COM1 (115200,E,8,1) Settings

3. Fill in MT500 PLC ID No. in [Parameter 1] (Refer to MT500 settings).

COM Port Settings		
COM : COM 1 -	Timeout (sec) :	1.0 •
Baud rate : 115200 🔻	Turn around delay (ms) :	0
Data bits : 8 Bits 🔹	Send ACK delay (ms) :	0
Parity : Even 🔻	Parameter 1 :	10
Stop bits : 1 Bit 🔹	Parameter 2 :	0
	Parameter 3 :	0
* OS version 20120920 or later support :	14400 baud rate OK	Cancel

28.4.2. Settings in EasyBuilder500

1. In [System Parameter Settings], set [Multiple HMI] to Slave, set [HMI-HMI link speed] to 115200.



[Baud rate] must be identical in EasyBuilder500 and EasyBuilder Pro.

	Parameter Sett						
PLC	General Indi	cator Security	Editor	r Hardware	Aux.		
	PLC type :	MITSUBISHI	FX0n/FX	(2	•		
	HMI model :	MT510T/MT5	508T (64	0 x 480)	•		
	PLC I/F port :	RS-485 defau	t –		Baud rate	9600	Ŧ
	Data bits :	7 Bits	-		Parity	Even	-
	Stop bits :	1 Bit	-				
	Parameter 1 :	0		Tur	m around delay	: 0	
	Parameter 3 :	0			Parameter 4	: 0	
	Parameter 5 :	0			Parameter 6	: 0	
H	HMI station no. :	0	•		PLC station no.	: 0	•
	Multiple HMI :	Slave	•	HMI-	HMI link speed	: 115200	•
	Connect I/F :	Serial	•				
	Local IF	address : 0	· [0 . 0	· 0]	
	Server IF	address : 0	· [0 . 0	· 0]	
	Subnetw	ork mask : 0		0.0	· 0]	
	Default route IF	address : 0		0 . 0	· 0]	
PI	LC time out const	ant (sec) : 3.0)	▼ F	LC block pack	: 0	•
			0		ancel	Apply	Help

 Double click PLC Address View.exe to check PLC ID No. and fill in [Parameter 1] of EasyBuilder.

MITSUBISHI FX0n/FX2						
PLC/Address Type ID	Bit/Word	Address Type	Addressing Format	Max	Min	Γ
MITSUBISHI FX0n/FX2	PLC ID=10					
0	Bit(HMI)	LB	ddd	9999	0	
1	Bit(PLC)	X	000	377	0	
2	Bit(PLC)	Y	000	377	0]
3	Bit(PLC)	М	ddd	9999	0	j
4	Bit(PLC)	Т	ddd	255	0	ĺ
5	Bit(PLC)	С	ddd	255	0	İ
8	Word(HMI)	LW	ddd	9999	0	İ
9	Word(PLC)	TV	ddd	255	0	ĺ
10	Word(PLC)	CV	ddd	199	0	İ
11	Word(PLC)	D	ddd	9999	0	ĺ
12	DWord(PLC)	CV2	ddd	255	200	İ
13	Word(PLC)	SD	ddd	9999	8000	İ
121	Word(HMI)	RWI	ddd	32767	0	İ
120	Bit(HMI)	RBI	ddd(h)	2047f	0	İ
140	Bit(HMI)	RB	ddd(h)	2047f	0	ĺ
141	Word(HMI)	RW	ddd	65535	0	İ
160	Bit(HMI)	Ms_RB	ddd (h)	4095f	0	İ
161	Bit(HMI)	Ms_LB	ddd	9999	0	j
180	Word(HMI)	Ms_RW	ddd	65535	0	ĺ

3. Connect HMIs via Com Port RS-232, the communication is then enabled.

28-7



Note

- There will always be a PLC selected in MT500 system parameter settings, in this case, even to read/write eMT3000 local data, the ID of the selected PLC of MT500 system parameters must also be filled in EasyBuilder [Parameter 1].
- When using S7-200, S7-300 drivers, since MT500 reverses the high bytes and the low bytes, this will cause MT500 to misread eMT3000 local data, therefore this way is not available in Master-Slave Mode.

Bit/Word	MT500	eMT3000	Range
В	Ms_RB	RW_Bit	dddd: 0~4095 (h): 0~f
В	Ms_LB	LB	ddd: 0~9999
W	Ms_RW	RW	dddd: 0~65535
W	Ms_LW	LW	ddd: 0~9999

The Comparison between MT500 and eMT3000:



29. Pass-through

This chapter explains how to set up Pass-through mode.

29.1.	Overview	. 29-2
29.2.	Ethernet Mode	. 29-2
29.3.	COM Port Mode	. 29-5
29.4.	Pass-through Control	. 29-7
29.5.	SIEMENS S7-200 PPI and S7-300 MPI Pass-through Settings	. 29-8



29.1. Overview

The Pass-through feature allows PC applications to control PLC via HMI. In this case the HMI is an adaptor.

The Pass-through feature provides two modes:

- Ethernet
- COM port

Click [Pass-through] in Utility Manager to open the setting dialog box.

29.2. Ethernet Mode

29.2.1. Steps to install virtual serial port driver

Before using [Ethernet] mode, please check if Weintek virtual serial port driver has been installed.

1. Open Utility Manager to check if the driver has been installed. If it shows [Please install weintek virtual serial port driver], please click [Install].

Pass-through	
Ethernet	C COM port
Virtual COM P	ort (PC <-> PLC)
	Please install weintek virtual serial port driver
PLC Connecti	on Port (HMI <-> PLC)
HMLIP	: Stop Pass-through
	V V
Install	Uninstall Apply
	Exit

2. If the dialog below pops up during installation asking for verification, please click

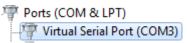
[Continue Anyway].



29-2



- When finished, the [Virtual COM Port (PC <-> PLC)] field displays the virtual COM port used.
- 29.2.2. Steps to Change the Virtual Serial Port
 - **1.** Open [Device Manager] to check the installed [Virtual Serial Port].



To change the number of virtual serial port, click [Virtual Serial Port] to open [Port Settings]
 » [Advanced].

(Virtual Serial Port (COM3) Properties
	General Port Settings Driver Details
Advanced Settings for COM3	<u>B</u> its per second: 9600 ▼
☑ Use FIFO buffers (requires 16550 compatible UAR	<u>D</u> ata bits: 8 ▼
Select lower settings to correct connection problem	Parity: None
Select higher settings for faster performance.	<u>S</u> top bits: 1 ▼
Receive Buffer: Low (1)	Eow control: None
<u>T</u> ransmit Buffer: Low (1)	Advanced Restore Defaults
COM <u>P</u> ort Number: COM3 -	
	OK Cancel

29.2.3. Settings of Ethernet mode

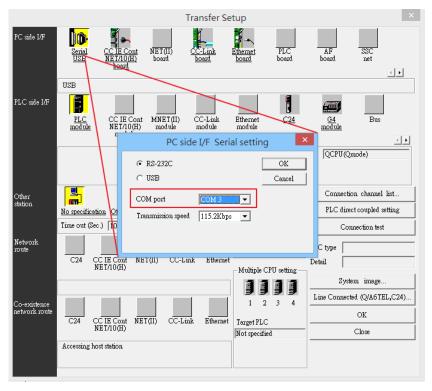
After installing the virtual serial port driver, follow the steps to use Ethernet mode of pass-through feature.

- 1. Set the IP address of the HMI connected with PLC.
- 2. Set the communication port and the serial port that connects HMI with PLC.
- 3. Click [Apply], to apply the settings.



Pa	ass-through			
	ethernet	© COM port		
	Virtual COM Por	t (PC <-> PLC)		
		СОМЗ		
		Install		
	Settings of Dest	ination HMI		
		Mode : Normal	•	
		IP: 192.168.	1 . 123	
	Commun	ication port : 8000	 (Default : 8000) 	
	Pass-th	nrough port:2000	•	
	PLC	connection : COM 1	 (LW-9902 on HMI)) 	
			Apply	
Destination C	OM Port			
			Ethernet	
DI C	→		← →	
PLC	·	- Brance		
		HMLIP		Virtual COM

4. When running PC application, set COM port to the used virtual serial port. For example, in Mitsubishi application, if the virtual serial port is COM 3, set [PC side I/F Serial setting] » [COM port] to COM 3.

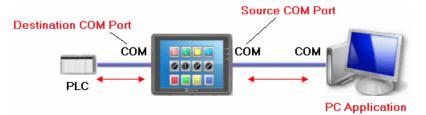


5. With the correct configurations, upon execution of PLC application on PC, HMI will be



automatically switched to Pass-through mode. During Pass-through, the PLC is controlled by PC via the virtual serial port. Pass-through mode will be turned off when the application ends.

29.3. COM Port Mode



[Source COM Port] The port connects HMI with PC.

[Destination COM Port] The port connects HMI with PLC.

To use [COM port] mode of Pass-through, please set the properties of Source COM Port and Destination COM Port correctly.

29.3.1. Settings of COM Port Mode

There are two ways to enable [COM port] mode of Pass-through feature.

- Using Utility Manager.
- Using system registers.
 LW-9901: pass-through source COM port (1 ~ 3: COM 1 ~ COM 3)
 LW-9902: pass-through destination COM port (1 ~ 3: COM 1 ~ COM 3)

29.3.2. Using Utility Manager

1. Click [Pass-through] button in Utility Manager to set the communication parameters as shown in the following figure.



ass-through		
C Ethernet	COM port	
HMI IP :	•	
	Get HMI Communication Parameters	
н	MI work mode : Unknown	
Source COM Po	t (PC -> HMI)	
	COM 1 TRS2	32 🔹
Baud rate :	9600 v Data bits : 7 Bit	s 🔹
Parity :	None Stop bits : 1 Bit	•
Destination CON	I Port (HMI -> PLC)	
	COM 3 RS2	32 🔻
Baud rate :	9600 💌 Data bits : 7 Bit	s 💌
Parity :	None Stop bits : 1 Bit	•
Start Pass-thro	ugh Stop Pass-through	
		Exit

Setting	Description
HMI IP	HMI IP address.
Get HMI Communication	Reads the settings of Source and Destination COM
Parameters	port. Click this button to update the communication
	parameters.
Source COM Port (PC->HMI) /	The communication parameters of Source and
Destination COM Port	Destination COM Port are displayed.
(HMI->PLC)	The settings will be applied when [Start
	Pass-through] is clicked.
Baud rate /	Source and Destination COM Port parameters should
Data bits /	be set to be same. Since [Source COM Port]
Parity /	connects PC, select RS-232 mode in most situations;
Stop bits	[Destination COM Port] connects PLC, so the setting
	depends on the PLC type, and can be one of RS-232,
	RS-485 2W, or RS-485 4W.

Note

When pass-through feature is no longer needed, click [Stop Pass-through] to stop it. HMI will then resume communication with PLC.

There are three work modes of HMI.



Mode	Description
Unknown	The work mode before reading the settings of HMI.
Normal	The work mode after reading the settings of HMI.
	The HMI does not accept any data form the Source
	COM Port.
Pass-through	The work mode is "Pass-through." the PC connected
	via Source COM Port can control the PLC connected
	via Destination COM Port.

29.3.3. Using System Registers

Another way of enabling pass-through is by writing to LW-9901(Source COM port) and LW-9902 (Destination COM port). When the values of LW-9901 and LW-9902 match the conditions below, HMI will start Pass-through automatically:

- The values of LW-9901 and LW-9902 are 1 to 3 (1 to 3: COM 1 to COM 3).
- The values of LW-9901 and LW-9902 are different.

To change the communication parameters, just change the value in the related registers and set ON the appropriate registers: [LB-9030: update COM 1 communication parameters], [LB-9031: update COM 2 communication parameters] and [LB-9032: update COM 3 communication parameters]. HMI will then update the settings.

Note

To stop Pass-through, change the values of LW-9901 and LW-9902 to 0.

29.4. Pass-through Control

Generally speaking, during pass-through, HMI closes its connection with the PLC until the pass-through mode ends. However, certain PLC drivers allow communications between HMI and PLC in pass-through mode.

To see whether a driver supports concurrent communication, see "PLC Connection Guide".

Pass-through control is controlled by LW-9903. The following table shows valid LW-9903 values and their features.

LW-9903	Description
0 (Default)	Normal Mode. Communications between HMI
	and PLC in pass-through mode is allowed.
2	Stop Mode. No communications between HMI
	and PLC in pass-through mode



Note

Due to speed limitation, users may wish to set LW-9903 to 2 to enhance the speed of program download/upload in pass-through mode.

29.5. SIEMENS S7-200 PPI and S7-300 MPI Pass-through Settings

EasyBuilder Pro supports SIEMENS S7-200 PPI and S7-300 MPI pass-through feature.

29.5.1. EasyBuilder Pro Settings

Launch EasyBuilder Pro, go to [System Parameter Settings] » [Device list], and then add SIEMENS S7-200 PPI or S7-300 MPI device. Click [Pass-Through Settings] and the following dialog box appears.

IEMENS PLC Pass-Through Settings	Settings
Disable pass-through	
☑ Designate client IP	
IP address: 192 . 168 . 0 . 119	
OK Cancel	
Interval of block pack (words) : 5	
Interval of block pack (words) : 5	-
	n Settings

Setting	Description
Disable pass-through	Select this check box to disable pass-through
	mode. By default this check box is not selected.
Designate client IP	Designate client HMI IP address used in
	pass-through mode.

29.5.2. S7-200 PPI Connection

Confirm that the HMI used in pass-through communication is started and connected to the network. Launch STEP 7 Micro/Win, open [Communications] dialog box, and then search for the HMI IP address. Connect the HMI to communicate.



nmunications		
Address Host: Remote: PLC Type:	FAE-PC1 192 . 168 . 1 . 28 Unknown	TCP/IP(Auto) -> NVIDIA nForce Networ Host: FAE-PC1 Unknown 192.168.1.28 Double-Click to Refresh
Vupdate PLC type	TCP/IP(Auto) -> NVIDIA nForce	
Protocol:	Networki TCP/IP	
Connection Timeout -		
Enter a timeout for re load may require a h	eceiving data. Connections with a high traffic igher timeout value.	
Timeout:	3 seconds	

29.5.3. S7-300 MPI Connection

Connect via virtual COM port or Ethernet.

29.5.3.1. Virtual COM Port

1. In Utility Manager run [Pass-Through], in [HMI Mode] select "MPI ISOTCP" to install virtual serial port driver. Set the HMI IP address and the COM port that connects PLC, and then start Pass-through.

Ethernet	🔿 COM port	
Virtual COM I	Port (PC <-> PLC)	
	СОМ4	
PLC Connect	ion Port (HMI <-> PLC)	
HMI Mode		Stop Pass-through
HMLIF	P: 192.168.1.235	
HMI Por	t: 102	
	СОМ 3	
Install	Uninstall	Apply

2. In STEP 7, go to [Option] » [Set PG/PC Interface]. Confirm that the interface used is "PC



Adapter(MPI)", and then click [Properties]. Select the same COM port as the virtuel serial port. In the example COM 4 is used.

Set PG/PC Interface	
Access Path LLDP / DCP	
Access Point of the Application:	
S70NLINE (STEP 7)> PC Adapter((MPI)
(Standard for STEP 7)	
Interface Parameter Assignment Used:	
PC Adapter(MPI)	P <u>r</u> operties
PC Adapter(Auto)	
E PC Adapter(PROFIBUS)	Сору
C/PPI cable(PPI)	Delete
(Parameter assignment of your PC adapter	
for an MPI network)	
⊂ Interfaces	
Add/Remove:	Select
	Cancel Help
OK	Cancel Help
	Cancel Help
Properties - PC Adapter(MPI)	Cancel Help
	Cancel Help
Properties - PC Adapter(MPI) MPI Local Connection	
Properties - PC Adapter (MPI) MPI Local Connection	
Properties - PC Adapter(MPI) MPI Local Connection	
Properties - PC Adapter (MPI) MPI Local Connection	
Properties - PC Adapter (MPI) MPI Local Connection	
Properties - PC Adapter (MPI) MPI Local Connection	
Properties - PC Adapter (MPI) MPI Local Connection	
Properties - PC Adapter (MPI) MPI Local Connection	
Properties - PC Adapter (MPI) MPI Local Connection	

3. When finished, STEP 7 can be used to Upload / Download PLC program via HMI.

29.5.3.2. Ethernet

1. In STEP 7 go to [Option] » [Set PG/PC Interface]. As shown in the following figure, select "TCP/IP(Auto) -> the name of the network interface card".





Set PG/PC Interface	×
Access Path LLDP / DCP	
Access Point of the Application:	-> NVIDIA nEorce Netv
(Standard for STEP 7)	
Interface Parameter Assignment Used: TCP/IP(Auto) -> NVIDIA nForce Networki	Properties
TCP/IP -> VirtualBox Host-Only E ▲ CP/IP(Auto) -> NVIDIA nForce Ne	Diagnostics
🕮 TCP/IP(Auto) -> TAP-Win32 Adapt 👝	Сору
CCP/IP(Auto) -> TAP-Windows Ada	Delete
(Assigning Parameters for the IE-PG access to your NDIS CPs with TCP/IP Protocol (RFC-1006))	
Interfaces	
Add/Remove:	Sele <u>c</u> t
ОК	Cancel Help

 Go to [PLC] » [Update station to PG], in [Target Station] select [Can be reached by means of gateway]. From left to right columns enter MPI, PLC station number, S7 Subnet ID, and HMI IP address. When finished, S7 can upload PLC program to STEP 7 via HMI.

Select Node	Address		×
Which module	e do you want to reach?		
<u>R</u> ack: <u>S</u> lot:			
Target Statior	n: C Local Can be react	ned by means of j	gateway
Enter conne	ection to target station:		1st gateway
Туре	Address	S7 subnet ID	IP address
MPI	3	0045-0001	192.168.1.235
Accessible No MPI MPI	odes		192 168 1.140 192 168 1.152
MPI			192.168.1.119
	(sure	<u>U</u> pdate	
ОК			Cancel Help

29.5.4. Registers of SIEMENS Pass-through

System registers from LW-10850 to LW-10864 are used to set or indicate pass-through status of SIEMENS devices.

For more information see "22 System Registers".

During pass-through mode, LW-10863 indicates errors and LW-10864 displays error code. The following table lists the error codes, the description of each code, and the possible reason.



(The client usually refers to STEP 7 PLC program)

Error Code	Description	Possible Reason
0	Successfully executed	
1	Prohibit client from connecting	HMI is already running
	HMI	pass-through and won't accept any
		request from other client.
2	Prohibit client from connecting	When LW-10850 is set to 1, the
	HMI	client IP for connecting HMI is
		different from the IP specified in
		LW-10858 ~ LW-10861.
3	Invalid communication	Incorrect setting in LW-10853.
	protocol	
4	Invalid PLC station number	The PLC station number specified in
		LW-10852 does not exist.
5	Delayed communication	PLC connection failure.
6	Busy communication	PLC does not accept pass-through
		request, please confirm PLC
		settings.
7	Invalid pass-through request	Environment setup failure.



30. Project Protection

This chapter explains the settings relevant to project protection.

30.1.	Overview	. 30-2
30.2.	EXOB Password	. 30-2
30.3.	Decompilation is Prohibited	. 30-3
30.4.	Disable EXOB Upload Function	. 30-3
30.5.	Project Key	. 30-3
30.6.	EMTP Password	. 30-4



EasyBuilder Pro V5.02.01

30.1. Overview

The copyright of program design must be protected. This chapter discusses how to protect the projects by settings in EasyBuilder Pro.

Note

The protected projects cannot be decrypted by the factory since they are encrypted by users, therefore, please remember your password.

30.2. EXOB Password

After editing a project (.emtp), users can compile the project to .exob format. The .exob file can be downloaded to HMI. Password can be set to protect the .exob file in [EXOB password] when compiling. (Password range: 0 ~ 4294967295)

A password will be required when attempting to decompile the .exob file back to .emtp file. If the password is entered incorrectly for three times, please restart EasyBuilder Pro.

Compiling	x
Project name : C:\Users\nicolas\Desktop\0604.cmtp EXOB file name : C:\Users\nicolas\Desktop\0604.cxob	
EXOB password : Setting (used in decompiler)	
Select the languages used on the HMI Startur Password Password : 111111 (range : 0 ~ 4294967295) OK Cancel	-
Shape size : 36 bytes Macro size : 14 bytes Address tag size : 962 bytes	^
Total size : 3496 bytes (0.00M) Free space : 33550936 bytes (32.00M) 0 error(s), 0 warning(s) succeeded	Ш
Double click error messages to modify the attributes of relative objects ! Compile Close	



30.3. Decompilation is Prohibited

After a project (.emtp) is done editing, users can compile the project to .exob format. The .exob file can be downloaded to HMI. If [Decompilation is prohibited] check box is selected when compiling, the setting in [EXOB password] will be ignored. Furthermore, the .exob file cannot be decompiled to .emtp file.

Compiling		×
Project name :	C:\Users\nicolas\Desktop\0604.cmtp	
EXOB file name :	C:\Users\nicolas\Desktop\0604.cxob	
	Decompilation is prohibited	
Select the languages	used on the HMI	
	Startup language after redownloading the project : Language 1	
✓ Language 1		
	: 36 bytes : 14 bytes	•
	962 bytes	
	: 3496 bytes (0.00M) : 33550936 bytes (32.00M)	E
0 error(s), 0 warning(s	3)	
succeeded	sages to modify the attributes of relative objects !	•
Compile		ose

30.4. Disable EXOB Upload Function

EasyBuilder Pro provides a system reserved register [LB-9033]. When this register is set ON, the .exob file cannot be uploaded. If attempting to upload an .exob file with this register set ON, the file obtained after uploading is 0 byte, and cannot be decompiled. Please reboot HMI for the changed setting to take effect.

30.5. Project Key

Projects can be restricted to run on a specific HMI. The setting is in [System Parameters Settings] » [General] » [Project protection].



Project protection			
Enable	Project key :	111111	(range : 0 ~ 4294901750)
* If this key is different from ⊦	IMI key, the projec	t won't be ex	xecuted normally.
* Use LW9046~9047 to chang	je HMI key. LB9046	indicates che	eck result (key error when status is on).

If the [Enable] check box is selected under [Project protection], please set the [Project key] (password range: 0 ~ 4294901750). System registers LW-9046 ~ LW-9047 (32-bit) can be used to set the [HMI key] for HMI. The values in LW-9046 and LW-9047 cannot be read or written by a remote device. The .exob file obtained after compiling can only be executed on HMI when [HMI key] and [Project key] match. If the keys don't match, LB-9046 is set ON. To change [HMI key], please reboot HMI.

Note

When [HMI key] and [Project key] don't match, HMI and PLC cannot communicate.

Lick the icon to download the demo project. Please confirm your internet connection.

30.6. EMTP Password

After a project (.emtp) is done editing, a password can be set to protect the .emtp file. In [System parameter] » [Security] tab, select [Enable] check box under [Project password] and click [Settings] (password range: 1 ~ 4294967295).

The password will be required when attempting to open the .emtp file.

password (EMTP file) 🔽 Enable		Setting
MTP file		×
Password :	116546	(range: 1 ~ 4294967295)
	ОК	Cancel

Note

When using "Window Copy" function, if the source file is protected by EMTP password, please enter the correct password for the system to execute window copy.



31. Memory Map

This chapter explains the settings relevant to Memory Map.

31.1.	Overview	31-2
31.2.	PIN Settings	
31.3.	Communication Flowchart	31-3
31.4.	Address Types	
31.5.	Settings	



31.1. Overview

Memory Map communication protocol is similar to IBM 3764R, and it is used when the memory data transferred seldom between two devices. When setting the two devices, one is set as Master, and another is Slave. Generally, Master and Slave do not communicate unless the data in the assigned address has changed. Once the data is synchronized, the communication will stop. The purpose of Memory Map is to keep the consistency of the assigned part of data between two devices (Master and Slave).

The corresponding addresses of Master and Slave devices should have the same property as MW (MB) address type. The size of MW (MB) in HMI is 10,000 words.

MB and MW indicate the same area of memory, for example, MB0~MBf correspond to the bits of MW0, MB10~MB1f correspond to MW1, as shown in the following table:

Device Type	Format	Range
МВ	DDDDh	DDDD:0~4095 h:0~f(hex)
MW	DDDD	DDDD:0~9999

31.2. PIN Settings

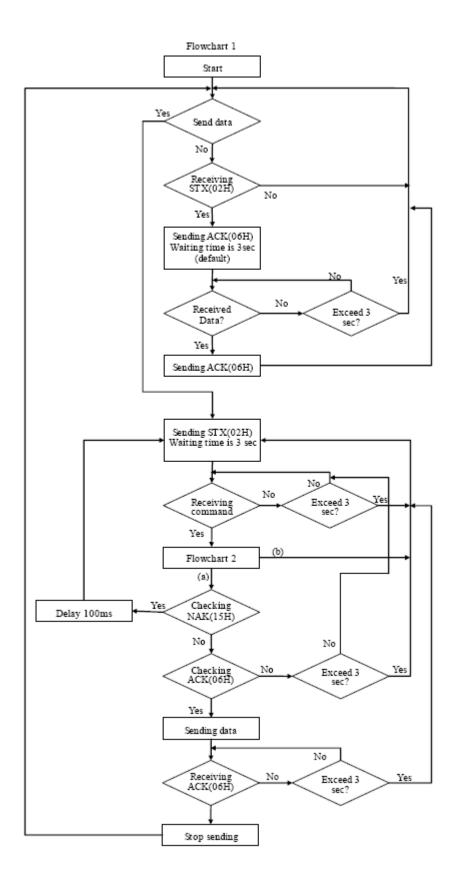
When using Memory Map communication protocol, the Master and Slave must have the same communication parameters. The wiring is shown in the following table:

(the # will be distinct	depends on the	type of PLC or controller.)
(.,

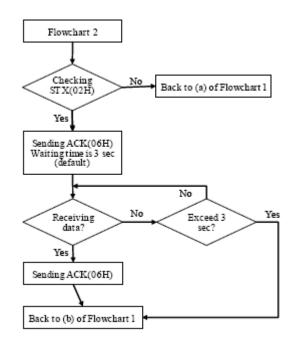
COM Port	RS-232	
Device	Master	Slave
Pin Mapping	TX(#)	RX(#)
	RX(#)	TX(#)
	GND(#)	GND(#)
COM Port	RS-485 (4W)	
Device	Master	Slave
Pin Mapping	TX+(#)	RX+(#)
	TX-(#)	RX-(#)
	RX+(#)	TX+(#)
	RX-(#)	TX-(#)
	GND(#)	GND(#)



31.3. Communication Flowchart



EasyBuilder Pro V5.02.01



Note

- Flowchart 2 works for Slave but not Master.
- STX: Start of Text, ACK: Acknowledge, NAK: Negative Acknowledge

31.4. Address Types

There are two address types, MB and MW.

The format of the commands that controls MB are listed in the following table:

MB Commands		
Offset(byte)	Format	Description
0	0x02	The operating sign to MB
1	0x##	Address (Low byte)
2	0x##	Bit Address (High byte)
		For example: MB-18 = 1*16 + 2 = 18 = 0x12 and 0x00
3	0x00	The data in MB address.
	(or 0x01)	(Bit type, must be 0 or 1)
4, 5	0x10, 0x03	Stop sign
6	0x##	The checksum. Calculate XOR from offset 0 to 5.





		MW Commands
Offset(byte)	Format	Description
0	0x01	The operating sign to MW
1	0x##	Address (Low byte)
2	0x##	Bit Address (High byte)
		If the address includes 0x10, insert another 0x10
		after it and all offsets after that are increased by 1.
		For example: 0x10, 0x04 will become
		0x10,0x10,0x04
3	0x##	Number of sending bytes (To control a word, the
		number of bytes must be even). If the number of
		bytes is 0x10, insert another 0x10 after it and all
		offsets after that are increased by 1.
4 to 4+n-1	0x##(L),0x##(H)	The address that the first and second bytes
	0x##(L),0x##(H)	correspond to is the initial address. "n" is the
		number of bytes. If the data includes 0x10, insert
		another 0x10 after it and the "Number of sending
		bytes" (offset 3) remains the same, but $n = n + 1$.
		Same thing applies to other 0x10 data.
4+n,	0x10	End sign
4+n+1	0x03	
4+n+2	0x##	The checksum. Calculate XOR from all above.

The format of the commands that controls MW are listed in the following table:

31.4.1. Communication Examples

Example 1

If Master sets the data of MW-3 to 0x0a, Master will build communication with Slave immediately due to the data changed, so Slave will update its MW-3 to 0x0a, the procedure is:

- 1. Master sends STX(0x02h).
- 2. Slave receives STX(0x02h) from Master, and sends ACK(0x06h) to Master.
- 3. Master receives ACK(0x06h) from Slave.
- 4. Master sends 0x01,0x03,0x00,0x02,0x0a,0x00,0x10,0x03,0x19, as shown in the following table:



Offset(byte)	Format	Description
0	0x01	The operating sign for MW
1	0x03	Address(Low byte)
2	0x00	Bit Address (High byte)
3	0x02	The number of bytes sent (MW-3= two bytes).
4, 5	0x0a, 0x00	Data in MW-3 is 0x0a and 0x00
6, 7	0x10, 0x03	End sign
8	0x19	The checksum
		0x01^0x03^0x00^0x02^0x0a^0x00^0x10^0x03=0x19

5. Slave receives data from Master and then sends ACK(0x06h).

6. Master receives ACK(0x06h) from Slave.

When finish communicating, Master sends the updated data in MW to Slave, and Slave synchronizes its MW data with Master.

Example 2

If the data includes 0x10; please notice the change in data format.

If MW-10 of Slave is set to 0x10, Slave will build communication with Master immediately, and Master will update its MW-10 to 0x10, the procedure is:

- 1. Slave sends STX(0x02h)
- 2. Master receives STX(0x02h) from Slave, and sends ACK(0x06h) to Slave.
- 3. Slave receives ACK(0x06h) from Master
- 4. Slave sends 0x01,0x10,0x10,0x00,0x02,0x10,0x10,0x00,0x10,0x03,0x10 as shown in the following table:

-		
Offset(byte)	Format	Description
0	0x01	The operating sign to MW
1	0x10	Address(Low byte)
2	0x10	Insert 0x10
3	0x00	Bit Address (High byte)
4	0x02	The number of bytes sent (MW-10= two bytes).
5	0x10	0x10 is the low byte in MW-10
6	0x10	Insert 0x10
7	0x00	0x00 is the high byte
8	0x10	End sign
9	0x03	
10	0x10	The checksum,
		0x01^0x10^0x10^0x00^0x02^0x10^0x
		10^0x00^0x10^0x03=0x10

5. Master receives data from Slave and sends ACK(0x06h) to Slave.

6. Slave receives ACK(0x06h) from Master.

Slave sends the updated data in MW to Master, and Master synchronizes its MW data with Slave.



31.5. Settings

The following explains how to connect two HMIs using Memory Map protocol.



If the type of these two HMIs are different, please create different project files, or, after setting the first HMI, directly change to the type of the second HMI in [Edit] » [System Parameter Settings] » [Model], and then compile and download the project to the second HMI.

31.5.1. Steps to Add a Memory Map Device

- 1. Launch EasyBuilder Pro, select [New], and the model of HMI.
- Click [Edit] form the main menu, click [System Parameter Settings], and select [Device] tab, then click [New] to add a new device.
- 3. In the [Name] field enter "Memory Map", and then select [PLC], set the [Location] to [Local].
- 4. Set [PLC type] to [Memory Map], and set [PLC I/F] to [RS-232].

Device Properties	
Name :	Memory Map
	○ HMI
Location :	Local Settings
PLC type :	Memory Map +
	V.1.10, MEMORY_MAP.e30
PLC I/F :	RS-232
COM :	COM1 (115200,E,8,1) Settings
	PLC default station no. : 0
	Default station no. use station no. variable
	Use broadcast command
	rval of block pack (words) : 5
	ad-command size (words) : 32
Max. wri	te-command size (words): 32
	OK Cancel

5. Click [Settings], and the setting is shown in the following figure.

COM Port Settings		
COM : COM 1 Baud rate : 115200 Data bits : 8 Bits Parity : Even Stop bits : 1 Bit	Timeout (sec) : Turn around delay (ms) : Send ACK delay (ms) : Parameter 1 : Parameter 2 :	0 0 0 0
	Parameter 2 . Parameter 3 : The number of resending com	0
* OS version 20120920 or later support 1	4400 baud rate OK	Cancel

- 6. After setting the COM port click [OK].
- 7. Click [OK] to finish setting.

Note

- Memory Map in MT500 is divided into [Memory Map_Master] and [MemoryMap_Slave]; please refer to the relevant manual.
- For eMT3000 and MT8000 Series, select [Memory Map] in the PLC type setting.
- [Data bit] must set to 8 bits.
- All the settings of the two HMIs must be the same.

31.5.2. Object Settings

Add two objects in window no. 10, a Toggle Switch and a Multi-state Switch:

Create a Toggle Switch Object as shown in the following steps.

- **1.** Set the [PLC name] of read address and write address to [Memory Map].
- 2. Set [Address] to MB-0.
- 3. Set [Switch style] to [Toggle]. (The picture and label of the object can be selected).

New Toggle Switch Object
General Security Shape Label
Comment :
Read Address
PLC name : Memory Map Setting
Address : MB 🔻 0
Invert signal
The Write when button is released
Switch style : Toggle



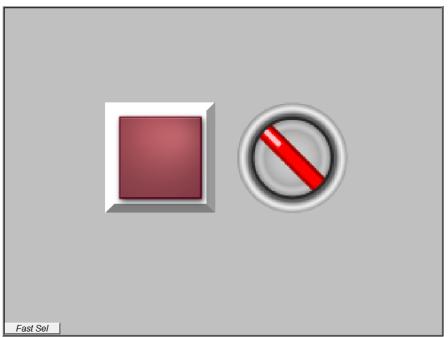
Create a Multi-state Object as shown in the following steps.

- 1. Set the [PLC name] of read address and write address to [Memory Map].
- 2. Set [Address] to MW-1.
- **3.** Set [Cyclical] to [Enable]. (The picture and label of the object can be selected).

New Multi-State Switch Object
General Security Shape Label
Comment :
Mode : Value Offset : 0
Read/Write use different addresses
PLC name : Memory Map Setting
Address : MW v 1 16-bit Unsigned
Write when button is released
Attribute
Switch style : JOG- 🔹 No. of states : 2 💌
Cyclical : Enable

31.5.3. Executing the Settings

Compile and download the same project to HMI 1 and HMI 2.



When pressing the button in one of the HMIs, the status of another one will also be changed. The way to connect a HMI with a controller is similar to the example above. The data in the same addresses of the two devices are kept identical.



32. FTP Server Application

This chapter explains how to use FTP Server.

32.1.	Overview	32-2
32.2.	Steps to Log in FTP Server	32-2
32.3.	Backup History Data and Update Recipe Data	32-3



32.1. Overview

Apart from saving the history data from an HMI to your PC by using SD card, USB disk or EasyPrinter, the FTP Server can also do the backup. After downloading a project to the HMI, the FTP Server can be used to backup or update the history data and the recipe data, but not able to delete those data.

32.2. Steps to Log in FTP Server

1. Before logging in FTP Server, please check the HMI IP address.

System information (Default HMI)					
Network\Version\					
IP Address:	192.168.1.123				
Net Mask:	255.255.255.0				
Route Address:	192.168.1.254				
Mac Address:	00:0C:26:04:51:4B				
	Ok				

2. On PC, enter the HMI IP address: ftp://192.168.1.123/ (example), then log in by the user name: uploadhis, and enter the HMI [history upload password] (if not changed, the default password is 111111). Or, directly enter "ftp://uploadhis:111111@192.168.1.123/"



 After entering the IP address, the ftp address: ftp://192.168.1.123 and its folders are shown as below:

					2 ×
Organize 🔻				₩= ▼	0
Favorites	datalog File folder	eventlog File folder			
Recent Places	operationlog File folder	pccard File folder			
📜 Libraries	File folder	usbdisk File folder			



32.3. Backup History Data and Update Recipe Data

- Steps to backup Data Sampling records
- **1.** Click "datalog" folder to view the files.
- 2. Click the file to check the records.
- 3. Copy and paste the needed files to your PC.



- Steps to backup Event (Alarm) Log records
- 1. Click "eventlog" folder to view the files.
- 2. Copy and paste the needed files to your PC.



- Steps to backup or update Recipe records
- 1. Click "recipe" folder to view the files.
- 2. Copy and paste the needed files to your PC.



Note

- Since the recipe data is automatically saved per minute, after updating "recipe.rcp" or "recipe_a.rcp", the HMI must be rebooted within one minute or the saving will be failed.
- The HMI can be rebooted by the system registers: [LB-9047] (reboot HMI) and [LB9048] (reboot HMI protection). You can set [LB-9048] ON first, and then set [LB-9047] ON to reboot the HMI.



33. EasyDiagnoser

This chapter explains how to use EasyDiagnoser.

33.1.	Overview	
33.2.	Configuration	
33.3.	EasyDiagnoser Settings	
33.4.	Error Code	33-9
33.5.	Window Adjustment	



33.1. Overview

EasyDiagnoser is a tool for detecting the error in the communication of HMI with PLC.

33.2. Configuration

The following steps explain how to configure EasyDiagnoser.

- 1. Open Utility Manager and click EasyDiagnoser.
- 2. Set the HMI IP address. Enter IP address or click [Search All], and then enter [Project Port].

n - I I III - M		
Select HMI		×
4 IP Name		4
ĮP:	192.168.1.131 -	
Project Port	8000 -	OK Exit
Select HMI		
1 IP Name		4
HMI Name:	eMT3105 -	192.168.1.118 (Default HMI)
Livii Name.		192.168.1.131 (eMT3105)
	<u>S</u> earch	192.168.1.162 (Default HMI)
		192.168.1.30 (Default HMI)

Or, during On-line simulation, right click and select [Run EasyDiagnoser] to open
EasyDiagnoser. EasyDiagnoser will monitor the communication between PC and PLC

•

ОК

Exit

Search All

Project Port: 8000

					- • •			
Siemens S7-300/ET200S (Ethernet)								
				Exit simulation				
				Run EasyDiagnoser				
				Screenshot				
96	56	3	4	9				
34	0	56	0	0				



- 🢖 Weintek HMI Diagnostic Tool EasyDiagnoser - - -🔚 🤞 🚊 🗹 <u>File View Options H</u>elp **p 🔝** Devices Command: Read + Write

 Device: All - <u>R</u>ange: 0 Index Address Type: All ~ 99999 <u>Capture</u> 0 Type Name MT8000 Series HMI No Cmd. PID Device St. Index Address / Length Time Е Location Local Block Interval 5 words Max. Read Length 256 words Max. Write Length 256 words 🗉 Sie Index SIEMENS S7/300 Ethernet Type Name Location Local Ethernet (192.168.1.170:102) PLC I/F Block Interval 5 words Max. Read Length 20 words Max. Write Length 20 words 🗣 🔝 Polling Packages 輸出 **p** 💽 Package ID Device Sta... Index Address / L... 7 (0) Local HMI -----[LB] 563 / 1 --8 (0) Local HMI [LB] 574 / 1 Siemens S... 1 Siemens S... 1 ---· 67 (16) [M] 0 / 1 ± 68 (10) --[MW] 0 / 10 Þ 📄 記錄器 📝 輸出 CAP NUM SCRL Ready
- 3. When finished, click [OK], EasyDiagnoser dialog box is shown in the following figure.

33.3. EasyDiagnoser Settings

33.3.1. Main Menu

Item	Description			
File	Save As			
	The communication data can be saved as .xls file which can be			
	opened by Excel.			
	Exit			
	Exits current file.			
View	Device Bar displays Device window.			
	Package Bar displays Package window.			
	Logger Bar displays Logger window.			
	Output Bar displays Output window.			
Options	Toolbars displays toolbar icons of Device Bar, Package Bar,			
	Logger Bar, and Output Bar.			



Help

33.3.2.

Activity Area

Status Bar displays information of CAP, NUM, and SCRL at the bottom of EasyDiagnoser window.

Update Package List displays the Polling Package information of current page.

Show Object ID (HMI) shows the ID of the objects on HMI as shown in the following figure.



Displays EasyDiagnoser version information.

communication.

Clear Activity List clears all the information recorded during

he a	ctivity	area, u	sers ca	in observe the cor	nmun	ication	between HMI and	d PLC.		
Cor	mmand:	Read +	Write	Device: All			<u>S</u> tation:	0		
<u>A</u> dd	lress Typ	e: All		Range:	0	~ 999	999	<u>C</u> aptur		
	No	Cmd.	PID	Device	St.	Index	Address / Length	Time	E.	4
	1270	R	67	Siemens S7-300/E	1		[M] 0 / 1	20	0	_
	1269	R	8	Local HMI			[LB] 574 / 1	20	0	
	1268	R	7	Local HMI			[LB] 563 / 1	20	0	
	1267	R	68	Siemens S7-300/E	1		[MW] 0 / 10	20	0	
	1266	R	8	Local HMI			[LB] 574 / 1	20	0	
	1265	R	7	Local HMI			[LB] 563 / 1	20	0	
	1264	R	67	Siemens S7-300/E	1		[M] 0 / 1	30	0	
	1263	R	68	Siemens S7-300/E	1		[MW] 0 / 10	30	0	
	1262	R	8	Local HMI			[LB] 574 / 1	20	0	
	1261	R	7	Local HMI			[LB] 563 / 1	20	0	
	1260	R	67	Siemens S7-300/E	1		[M] 0 / 1	10	0	
	4050	n	60	Ciamana 07 200/E	4		0.000 0 / 40	40	٨	Ľ

ltem	Description
Command	Read + Write
	Displays Read and Write information in activity area.
	Read
	Displays only Read information in activity area.

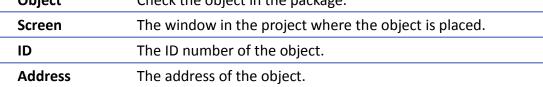


	Write
	Displays only Write information in activity area.
Device	All
	Displays the information of Local HMI and PLC.
	 If command is set to Read + Write, the Read and Write
	information of Local HMI and PLC will be displayed in the
	activity area.
	 If command is set to Read, the Read information of Local
	HMI and PLC will be displayed in the activity area.
	 If command is set to Write, the Write information of Local
	HMI and PLC will be displayed in the activity area.
	Local HMI
	Displays the information of Local HMI.
	 If command is set to Read + Write, the Read and Write
	information of Local HMI will be displayed in the activity
	area.
	 If command is set to Read, the Read information of Local
	HMI will be displayed in the activity area.
	 If command is set to Write, the Write information of Local
	HMI will be displayed in the activity area.
	PLC
	Displays the information of PLC.
	 If command is set to Read + Write, the Read and Write
	information of PLC will be displayed in the activity area.
	 If command is set to Read, the Read information of PLC
	will be displayed in the activity area.
	 If command is set to Write, the Write information of PLC
	will be displayed in the activity area.
Station	Selects the PLC station number to be displayed.
	(This function is disabled when selecting [All] in [Device]).
Address	Selects all or a preferred address type to be displayed.
Туре	(This function is disabled when selecting [All] in [Device]).
Range	Sets the range of address types.
	(This function is disabled when selecting [All] in [Device]).
Capture	Click to start/stop capturing the communication message.
Error	Please see "33.4 Error Code".



33.3.3. Polling Packages

Polling Packages							
Package ID	Device	Station	Index	Address / Lengt	h		
7 (0)	Local HMI			[LB] 563 / 1			
8 (0)	Local HMI			[LB] 574 / 1			
 67 (16) 	Siemens S7-300/ET20	1		[M] 0 / 1			
	Siemens S7-300/ET20	1		[MW] 0 / 10			
tem	Description						
ackage ID	Uses the Package ID to	check t	he erro	r of the object			
evice	Displays HMI and PLC t	ype.					
tation	Displays PLC station nu	ımber.					
ndex	Displays the index regi	ster nun	nbers o	f the objects.			
ddress /	Displays the device typ	Displays the device type and the size of the package (in					
ength	words).						
Polling Packages					×		
Object		Screen	ID	Address	*		
7 (0)	Local HMI			[LB] 563 / 1			
8 (0)	Local HMI			[LB] 574 / 1			
67 (16)	Siemens S7-300/ET	1		[M] 0 / 1			
Toggle Swite	ch 🛛	10	1	[M] 0			
Toggle Swite		10	1	[M] O	E		
	sh .	10	40	10 41 O			
Toggle Swite			12	[M] 0			
Toggle Swite	ch in the second s	10	12	[M] O			
Toggle Swite Toggle Swite	ch ch	10 10	12 13	[M] 0 [M] 0			
Toggle Swit Toggle Swit Toggle Swit	ch ch ch	10 10 10	12 13 13	[M] 0 [M] 0 [M] 0			
Toggle Swit Toggle Swit Toggle Swit Toggle Swit	ch chaile	10 10 10 10	12 13 13 14	[M] 0 [M] 0 [M] 0 [M] 0			
Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit	ch chaile	10 10 10 10 10	12 13 13 14 14	[M] 0 [M] 0 [M] 0 [M] 0 [M] 0			
Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit	ch Charles Cha	10 10 10 10 10 10 10	12 13 13 14 14 14 15	[M] 0 [M] 0 [M] 0 [M] 0 [M] 0 [M] 0			
Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit	ch chaile	10 10 10 10 10 10 10 10	12 13 13 14 14 15 15	[M] 0 [M] 0 [M] 0 [M] 0 [M] 0 [M] 0 [M] 0			
Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit	ch chaile	10 10 10 10 10 10 10	12 13 13 14 14 14 15	[M] 0 [M] 0 [M] 0 [M] 0 [M] 0 [M] 0			
Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit	ch chaile	10 10 10 10 10 10 10 10	12 13 13 14 14 15 15	[M] 0 [M] 0 [M] 0 [M] 0 [M] 0 [M] 0 [M] 0			
Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit Toggle Swit	ch chaile	10 10 10 10 10 10 10 10	12 13 13 14 14 15 15 15 16	[M] 0 [M] 0 [M] 0 [M] 0 [M] 0 [M] 0 [M] 0			





33-7



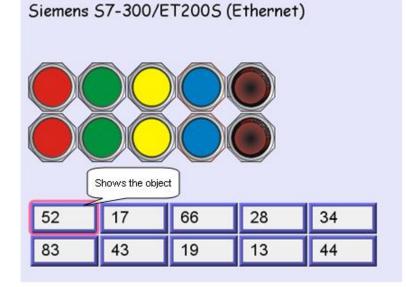
Click [Package ID], the device station number will be displayed in the 3rd column.

	Package ID	Device	Station	Index	Address / Length	
	7 (0)	Local HMI			[LB] 563 / 1	
	8 (0)	Local HMI			[LB] 574 / 1	
Ŧ	67 (16)	Siemens S7-300/ET2	1		[M] 0 / 1	
÷	68 (10)	Siemens S7-300/ET2	1		[MW] 0 / 10	

Double click [Package ID] then select [object] to display the position of the object.
 For example, select [Numeric Input] and the screen no. displays 10.

This shows that this object is in window no. 10 in the project and will be marked with pink frame on HMI as shown in the following figures

	Object		Screen	ID	Address
	7 (0)	Local HMI			[LB] 563 / 1
	8 (0)	Local HMI			[LB] 574 / 1
+	67 (16)	Siemens S7-300/ET20	1		[M] 0 / 1
	68 (10)	Siemens S7-300/ET20	1		[MW] 0 / 10
	Numeric Input		10	2	[MW] 0
	Numeric Input		10	3	[MW] 2
	Numeric Input		10	4	[MW] 4
	Numeric Input		10	5	[MW] 6
	Numeric Input		10	6	[MW] 8
	Numeric Input		10	7	[MW] 10
	Numeric Input		10	8	[MW] 12
	Numeric Input		10	9	[MW] 14
	Numeric Input		10	10	[MW] 16
	Numeric Input		10	11	[MW] 18





33.3.4. Devices

Displays the information of HMI and PLC.

Dev	vices					
	Local HMI					
	Index	0				
	Type Name	MT8000 Series HMI				
	Location	Local				
	Block Interval	5 words				
	Max. Read Length	256 words				
	Max. Write Length	256 words				
	Siemens S7-300/ET200S (Ethernet)					
	Index	1				
	Type Name	SIEMENS S7/300 Ethernet				
	Location	Local				
	PLC I/F	Ethernet (192.168.1.170:102)				
	Block Interval	5 words				
	Max. Read Length	20 words				
	Max. Write Length	20 words				

33.3.5. Output (Macro debug)

With Macro Trace function, the executing status of Macro can be seen.

In the illustration below, for [ID 1, Ln 7] and [ID 1, Ln 12]

ID 1 represents Macro name.

Ln 7 and Ln 12 represent that data are in the 7th and 12th line of Macro.

Output	$\mathbf{\Sigma}$
[ID 1, Ln 7] LW0 = 1	*
[ID 1, Ln 12] LW0 = 2 [ID 1, Ln 7] LW0 = 2	
[ID 1, Ln 12] LW0 = 3	
[ID 1, Ln 7] LW0 = 3	
[ID 1, Ln 12] LW0 = 4	
	Ŧ
 • 	- F

For more information, see "18 Macro Reference".



33.4. Error Code

In the activity area, users can find the reason of error through the error codes listed below.

- 0: Normal
- 1: Time out
- 2: Fail Error
- 12: Ignore

When error occurs, error message will be shaded red as shown in the following figure.

The error code is 1 since the PLC is disconnected with HMI.

The error code is 12 since "PLC No Response" message window is shown.

🖩 🥌 📜 🖸 Eile <u>V</u> iew <u>C</u>	ptions <u>H</u> elp							
mmand: Re	ad + Write		<u>D</u> evice: Siemen	s S7-300/ET2	00S (Ethernet)	▼ Station:	0	
dress Type:	All			~ 9999	99		<u>C</u> apture	
No	Cmd.	PID	Device	St.	Index	Address / Length	Time	E
1310	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	2063	2
1309	R	67	Siemens S7-300/ET200S (Ether	1		[M] 0 / 1	2033	2
1308	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	2023	2
1307	R	67	Siemens S7-300/ET200S (Ether	1		[M] 0 / 1	2023	2
1306	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	2023	2
1305	R	67	Siemens S7-300/ET200S (Ether	1		[M] 0 / 1	2013	2
1304	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	2023	2
1303	R	67	Siemens S7-300/ET200S (Ether	1		[M] 0 / 1	40	0
1302	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	40	0
1301	R	67	Siemens S7-300/ET200S (Ether	1		[M] 0 / 1	20	0
1300	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	30	0
1299	R	67	Siemens S7-300/ET200S (Ether	1		[M] 0 / 1	30	0
1298	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	20	0
1297	R	67	Siemens S7-300/ET200S (Ether	1		[M] 0 / 1	20	0
1296	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	50	0
1295	R	67	Siemens S7-300/ET200S (Ether	1		[M] 0 / 1	50	0
1294	R	68	Siemens S7-300/ET200S (Ether	1		[MW] 0 / 10	20	0
	_							-





33.5. Window Adjustment

Users can drag or use the smart docking icons in editing window to place the windows to a desired position.



Note

 EasyDiagnoser doesn't support Siemens S7/1200 (Ethernet) and Allen-Bradley Ethernet/IP (CompactLogix/ControlLogix) – Free Tag Names since both of the PLCs use tag.



34. Rockwell EtherNet/IP Free Tag Names

This chapter explains how to use Rockwell EtherNet / IP Free Tag Names.

34.1.	Overview	34-2
34.2.	Steps to Import User-Defined AB Tag CSV File to EasyBuilder Pro	34-2
34.3.	Steps to Add a New Data Type	34-4
34.4.	Steps to Paste	34-6
34.5.	Miscellaneous Functions	34-7
34.6.	Module-Defined	34-8



34.1. Overview

When using Rockwell EtherNet/IP Free Tags (CompactLogix/ControlLogix) driver, the User-defined tag in RSLogix5000 can be exported to .csv file, and then imported to EasyBuilder Pro.

However, the tags in User-Defined, Predefined and Module-Defined will not be exported. Structure Editor in EasyBuilder Pro is then used for importing and editing tags of data structures in User-Defined, Predefined and Module-Defined.

	A	В	С	D	E	F	
7	TYPE	SCOPE	NAME	DESCRIPT	DATATYPE	SPECIFIER	ATTRIBUTES
8	TAG		Local:1:C		AB:Embedded_IQ16F:C:0		
9	TAG		Local:1:I		AB:Embedded_IQ16F:I:0		
10	TAG		Local:2:C		AB:Embedded_OB16:C:0		
11	TAG		Local:2:I		AB:Embedded_OB16:I:0		
12	TAG		Local:2:0		AB:Embedded_OB16:0:0		
13	TAG		Array2D		DINT[25,5]		(RADIX := Decimal, Cons
14	TAG		ArrayBool		BOOL[256]		(RADIX := Decimal, Cons
15	TAG		ArrayDINT	•	DINT[130]		(RADIX := Decimal, Cons
16	TAG		ArrayReal		REAL[125]		(RADIX := Float, Constant
17	TAG		B001		INT[15]		(RADIX := Decimal, PLC)
18	TAG		b003		INT[255]		(RADIX := Decimal, PLC)
10	TAC		L1		POOT		(PADIV - Desimal Cone

34.2. Steps to Import User-Defined AB Tag CSV File to EasyBuilder Pro

1. Create Tags in RSLogix5000.

RSLogix 5000 - AB [1769-L23E-QB1 18.11]* - File Edit View Search Logic Communications					
	-				
	<u> </u>	. 🕒 🗹 🖤 🗉	र 🔍 🛛 Select a	Language	✓ Ø
Rem Run 📜 🗖 Run Mode 🛛 🛤	Path: AB_ETH-1\192.168.1.13	0\Backplane\0*			
No Forces					
No Edits		-()(U)(L)-	▶		
	Favorites Add-On A S	afety 🔏 Alarms 🔏	Bit 👗 Timer/Ci		
Controller Organizer - 4 ×	Scope: 🛐 AB 🛛 🖌 Sh	ow: All Tags			~
Controller Tags	Name <u>18</u> 4	Value 🔸	Force Mask 🛛 🗲	Style	Data Type
Controller Fault Handler	+ ABC	56		Decimal	DINT
Power-Up Handler	+ Array2D	{}	{}	Decimal	DINT[25,5]
🖨 🔄 Tasks	+ ArrayBool	{}	{}	Decimal	BOOL[256]
🖻 👼 Main Task	+ ArrayDINT	{}	{}	Decimal	DINT[130]
Hereit MainProgram Unscheduled Programs / Phases	+-ArrayReal	{}	{}	Float	REAL[125]
Motion Groups	61	0		Decimal	BOOL
Ungrouped Axes	F-INT	{}	{}	Decimal	INT[360]
🗀 Add-On Instructions		{}	{}		AB:Embedded IQ
🚍 😋 Data Types	T-Local 1:	()	{}		AB:Embedded IQ
😟 🙀 User-Defined	+ Local 2:C	{}	{}		AB:Embedded 0
🕀 🛄 Strings	+ Local 2:1	{}	{}		AB:Embedded_0
Add-On-Defined	+ Local 2:0	{}	{}		AB:Embedded_0
H Module-Defined	VarBool	{}	{}	Decimal	BOOL
Trends					
I/O Configuration	VarDint	21862		Decimal	DINT
CompactLogix5323E-QB1 System	+-Varint	0		Decimal	INT
1769-L23E-QB1 AB	VarReal	0.0		Float	REAL
😑 🛷 1769-L23E-QB1 Ethernet Port LocalEN	+-VarSint	-128		Decimal	SINT

2. Export Tags to .csv file.



🔀 RSLogix 5000 - AB [1769-L23E-QB1 18.11]*	- [Co	ontroller Tags - AB(contr	oll	er)]
🦻 File Edit View Search Logic Communications	Too	s Window Help		
Image: Controller OK Image: Controller OK	9	Options Security Documentation Languages		▶ 🙀 📴 📝 🖭 🔍 Q. Q. 3.1.130\Backplane\0* 🗸 🗸
No Edits El 1/0 0K		Iranslate PLC5/SLC Import Export	•	+/+ -()(U)(L)- X Safety X Alarms X Bit X Tir Tags and Logic Comments
Controller AB Controller Tags Controller Fault Handler Controller Fault Handler Controller Tags Controller Tags Controller Additional Controller Controll		Motion Monitor Equipment Phases Custom Tools ControlFLASH Online Books	•	<u>Component</u> 56 () () () () 0

3. In EasyBuilder Pro, add Rockwell EtherNet/IP-Tag (CompactLogix/ControlLogix) driver. Enter PLC IP address and click [Import Tag].

System Parameter	r Settings					×	💽 Open		—
Extended Me		Printer/Backu	-	e-Mail		Recipes	🔾 🗢 📕 🕨 Tags		✓ 4y Search Tags
Device	Model	General	System Sett	ing Se	curity	Font			
Device list :							Organize 🔻 New folde		8== 🔻 🗔 🔞
No.	Name	Loca			Interface	L	☆ Favorites	Name	Date modified Type
Local HMI Local PLC	4 Rockwell			120/eMT3 well Ether		- P-1 T	Desktop	AB-Controller-Tags.CSV	2013/7/24 下午 03: CSV File
EUCATTEC	+ INOCKWEII			wen Eurei	Euremer(i		Downloads		
							🔛 Recent Places		
<new< th=""><th>Delete</th><th></th><th>asyBuilder Pro</th><th>Import Tag ort tag inform</th><th></th><th>F Type</th><th>□ Libraries □ Documents □ Music □ Pictures □ Videos ● Computer ▲ Local Disk (C:) ♀ VM-Share (\\vbo ▼</th><th>4 [</th><th></th></new<>	Delete		asyBuilder Pro	Import Tag ort tag inform		F Type	□ Libraries □ Documents □ Music □ Pictures □ Videos ● Computer ▲ Local Disk (C:) ♀ VM-Share (\\vbo ▼	4 [
			<u> </u>	one may inform	actor succ	.costanyt	File <u>n</u>	ame: AB-Controller-Tags.CSV	✓ Import/Export Files (*.CSV) ▼
						OK			Open Cancel

4. In the object setting dialog, select the PLC type, and select a controller tag.

Tag : 0		Data Type	Description
	ags Test_DINT_Array	DINT[10]	Description
6	- Test_DINT_DEC - Test_DINT_DEC - Test_INT_Array - Test_INT_BIN_Array - Test_INT_DEC - Test_REAL - NEW_TEST	DINT DINT INT[6] INT INT[6] INT REAL INT	
Tag	1		- OK Cance



34.3. Steps to Add a New Data Type

Structure Editor is in the installation directory of EasyBuilder Pro. Double-click Structure Editor.exe and the editor window will appear. See the following steps.

 Right click the assigned data type (usually labeled as User-Defined), then click [New Data Type] to start editing.

🖭 Structure Editor			×
Data Types User-De New data typ ⊕-Strings New data typ	0		
			Ŧ
	Name	Data type	Description
	O member(s) Add Paste	III Edit Dele	te OK
Load from default Save to default	Help		Exit

2. Enter the name of the data type. [Description] field can be left blank. To add a member, click [Add].

Structure Editor	New data type Name TestStr Description	uct	
	Name	Data type	Description
Load from default Save to default	 0 member(s) Add Past Help 	III te Edit Dek	ete OK Exit



3. Enter the name and the data type then click [OK].

Add Data Membervoid	×
Name : Data 1	
Comment :	
·	
Data type :	
AB:1769_DI16:I:0 AB:1769_D0132:I:0 AB:1769_D016:0:0 AB:1769_D032:0:0 AB:1769_IF16:I:0 AB:Embedded_IQ16F:C:0 AB:Embedded_0B16:C:0 AB:Embedded_0B16:I:0 AB:Embedded_0B16:0:0	
Array dimensions	
Dim <u>2</u> Dim <u>1</u> Dim <u>0</u>	
Show data types by groups	
OK Cancel	

4. After adding all members, click [OK]. The new data type will be added to the list of data types.

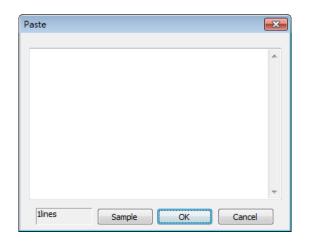
Coop MainProgram MultistateInput MultistateOutput MultistateValue NotificationClass Program PulseConverter Schedule	Structure Editor ConveyorProgram Device EventEnrolment File Group LifeSafetyPoint LifeSafetyZone	A Name Description	TestStruct			*
Schedule	MainProgram MultistateInput MultistateOutput MultistateValue NotificationClass Program	Data 1			Description	
	TrendLog TestStruct ⊕ Strings ⊕ Predefined		nember(s)	Edit Delet	te OK	, >

5. After changing the name or description of a data type, click [OK] to update.



34.4. Steps to Paste

1. When adding a patch of members, this function allows users to add multiple data in one step. First, click the [Paste] button.

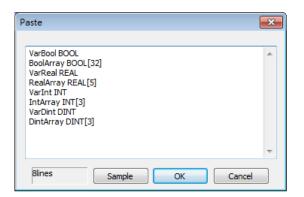


 Type in data name, and then enter data type, separated by Space key or Tab key. It is recommended that data be directly copied and pasted from RSLogix5000 to avoid errors. Users can click [Sample] for a formatting example.

escription:				
embers:	Data Type	ata Type Size: 60 byte	(s)	External Access
VarBool	BOOL	Decimal		Read/Write
BoolArray	BOOL[32]	Decimal		Read/Write
VarReal	REAL	Float		Read/Write
RealArray	REAL[5]	Float		Read/Write
VarInt	INT	Decimal		Read/Write
IntArray	INT[3]	Decimal		Read/Write
VarDint	DINT	Decimal		Read/Write
	DINT[3]	Decimal		Read/Write
DintArray				

3. Copy the needed Name and Data defined in RSLogix such as one shown above. Paste the content in the editing window, as shown in the following figure.





4. Click [OK] to finish setting, and return to the main menu to check the added data.

💽 Structure Editor							×
Device EventEnrolment File Group LifeSafetyPoint LifeSafetyZone Loop	•	Name Description	TestStruct				* *
···· MainProgram ···· MultistateInput		Name		Data type		Description	^
MultistateOutput		Data1		INT			
MultistateValue		VarBool		BOOL			=
···· NotificationClass		BoolArray		BOOL[32]			
··· Program		VarReal		REAL			
···· PulseConverter	=	RealArray		REAL[5]			
Schedule		VarInt		INT			+
TestStruct		1				+	-
····· TrendLog ⊕·· Strings ⊕·· Predefined ⊕·· Module-Defined	-	9 mer Add	nber(s) Paste	Edit	Delet	e OK	
Load from default Save to default	efault	Help				Exit	

34.5. Miscellaneous Functions

- Revising member data
 Double click the member to be revised, or click the member then click [Edit].
- Deleting member data
 Select the data to be deleted then click [Delete]. To delete all members of a data type, press and hold the Delete button on the keyboard and then click the [Delete] button in the dialog box.
- Deleting a data type
 Select the data type from the list on the left and then press the Delete key on the keyboard.
- Load from Default

To start over from default settings, click [Load from Default] button.

Save to Default

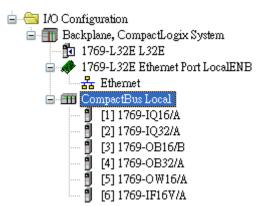
Saves data type settings to default for use in other projects.



34.6. Module-Defined

Module-Defined is a default structure of a module.

Here is an example showing how to define the default structure of a module. In RSLogix5000 [I/O Configuration], the I/O module is set.

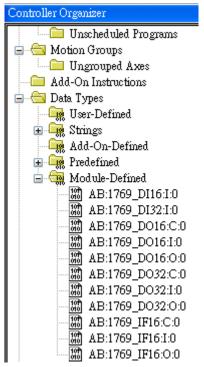


The exported CSV file will not list tags that are associated with module-defined structure. Please define the tags manually by following the steps below.

	А	В	С	D	E	F	G	Н
7	TYPE	SCOPE	NAME	DESCRIPT	DATATYPE	SPECIFIER	ATTRIBUT	TES
8	TAG		Local:1:I		AB:1769_DI16:I:0			
9	TAG		Local:2:I		AB:1769_DI32:I:0			
10	TAG		Local:3:C		AB:1769_D016:C:0			
11	TAG		Local:3:I		AB:1769_D016:I:0			
12	TAG		Local:3:0		AB:1769_D016:0:0			
13	TAG		Local:4:C		AB:1769_D032:C:0			
14	TAG		Local:4:I		AB:1769_D032:I:0			
15	TAG		Local:4:0		AB:1769_D032:0:0			
16	TAG		Local:5:C		AB:1769_D016:C:0			
17	TAG		Local:5:I		AB:1769_D016:I:0			
18	TAG		Local:5:0		AB:1769_D016:0:0			
19	TAG		Local:6:C		AB:1769_IF16:C:0			
20	TAG		Local:6:I		AB:1769_IF16:I:0			
21	TAG		Local:6:0		AB:1769_IF16:0:0			
22								

 In RSLogix5000 [Controller Organizer] » [Data Types] » [Module-Defined], double click Data Type of the module. Members of the module will be shown in a popup dialog. Copy the Name and Data Type of the members.





In Structure Editor, right click [Module-Defined], and then click [New Data Type]. In [New Data Type] » [Name], enter the Module-Defined name.

E Structure Editor MultistateValue NotificationClass Program PulseConverter Schedule TrendLog Strings	New data type Name Description		
	pe	Data type	Description
AB:Embedded_OB16 AB:Embedded_OB16 AB:Embedded_OB16 AB:Embedded_OB16 Control AB:Embedded_OB16 Control AB:Embedded_OB16	O member(s) Add Paste Help	III Edit De	lete OK



3. Click [Paste], and paste the data type information in the dialog box.

Controller Organizer 2 3 × Controller EGG Controller Tags Controller Fault Handler Power-Up Handler Tasks GMainTask MainProgram Unscheduled Programs Mon Groups	Name: Description:	AB:1769_IF4:C:0		2		
Ungrouped Axes						
🗀 Add-On Instructions 	Members:		Data 1	Type Size: 20 byte(s)		
User-Defined	Name	Data Type	Style	Description	External Access	~
🗊 🙀 Strings	Ch0Config	INT	Binary		Read/Write	
Add-On-Defined	Ch0Filter_0	BOOL	Decimal		Read/Write	=
🖬 🖙 Predefined	Ch0Filter_1	BOOL	Decimal		Read/Write	
AB:1769 IF4:C:0	Ch0Filter_2	BOOL	Decimal		Read/Write	
M AB:1769_IF4:I:U	Ch0Filter 3	BOOL	Decimal		Read/Write	
Trends	ChORange_8	BOOL	Decimal		Read/Write	
😑 😋 I/O Configuration	ChORange_9	BOOL	Decimal		Read/Write	
😑 🎹 Backplane, CompactLogix System	ChORange_10	BOOL	Decimal		Read/Write	
- 📅 1769-L35E EGG - 🛷 1769-L35E Ethernet Port LocalENB	ChORange 11		Decimal		Read/Write	
HING IN SECTOR FOR LOCALINE	Ch0DataForm		Decimal		Read/Write	
CompactBus Local	ChOD ataForm	-	Decimal		Read/Write	
[1] 1769-IF4/B	ChOD ataForm	-	Decimal		Read/Write	~
Description				ОК	Cancel Apply	Help

R Structure Editor						×
NotificationClass Program PulseConverter Schedule TrendLog Strings	•	Name Description	AB:1769_IF4:	C:0		^
		Name Ch0Config Ch0Filter_(D	Data type INT BOOL BOOL	Description	
AB: 1769_TF16:I:0 AB:Embedded_IQ16F:C:0 AB:Embedded_IQ16F:I:0 AB:Embedded_IQ16F:I:0 AB:Embedded_OB16:C:0	ш	Ch0Filter_ Ch0Filter_ Ch0Filter_ Ch0Range	2 3 _8	BOOL BOOL BOOL BOOL	Þ	-
AB:Embedded_OB16:I:0 AB:Embedded_OB16:O:0 AB:1769_IF4:C:0	-	52 mer	mber(s)	Edit	Delete OK	
Load from default Save to de	efault	Help	•		Exit	



4. Select a member and then click [Edit]. Since the data of the modules allows bit-wise operation, [Binary Access] should be selected, then click [OK] to return to Structure Editor.

Edit data mem	nber	×
Name :	Ch0Config	
Comment :		*
Data hara a	INT. 🗐 Pierrer	~
Data type :	INT V Binary acc	255
	AB: 1769_DI 16:I:0 . AB: 1769_DI 32:I:0 . AB: 1769_DO 16:0:0 . AB: 1769_DO 32:0:0 . AB: 1769_IF 16:I:0 . AB: 1769_IF 4:C:0 . AB:Embedded_IQ16F:C:0 . AB:Embedded_OB16:C:0 . AB:Embedded_OB16:C:0 .	•
Array	/ dimensions	ו ר
	Dim <u>2</u> Dim <u>1</u> Dim <u>0</u>	
0		
Short	w data types by groups	
	OK Cancel	

5. Click [OK] to finish setting.



35. EasyWatch

This chapter explains how to use EasyWatch.

Overview	35-2
Configuration	
Monitor Settings	
Macro Settings	
HMI Manager	
Object List	
	Configuration Monitor Settings Macro Settings HMI Manager



35.1. Overview

EasyWatch allows users to monitor the HMI or the PLC address values via Ethernet from the PC, or to invoke the Macro for debugging, remote monitoring, and controlling.

For example, In EasyBuilder Pro, after creating a Numeric Input Object, you can set its address to LW-10, and set the same address in EasyWatch. The value of LW-10 will be shown in EasyWatch when successfully connecting.

	Numeric Input	Object's Properties			—X —	
	General Data Descrij	Entry Numeric Format Security Shape	Font	Profile		
	a 🗔	ead/Write use different addresses				
	Read addres	Local HMI		Setting		
		Untitled - EasyWatch			-	= x
<u>F</u> ile <u>E</u> dit Obje	ects <u>H</u> elp					
🕒 🍐 🔷	12 1	🛍 🕨 🔳 🗶 🖂 🤤 🛛	3 19			
New Page 🚺						▼ X
Name	Status	HMI/PLC	Address	Address Type	Update Cycle	Value
New Monit	or 🕨 Connected	192.168.1.85 (8000) - Default HMI : HMI	LW:10	16-bit Unsigned	2500 ms	0
<		Ш.				>
or Help, press F1					CAP NUM	SCRL .:

Note

When the system register [LB-9044 (disable remote control)] or [System Parameter Settings] » [System Setting] » [Prohibit remote HMI connecting to this machine] is enabled, the feature of monitoring in EasyWatch will be unavailable.

35.2. Configuration

35.2.1. Basic Functions

Item	Description
File	New: Opens a new EasyWatch file.
	Open: Opens an existing EasyWatch file.
	Save: Saves an EasyWatch file.
	Save As: Saves an EasyWatch file to .ewt format.



	Exit: Exits EasyWatch.
Edit	Cut: Relocates the selected items to the clipboard.
	Copy: Copies the selected items to the clipboard.
	Paste: Pastes the items in the clipboard at the selected location.
Objects	Add Object: Adds new Monitor or Macro objects.
	Delete Objects : Selects the objects to be deleted, a dialog box
	appears, click [Yes] to delete.
	Modify Object: Changes the settings of the selected object.
	HMI Manager: Adds, modifies, or removes HMI settings.
	Run: Executes the selected object.
	Stop: Stops executing the selected object.
Help	Help Topics: Reference of how to operate EasyWatch.
	About EasyWatch: EasyWatch version information.

35.2.2. Quick Selection Tools

	Untitled - EasyWatch	- = X
<u>F</u> ile <u>E</u> dit	Objects Help	
P 🍰	🔶 🗶 🖻 👘 🕨 🔳 🗶 🚧 🤐 🗖	

Item	Description		
New	Opens a new EasyWatch file.		
Dpen	Opens an existing EasyWatch file.		
Save	Saves an EasyWatch file.		
🐱 Cut	Relocates the selected items to the clipboard.		
Сору	Copies the selected items to the clipboard.		
Paste	Pastes the items in the clipboard at the selected location.		
Run	Executes the selected object.		
Stop	Stops executing the selected object.		



🗱 Delete	Deletes the selected object.
Add Monitor	Adds a new address monitor.
🙀 Add Marco	Adds a new macro.
	Opens HMI Manager to add, modify, or remove
нмі	HMI settings.
Help Topics	Reference of how to operate the EasyWatch.

35.3. Monitor Settings

35.3.1. Adding a Monitor

There are two ways to create a Monitor Object.

- Select from the toolbar: [Objects] » [Add Object] » [Add Monitor].
- Select from the quick selection toolbar: [Add new address monitor].

35.3.2. Monitor Settings

Monitor Sett	ngs	
	Name : New Monitor	Read Only
HMI — Local I	Host (8000)	Open HMI Manager
	Setting	Station No. : 0 None
Address	Address : LW I0 Address : DDDDD [range : 0 ~ 10799]	
Address Ty	pe	16-bit BCD 32-bit BCD 16-bit HEX
 Num Strin 		32-bit HEX 16-bit Binary 32-bit Binary 16-bit Unsigned 16-bit Signed
Uj	odate Cyde : 2500 ms	Ok Cancel Help
ng	Description	
e	Enters an object nam	ne which is an unique name.

Enters an object name which is an unique name. **Read Only:** If an object is set to read only, its address value can't be edited.



НМІ	Select a HMI to monitor.
PLC	Selects the desired PLC to monitor and sets its type, station number, and connection method.
Address	Selects the desired object address to monitor and sets its address type.
Address Type	When the address type is set, the available formats of the address can be selected. When executing, the address will bedisplayed according to the selected format.
Update Cycle	Sets the update interval of the monitoring object. If many objects are executed simultaneously, the errors or delays may appear.

35.3.3. Steps to Add a New Monitor

 Select a target HMI. If the target HMI does not exist, click [Open HMI Manager] and click [Add] to search the HMI for adding.

	Monitor Settings	8
	Name : New Monitor	d Only
	192. 168. 1.85 (8000) - Default HMI	Open HMI Manager
	HMI Manager	
	192.168.1.85 (8000) - Default HMI	Add
		Modify
		Remove
HMI Setting	s	
⊲ IP	Name	Þ
	HMI Name: Default HMI I Search 192.168.1.58 (Default 192.168.1.63 (dMT-76 192.168.1.65 (Default 192.168.1.65 (Default 192.168.1.66 (dMT-18 192.168.1.69 (dMT-18 192.168.1.69 (dMT-18 192.168.1.69 (dMT-18 192.168.1.69 (dMT-18 192.168.1.69 (DMT-18 192.168.1.89 (DMT-18 192.188 192.180 192.180 192.189 (DMT-18 192.180 192.180 192.180 192.180 192.180 192.180 192.180	E6) HMI) i999 2D) EE) E
🔲 Use Loc	al HMI HMI Port No. : 8000 🗸	OK Cancel

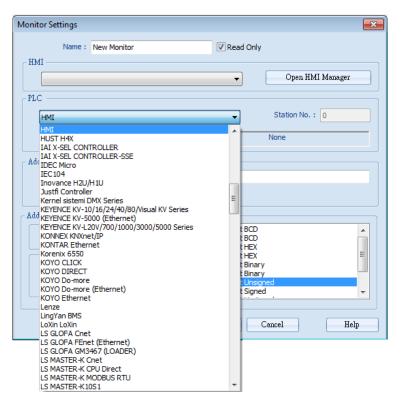
Or, select [Use Local HMI] checkbox to use the project on PC as a monitoring device.

HMI Settings		—
V Use Local HMI	HMI Port No. : 8000 -	OK Cancel

2. Select a target HMI or PLC. If one HMI is selected, you can directly control the local HMI.







If one PLC is selected, click [I/F Setting] and select [Com Port] to choose a COM port.

Monitor Settings			[83
Name : New M	onitor	Read Only		
C HMI				
			Open HMI Manager	
PLC				
FATEK FB Series		-	Station No.: 0	
I/F Setting	COM Port Settings	? 💌	rt : COM 1	-
- Address	COM 1			
Address : X	COM 2 COM 3			
Address Format : DDD	COM 4 COM 5 COM 6			
– Address Type –	COM 6 COM 7 COM 8			
Bit	COM 9 COM 10			
	COM IO			
Numeric				

Or, click [I/F Setting] and select [Ethernet] to set the IP address.

N	onitor Settings	×
	Name : New Monitor	
	HMI	
	Open HMI Manager	
	PLC	
	FATEK FB Series Station No. : 0	
	I/F Setting Ethernet Settings	_
	Address IP Address : 192 . 168 . 1 . 85	_
	Address : X	
	Address Format : DD OK Cancel	

3. Set the object address and its address type for being monitored.



Monitor Settings	×
Name : New Monitor	Read Only
	Open HMI Manager
PLC HMI	Station No. : 0
I/F Setting	None
Address	10
Address Format : DDDDD [range : 0 ~ 1	0799]

When a Word address is selected, set the address type to [Numeric] or [String].[Numeric]: Select the data format of the monitor address.

Address Type ———			
🔘 Bit		16-bit BCD 32-bit BCD 16-bit HEX	Â
Numeric		32-bit HEX 16-bit Binary 32-bit Binary	=
🔘 String	No. of Word : 1	16-bit Unsigned 16-bit Signed	-

[String]: Select the data format from [ANSI], [UNICODE], and [High/Reversed]. Set [No. of Word] for reading the number of WORD.

- Address Type				
Bit			ANSI UNICODE High/Low Reversed	
O Numeric				
String	No. of Word :	1		

5. Set the update interval of the monitor object. The range can be set from 500ms to 5000ms.

- Address Type	ANSI UNICODE High/Low Reversed
Numeric String No. of Word : 1	
Update Cycle : 2500 ms	Ok Cancel Help



35.4. Macro Settings

35.4.1. Adding a Macro

There are two ways to create a Macro object.

- Select from the toolbar: [Objects] » [Add Object] » [Add Macro].
- Select from the quick selection toolbar: [Add new macro].

35.4.2. Macro Settings

Name : New Macr	0		
∩ ^{HMI}			
		- Open	HMI Manager
_ Масто Туре			
Active Type 🕨			
	Direct Ac	tive	
1			
Macro			
		Up	
Macro		Macro ID : 0	
Macro ID :	0 🔻		
Sleep			
	00 ms 🔻		
Sleen Time • E	oo nis		
Sleep Time : 5		Down	
Sleep Time : 5			
	Add Replace	Cle	an Remove

Setting	Description						
Name	Enters an object name which is an unique name.						
НМІ	Selects a HMI to monitor.						
Macro Type	The ways to execute the Macro included Direct Active and						
	Cycle Active.						
Macro	Each Macro Object can execute multiple macros. The time						
	interval between the executions of two macros can be						
	changed.						

35.4.3. Steps to Add a New Macro

- 1. Select a target HMI. If the target HMI does not exist, add a new device, see "35.3.3 Steps to Add a New Monitor".
- Set Active Type under Macro Type to [Direct Active] or [Cycle Active].
 [Direct Active]: Directly executes Macro once.
 [Cycle Active]: Set the interval of executing Macro.



Mad	cro Settings	x
	Name : New Macro	
ſ	HMI Open HMI Manager	
_ p	facro Type ▲ctive Type → Direct Active	
	Cycle Active e (Cycle : 1 secs)	_

For example, if [Cycle Active] is set to 5 seconds, when executing a macro, the next time to execute the macro object will be 5 seconds later.

Active Cycle Settings	
Active Cycle : 5 secs	
OK Cancel	

3. Set Macro to [Macro] or [Sleep].

[Macro]: Select the Macro ID for execution, and click [Add] to add the Macro to the list.

Macro Macro ID : 0	Up Macro ID : 0
© Sleep Sleep Time : 500 ms ▼	Down
Add Replace	Clean Remove

[Sleep]: Set the time interval between the executions of two Macros. Click [Add] or [Replace] to add or replace the Macros from the list.

Macro	
Macro	Up
	Macro ID : 0 Sleep : 1000 ms
Isleep Sleep Time : 1000 ms ▼	Down
Add Replace	Clean Remove

35.5. HMI Manager

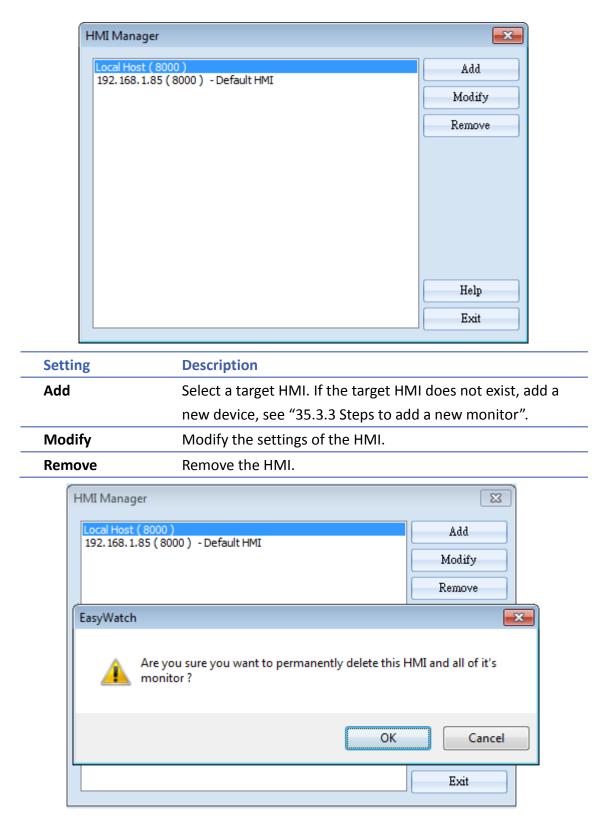
35.5.1. Opening HMI Setting

There are two ways to open the HMI Setting.

- Select from the toolbar: [Objects] » [HMI Manager].
- Select from the quick selection toolbar: [Open HMI manager].



35.5.2. HMI Manager



EasyBuilder Pro V5.02.01

35.6. Object List

35.6.1. Object List Columns

-	Untitled - EasyWatch _ 🗖 🗙
<u>File E</u> dit Objects <u>H</u>	Jelp
🕒 🌰 🔷 I 🐰	(🗈 💼 🕨 💻 🗶 🜌 🤽 🗔 😰
New Page New Page1	• × ×
	ttus HMI/PLC Address Address Type Update Cycle Value Stop Local Host (8000) : HMI LW : 10 16-bit BCD 2500 ms
<	
For Help, press F1	CAP NUM SCRL
Setting	Description
Name	Displays the object name. The small icons of the names
	are for users to identify the objects.
Status	Displays the status of the objects: [Connecting],
	[Connected], and [Stop]. If HMI is not connected or Port
	No. is incorrect, the error message "HMI Not Found" will
	appear. For Monitor Objects, if the address is incorrect,
	"Address Error" message will appear.
HMI / PLC	Displays the information of HMI / PLC that is currently
	operated by the objects.
Address	For Monitor Objects, the relevant address setting is
Address Type	displayed.
Update Cycle	Set the update interval of the monitor object.
Value	For Monitor Object, if the status shows [Connected], the
	current HMI address value will be displayed. Modifying
	the value is also available when the Read-Only checkbox
	is unselected. For Macro Object, if set to [Direct Active],
	there will be an [Active] button in this column for clicking
	to directly execute a macro.



35.6.2. Editing Object List

• Adding a new page: Click the icon to add a new page.

New Page New	Page1 💽	▼ 3	×
Name	Status	HMI/PLC	
🚧 New Monitor	📲 Stop	Local Host (8000) : HMI	

• Deleting a page: Click the icon and confirm the deletion.

New Page N	lew Page1 🚦		×X
Name	Status	HMI/PLC	
🔥 🗠 New Monit	or 📲 Stop	Local Host (8000) : HMI	
	EasyWatch		
Are		ou sure you want to permanently delete this page ?	
		OK Cancel	

• Renaming the page: Double-click on the page name and enter a new name.

New Page	New Page1						•	×
Name	Status	HMI/PLC	Address	Address Type	Update Cycle	Value		

• Positioning the column headers: Drag and drop the column headers to the desired location.

New Page	New Page1	Address Ty	лое					*>	×
Name	Status	HMI/PLC	Addr	ress	Address Type	Update Cycle	Value		
🔥 🗠 New Mon	i 💵 Stop 1	Local Hos	LW :	10	16-bit Unsigned	2500 ms			





36. Administrator Tools

This chapter explains how to setup Administrator Tools.

36.1.	Overview	
36.2.	User Accounts	
36.3.	USB Security Key	
36.4.	e-Mail SMTP Server Settings	
36.5.	e-Mail Contacts	



EasyBuilder Pro V5.02.01

36.1. Overview

Administrator Tools allows storing the data of [User Accounts], [USB Security Key], [e-Mail SMTP Server Settings], and [e-Mail Contacts] to USB. With EasyBuilder Pro user accounts and e-Mail function, the data built can be imported to HMI by Function Key object set to "Import user data / Use [USB Security Key]". The portability and convenience is greatly improved. Launch Administrator Tools, select the check boxes in [Save] column to enable the selected functions introduced in this chapter.

\iint Administrator 1	Administrator Tools		
Save	Contents of the USB data User Accounts USB Security Key e-Mail SMTP Server Settings e-Mail Contacts		
]	
Help Topics	Save to folder	•	

36.2. User Accounts

36.2.1. User Accounts Settings

Select [User Accounts] check box and complete the relevant settings.



Save		ontents of the US	B data							
	-	ser Accounts								
		SB Security Key	0							
		Mail SMTP Server Mail Contacts	r Settings							
	e-	Mail Contacts								
Jser Accou	int Settings									
No.	Secret	User name	Password	Class A	Class B	Class C	Class D	Class E	Class F	Cla
1		001001	001001							
			77							
٢										4
<	Add		III Remove			Impo	rt		Export	4
<						Impo	rt		Export	4
Effective 1) terms				Impo	rt		Export	•
Effective 1	Time ict the usin	g terms	Remove	09/2013 10:51		Impo	rt	Save to		4

Settings	Description
Secret	Select the check box to set secret accounts.
User Name	Sets User Name. *Note 1
Password	Sets User Password. *Note 1
Class A to L	Sets user privilege.
Add	Adds a new account. *Note 2
Remove	Deletes an existing account.
Import	Imports user account data.
Export	Exports user account data.
Effective Time	If [Restrict the using terms] is not selected, it is
	available to import data anytime.
	If select [Restrict the using terms] check box, and set
	an effective time period, the importing of data to
	HMI can only be done in the time limit specified.
	When time expired, data cannot be imported, please
	reset the data with this tool again.
Save to USB	Saves data to USB. To save to a specific folder, click
	"
	Save to USB





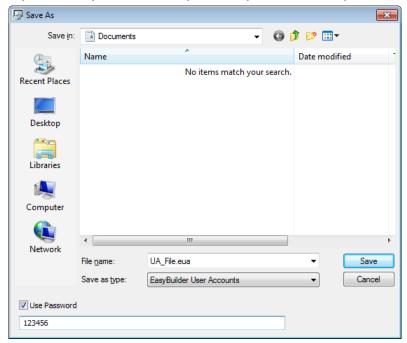
- 1. Only allows letters, numbers, "-" or "_", case-sensitive.
- 2. A maximum of 127 user accounts can be added.

36.2.2. Steps to Set User Accounts

 Click [Add] to create a new account. Click [Remove] to delete the selected account. Select [Secret] check box to define the account as a secret user. Type in [User name] and [Password] and check the privilege from [Class A] to [Class L] check boxes.

No.	Secret	User name	Password	Class A	Class B	Class C	Class D	Class E	Class F	Cla
1		Kevin	001001			V				
2	V	Fiona	002002	V			V	✓	V	
3		Katte	003003	V			V		V	
4	1	Barry	004004	V		V				
5		Susan	005005	V						
6		Carey	006006	V						
7		Paul	007007	V	V	V			V	
8	1	Ted	008008	V		1		V		
9		Jim	009009	V						
10		Mark	010010	V		1		1	V	
										,

2. To back up the data, click [Export]. Click [Use Password] to protect the data, next time when click [Import] to import the backup data, a password is required.



3. If under [Effective Time], the [Restrict the using terms] check box is selected, only during the specified time period can the users import account data to HMI. If not selected, users



can import data to HMI at any time.

Effective Time	2		
Restrict t	he using terms		
	Sep /09/2013 10:51	~	Oct /09/2013 00:00

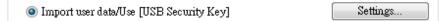
4. When finished, click [Save to USB], select the location of USB and then click [Create]. If successful, the "Generated successfully!" message is shown.

	Save to USB	
Save to USB		×
Select your H:\ <u>If your USB</u>	JSB device device is not displayed, click HERE	•
Create		Exit
	Administrator Tools	×
	Generated successfully	v !
	ОК	

36.2.3. Steps to Import Accounts Using EasyBuilder Pro

The following steps explain how to create a Function Key to import data in EasyBuilder Pro.

1. Select "Import user data/Use [USB Security Key]" in Function Key setting dialog box, and then click [Settings].



2. Under [Function mode] select [Import user accounts]. Select the device that stores the data in [Data position]. Select [Overwrite] under [Account import mode]; HMI will only store the account data imported this time. Select [Append], HMI will store the accounts imported this time and those already exist. Select [Delete file after importing user accounts] check box to delete the source files after importing.



Function mode
Import e-mail settings and contacts
Import user accounts
© Use [USB Security Key] to Login
Data position
○ SD card
Account import mode
Overwrite
Delete file after importing user accounts
OK Cancel

Click the icon to download the demo project that explains how to import user accounts by using Function Key. Please confirm your internet connection.

36.3. USB Security Key

36.3.1. USB Security Key Settings

With the predefined user login information, the USB Security Key can be used to log in directly.

😼 Administrator To	ols		
Save	Contents of the USB data		
	User Accounts		
▶ 🔽	USB Security Key		
	e-Mail SMTP Server Settings		
	e-Mail Contacts		
USB Security Key			
	User name :	Tony	
	User hame :	Tony	
	Password :	•••	
	Confirm :	•••	
Effective Time			
Restrict the	using terms		
			Save to USB
Se	p/09/2013 10:51 💌 🖶 🛛 ~	Oct /09/2013 00:00	Save to USB
Help Topics			
Help Topics			



Save to USB	Saves data to USB.
	any time.
	time period. If not specifying Effective Time, log in at
Effective Time	Logs in using USB Security Key during the specified
Confirm	Confirms User Password.
Password	Sets User Password. *Note 1
User Name	Sets User Name. *Note 1
Settings	Description



1. Only allows letters, numbers, "-" or "_", case-sensitive.

36.3.2. Steps to Set USB Security Key

1. Type in the existing user name and password. Type the password again in [Confirm] field for password confirmation.

USB Security Key		
User name :	Tony	
Password :	•••	
Confirm :	•••	

2. Under [Effective Time] if [Restrict the using terms] check box is selected, only during the specified time period can users log in using USB Security Key. If not selected, users can log in using USB Security Key at any time.

Effective Time		
Restrict the using terms		
Sep /09/2013 10:51	~	Oct /09/2013 00:00

 When finished, click [Save to USB], select the location of USB and then click [Create]. If successful, the "Generated successfully!" massage is shown.



	Save to USB	-
	1	
Save to USB		×
Select your USB dev	ice	
If your USB device i	s not displayed, click HERE	
Create		Exit
	Ļ	
Admini	strator Tools	×
	Generated successfully	1
	ОК	

36.3.3. Steps to Set USB Security Key Using EasyBuilder Pro

The following steps explain how to create a Function Key to enable USB Security Key in EasyBuilder Pro. By touching the object, the USB Security Key is enabled for login.

 Select "Import user data/Use [USB Security Key]" in Function Key setting dialog box, and then click [Settings].



2. Under [Function mode] select [Use USB Security Key to Login]. Select the device that stores the data in [Data position].





Click the icon to download the demo project that explains how to enable login using USB Security Key by using Function Key. Please confirm your internet connection.

36.4. e-Mail SMTP Server Settings

Select [e-Mail SMTP Server Settings] check box to complete the relevant settings.

Administrator Tool	S			×
Save	Contents of the USB data			
	User Accounts USB Security Key			
	e-Mail SMTP Server Settings			
	e-Mail Contacts			
Mail Settings				
SMTP Server :	smtp.example.com]		
Port :	25	Sender information		
User name :	eMT3000@example.com		еМТ3000	
Password :	•••••	Mail address :	eMT3000@example.com	
Confirm :	•••••]		
	V Log on using Secure Password Authenticat			
	Use the following type of encrypted conne	ection		
	TLS 🔻			
Help Topics			Save to USB	
ail Settings	Descriptio	on		
ail Settings ITP Server		on SMTP Server	· ·	
	Specifies	SMTP Server	port number.	
ITP Server	Specifies Specifies	SMTP Server	port number.	
ITP Server	Specifies Specifies User e-ma	SMTP Server SMTP Server	port number. ame.	
ITP Server rt er name	Specifies Specifies User e-ma User e-ma	SMTP Server SMTP Server ail account n ail account p	port number. ame.	
ITP Server rt er name ssword	Specifies Specifies User e-ma User e-ma Confirm u	SMTP Server SMTP Server ail account n ail account p iser e-mail a	port number. ame. assword.	
ITP Server rt er name ssword nfirm	Specifies Specifies User e-ma User e-ma Confirm u cion Descriptio	SMTP Server SMTP Server ail account n ail account p iser e-mail a on	port number. ame. assword.	/ed.
ITP Server rt er name ssword nfirm nder Informat	Specifies Specifies User e-ma User e-ma Confirm u tion Descriptio The sende	SMTP Server SMTP Server ail account n ail account p iser e-mail a on er name disp	port number. ame. assword. ccount password.	
ATP Server rt er name ssword nfirm nder Informat	Specifies Specifies User e-ma User e-ma Confirm u tion Descriptio The sende	SMTP Server SMTP Server ail account n ail account p iser e-mail a on er name disp er address di	port number. ame. assword. ccount password. layed when mail receiv	

36.4.1. Steps to set e-Mail SMTP Server Settings

1. Set the settings as shown in the following figure.



Mail Settings			
SMTP Server :	smtp.example.com		
Port :	25	Sender information	
User name :	eMT3000@example.com	Name :	eMT3000
Password :	•••••	Mail address :	eMT3000@example.com
Confirm :	•••••		
	V Log on using Secure Password Authentication	n (SPA)	
	Use the following type of encrypted connecti	ion	
	TLS 🔻		

2. When finished, click [Save to USB], select the location of USB and then click [Create]. If successful, the "Generated successfully!" massage is shown.

	Save to USB	•
	Ļ	
Save to USB		×
Select your USB device		
H:\		-
If your USB device is no	t displayed, click HERE	
Create		Exit
	Ļ	
Administra	tor Tools	x
0	Generated successfully	1
	ОК	

36.5. e-Mail Contacts

36.5.1. e-Mail Contacts Settings

Select [e-Mail Contacts] check box to complete the relevant settings.



Save	Contents of	the USB data						
	User Accour	nts						
	USB Security	у Кеу						
\checkmark		^o Server Settings						
▶ 🗸	e-Mail Conta	cts						
Contacts					No	of groups :	1	
				Group inform			•	
Contact	Name Mail	Address				Group A		-
John Doe		@example.com				: Basic Conta		-
Smith Do		@exmaple.com		De	scription	: Basic Conta	icts	4
John Smit		@exmaple.com		Cont	act I	Mail Address		
Mary Smit	th msmit	th@example.com		John	Doe jo	loe@example	com	-
				Smith	Doe s	doe@exmaple	.com	
			>>					
			<<					
								_
Add	Remo	ve			E	Import	Export	
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,						Inportin		

Settings	Description
Add	Adds a new contact. *Note1
Remove	Removes a contact.
No. of groups	The number of groups. *Note2
Current group	The name of current group. *Note3
Description	Group description.
Import	Imports contact information.
Export	Exports contact information.
Save to USB	Saves data to USB.

Note

- **1.** A maximum of 256 contacts can be added.
- 2. A maximum of 16 groups can be added. (Group A to Group P)
- **3.** From Group A to P, when [No. of groups] is "1", only Group A will exist, when added to "2", Group A and Group B will exist, and so on.

36.5.2. Steps to set e-Mail Contacts

- **1.** Click [Add] to add in all contacts.
- 2. Add the contacts to Group A, the added contacts are displayed in red font.



	Save Co	ontents of the USB data	
	🔲 Use	er Accounts	
		B Security Key	
		Mail SMTP Server Settings	
•	V e-N	Mail Contacts	
nt	acts		
			No. of groups : 1
	Contact Name	Mail Address	Group information
	John Doe	jdoe@example.com	Current group : Group A
	Smith Doe	sdoe@exmaple.com	Description : Basic Contacts
	John Smith	jsmith@exmaple.com	Contact Mail Address
	Mary Smith	msmith@example.com	John Doe jdoe@example.com
		_	Smith Doe sdoe@exmaple.com
			>>
			<<
	Add	Remove	Import Export

3. Press the up or down arrows of the spin box of [No. of groups] to add new groups. If add to "2", Group B can be found. Repeat step 1 and step 2 to add contacts into groups.

Smith Doe sdoe@exmaple.com John Smith jsmith@exmaple.com Mary Smith msmith@example.com >> <	Contact Name John Doe	Mail Address jdoe@example.com	No. of groups : 2 Group information Current group : Group B Description : Group A
	John Smith	jsmith@exmaple.com	Contact Mail Address John Smith jsmith@exmaple.com Mary Smith msmith@example.com

- **4.** After adding all the e-mail contacts, click [Export] to back up the data for future use and modification. Next time when needed, click [Import] to import the backup data.
- 5. When finished, click [Save to USB], select the location of USB and then click [Create]. If successful, the "Generated successfully!" massage is shown.



	Save to USB		
	1		
Save to USB			×
Select your U	JSB device		
H:\			-
If your USB of	device is not displayed, click H	<u>ERE</u>	
Create		Exi	t
	Ļ		
A	dministrator Tools	×	
	Generated succ	cessfully !	
		ок	

36.5.3. Steps to Import e-Mail Settings and Contacts Using EasyBuilder Pro

The following steps explain how to create a Function Key to import e-Mail contacts.

- 1. Select "Import user data/Use [USB Security Key]" in Function Key setting dialog box, and then click [Settings].
- 2. Under [Function mode] select [Import e-mail settings and contacts]. Select the device that stores the data in [Data position].



Click the icon to download the demo project that explains how to import e-mail settings and contacts by using Function Key. Please confirm your internet connection.



37. MODBUS TCP/IP Gateway

This chapter explains how to use MODBUS TCP/IP Gateway and configure address mapping tables.

37.1.	Overview	37-2
37.2.	Steps to Create an Address Mapping Table	37-2
37.3.	Notes about Configuring Address Mapping	37-5



37.1. Overview

To access the data of the PLC connected to HMI with SCADA software (Supervisory Control and Data Acquisition), the former way was to transfer PLC data to the HMI's local address first, and then use MODBUS TCP/IP protocol on PC to read HMI local address to get PLC data. Now by using MODBUS TCP/IP Gateway provided by EasyBuilder, the mapping of MODBUS address to PLC address can be defined first, and then one can directly use MODBUS TCP/IP protocol to access PLC data.



37.2. Steps to Create an Address Mapping Table

To create an Address Mapping Table, please follow the steps:

- 1. In [System Parameter Settings] » [Device] tab, add the PLC device. (In the example FATEK FB Series is used).
- 2. Add MODBUS Server (Ethernet), select [Enable] check box under [MODBUS TCP/IP Gateway] as shown in the following figure.



Device Properties	
Name :	MODBUS Server
	○ HMI
Location :	Local Settings
PLC type :	MODBUS Server >
	V.1.00, MODBUS_SERVER.e30
PLC I/F :	Ethernet 🔹
IP :	Local,Port=502(=HMI Port) Settings
	Station no. : 1
	Use broadcast command
-MODBUS TCP/IP Gate	way
	Enable Address Mapping Tables
	OK Cancel

3. Click [Address Mapping Tables] button and the following default tables will be displayed. Modify the tables if needed or add new tables.

Table	Description	MODBUS Address		PLC Name	Mapped PLC Address	Table Size	Read/Write
1	0x <==> LB	0x-1	<==>	Local HMI	LB-0	12400 Bit(s)	Read/Write
2	1x <==> LB	1x-1	<==	Local HMI	LB-0	12400 Bit(s)	Read only
3	3x <==> LW	3x-1	<==	Local HMI	LW-0	9999 Word(s)	Read only
4	4x <==> LW	4x-1	<==>	Local HMI	LW-0	9999 Word(s)	Read/Write
5	3x <==> RW	3x-10000	<==	Local HMI	RW-0	55536 Word(s)	Read only
5	4x <==> RW	4x-10000	<==>	Local HMI	RW-0	55536 Word(s)	Read/Write
*Note:	: No cross-table	reading/writing, i.e	. access	ing data fron	n multiple tables in one	command.	
		reading/writing, i.e the last communicat		-	· .	command. the following fur	nction codes :



4. For example, to access the data in the 50 consecutive registers of FATEK FB Series PLC starting from register D-0, configure the settings as shown in the following figure.

able Settings					
Description (1) Device type	:				
	🔘 Bit	۲	Word		
(2) Mode					
 Read, 	Write	Read only	(O Write on	ly
(3)					
MODBUS addres	s				
PLC name : M	DDBUS Server			-	
Address : 4x	:	▼ 1			
(4) Mapped PLC add	ress				
PLC name : FA	TEK FB Series			•	Setting
Address : D		▼ 0			
(5)					
Table size					
	50	Word(s)			
(6)					
Conversion					
	AB -> BA		ABCD -> 0	CDAB	
			ОК		Cancel

- (1) Select the device type of the registers to be mapped, in the example select [Word].
- (2) Select the mode to access the data in the mapped register, in the example set to [Read/Write].
- (3) Set the start address of MODBUS, in the example set to "4x-1".
- (4) Set the start address of the mapped PLC, in the example set to "D-0".
- (5) Set the range size of address mapping, in the example set to "50".
- (6) If needed, select high/low byte swap (AB->BA) or high/low word swap (ABCD->CDAB).

Table	Description	MODBUS Address		PLC Name	Mapped PLC Address	Table Size	Read/Write
1	Access D0 ~ D49	4x-1	<==>	FATEK FB Series	D-0	50 Word(s)	Read/Write

The above figure shows that MODBUS Server $4x-1 \sim 4x-50$ registers are mapped to FATEK FB Series PLC D-0 \sim D-49 registers.

5. When finished, the data of FATEK FB Series PLC D-0 ~ D-49 registers are now accessible by using MODBUS TCP/IP protocol to send read / write command to 4x-1 ~ 4x-50 registers.



37.3. Notes about Configuring Address Mapping

- UDP is not supported when using the MODBUS TCP/IP Gateway feature.
- This feature is only supported by MODBUS Server (Ethernet) interface.
- System register LW-9288 is used to indicate if data transfer has been correctly executed.
 The following error codes represent:

Value	Definition
0	Normal
1	Read or write the register that is not defined in the
	Address Mapping Table.
2	Read or write a range of registers that is not within
	the range defined in a single Address Mapping Table.
	(Or, read / write a register that is defined in other
	Address Mapping Table.)
3	The command format does not follow MODBUS
	TCP/IP protocol.
4	Modify a read-only register.
5	Read a write-only register.
6	Cannot get the correct reply from PLC within the
	specified time range.
7	Use a function code that is not supported by
	MODBUS Server.

- The defined register range must not overlap between different mapping tables.
- If [MODBUS TCP/IP Gateway] is enabled, EasyBuilder will cancel the original mapping between MODBUS Server and HMI register. That includes:
 - (1) 0x, 1x mapped to LB
 - (2) 3x, 4x mapped to LW, RW

Therefore, to access data in LB or LW register via 0x, 1x, 3x, 4x, configure the Address Mapping Table again. The following figure is an example.

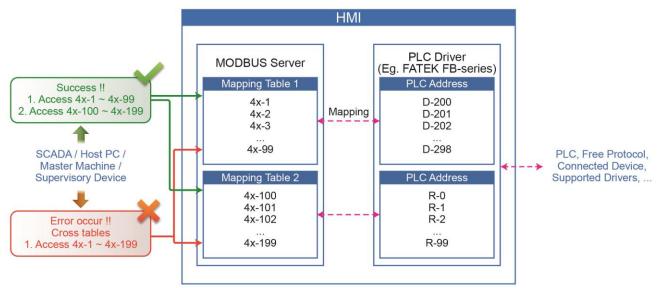
Table	Description	MODBUS Address		PLC Name	Mapped PLC Address	Table Size	Read/Write
1	0x <==> LB	0x-1	<==>	Local HMI	LB-0	12400 Bit(s)	Read/Write
2	1x <==> LB	1x-1	<==	Local HMI	LB-0	12400 Bit(s)	Read only
3	3x <==> LW	3x-1	<==	Local HMI	LW-0	9999 Word(s)	Read only
4	4x <==> LW	4x-1	<==>	Local HMI	LW-0	9999 Word(s)	Read/Write
5	3x <==> RW	3x-10000	<==	Local HMI	RW-0	55536 Word(s)	Read only
6	4x <==> RW	4x-10000	<==>	Local HMI	RW-0	55536 Word(s)	Read/Write



SCADA can only read / write the register defined in one Address Mapping Table at one time, that is, the same MODBUS command cannot access the data in the registers defined in different Address Mapping Tables.

Table	Description	MODBUS Address		PLC Name	Mapped PLC Address	Table Size	Read/Write
1	Access D200 ~ D298	4x-1	<==>	FATEK FB Series	D-200	99 Word(s)	Read/Write
2	Access R0 ~ R99	4x-100	<==>	FATEK FB Series	R-0	100 Word(s)	Read/Write

As shown in the above figure, in Mapping Table 1 set MODBUS 4x-1 to access register D-200, table size 99 words, and in Mapping Table 2 set MODBUS 4x-100 to access register R-0, table size 100 words. If using SCADA to send a command to read from 4x-1 to 4x-199, table size 199 words, since the range spans two different tables, the command will not be accepted by HMI. Instead, access the data with two separate commands (4x-1~4x-99 and 4x-100~4x-199), each reading only from one table as shown in the following figure.



38. EasyDownload

This chapter explains how to setup EasyDownload.

38.1.	Overview	. 38-2
38.2.	Configuration	. 38-2



38.1. Overview

EasyDownload allows downloading the project data file built in EasyBuilder Pro via Ethernet or USB cable. In EasyBuilder Pro main menu select [Tools] and then select [Build Data for USB Disk or SD Card Download] to build the data file before running EasyDownload.

38.2. Configuration

EasyDownload	
Download data folder	C:\Users\nicolas\Desktop\emt3000 Browse
I RW	Browse
₩_RW_A	Browse
Recipe database	Browse
 ✓ Reboot HMI after down ✓ Reset recipe ✓ Reset event log 	nload V Reset recipe database V Reset data log V Reset operation log
C USB cable	© Ethernet
IP Name HMI Name: Image: Compare the second	Add Image: Search All 192.168.1.102 (cMT-1BEE) Image: Search All Add All Search All 192.168.1.118 (c7) Image: Search All Image: Search All Image: Search All Add All Search All 192.168.1.12 (cMT-6559) Image: Search All Image
Download Down	Delete All
igs	Description
lload data folder	Browse for the project data file to download.
	Browse for the recipe data (.rcp) to download.
4	Browse for the recipe data (.rcp) to download.
e database	Browse for the recipe database (.db) to downlo
ot HMI after Ioad	If selected, HMI will reboot after downloading.



EasyBuilder Pro V5.02.01

Reset recipe Reset recipe database Delete startup screen Reset event log Reset data log Reset Operation Log	The selected files will be erased before downloading.
USB cable	Download the file to HMI via USB cable. Please make
OSD cable	sure that the USB driver is correctly installed.
Ethernet	Download the file to HMI via Ethernet.
IP	Enter the target HMI IP address.
Name	Enter the target HMI name.
Search	Search by HMI name.
Search All	Search from all the HMI names on the same network.
Add	Add the selected HMI in the Search field to the
	Destination field.
Add All	Add all the HMI in the Search field to the Destination
	field.
Destination	A list of all the HMI for download.
Delete	Delete the selected HMI in the Destination field.
Delete All	Delete all the HMI in the Destination field.
Download	Click to start download to the selected HMI in the
	Destination field.
Download All	Click to start download to all the HMI in the Destination
	field.
Password / Port No.	Enter the password and the port number of download
	set in HMI system settings.
	Download Password, Port No.
	Password : 111111 Port no. of download : 20248 (default : 20248) OK Cancel





Note

Take eMT3000 Series as an example, the built download data directory has the following structure. Please select the parent directory of the generated file when download.

Parent directory	First subdirectory	Second subdirectory
emt3000	001	
	002	
	Pub	driver
		font

- The parent directory name changes according to the model used.
- Downloading to multiple HMI is only supported by Ethernet.
- When downloading the project to multiple HMI at a time, all the HMI must use the same password and port number.
- The downloading process goes from the top of Destination list to the bottom. Only when the downloading of one HMI is done will the downloading of next HMI start.
- To avoid long waiting time for an off-line HMI, the wait time is 3 seconds.



39. Data Security

This chapter explains how to setup Data Security.

39.1.	Overview	. 39-2
39.2.	Configuration	. 39-2



39.1. Overview

Data Security allows setting restrictions on the write operations that modify local Word or Bit register data. To do so, open [System Parameter Settings] » [Device] tab, select [Local HMI] and then click [Security...] button.

191100110000 1	femory	Printer/E	Backup Serve	er e	e-Mail	Recipes
Device	Model	General	Sy	stem Setting	Security	Font
evice list :						
No.	Name		Location	Device type	Interfa	ce I.
Local HM	I Local I	HMI	Local	eMT3070 (8	00	5
•		ш				Þ
New	Dele	te	Security			
roject descript	ion :					
iojeci descript	1011.					
						*
						~ +
(N.C.1			mar a	* *
¢ CADA softwa fODBUS TCP	re can indirectly AP Server first a	access PLC da ind enable [MO	ata via MOD: DBUS TCP	BUS TCP/IP Ser /IP Gateway])	ver on HMI. (A	+ +
≪ CADA ∞ftwa fODBUS TCP	re can indirectly AP Server first a	access PLC da ind enable [MC	ata via MOD. DDBUS TCP	BUS TCP/IP Ser /IP Gateway])	ver on HMI. (A	+ Hdd a
€ CADA ∞ftwa IODBUS TCP	re can indirectly AP Server first a	access PLC da and enable [MC	DBUS TCP	/IP Gateway])		+ Hdd a
CADA softwa fodBUS TCP	/IP Server first a	raccess PLC de and enable [MC	DBUS TCP	BUS TCP/IP Ser /IP Gateway]) ss Mapping Tabl		, dd a
CADA softwa foDBUS TCP	re can indirectly AP Server first a PLC	access PLC de ind enable [MG	DBUS TCP	/IP Gateway])		dd a
<pre> « CADA softwa foDBUS TCP </pre>	/IP Server first a	access PLC da ind enable [MC	DBUS TCP	/IP Gateway])		dd a
« CADA softwa IODBUS TCP	/IP Server first a	nd enable [MO	Addre	/IP Gateway]) ss Mapping Tabl		dd a
« CADA softwa IODBUS TCP	/IP Server first's	nd enable [MO	Addre	/IP Gateway])		, dd a

39.2. Configuration

The following is the settings dialog box:



Data Security

Name : Local HMI		
e HMI	O PLC	
Location : Local	← Settings	
W protection		
V Prohibit remote-write oper	ation of remote HMI or MODBUS client	
LW range : 0	~ 0	
RW protection		
📝 Prohibit remote-write oper	ation of remote HMI or MODBUS client	
RW range : 0	~ 0	

Setting	Description
LW protection \ Prohibit remote-write operation of remote HMI or MODBUS client	If selected, a remote HMI or MODBUS client will not be able to write to the specified LW addresses.
RW protection \ Prohibit remote-write operation of remote HMI or MODBUS client	If selected, a remote HMI or MODBUS client will not be able to write to the specified RW addresses.

Click [Data Security] button to set the restrictions on the write operations that modify local Word or Bit register data.

39.2.1. Word address settings

Set the restrictions relevant to local Word addresses.



	LW-0			
Address Mode :	Word 🗸	}		
PLC name :	Local HMI		*	Setting
Address :	LW	• 0		16-bit Unsigned
Min. value :	0	Max	value: 1	0
☑ Disable re	mote-write operation			
Used only	in remote-write opera	ition		
	[,] in remote-write opera value when write value		nin. <mark>value</mark>	
Use min.		e is less than n		
Use min.	value when write value	e is less than n		
☑ Use min. ☑ Use max.	value when write value	e is less than n e is more thar	i max. value	(
Use min. • Use max. otification	value when write value value when write valu	e is less than n e is more thar	i max. value	Setting

Setting	Description	
Description	Enter the description or memo about this setting.	
Address Mode	Select [Word] to set the relevant attributes.	
Min. value	Set the minimum value that can be written to the	
	designated word address.	
Max. value	Set the maximum value that can be written to th	
	designated word address.	
Disable remote-write	If selected, the remote HMI will not be able to	
operation	write to the protected address.	
Used only in	If selected, the range between [Min. value] and	
remote-write	[Max. value] is only used to restrict the value	
operation	written by a remote device.	
Use min. value when	If selected, when the written value is less than	
write value is less	[Min. value], the system will write the specified	
than min. value	minimum value instead.	
	If not selected, when the written value is less	
	than [Min. value], the system will keep the	
	original value.	
Use max. value when	If selected, when the written value is greater than	
write value is more	[Max. value], the system will write the specified	
than max. value	maximum value instead.	



	If not selected, when the written value is greater
	than [Max. value], the system will keep the
	original value.
Notification	When the written value is not within the specified
	range between [Min. value] and [Max. value], the
	system will trigger the designated notification bit
	address.

As shown in the preceding figure, the remote HMI will not be able to write to LW-0, and when the value written to the local address is greater than 10, the value 10 is written instead, and the notification bit LB-0 will be set ON.

39.2.2. Bit address settings

Set the restrictions relevant to local Bit addresses.

Description :	LB-10			
Address Mode :	Bit 👻			
PLC name :	Local HMI		-	Setting
Address :	LB 🔻	10		
	emote-write operation			
☑ Used on	emote-write operation ly in remote-write operatio	n		
☑ Used on Notification	ly in remote-write operatio			
✓ Used on Notification ✓ Enable	ly in remote-write operatio	n © Set OFF		
☑ Used on Notification	ly in remote-write operatio	© Set OFF		Setting

Setting	Description
Description	Enter the description or memo about this setting.
Address Mode	Select [Bit] to set the relevant attributes.
Set ON only	If selected, the designated bit address can only be
	set ON.
Set OFF only	If selected, the designated bit address can only be
	set OFF.



Disable remote-write	If selected, the remote HMI will not be able to				
operation	write to the protected address.				
Used only in	If colored, the energified condition is only used to				
remote-write	If selected, the specified condition is only used to				
operation	restrict the write operation by a remote device.				
Notification	When enabled and:				
	 Select [Set ON], the system will trigger the 				
	notification bit address when attempting to				
	set OFF the protected bit address.				
	 Select [Set OFF], the system will trigger the 				
	notification bit address when attempting to				
	set ON the protected bit address.				

As shown in the preceding figure, the remote HMI can only set ON LB-10, while the local HMI is not restricted. If the remote HMI attempts to set LB-10 OFF, the system will trigger notification bit LB-0 ON.





Appendix A. Comparison of HMI Software Features

- eMT Series: eMT3070A, eMT3070B, eMT3105P, eMT3120A, eMT3150A
- cMT Series: cMT-SVR100, cMT-iV5
- mTV Series: mTV-100
- iE Series: MT8050iE, MT8070iE, MT8071iE, MT8100iE, MT8101iE
- XE Series: MT8090XE, MT8091XE, MT8121XE, MT8150XE

Series	eMT	cMT	mTV	iE	XE
Project Size	64MB	32MB	64MB	16MB	64MB
History Data Size	64MB	(*3)	64MB	16MB	(*6)
Embed Pictures in Project	✓	\checkmark	\checkmark	\checkmark	\checkmark
Embed PLC Tag Information in Project	✓	\checkmark	\checkmark	\checkmark	\checkmark
Project Protection	✓	(*4)	\checkmark	\checkmark	\checkmark
Enhanced Security	✓	\checkmark	\checkmark	\checkmark	\checkmark
Recipe Database / View Database	✓	\checkmark	\checkmark	\checkmark	\checkmark
Operation Log / Operation View	✓	N/A	N/A	\checkmark	\checkmark
e-Mail	✓	\checkmark	\checkmark	\checkmark	\checkmark
Media Player	(*1)	N/A	N/A	N/A	\checkmark
Audio Output	✓	(*5)	N/A	N/A	N/A
Video Input	(*2)	N/A	N/A	N/A	N/A
USB CAM	✓	N/A	\checkmark	N/A	\checkmark
Flow Block	✓	\checkmark	\checkmark	\checkmark	\checkmark
Combo Button	✓	\checkmark	\checkmark	\checkmark	\checkmark
Circular Trend Display	✓	\checkmark	\checkmark	\checkmark	\checkmark
Picture View	\checkmark	N/A	\checkmark	\checkmark	\checkmark
File Browser	✓	\checkmark	\checkmark	\checkmark	\checkmark
Recipe Import/Export	\checkmark	\checkmark	\checkmark	\checkmark	\checkmark
CAN Bus	✓	N/A	N/A	N/A	(*7)
VNC	✓	N/A	\checkmark	\checkmark	\checkmark
Download Project via USB Cable	✓	N/A	N/A	N/A	(*8)
Download Project via USB Disk	✓	N/A	\checkmark	\checkmark	\checkmark
Download Project via Ethernet	✓	\checkmark	\checkmark	\checkmark	\checkmark
User-Defined Boot Screen	~	N/A	~	\checkmark	\checkmark
EasySystemSetting	~	N/A	\checkmark	\checkmark	\checkmark





- 1. eMT3105P/eMT3120A/eMT3150A support Media Player.
- 2. eMT3120A/eMT3150A support NTSC and PAL analog video systems.
- 3. Maximum: 40 Data Samplings, 10,000 records for each. cMT-iV5 is not included.
- 4. cMT-iV5 does not support Project Protection.
- 5. cMT-iV5 has built-in mono speaker.
- The History Data Size of MT8090/8091XE is 120 MB; the History Data Size of MT8121/8150XE is 64MB.
- 7. MT8091XE supports CAN Bus.
- 8. MT8121/8150XE support downloading project via USB cable.

