

CNC

MELDAS 600 Series

MELDAS 60/60S Series

**PLC DEVELOPMENT SOFTWARE MANUAL
(MELSEC TOOL SECTION)**



MELSEC and MELDAS are the registered trademarks of Mitsubishi Electric Corporation.

Microsoft, Windows and Microsoft Windows NT are the registered trademarks of Microsoft Corporation in the United States and/or other countries.

Other company and product names herein may be the trademarks or registered trademarks of their respective owners.

INTRODUCTION

In the MELDAS600, 60/60S Series, the user PLC development environment is supported using MELSEC PLC development tool, which is Mitsubishi integrated FA software MELSOFT series (GX Developer), in addition to the conventional MELDAS PLC programming tool (PLC4B) and onboard ladder tool. This manual explains user PLC development environment using GX Developer, mainly usage specific to MELDAS.

Details described in this manual

CAUTION

-  Items not described in this manual must be interpreted as "not possible".
-  Some screens and functions may differ depending on the NC system or its version, and some functions may not be possible. Please confirm the specifications before use.

General precautions

Refer to the respective manuals for details of the MELDAS series PLCs and for details of various tools described in this manual. Refer to the MELSEC Series manual for details on the various tools and instructions for the MELSEC Series.

[MELDAS Series PLC related documents]

MELDAS 60/60S Series	PLC Onboard Instruction Manual	BNP-B2213
MELDAS 60/60S Series	PLC Programming Manual (Ladder section with MELSEC tool)	BNP-B2269

[MELSEC Series Software Package Manual]

GX Developer Version 8 Operating Manual (Startup Section)			
	GXDEV8-0-IN-E	13JU40	SH-080372E
GX Developer Version 8 Operating Manual			
	GXDEV8-0-E	13JU41	SH-080373E
GX Converter Version 1 Operating Manual			
	SW0D5-CNVW (OPE)-E	13J949	IB-080004E

(Caution)

- The version numbers are current as of the editing of this manual, but may be updated in the future.
- GX Developer Version 8 (Model SW8D5C-GPPW) is the new name of the old "Windows Version GPP Function Software package" (common name GPPW).

Precautions for Safety

Always read the specifications issued by the machine maker, this manual, related manuals and attached documents before installation, operation, programming, maintenance or inspection to ensure correct use.

Understand this numerical controller, safety items and cautions before using the unit. This manual ranks the safety precautions into "Danger", "Warning" and "Caution".

 DANGER	When the user may be subject to imminent fatalities or major injuries if handling is mistaken.
 WARNING	When the user may be subject to fatalities or major injuries if handling is mistaken.
 CAUTION	When the user may be subject to bodily injury or when physical damage may occur if handling is mistaken.

Note that even items ranked as " **CAUTION**", may lead to major results depending on the situation. In any case, important information that must always be observed is described.

 DANGER
Not applicable in this manual.

 WARNING
Not applicable in this manual.

 CAUTION
<p>1. Items related to product and manual</p> <ul style="list-style-type: none"> For items described as "Restrictions" or "Usable State" in this manual, the instruction manual issued by the machine maker takes precedence over this manual. An effort has been made to describe special handling of this machine, but items that are not described must be interpreted as "not possible". This manual is written on the assumption that all option functions are added. Refer to the specifications issued by the machine maker before starting use. Refer to the Instruction Manual issued by each machine maker for details on each machine tool. Some screens and functions may differ depending on the NC system or its version, and some functions may not be possible. Please confirm the specifications before use. <p>2. Precautions for startup and maintenance</p> <ul style="list-style-type: none"> Before starting program modification, forced output, RUN, STOP or similar operation during running, read the manual carefully and ensure safety fully. Not doing so can cause machine damage or accidents due to operation mistakes. <p style="text-align: right;">(Continued on next page)</p>

CAUTION

3. Items related to program development

-  Always observe the cautions before development to develop a program.
-  If the data transferred does not follow the file name rule, the NC will mistake it for another data, resulting in unexpected operation, e.g. PLC program erasure.
-  Do not read a ladder file on which a conversion error occurred into the GX Developer side. The file may include unexpected contents to result an illegal operation.
-  When an error occurred at GX Developer On-line function, the error message may not explain exactly the state in the CNC controller side.
Always refer to the error list.

CONTENTS

1. PLC DEVELOPMENT ENVIRONMENT.....	1
1.1 System Configuration	1
1.2 Software Configuration	2
1.3 Outline of GX Developer Functions and List of Supported Functions.....	4
1.3.1 Function support conditions (general section)	4
1.3.2 Function support conditions (on-line section).....	8
1.4 Memory Specifications	11
1.4.1 Memory configuration	11
1.4.2 PLC data storage method.....	11
1.4.3 Selecting the memory with GX Developer	12
2. SETUP	14
2.1 Setting up Each Tool	14
2.2 RS-232C Connection Procedures.....	14
2.3 Setting the CNC Parameters.....	15
3. PLC PROGRAM DEVELOPMENT.....	17
3.1 User PLC (Ladder) Development Procedure	18
3.2 Precautions before Development.....	19
3.3 Newly Creating a PLC Program	21
3.4 Specifying the Connection Target	22
3.4.1 Operation procedure.....	22
3.5 Starting/Stopping the PLC of the CNC Controller	23
3.5.1 Operation procedure.....	23
3.6 Writing the PLC Program to the CNC Controller.....	24
3.6.1 Operation procedure.....	24
3.6.2 Operation to be performed at write error	24
3.6.3 How to confirm the error step number.....	25
3.7 Reading the PLC Program from the CNC Controller	26
3.7.1 Operation procedure	26
3.8 Verifying the PLC Programs	28
3.8.1 Operation procedure.....	28
3.9 Monitoring the PLC Program.....	29
3.9.1 Operation procedure.....	29
3.10 Using PLC4B Type PLC Programs.....	30
3.10.1 Development procedure	30
3.10.2 Starting GX Converter and specifying the file to be converted.....	31
3.10.3 Conversion format setting.....	31

4. PLC MESSAGE DEVELOPMENT	34
4.1 Development Procedure.....	34
4.1.1 Using a general text editor.....	35
4.1.2 Entering messages directly from GX Developer	35
4.2 Message Data Description Method	36
4.2.1 Description format.....	36
4.2.2 Description method.....	37
4.2.3 Precautions	38
4.3 Converting Data into GX Developer Format	39
4.3.1 Starting GX Converter and specifying the file to be converted.....	39
4.3.2 Conversion format setting.....	39
4.4 Entering/Editing Data Using GX Developer	41
4.4.1 Interlinear statement display using circuit display	41
4.4.2 Interlinear statement display using list display	42
4.4.3 Editing of integrated type interlinear statements	43
4.5 Writing to the CNC Controller.....	44
4.6 Reading and Verifying from the CNC Controller	44
4.6.1 Menu selection/screen operation.....	44
4.6.2 Message read format.....	45
5. DEVICE COMMENT CREATION	46
5.1 Development Procedure.....	46
5.2 Description Method for Indirect Entry	47
5.3 Converting Device Comments.....	48
5.3.1 Starting GX Converter and specifying the file to be converted.....	48
5.3.2 Conversion format setting.....	48
6. RELATIONSHIPS BETWEEN GX DEVELOPER AND ONBOARD FUNCTIONS	51
6.1 Function Differences.....	51
6.2 Specification Differences	51
7. PROCEDURE FOR WRITING PLC PROGRAMS, ETC. TO ROM.....	53
7.1 Data That Can Be Written to ROM.....	53
7.2 ROM Writing Operation Procedure	53
8. LIST OF ERROR STATES	55

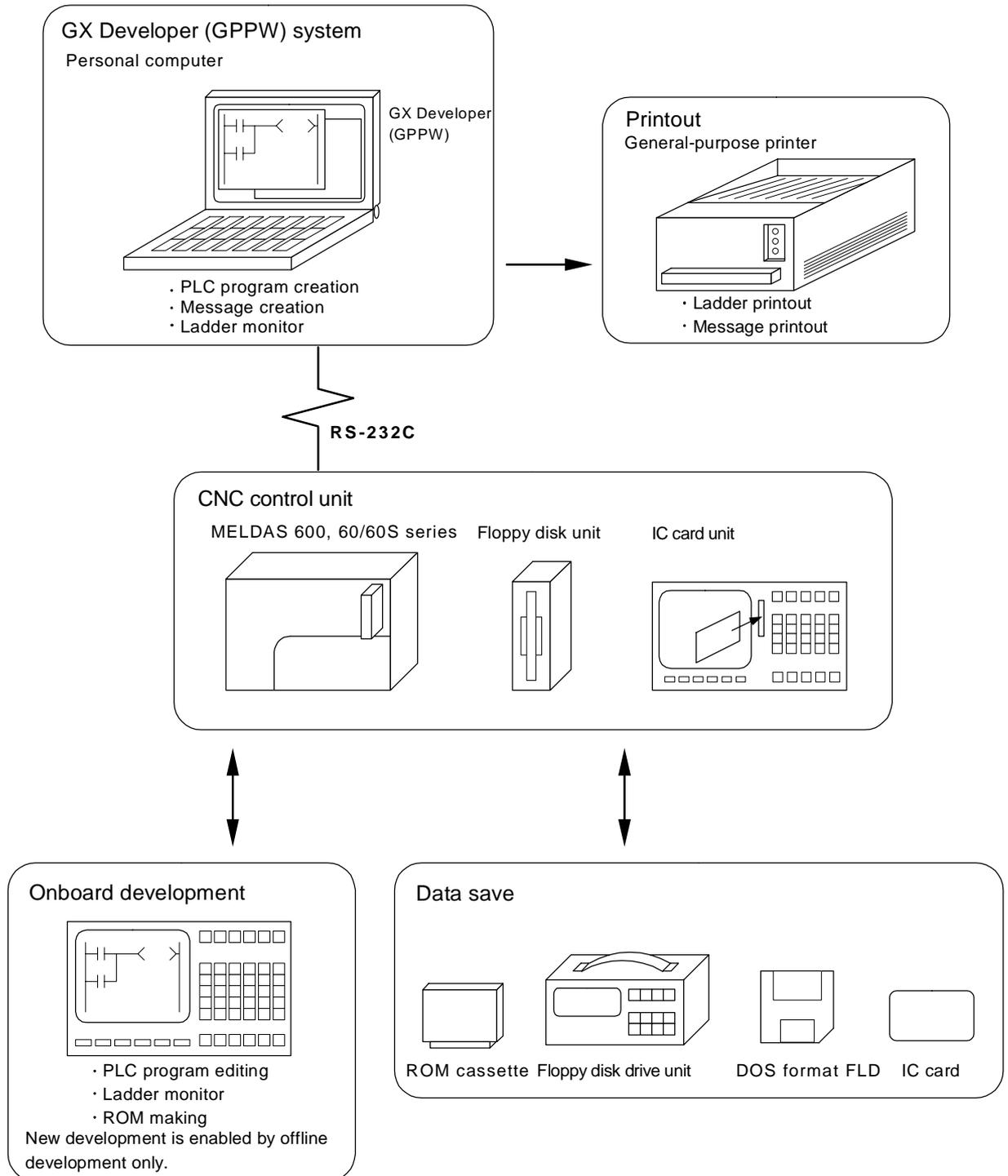
1. PLC DEVELOPMENT ENVIRONMENT
1.1 System Configuration

1. PLC DEVELOPMENT ENVIRONMENT

1.1 System Configuration

Most of the development work can be done by connecting a personal computer and a CNC unit by an RS-232C cable.

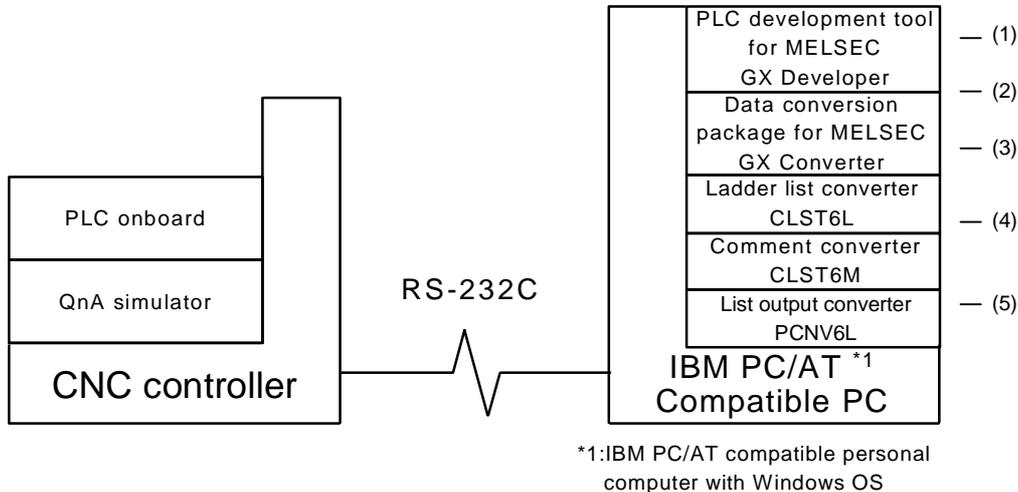
[Note] The supported peripheral devices and like change depending on the model.



1. PLC DEVELOPMENT ENVIRONMENT

1.2 Software Configuration

1.2 Software Configuration



(1) GX Developer (PLC development software package)

GX Developer is a programming software package (model name: SW8D5C-GPPW) designed for Mitsubishi Electric's MELSEC series programmable logic controllers. By performing operations similar to those of the MELSEC series, you can develop user PLC ladders for the MELDAS series. Note that some functions specific to the "MELSEC series" may not be unavailable.

For MELDAS series ladder development, we recommend you to use GX Developer Version 4 (SW4D5C-GPPW) or later. For function details, refer to the Operating Manual supplied. The DOS version "GPPQ" (SW2IVD/NX-GPPQ GPP function software package) of this package is also usable. Refer to Appendix "OPERATION METHODS USING GPPQ" for details.

(2) GX Converter (data conversion software package)

The GX Converter is a tool that carries out data conversion of GX Developer data files and the following (Start up GX converter from the GX Developer menu as a add-on tool for GX Developer):

- Ladder list files and comment text files output by the CLST6L
- Alarms and operator messages created by the text editor
- Data files of commercially available spreadsheet software, word processors and editors

This tool is a software package for various MELSEC support. GX Converter needs to be used with the versions following GX Developer Version 3 (SW3D5C-GPPW). Refer to the enclosed Operating Manual for function details.

The DOS version "CNVQ" (SW0IVD/NX-CNVQ data conversion software package) of this tool can also be used. Refer to Appendix "OPERATION METHODS USING GPPQ" for details.

(3) CLST6L (Ladder list converter)

This tool converts the user PLC ladder list data developed for the PLC4B to M600, M60/60S specification instructions and devices, and outputs the data in a ladder list format. The user PLC ladder developed in the PLC4B can be used for M600, M60/60S Series with the GX Developer by using the GX Converter to further convert the conversion results of this tool. This tool runs on Windows DOS.

1. PLC DEVELOPMENT ENVIRONMENT

1.2 Software Configuration

(4) CLST6M (Device comment converter)

This tool outputs the contact/coil comment data of a user PLC ladder developed for the PLC4B in the text format of the M600, M60/M60S series device specifications. The contact/coil comment data developed in the PLC4B can be used with GX Developer by using GX Converter to further convert the conversion results of this tool.

This tool runs on Windows DOS.

(5) PCNV6L (List output converter)

This tool outputs a MELDAS specification ladder printout image with cross information in a text format from the GX Developer specification ladder list and comment data.

This tool runs on Windows DOS.

1. PLC DEVELOPMENT ENVIRONMENT
1.3 Outline of GX Developer Functions and List of Supported Functions

1.3 Outline of GX Developer Functions and List of Supported Functions

The GX Developer functions explained here are those supported by the M600, M60/60S Series in the "off-line functions" operated with the GX Developer independently and "on-line functions" carried out connected to the M600, M60/60S controller.

Refer to the enclosed Operating Manual for function details.

Refer to "APPENDIX 1. OPERATION METHODS USING GPPQ" for the GPPQ-specific functions and operations when the DOS version "GPPQ" (SW21VD/NX-GPPQ GPP function software package) of this package is used.

1.3.1 Function support conditions (general section)

The following shows a list of GX Developer outline functions supported by the M600, M60/60S Series.

A ☉ mark indicates functions that can be used by the M600, M60/60S Series. An ✕ mark indicates that the function cannot be used because it is related to "MELSEC Series" characteristic functions. The function details during on-line are described in the next section.

List of general section functions (1) ☉ : Possible, △ : Limitedly possible, ✕ : Not possible

Program type	Support	Remarks
Ladder	☉	
List	☉	
SFC	✕	
MELSAP-L	✕	
Function block	✕	

Function	Menu	Sub menu	Support	Remarks	
Project	New project		☉		
	Open project		☉		
	Close project		☉		
	Save		☉		
	Save as		☉		
	Delete project		☉		
	Verify		☉		
	Copy		☉		
	Edit Data	New		☉	
		Copy		☉	
		Delete		☉	
		Rename		☉	
	Change PLC type		△	Fixed Q4A	
	Import file	Import from GPPQ format file		☉	
		Import from GPPA format file		✕	
		Import from FXGP(DOS) format file		✕	
		Import from FXGP(WIN) format file		✕	
		Import from TEXT ,CSV format file		☉	
	Export file	Export to GPPQ format files		☉	
		Export to GPPA format files		✕	
Export to FXGP(DOS) format file			✕		
Export to FXGP(WIN) format file			✕		
Export to TEXT ,CSV format file			☉		

1. PLC DEVELOPMENT ENVIRONMENT
1.3 Outline of GX Developer Functions and List of Supported Functions

List of general section functions (2) ◎ : Possible, △ : Limitedly possible, × : Not possible

Function	Menu	Sub menu	Support	Remarks	
(Project)	Macro	Registration macros	◎		
		Macro utilize	◎		
		Delete macros	◎		
		Macro reference path	◎		
	Printer setup		◎		
	Print		◎		
	Start new GX Developer session		◎		
	Exit GX Developer		◎		
Edit	Undo		◎		
	Restore after ladder conversion		◎		
	Cut		◎		
	Copy		◎		
	Paste		◎		
	Insert line		◎		
	Delete line		◎		
	Insert row		◎		
	Delete row		◎		
	Insert NOP batch		◎		
	Delete NOP batch		◎		
	Draw line		◎		
	Delete line		◎		
	Change TC setting		◎		
	Read mode		◎		
	Write mode		◎		
	Ladder symbol	Open contact		◎	
		Close contact		◎	
		Open branch		◎	
		Close branch		◎	
		Coil		◎	
		Application instruction		◎	
		Vertical line		◎	
		Horizontal line		◎	
		Delete vertical line		◎	
		Delete horizontal line		◎	
		Rising pulse		△	Instead of DEFR
		Falling pulse		×	
		Rising pulse open branch		×	
		Falling pulse close branch		×	
		Invert operation results		×	
		Convert operation results to rising pulse		×	
		Convert operation results to falling pulse		×	
Documentation	Comment		◎		
	Statement		◎		
	Note		◎		
	Statement/Note block edit		◎		

1. PLC DEVELOPMENT ENVIRONMENT
1.3 Outline of GX Developer Functions and List of Supported Functions

List of general section functions (3) ◎ : Possible, △ : Limitedly possible, ✕ : Not possible

Function	Menu	Sub menu	Support	Remarks	
Find/Replace	Find device		◎		
	Find instruction		◎		
	Find step no.		◎		
	Find character string		◎		
	Find contact or coil		◎		
	Replace device		◎		
	Replace instruction		◎		
	Change open/close contact		◎		
	Replace character string		◎		
	Replace unit head I/O No.		✕		
	Replace statement/note type		◎		
	List of used contact coils		◎		
	List of used devices		◎		
Convert	Convert		◎		
	Convert (All programs being edited)		◎		
	Convert (Online change)		✕		
View	Comment		◎		
	Statement		◎		
	Note		◎		
	Alias		◎		
	Macro instruction format display		◎		
	Comment format	4*8 characters		◎	
		3*5 characters		◎	
	Alias format display	Replace device name and display		◎	
		Arrange with device and display		◎	
	Toolbar		◎		
	Status bar		◎		
	Zoom	50%		◎	
		75%		◎	
		100%		◎	
		150%		◎	
		Specify		◎	
		Auto		◎	
	Project data list		◎		
	Instruction list		◎		
Set the contact	9 contacts		◎		
	11 contacts		◎		
Elapsed time		✕			
Online	Refer to "List of on-line section functions"	Refer to "List of on-line section functions"			
Diagnostics	PLC diagnostics		✕		
	MELSECNET(II)/10/H diagnostics		✕		
	Ethernet diagnostics		✕		
	CC-Link/CC-Link/LT diagnostics		✕		
	System monitor		✕		
	Online module change		✕		

1. PLC DEVELOPMENT ENVIRONMENT
1.3 Outline of GX Developer Functions and List of Supported Functions

List of general section functions (4) ◎ : Possible, △ : Limitedly possible, ✕ : Not possible

Function	Menu	Sub menu	Support	Remarks	
Tools	Check program		◎		
	Merge data		◎		
	Check parameter		✕		
	Transfer ROM		Read	✕	
			Write	✕	
			Verify	✕	
			Write to file	✕	
	Delete unused comments		◎		
	Clear all parameters		✕		
	IC memory card		Read IC memory card	✕	
			Write IC memory card	✕	
			Read image data	✕	
			Write image data	✕	
	Start ladder logic test		✕		
	Set TEL data		Connection	✕	
			Disconnection	✕	
			TEL data	✕	
AT command			✕		
Call book			✕		
Intelligent function utility	Utility list	✕			
Customize keys		◎			
Change display color		◎			
Options		△	Limited partly		
Create start-up setting file		◎			
Window	Cascade		◎		
	Tile vertically		◎		
	Tile horizontally		◎		
	Arrange icons		◎		
	Close all windows		◎		
Help	CPU error		✕		
	Special relay/register		✕		
	Key operation list		◎		
	Product information		◎		
	Connect to MELFANSweb		◎		

1. PLC DEVELOPMENT ENVIRONMENT

1.3 Outline of GX Developer Functions and List of Supported Functions

1.3.2 Function support conditions (on-line section)

The following shows a list of GX Developer on-line functions supported by the M600, M60/60S Series.

A \odot mark indicates functions that can currently be used by the M600, M60/60S Series. An \times mark indicates that the function cannot be used because it is related to "MELSEC Series" characteristic functions.

List of on-line section functions (1) \odot : Possible, Δ : Limitedly possible, \times : Not possible

Menu	Sub menu	Detailed function	Support	Remarks
Transfer setup		PC side I/F	\odot	
		PLC side I/F	Δ	Only for QnACPU
		Other station	\times	
		Network route	\times	
		Co-existence network route	\times	
Read from PLC		Target memory	\odot	
		Title	\odot	
		File selection	\odot	
		Device data	\times	
		Program	\times	
		Common	\times	
		Local	\times	
		Refresh view	\odot	
		Free space volume	\odot	
		Create title	\times	
Write to PLC		Target memory	Δ	
		Title	\odot	
		File selection	\odot	
		Device data	\times	
		Program	\times	
		Common	\times	
		Local	\times	
		Free space volume	\odot	
		Create title	\times	
Verify with PLC		Target memory	\odot	
		Title	\odot	
		File selection	\odot	
		Program	\times	
		Refresh view	\odot	
		Free space volume	\odot	
		Create title	\times	
Write to PLC (Flash ROM)	Write the program memory to ROM		\times	
	Write to PLC (Flash ROM)		\times	
Delete PLC data		Target memory	\odot	
		Title	\odot	
		File selection	\odot	
		Refresh view	\odot	
		Free space volume	\odot	
		Create title	\times	
Change PLC data attributes			\times	
PLC user data	Read PLC user data		\times	
	Write PLC user data		\times	
	Delete PLC user data		\times	

1. PLC DEVELOPMENT ENVIRONMENT
1.3 Outline of GX Developer Functions and List of Supported Functions

List of on-line section functions (2) ◎ : Possible, △ : Limitedly possible, ✕ : Not possible

Menu	Sub menu	Detailed function	Support	Remarks	
Monitor	Monitor mode	ON/OFF state	◎		
		Scan time display	✕		
		CPU state display	◎		
	Monitor [Write mode]		✕		
	Start monitor [All windows]		◎		
	Stop monitor [All windows]		◎		
	Start monitor		◎		
	Stop monitor		◎		
	Change current value monitor [Decimal]		◎		
	Change current value monitor [Hexadecimal]		◎		
	Local device monitor				
	Device batch	Device		◎	
		Connect		◎	
		Coil		◎	
		Setting value		◎	
		Current value		◎	
		Monitor format : Bit & word		◎	
		Monitor format : Bit		◎	
		Monitor format : word		◎	
		Display : 16bit integer		◎	
		Display : 32bit integer		◎	
		Display : Real number		✕	
		Display : ASCII character		✕	
		Value : DEC		◎	
		Value : HEX		◎	
		T/C set value Reference program		◎	
		Device test		◎	
		Entry data monitor	Device		◎
	ON/OFF/Current			◎	
	Setting value			◎	
	Connect			◎	
	Coil			◎	
	Display : 16bit integer			◎	
	Display : 32bit integer			◎	
	Display : Real number			✕	
	Display : ASCII character			✕	
	Value : DEC			◎	
	Value : HEX			◎	
	T/C setting value, Local label Reference program			◎	
	Device test			◎	
	Buffer memory batch		✕		
	Monitor condition setup	Device		◎	
		Step No.		◎	
	Monitor stop condition setup	Device		◎	
		Step No.		◎	
	Program monitor list		✕		
	Interrupt program monitor list		✕		
	Scan time measurement		✕		
	Entry ladder monitor		◎		
	Delete all entry ladder		◎		

1. PLC DEVELOPMENT ENVIRONMENT
1.3 Outline of GX Developer Functions and List of Supported Functions

List of on-line section functions (3) ◎ : Possible, △ : Limitedly possible, ✕ : Not possible

Menu	Sub menu	Detailed function	Support	Remarks	
Debug	Device test	FORCE ON	◎		
		FORCE OFF	◎		
		Toggle force	◎		
		Device	◎		
		Buffer memory	✕		
		Forced input output registration/cancellation		✕	
		Debug		✕	
		Skip execution		✕	
	Partial execution		✕		
	Step execution		✕		
Trace	Sampling trace		✕		
Remote operation		PLC status	◎		
		RUN	◎		
		STOP	◎		
		PAUSE	✕		
		Latch clear	✕		
		STEP-RUN	✕		
		Reset	✕		
		Operation during RUN, STEP-RUN	✕		
		Specify execution destination	✕		
Keyword setup	Register		✕		
	Delete		✕		
	Disable		✕		
Clear PLC memory			✕		
Format PLC memory		Target memory	△	Only internal RAM other than ladder storage area.	
		Format Type	✕		
Arrange PLC memory			✕		
Set time		YY MM DD Hr. Min. Sec.	◎		
		Day of week	✕		
		Specify execution destination	✕		

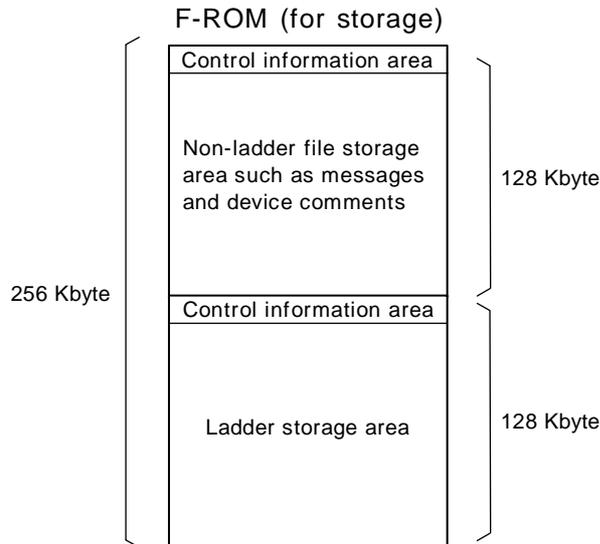
1. PLC DEVELOPMENT ENVIRONMENT

1.4 Memory Specifications

1.4 Memory Specifications

1.4.1 Memory configuration

The M600, M60/60S series stores PLC data onto flash ROM (hereafter abbreviated to the F-ROM). The following shows the storage area structure.

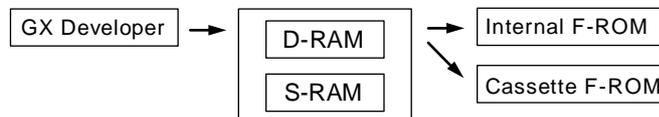


1.4.2 PLC data storage method

The PLC data transferred from GX Developer are stored in either of the following two storage methods, which differs depending on the model.

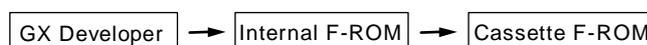
(1) Two-step storage method (M600 series)

As for the transferred data, the PLC ladder code is first stored into the nonvolatile RAM area (hereafter abbreviated to the S-RAM) designed for execution. The non-ladder PLC data are stored into the normal RAM area (hereafter abbreviated to the D-RAM). Write them to the F-ROM on the CNC controller side I/O screen. At power-ON of the CNC controller, the data are transferred from the F-ROM to the S-RAM and D-RAM areas, where they will be executed.



(2) Direct storage method (M60/60S series)

The transferred data are stored into the F-ROM area designed for both storage and execution. They can also be copied to the cassette F-ROM area for maintenance purpose.



1. PLC DEVELOPMENT ENVIRONMENT

1.4 Memory Specifications

1.4.3 Selecting the memory with GX Developer

(1) Areas that can be selected on GX Developer

The following table indicates the storage areas that can be selected for the online functions (Write to PLC, Read from PLC, Verify with PLC) of GX Developer.

Function	M600 series	M60/60S series
Write to PLC	S-RAM	Internal F-ROM
Read from PLC, Verify with PLC	S-RAM Internal F-ROM Cassette F-ROM	Internal F-ROM

1. PLC DEVELOPMENT ENVIRONMENT

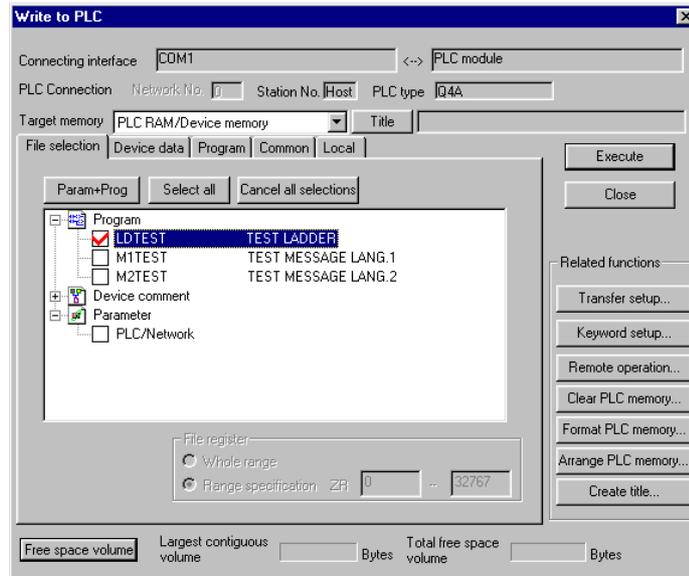
1.4 Memory Specifications

(2) Display of storage area on GX Developer

Any of the storage areas that can be selected for the online functions (Write to PLC, Read from PLC, Verify with PLC) of GX Developer can be specified as a [Target memory] item on the corresponding operation screen.

Also, pressing the [Title] button displays the comment of that storage area if PLC data exists. It is not displayed if the storage memory is not fitted or the data does not exist.

Further, pressing the [Free space volume] button displays [Total free space volume] and [Largest continuous volume] of the selected area.



Some storage areas displayed as the target memory differ from the actual storage areas of the CNC controller. The following tables indicates the relationships between the [Target memory] items and storage areas.

Table 1.4.3.1 M600 series

Screen indication	Meaning in M600 series	Title indication	Free area indication
Internal RAM	Internal S-RAM (ladder program)	ON BOARD S-RAM + (D-RAM)	Target : All storage areas of Internal RAM
IC Card A[RAM]	Internal D-RAM (non-ladder data)	ON BOARD D-RAM	Target : Non-ladder file storage area
IC Card A[ROM]	Internal F-ROM	ON BOARD F-ROM	Target : All storage areas of Internal F-ROM
IC Card B[RAM]	Not used		
IC Card B[ROM]	Cassette F-ROM	ADD ON F-ROM	Target : All storage areas of cassette F-ROM

Table 1.4.3.2 M60/60S series

Screen indication	Meaning in M60/60S series	Title indication	Free area indication
Internal RAM/ Device memory	Internal F-ROM	ON BOARD F-ROM	Target : All storage areas of Internal F-ROM
IC Card A[RAM]	Not used		
IC Card A[ROM]	Not used		
IC Card B[RAM]	Not used		
IC Card B[ROM]	Not used		

2. SETUP
2.1 Setting up Each Tool

2. SETUP

2.1 Setting up Each Tool

In the M600 M60/60S Series PLC development environment, it is assumed that the various tools are used with an IBM PC/AT compatible personal computer. Prepare each tool so that it is IBM PC/AT compatible personal computer. Refer to the enclosed Operating Manual (Startup) and Operating Manual for the setup and start procedures of each tool.

2.2 RS-232C Connection Procedures

The serial port connected with the CNC controller differs depending on the model. Also, it may require a special branch cable. Refer to the connection manual of that model.

Between the IBM PC/AT compatible personal computer that uses GX Developer and the CNC controller, use an RS-232C serial cable equivalent to the one shown below in the RS-232C connection diagram.

[Note]

The cables given in the connection diagrams of the GX Developer Operating Manual cannot be used.

Settings such as the GPPW communication speed are not required on the NC side.

NC side (25-pin D-SUB)		Cable connection and signal direction	Personal computer side (9-pin D-SUB)	
Signal name	Pin No.		Pin No.	Signal name
CD	8	-----·	1	DC
SD	2	—————→	2	RD
RD	3	←————	3	SD
DR (DSR)	6	←————	4	ER (DTR)
SG	7	←————→	5	SG
ER (DTR)	20	—————→	6	DR (DSR)
CS (CTS)	5	←————	7	RS (RTS)
RS (RTS)	4	—————→	8	CS (CTS)
	22	-----·	9	RI

* The above shows a general RS-CS method connection format.

* The pin Nos. of dotted lines are not used.

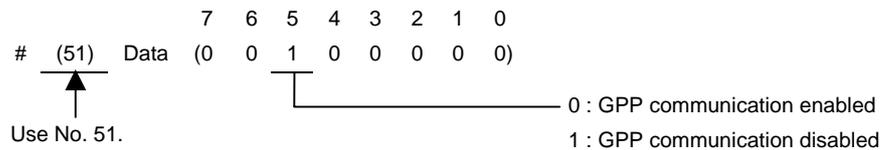
2. SETUP

2.3 Setting the CNC Parameters

2.3 Setting the CNC Parameters

Each model of the M600, M60/60S series has bit selection parameters related to GX Developer. If an appropriate value is not set in the parameters, an error will occur in communication with GX Developer.

(1) M600M series



[Reference] #51 corresponds to the low side of the file register R5025.

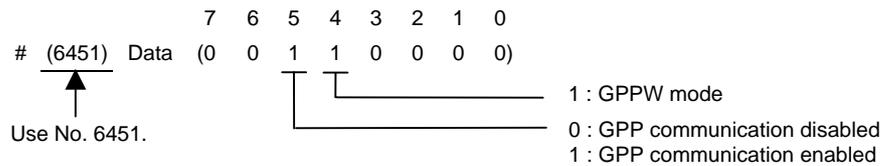
Bit 5 = 0

The serial port is used for communication with GX Developer.
(This serial port cannot be used concurrently with the other function.)

Bit 5 = 1

The serial port is not used for communication with GX Developer.
(When the serial port is used for another function)

(2) M600L series



[Reference] #6451 corresponds to the low side of the file register R2925.

Bit 4 = 1

The PLC development environment of GX Developer is used.

Bit 5 = 0

The serial port is not used for communication with GX Developer.
(When the serial port is used for another function)

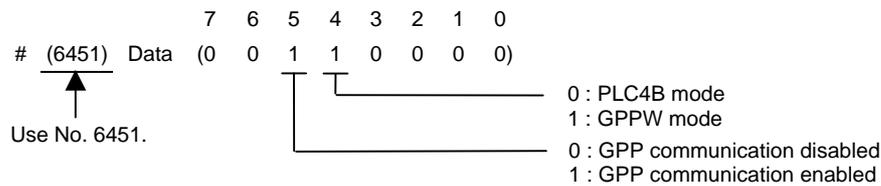
Bit 5 = 1

The serial port is used for communication with GX Developer.
(This serial port cannot be used concurrently with the other function.)

2. SETUP

2.3 Setting the CNC Parameters

(3) M60/60S series



[Reference] #6451 corresponds to the low side of the file register R2925.

(a) PLC environment selection parameter

Choose the PLC development environment. Independently of the setting of this bit, the currently stored ladder of either format operates.

Note that the onboard function is inactive if the format of the currently stored ladder differs from the environment selected here.

Bit 4 = 0

The PLC development environment of PLC4B is used.

Bit 4 = 1

The PLC development environment of GX Developer is used.

(b) GPPW serial port use selection parameter

When using the PLC development environment of GX Developer, choose whether the serial port used for communication with GX Developer will be used or not.

Bit 5 = 0

The serial port is not used for communication with GX Developer.
(When the serial port is used for another function)

Bit 5 = 1

The serial port is used for communication with GX Developer.

At this time, the onboard function is not started if bit 4 = 1.

When the GPPW serial port use selection parameter is set to "1" (enabled), GPPW occupies the communication so communication (input/output) other than GPPW cannot be used.

3. PLC PROGRAM DEVELOPMENT

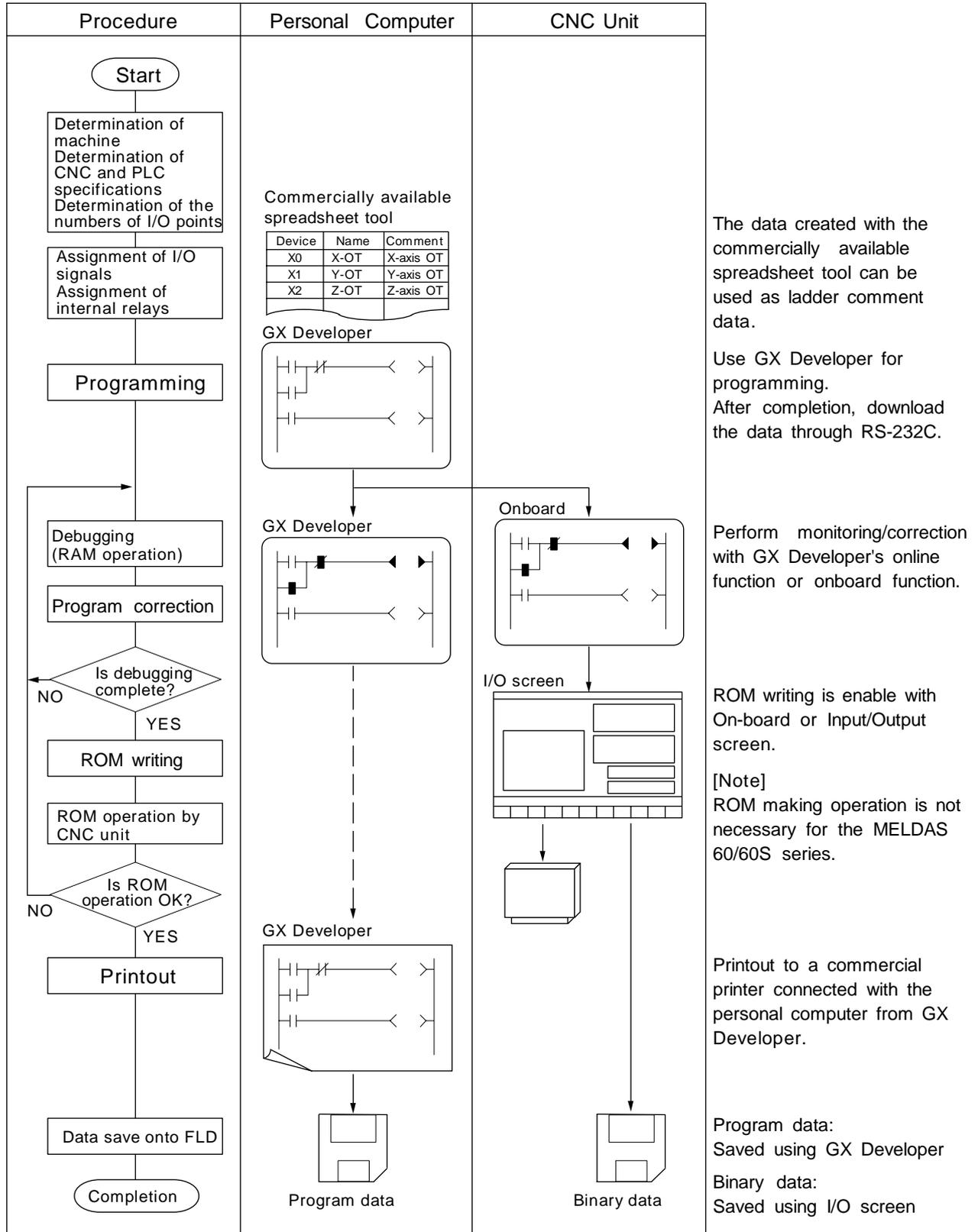
This chapter mainly describes MELDAS-specific usage about the PLC program development procedure.

3. PLC PROGRAM DEVELOPMENT

3.1 User PLC (Ladder) Development Procedure

3.1 User PLC (Ladder) Development Procedure

The following indicates a procedure for creating a general user PLC ladder.



3.2 Precautions before Development

Pay careful attention to the following items before developing ladder programs using the GX Developer.

(1) PLC Type Selection

The PLC type must be set when newly creating programs, etc. Select the following CPU type when requested to select the PLC type by the GX Developer. An error will occur during transfer of the ladder program to the CNC controller if another PLC type is selected.

 **CAUTION** Select "Q4A" for CPU type.

(2) Device Setting

Do not set the devices when developing the ladder program for the CNC controller. Develop the program with the device settings (No. of points, etc.) left at their default values applied when GX Developer was started. The ladder program cannot be transferred to the CNC controller normally when it is developed with settings other than the default values.

 **CAUTION** Do not set the devices.

(3) PLC Instructions

MELSEC-specific PLC instructions cannot be used in the ladder program development for the CNC controller. Only PLC instructions and formats in PLC programming manual can be used. The format, etc., are changed with some instructions. Refer to "Appendix 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B AND M600, M60/60S" for details.

 **CAUTION** MELSEC-specific PLC instructions cannot be used.

(4) Label at the beginning of ladder program

In a MELDAS ladder program, a processing unit is differentiated by specifying a reserved label number at the beginning of processing. There are the following different processing units. (Since programmable processings are changed depending on the model, refer to the PLC programming manual of that model.)

P251 : PLC high-speed processing program starting label

P252 : PLC main processing program starting label

P253 : Initial processing program starting label

If only the PLC main processing is to be performed, do not omit but describe the above label at the beginning of a ladder program. Unless the label is described, normal RUN cannot be performed.

 **CAUTION** Specify a label at the beginning of a ladder program.

3. PLC PROGRAM DEVELOPMENT

3.2 Precautions before Development

(5) Statements and notes

GX Developer allows a ladder program to be commented (with interlinear statements and notes). They are available in two types: integrated and peripheral.

Integrated type : Can be downloaded together with a ladder program to the CNC controller.

Peripheral type : Cannot be downloaded.

The integrated type cannot be used with the M600, M60/60S series. If it is used, a ladder program cannot be transferred to the CNC controller properly.

 **CAUTION** Do not use integrated type interlinear statements and notes.

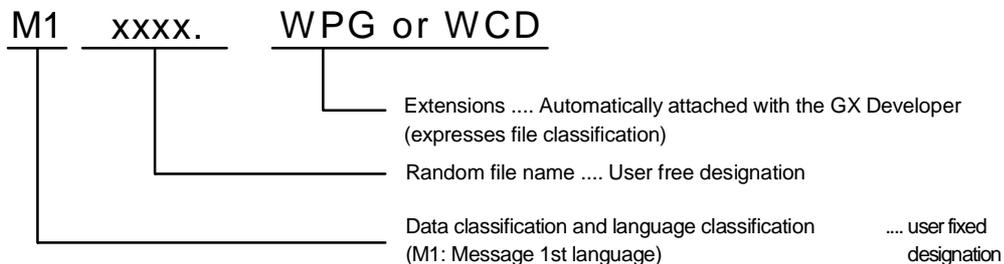
(6) File Name

Inside the M600, M60/60S series, PLC-related data are controlled and stored in the following categories. Therefore, they are also developed in the same categories.

 **CAUTION**

 If the data transferred does not follow the file name rule, the NC will mistake it for another data, resulting in unexpected operation, e.g. PLC program erasure.

File name rule



When data is transferred by GX Developer, its data type is distinguished by the file name.

An extension indicates a file type, and the first two characters denote a data type and a language type.

The file name can be designated with eight characters (excluding the extension) which contain the data type and language type.

[Note] Unusable file name

Data type : "M", "C" or "H" (alphabet)

Language type : "0" to "9" (numeral)

at the beginning of a file name are reserved on the NC side. Do not use the file name of this combination.

3. PLC PROGRAM DEVELOPMENT

3.2 Precautions before Development

List of PLC related data

	Related data classification	File name (GX Developer)	Remarks
1	PLC program (ladder)	zzzzzz.WPG	PLC ladder code
2	PLC program comment	zzzzzz.WCD	Comment data for GX Developer
3	Message 1st language	M1xxxx.WPG	1st language for alarm/operator messages, PLC switches, tool registration comments/load meter comments
4	Message 2nd language	M2xxxx.WPG	Same as above (2nd language data)

(a) PLC program (ladder)

- Ladder program developed using GX Developer.
- Only one file can be stored in the NC.

(b) PLC program comment

- Program comment for GX Developer display
- Only one file can be stored in the NC with the same file name as the ladder program.
- A device comment (32 characters) and a device name (10 characters) can be defined for each device.
- Stored mainly when it is read to GX Developer and used as a comment.

(c) Message 1st language, (d) Message 2nd language

- Alarm message/operator message/PLC switch/comment message data.
- One 1st language file and one 2nd language file can be stored in the NC.
- The messages can be handled and edited as "integrated type interlinear statements" by GX Developer.
- The maximum message length and the number of messages can be specified for each message type.

3.3 Newly Creating a PLC Program

The ladder program is newly created with GX Developer.
Refer to the Operating Manual for details on newly creating a program.

3. PLC PROGRAM DEVELOPMENT

3.4 Specifying the Connection Target

3.4 Specifying the Connection Target

You must specify the connection target before performing online operations from GX Developer to the CNC controller.

3.4.1 Operation procedure

Perform the following operation from GX Developer to start the setting screen.

[Online] → [Transfer setup]

Set only the following items. Leave the other items unchanged from the initial values.

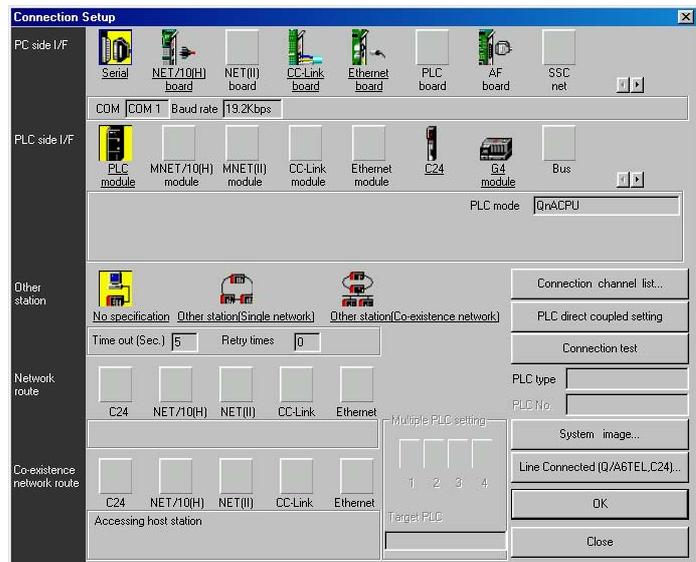
(1) Personal computer side

Interface : [Serial interface]
Serial port name : [COM1] or [COM2]
Baudrate : [19.2Kbps]

(2) PLC side

Interface : [CPU unit]

■ Setting screen of SW4D5-GPPW



3. PLC PROGRAM DEVELOPMENT

3.5 Starting/Stopping the PLC of the CNC Controller

3.5 Starting/Stopping the PLC of the CNC Controller

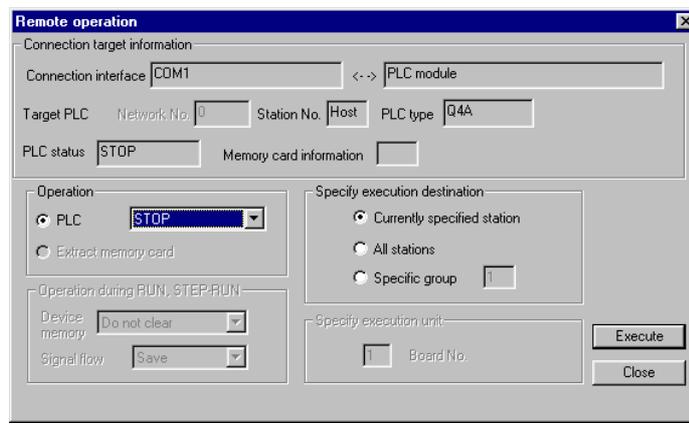
Before writing a PLC program, you must stop the PLC of the CNC controller.

3.5.1 Operation procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Remote operation] or **[Alt] + [6]**

On the following screen, set "STOP" or "RUN" in the [PLC] part under [Operation] and click [Execute]. The current status is displayed in [PLC status] under [Connection target information].



[Note] Operations other than RUN and STOP are not possible.

The operation is completed when the following dialog appears. Click [OK]. The status after completion appears in [PLC status] on the remote operation screen displayed behind. If the status does not change, check whether an alarm is displayed or not on the CNC controller side.



3. PLC PROGRAM DEVELOPMENT

3.6 Writing the PLC Program to the CNC Controller

3.6 Writing the PLC Program to the CNC Controller

The following indicates how to write ladders from GX Developer to the CNC controller (especially the restrictions and M600, M60/60S series-specific operations).

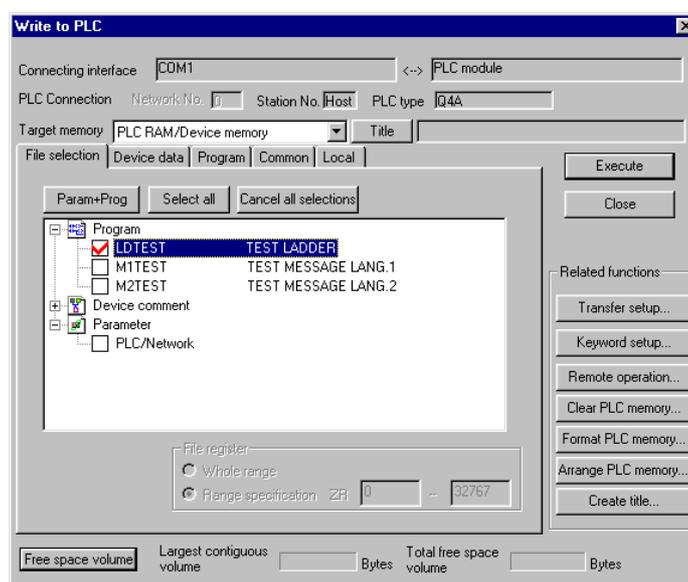
3.6.1 Operation procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Write to PLC]

On the following screen, choose the ladder file to be written from the [File selection] tab and click [Execute].

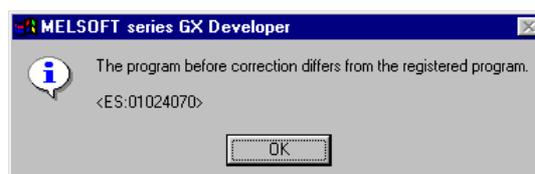
You can command RUN/STOP of the PLC using [Remote operation] under [Related functions].



[Note] As [Target memory], only [PLC RAM/Device memory] is valid. Do not set the other tabs ([Device data], [Program], [Common], [Local]) than [File selection].

3.6.2 Operation to be performed at write error

As soon as a ladder is written from GX Developer to the CNC controller, the CNC controller converts it into the CNC-specific ladder machine code. A conversion error occurs if any of the devices and instruction formats not supported by the M600, M60/60S series is used. At a conversion error, the CNC side ladder machine code is converted into the "NOP code", the ladder up to the last step is transferred, and the following dialog is then displayed on the GX Developer screen.



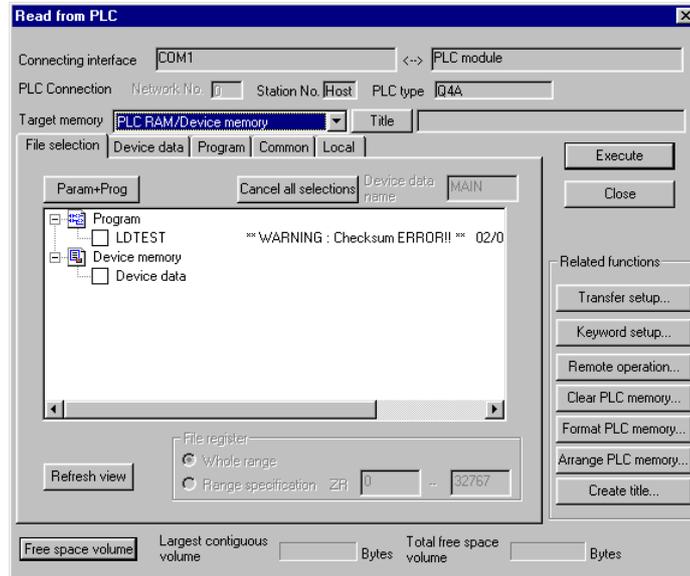
3. PLC PROGRAM DEVELOPMENT

3.6 Writing the PLC Program to the CNC Controller

When the file that resulted in a conversion error is displayed with the [File selection] tab of the [Read from PLC] screen, the following warning appears in the title field.

**** WARNING Checksum ERROR!! ****

If you execute RUN the PLC as-is, an alarm occurs on the CNC side and the PLC does not RUN.



⚠ CAUTION

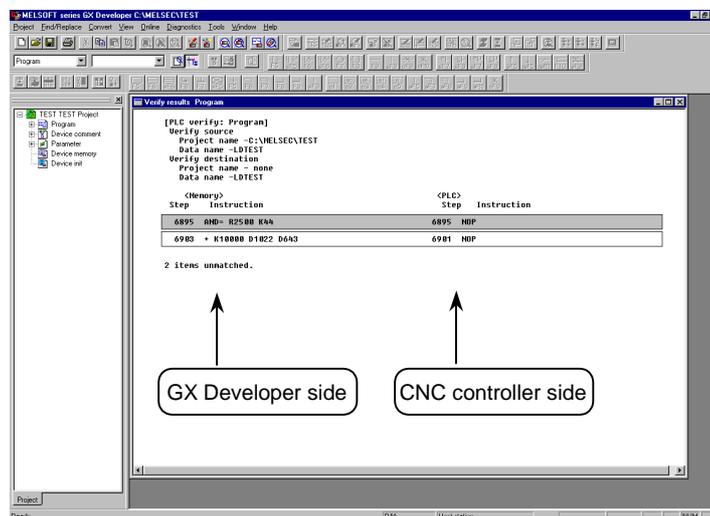
⚠ Do not read a ladder file on which a conversion error occurred into the GX Developer side. The file may include unexpected contents to result an illegal operation.

3.6.3 How to confirm the error step number

The PLC verification function can be used to confirm the error step. Executing verification with PLC displays mismatches as in the following example. For details of the PLC verification function, refer to "3.8 Verifying the PLC Programs".

Following example shows the step where CNC controller outputs a conversion error because NOP instructions mismatched between CNC and GX Developer.

Double-click the mismatch to display and to edit the corresponding part of the GX Developer side. <Memory> indicates the GX Developer side, and <PLC> the CNC controller side.



3. PLC PROGRAM DEVELOPMENT

3.7 Reading the PLC Program from the CNC Controller

3.7 Reading the PLC Program from the CNC Controller

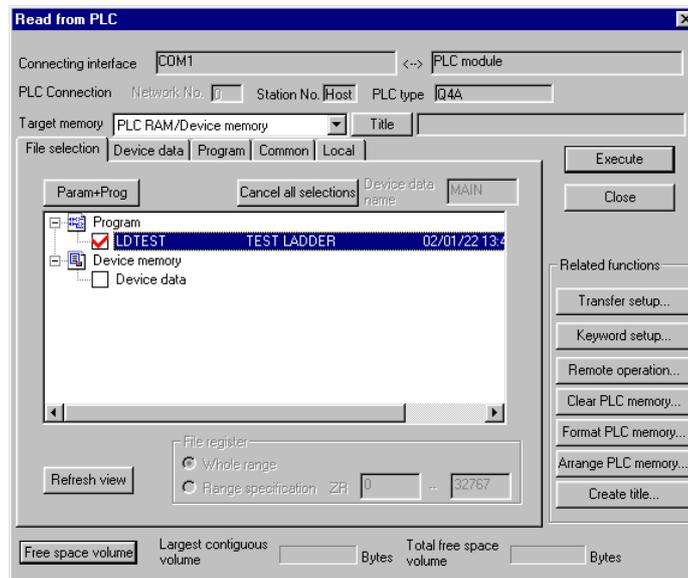
The following indicates how to read a ladder from the CNC controller to GX Developer.

3.7.1 Operation procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Read from PLC]

On the following screen, choose the ladder file to be read from the [File selection] tab, and click [Execute].



[Note] As [Target memory], the fitted memory is valid.
Do not set the other tabs ([Device data], [Program], [Common], [Local]) than [File selection].

If a ladder file with the same name already exists in the GX Developer side, following dialogue will appear.

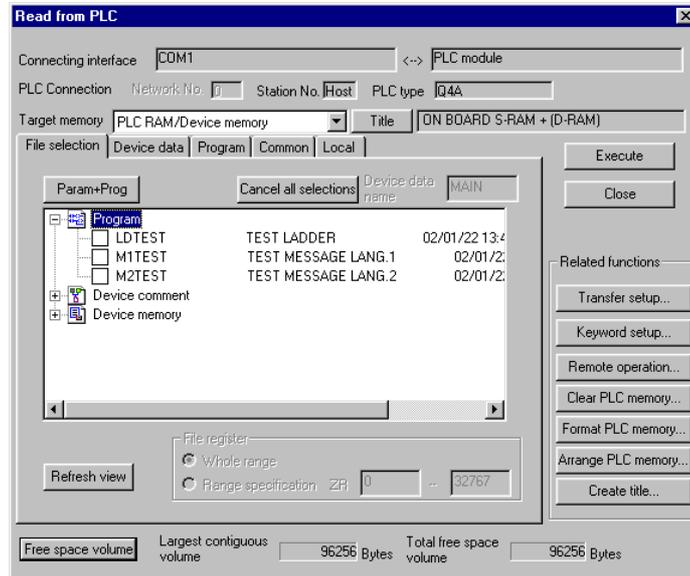


[Note] Choosing [Yes (Y)] in the dialogue will overwrite the GX Developer side ladder file. The file before overwriting will be erased.
Confirm the file enough before choosing [Yes (Y)].

3. PLC PROGRAM DEVELOPMENT

3.7 Reading the PLC Program from the CNC Controller

The [Read from PLC] screen can also be used as a CNC controller side file listing function. Move the scroll bar of the [File selection] tab to the right to display the write date and size of each file. Click [Free space volume] to display the free area of the target memory.



3. PLC PROGRAM DEVELOPMENT

3.8 Verifying the PLC Programs

3.8 Verifying the PLC Programs

The following indicates how to verify ladders between the CNC controller and GX Developer.

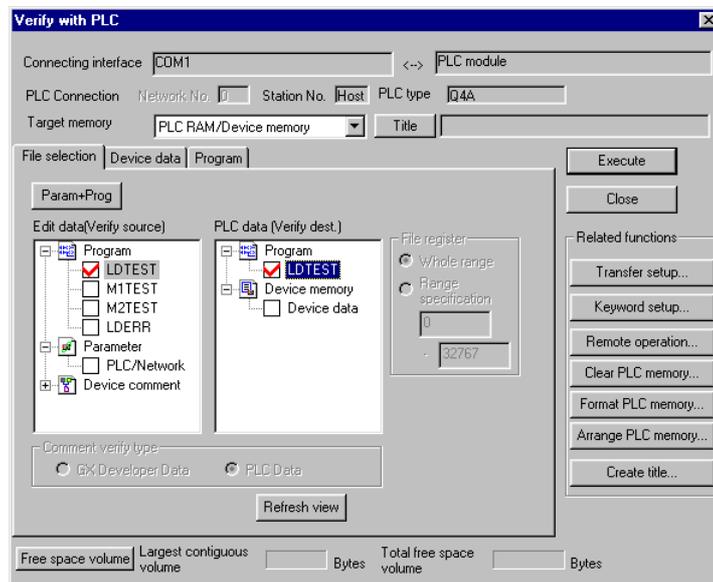
3.8.1 Operation procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Verify with PLC]

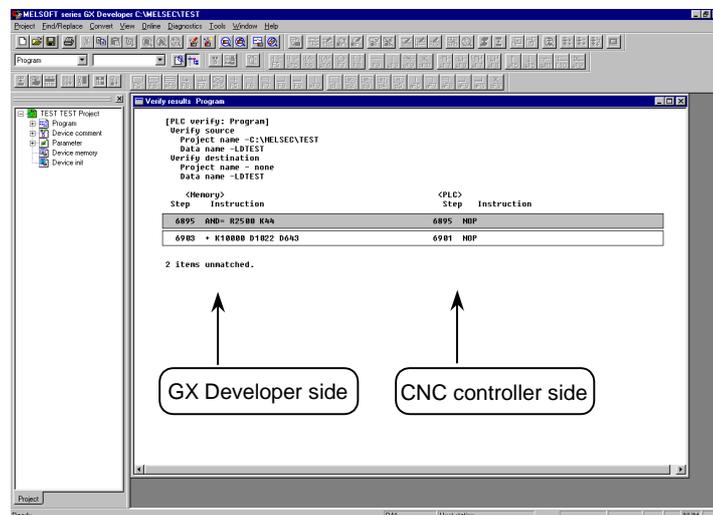
On the following screen, choose the ladder files to be verified from the [File selection] tab, and click [Execute].

[Verify source] : GX Developer side [Verify dest] : CNC side



[Note] As [Target memory], the fitted memory is valid.
Do not set tabs ([Program]) other than [File Selection].

If verification mismatches occur, the following mismatch screen appears. Double-click the mismatch to display the corresponding part of the GX Developer side file.



3. PLC PROGRAM DEVELOPMENT

3.9 Monitoring the PLC Program

3.9 Monitoring the PLC Program

There are no MELDAS-specific operations to monitor a PLC program. Refer to the Operating Manual for the operation methods. For usable functions, refer to "1.3.2 Function support conditions (on-line section)". This section explains the operation procedure outline and precautions.

3.9.1 Operation procedure

Perform the following operation from GX Developer to start monitoring.

- (1) Display the ladder program to be monitored and move to the circuit part to be monitored.
- (2) Perform the following operation to start monitoring.

[Online] → [Monitor] → [Monitor mode] or F3

- (3) Perform the following operation to stop monitoring.

[Online] → [Monitor] → [Stop monitor] or Alt + F3

[Note] If the ladder program being run by the CNC controller differs from the one being displayed on GX Developer, monitoring will not result in an error but will continue.

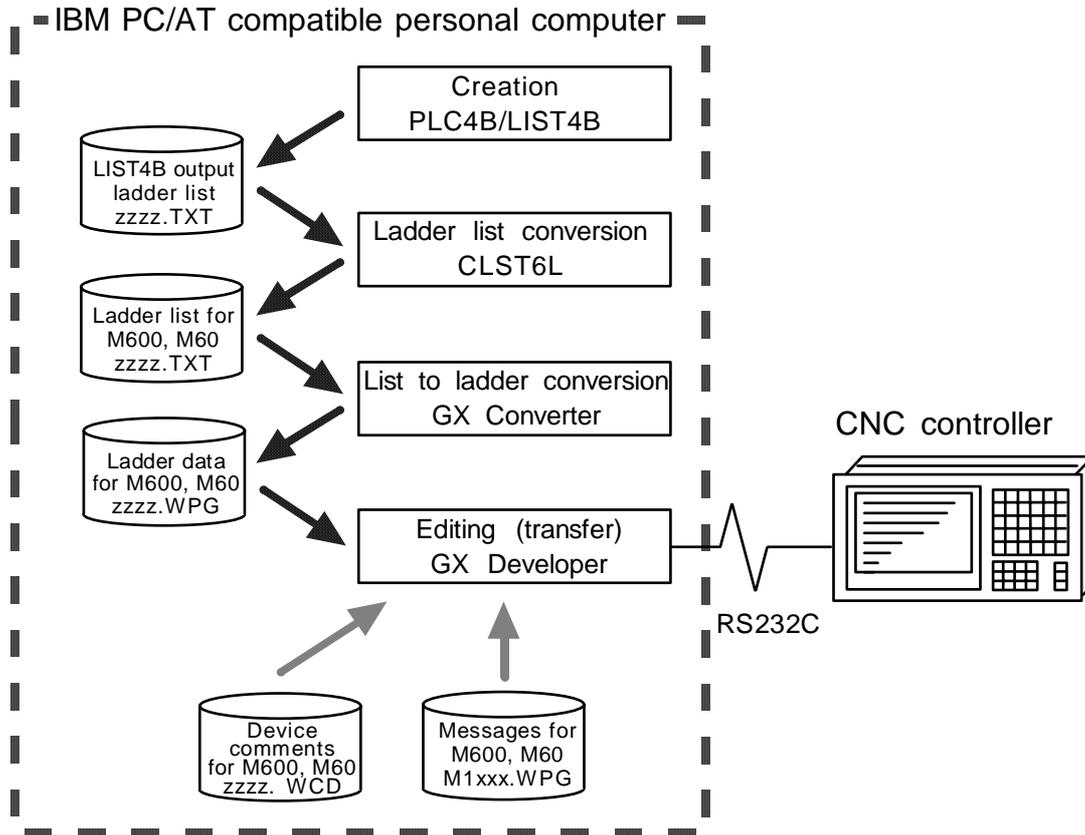
3. PLC PROGRAM DEVELOPMENT

3.10 Using PLC46 Type PLC Programs

3.10 Using PLC4B Type PLC Programs

The PLC program developed with the conventional MELDAS PLC programming tool (PLC4B) can be used.

3.10.1 Development procedure



(1) Creation

The PLC program created for the old model is output in a list format.

(2) Conversion

Using CLST6L (ladder list converter), the output program is converted into a PLC program for M600, M60/60S series (list format). Using GX Converter (data conversion software package), the list format program is converted into the GX Developer data.

(3) Editing/transfer

The resultant program can be handled like a newly created PLC program.

3. PLC PROGRAM DEVELOPMENT

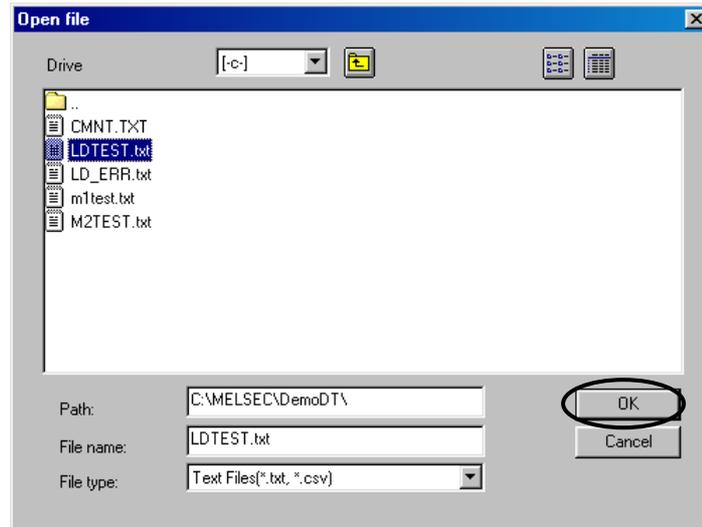
3.10 Using PLC46 Type PLC Programs

3.10.2 Starting GX Converter and specifying the file to be converted

Perform the following operation from GX Developer to start GX Converter (read).

[Project] → [Import file] → [Import to TEXT ,CSV format file]

On the following screen, choose the file to be converted (LDTEST.TXT) and click [OK].

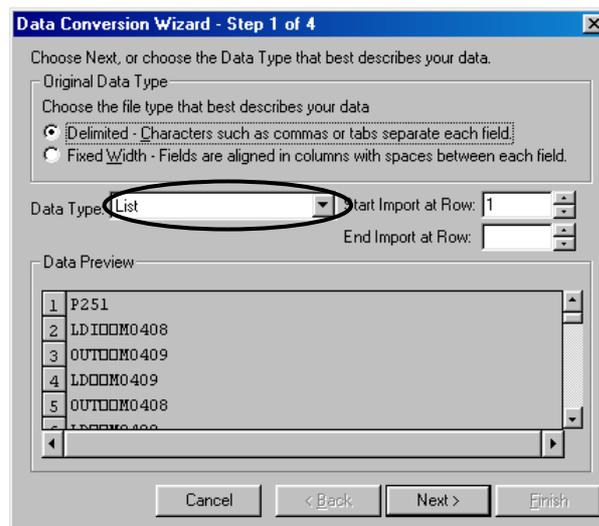


3.10.3 Conversion format setting

Set the conversion format on the following data conversion wizard screen.

(1) Data conversion wizard 1/4

Choose [Original Data Type]-[Delimited] and [Data Type]-[List], and click [Next>].

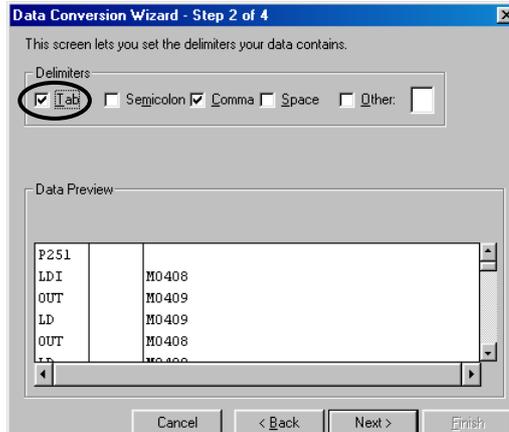


3. PLC PROGRAM DEVELOPMENT

3.10 Using PLC46 Type PLC Programs

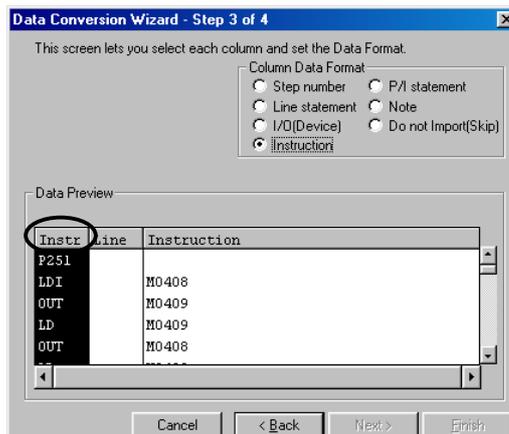
(2) Data conversion wizard 2/4

Choose [Delimiters]-[Tab] and click [Next>].



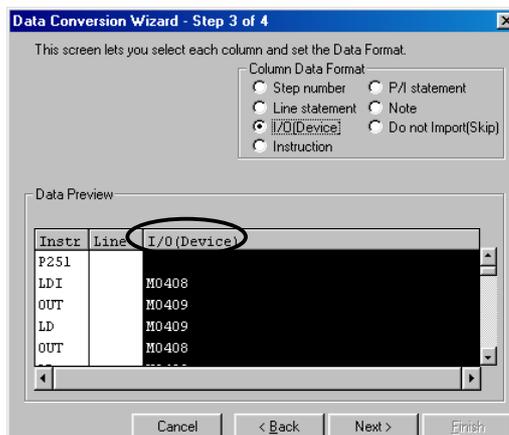
(3) Data conversion wizard 3/4

Choose to highlight the Instruction column part in the [Data Preview] list and choose [Column Data Format]-[Instr].



(4) Data conversion wizard 3/4

Further, choose to highlight the Argument column part in the [Data Preview] list and choose [Column Data Format]-[I/O(Device)]. Click [Next>].

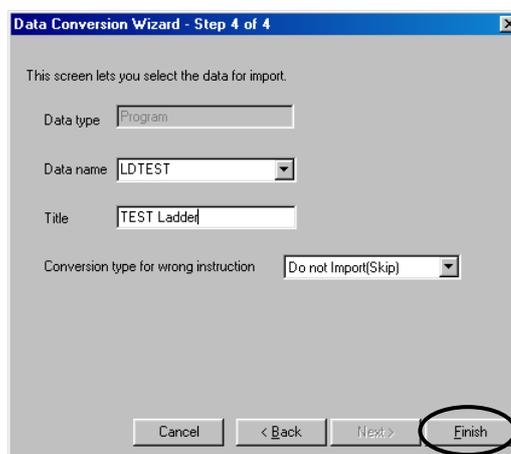


3. PLC PROGRAM DEVELOPMENT

3.10 Using PLC46 Type PLC Programs

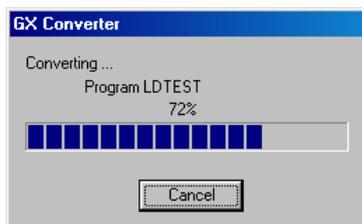
(5) Data conversion wizard 4/4

Set the program name used on GX Developer at [Data name] column and a ladder annotation at [Title] column, and click [Finish].



(6) Completion

The setting is complete when the following completed dialog appears after the converting dialog. Click [OK].



4. PLC MESSAGE DEVELOPMENT

4.1 Development Procedure

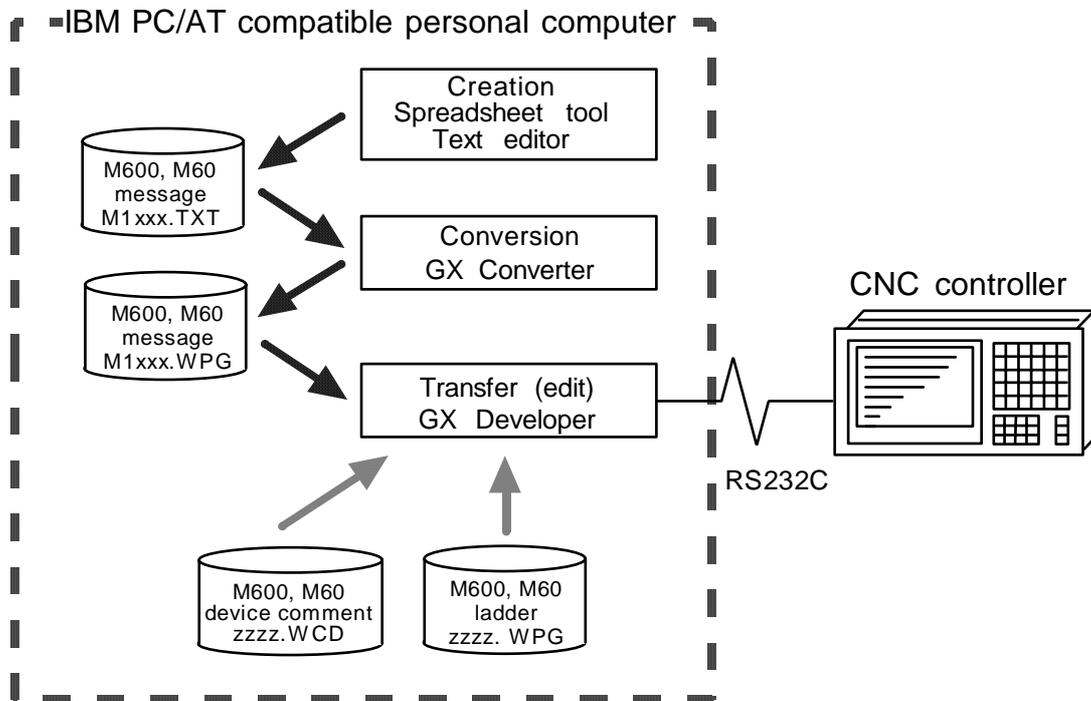
4. PLC MESSAGE DEVELOPMENT

This chapter describes a procedure for developing PLC-related data such as alarm messages, operator messages, and PLC switches.

4.1 Development Procedure

There are the following two methods as a general development procedure of message data.

- (1) Making conversion into GX Developer data using a general text editor or spreadsheet tool and data conversion package.
(When there is a large volume of message data and you want to control them with a commercially available tool, for example)
- (2) Entering messages directly from GX Developer
(When there is a small volume of message data or when addition or correction is to be made, for example)



4. PLC MESSAGE DEVELOPMENT

4.1 Development Procedure

4.1.1 Using a general text editor

(1) Creation

The message data is described using a general text editor. The description method and format will be described later.

(2) Conversion

The conversion from text data to GX Developer data is carried out using the "GX Converter (data conversion software package)".

(3) Transfer

With the GX Developer, the message data is handled as a PLC program interlinear comment, and can also be edited.

The message data is transferred to the CNC controller using the GX Developer, in the same manner as the ladder program.

4.1.2 Entering messages directly from GX Developer

(1) Creation

The message data is described directly from GX Developer. The message data is handled as a PLC program interlinear comment by GX Developer. The description method and format will be described later.

(2) Transfer

The message data is transferred from GX Developer to the CNC controller in the same manner as the ladder program.

4. PLC MESSAGE DEVELOPMENT
4.2 Message Data Description Method

4.2 Message Data Description Method

The message data can be described as text data by a general text editor and also by commercially available spreadsheet software in addition to the direct input with GX Developer.

4.2.1 Description format

Message data is classified into setting areas to store the setting for each message and message areas to store message data. It is described in the following respective description format.

(1) Setting area

The message length and No. of messages are set for each message in the setting area. The message data region secured by the CNC controller can be adjusted to the most efficient status using these settings. The respective maximum values are set if nothing is set. (Refer to "4.2.3 Precautions" for the maximum values.)

;\$, message classification code, maximum message length, No. of messages [CR]

Alarm message : A	Operator message : O
PLC switch : P	Comment message : M

(2) Message area

The message area is described using the following description format.

The description format cannot be abbreviated. Comma(,) and [CR] must be described, even the message character string is blank.

Message classification	Description format
Alarm message	;A, index No., data register No., message character string [CR]
Operator message	;O, index No., data register No., message character string [CR]
PLC switch	;P, switch No., message character string [CR]
Comment message	;M, device, device No., message character string [CR]

Message classification code	: A one-byte alphabetic character expressing each message classification
Index No.	: One-byte number (0 to No. of messages in the setting area - 1)
Switch No.	: One-byte number (0 to No. of messages in the setting area - 1)
Data register No.	: One-byte number
Device	: One-byte number (1 or 2)
Device No.	: One-byte number (0 to 10)
Message character string	: One-byte alphanumeric character, No. of characters in the setting area message length. Semicolons, commas, spaces and tabs can also be used. Note that the tab at the head of the message character string is ignored.
Semicolon(;)	: Message data identification code
Comma(,)	: Separator between each description (a comma only is used to leave a message character string blank)
[CR]	: Line feed code, (CR/LF) or (LF).

4. PLC MESSAGE DEVELOPMENT

4.2 Message Data Description Method

4.2.2 Description method

The message data is described as text data by the following description format.

<pre>#M635 ladder ver1. '97.08.01</pre>	... Comment
<pre>;\$, A,32, 200</pre>	... Setting area
<pre>;\$, 0, 40, 200</pre>	
<pre>;\$, P, 14, 32</pre>	
<pre>;\$, M, 60, 20</pre>	
<pre>NOPLF</pre>	
<pre>;A, 0, 0, Emergency stop</pre>	
<pre>;A, 11, 1, Spindle alarm</pre>	
<pre>...</pre>	
<pre>NOPLF</pre>	... Message area (alarm messages)
<pre>...</pre>	
<pre>NOPLF</pre>	
<pre>...</pre>	
<pre>NOPLF</pre>	... Page break code
<pre>;0, 1, 9000, MELDAS 600LADDER Ver1.0</pre>	... Message area (operator messages)
<pre>;0, 20, 9000, BND-400W000-A0</pre>	
<pre>...</pre>	
<pre>NOPLF</pre>	
<pre>;P, 1, Program restart</pre>	
<pre>;P, 2, Automatic power OFF</pre>	
<pre>...</pre>	
<pre>NOPLF</pre>	... Message area (PLC switches)
<pre>...</pre>	
<pre>NOPLF</pre>	
<pre>;M, 1, 0, [Spindle]</pre>	... Message area (comments)
<pre>;M, 1, 0, [Standby 1]</pre>	
<pre>...</pre>	
<pre>END</pre>	... End code

(1) Comment

Statements having a semicolon (;) at the head of the line, in a different format than described in "4.2.1 Description format", are regarded as comments. These comments are handled as comment data in the GX Developer also, but are erased during the transfer to the CNC controller. An error will occur if there is no semicolon at the head of the line.

(2) Setting area

Each message is set here. This area must be described before the message area of the relevant message. That setting will be ignored if it is described in the middle of or after the relevant message description.

(3) Message area

Collect similar messages in a group and describe them. There is no description order in the respective messages, but the latter description is validated if there are descriptions with the same factors (index No., etc.).

(4) Page break code

A page break code is described at one or more places approx. every 15 lines in the setting area and message area. The message data may skip if there is no page break code.

4. PLC MESSAGE DEVELOPMENT

4.2 Message Data Description Method

(5) End code

An end code is described at the end of the description. Description after the end code are ignored.

An error will occur if there is no end code.

4.2.3 Precautions

No. of characters, quantity limitations, handling of information other than settings, handling of information other than format are described below.

(1) Message data maximum value

Processing will be carried out with the following values considered as the maximum values if the setting is not carried out in the setting area, or if the description position in the setting area is illegal.

Message classification	Max. message length	Max. No. of messages	Data size by multiplying max. message length by max. No. of messages
Alarm messages	32 byte	512	16 Kbyte
Operator messages	60 byte	512	30 Kbyte
PLC switches	14 byte	32	0.5 Kbyte
Comments	60 byte	100	6 Kbyte

[Note] Two-byte data in the message character string is handled as two characters. GX Developer accepts 64 characters as an interlinear comment. However, since that includes information other than a message character string (e.g. message classification code, index No. and data register No.), the message character string is actually up to 58 characters long.

(2) When the setting value and message data do not match

When the message data contents (such as index No., switch No. and message character string) overflows from the settings in the setting area, the data that overflowed is ignored.

4. PLC MESSAGE DEVELOPMENT

4.3 Converting Data into GX Developer Format

4.3 Converting Data into GX Developer Format

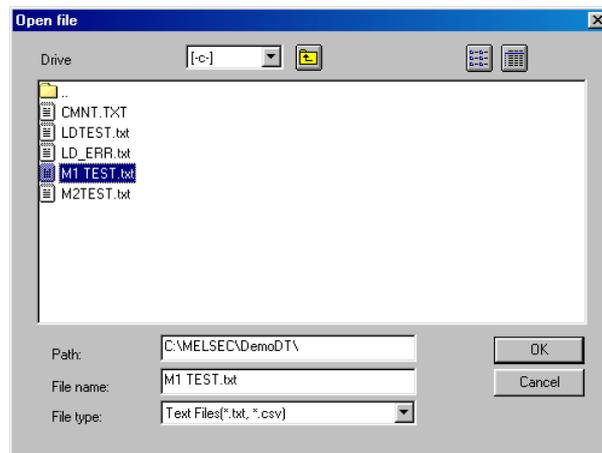
Convert the message data, which was described using a text editor or like, into GX Developer data in the following method. Use "GX Converter (data conversion software package)" for conversion. GX Converter can be started from the GX Developer menu.

4.3.1 Starting GX Converter and specifying the file to be converted

Perform the following operation from GX Developer to start GX Converter (read).

[Project] → [Import file] → [Import from TEXT ,CSV format file]

On the following screen, specify the file to be converted (M1TEST.TXT) and click [OK].

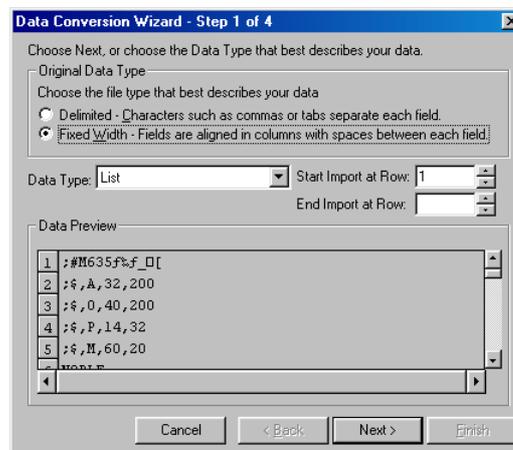


4.3.2 Conversion format setting

Set the conversion format on the following data conversion wizard screen.

(1) Data conversion wizard 1/4

Choose [Original Data Type]-[Fixed Width] and [Data Type]-[List], and click [Next>].

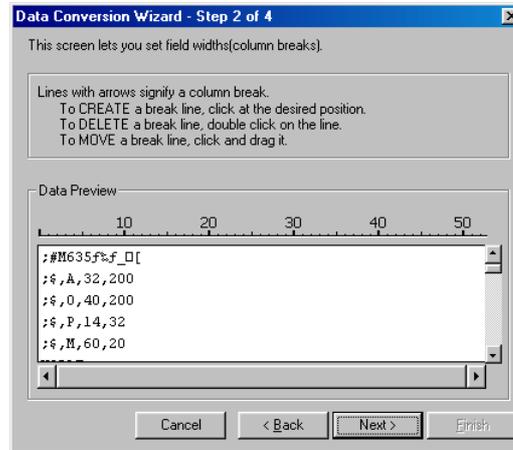


4. PLC MESSAGE DEVELOPMENT

4.3 Converting Data into GX Developer Format

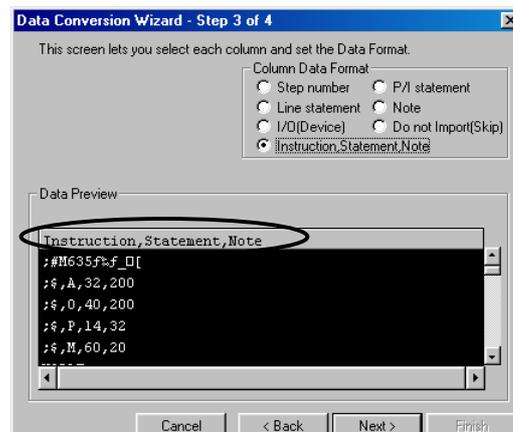
(2) Data conversion wizard 2/4

Just click [Next>].



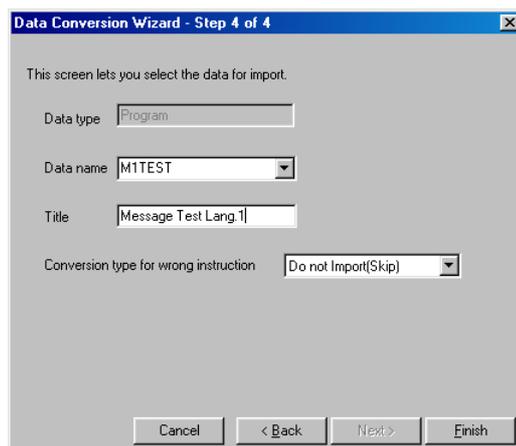
(3) Data conversion wizard 3/4

Choose to highlight the Instruction column part in the [Data Preview] list and choose [Column Data Format]-[Instruction ,Statement ,Note]. Click [Next>].



(4) Data conversion wizard 4/4

Set the program name used on GX Developer in [Data name] and a data annotation in [Title], and click [Finish]. Data conversion is completed when the Completed dialog appears. Click [OK].



4. PLC MESSAGE DEVELOPMENT

4.4 Entering/Editing Data Using GX Developer

4.4 Entering/Editing Data Using GX Developer

The message data in GX Developer are handled as the "integrated type interlinear statements" of a PLC program. "Integrated type interlinear statements" are interlinear comments provided to assist the understanding of the PLC program, and those transferred to the controller together with the PLC program are called the "integrated type".

"Interlinear statements" can be displayed and edited using [Ladder] or [Instruction list].

4.4.1 Interlinear statement display using circuit display

(1) Display of project data list

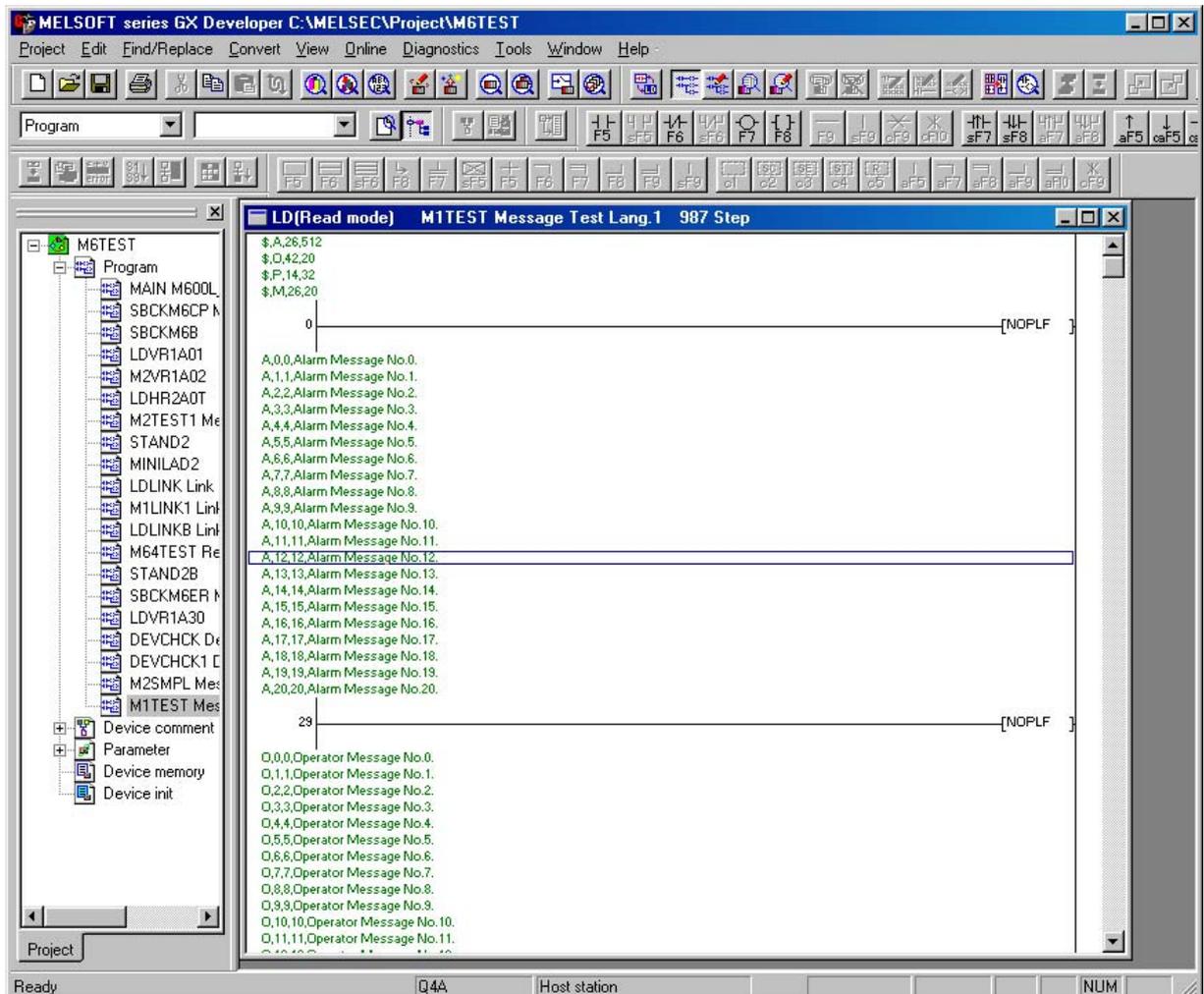
Perform the following operation to display the "Project data list" window and double-click the file name to display the edit screen. First, the normal ladder screen appears.

[View] → [Project data list], then double-click [File name you want to display].

(2) Display of message data

Perform the following operation to display the message data that are integrated type interlinear statements.

[View] → [Statement]



4. PLC MESSAGE DEVELOPMENT

4.4 Entering/Editing Data Using GX Developer

4.4.2 Interlinear statement display using list display

(1) Display of project data list

Perform the following operation to display the "Project data list" window and double-click the file name to display the edit screen. First, the normal ladder screen appears.

[View] → [Project data list], then double-click [File name you want to display].

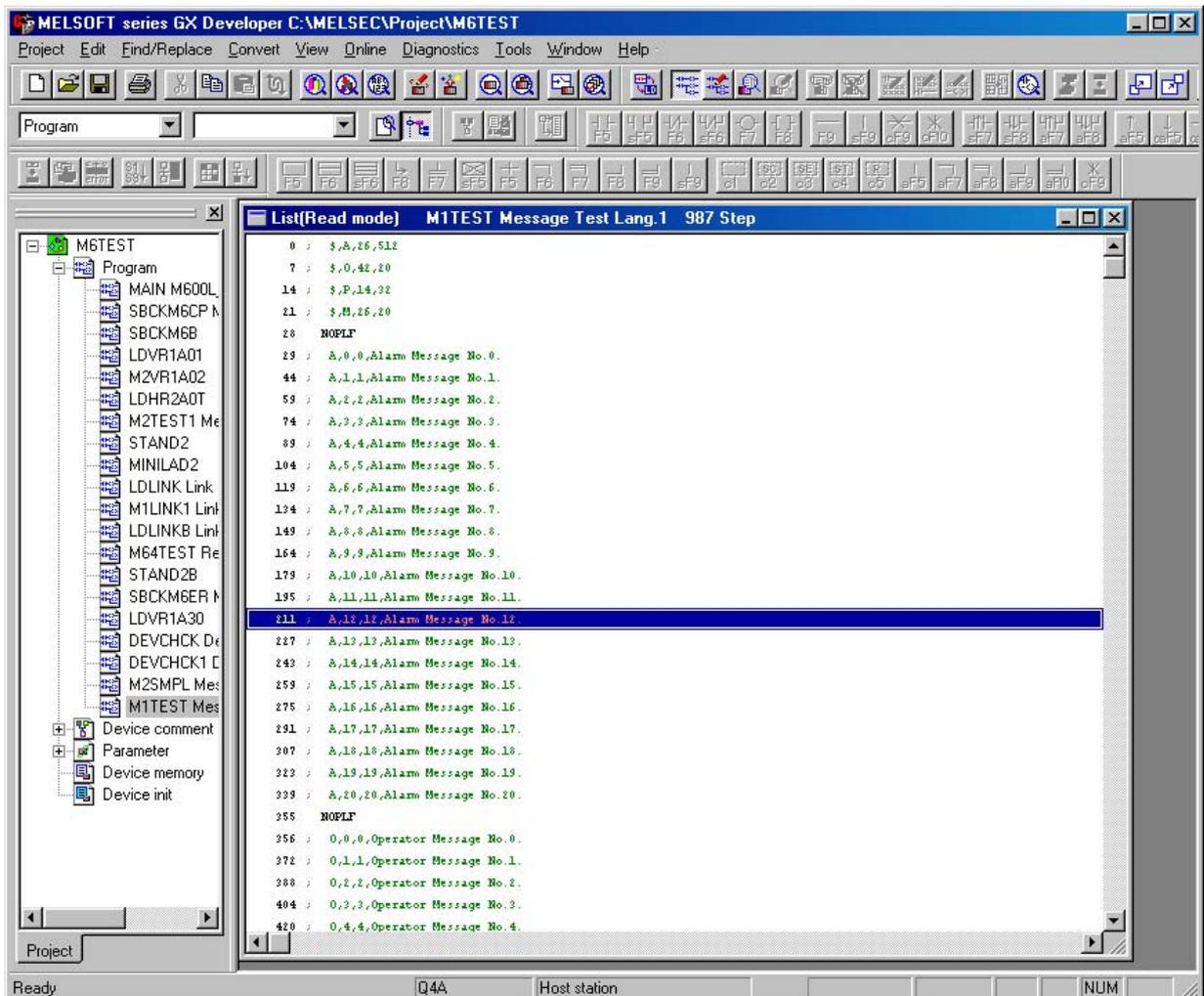
(2) Display of list data

Perform the following operation to display the list data. The list display also shows the message data that are integrated type interlinear statements.

[View] → [Instruction list]

Perform the following operation to return to the circuit display.

[View] → [Ladder]



4. PLC MESSAGE DEVELOPMENT

4.4 Entering/Editing Data Using GX Developer

4.4.3 Editing of integrated type interlinear statements

(1) Circuit display

On the circuit display screen that shows the integrated type interlinear statements, double-clicking the interlinear statement you want to edit displays the following dialog. Perform editing operation on the dialog and click [OK] or press [Enter].



(2) List display

On the list display screen, double-clicking the interlinear statement you want to edit displays the following dialog. Perform editing operation on the dialog and click [OK] or press [Enter].

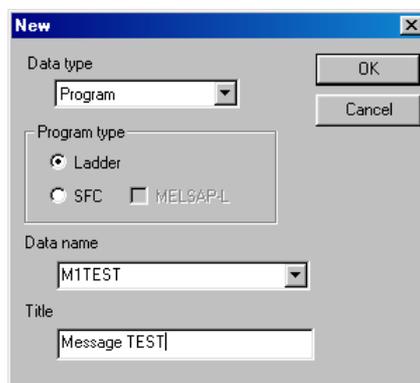


(3) Entering new message data

- Displaying new edit screen

Perform the following operation to display the [New] dialog, and set the [Data name] and [Title]. After setting, click [OK].

[Project] → [Edit Data] → [New]



- Changing to list display mode

Perform the following operation to display the list data.

[View] → [Instruction list]

- Entering message data

Press "Enter" at the "END" line, and input the data in the same manner as 4.4.3 (2) above. Then, press "Enter" on the next line, and input the message data.



4. PLC MESSAGE DEVELOPMENT

4.5 Writing to the CNC Controller

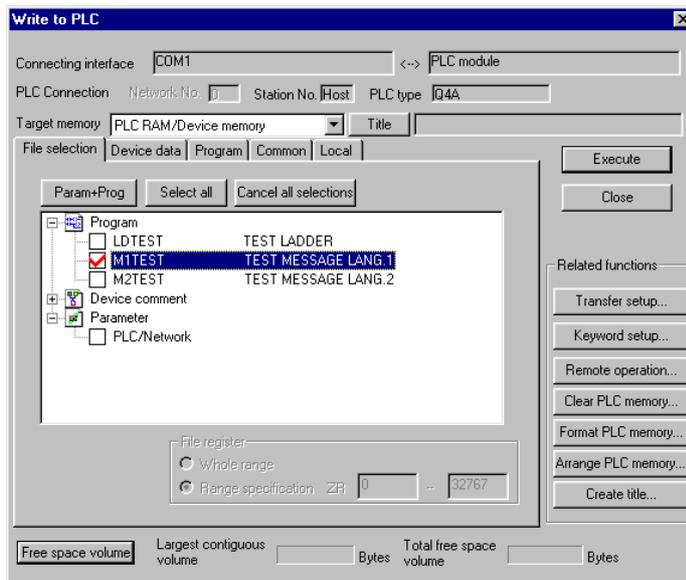
4.5 Writing to the CNC Controller

The following shows the method of transferring a message from the GX Developer to the CNC controller. The transfer method is the same as the ladder code transfer method. Ladder codes and message data are distinguished by their file names only.

Perform the following operation to display the [Write to PLC] screen, and choose the file to be written.

[Online] → [Write to PLC]

The following example transfers a message first language file "M1TEST.WPG".



4.6 Reading and Verifying from the CNC Controller

The following shows the method of reading and verifying a message from the CNC controller to the GX Developer. The method of reading and verifying is the same as that of ladder codes. Ladder codes and message data are distinguished by their file names only.

4.6.1 Menu selection/screen operation

Refer to the following sections for operation methods.

- For read : "3.7 Reading the PLC Program from the CNC Controller"
- For verification : "3.8 Verifying the PLC Programs"

4. PLC MESSAGE DEVELOPMENT

4.6 Reading and Verifying from the CNC Controller

4.6.2 Message read format

The message description format was shown in "4.2.1 Description format", but there are no special rules concerning provision of descriptions in the setting area or the order of message description in the message area. For that reason, the description format may differ between transfer and reading of the message data.

The following shows the format during reading as the "Standard description format".

Standard description format of message data

Alarm message setting	...(1)
Operator message setting	
PLC switch setting	
Comment message setting	
Alarm messages	...(2)
Operator messages	...(3)
PLC switches	...(4)
Comment messages	...(5)
NOPLF	
END	

(1) Setting area

The settings are described in order of alarm, operator, PLC switch and comment.
The maximum value is described if the setting is abbreviated.

(2) Alarm messages

Each message data is described in order of the index Nos.

(3) Operator messages

The same as the alarm messages.

(4) PLC switches

Each message data is described in order of the switch Nos.

(5) Comment messages

These messages are described in the same order as described before transfer.

(6) Others

- Spaces and tabs are not included before and after the comma(,) separating the message data factors.
- The message character string is handled the same as normal data even when blank.
- The NOPLF code between messages is described to the position to which the message data following the NOPLF code during transfer moved.

5. DEVICE COMMENT CREATION

5.1 Development Procedure

5. DEVICE COMMENT CREATION

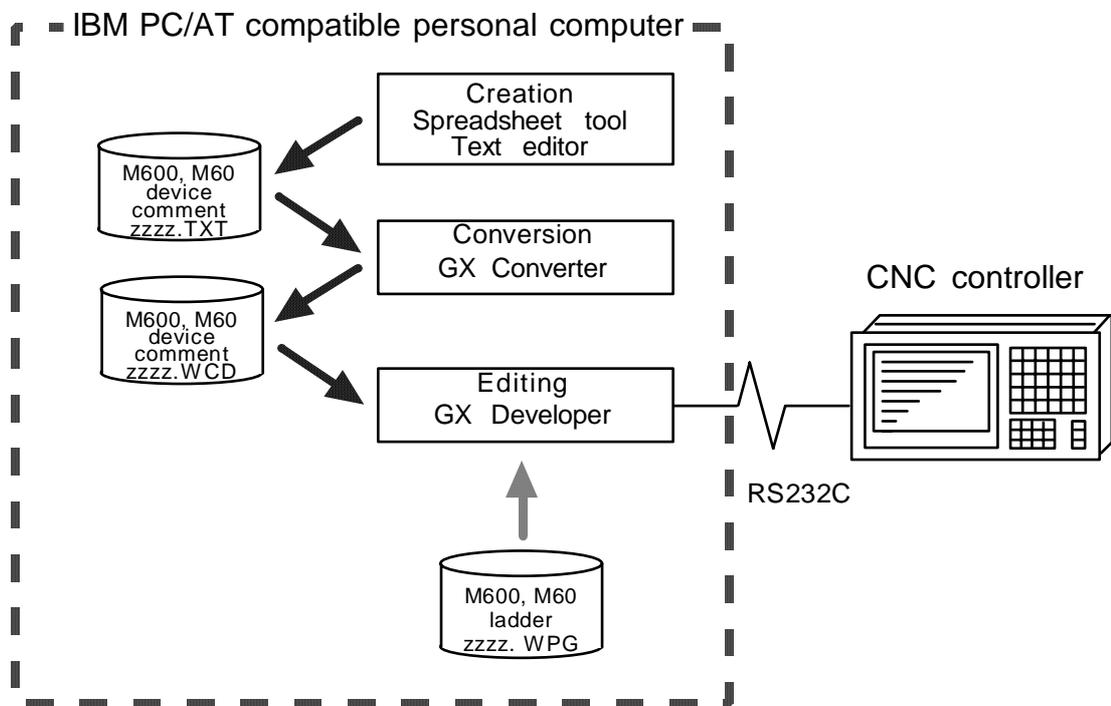
There are no MELDAS-specific operations for device comments. Therefore, refer to the Operating Manual for the development method. This section describes the device comment development procedure outline and the development method using a general-purpose tool.

5.1 Development Procedure

The following two methods can be used as general development procedures of message data.

(1) Indirect entry

In this method, device comments are converted into GX Developer data using a general text editor or spreadsheet tool and data conversion package. Use this method when you want to divert the device comments of the old model or when a device comment volume is large and you want to control them with a commercially available tool, for example.



(2) Direct entry

In this method, device comments are entered directly from GX Developer. Use this method when a device comment volume is small or when addition or correction is to be made, for example.

There are the following three methods for direct entry from GX Developer. Refer to the Operating Manual for details.

- Creating comments on the device comment edit screen
- Creating device comments after circuit creation during ladder circuit creation
- Making addition/correction to device comments in the created ladder circuit

5. DEVICE COMMENT CREATION
5.2 Description Method for Indirect Entry

5.2 Description Method for Indirect Entry

The following explains the description method for creating device comments using a spreadsheet tool or like. The following example describes device comments using a spreadsheet tool.

	A	B	C	
1	X0	SAFETY	Safety unit run	Describe device, equipment name and comment on the same line.
2	X1	COVER	Safety cover close	
3	X2	READY	Operation ready complete	
4	X3	OIL-M	Oil pressure motor	
5	X4	PARTS-A	Parts A ready complete	

↑ Device-only column	↑ Equipment name-only column	↑ Comment-only column
-------------------------	---------------------------------	--------------------------

Column data format	Explanation
Device	(1) Describe a device. <ul style="list-style-type: none"> Conversion cannot be made if a device has not been described. Always describe a device. (2) A device is a required item. Describe it in one-byte code.
Equipment name	(1) Describe an equipment name. <ul style="list-style-type: none"> It is not registered if the device part on the same row is blank or the device is illegal. (2) You can describe an equipment name of up to 8 characters.
Comment	(1) Describe a comment. <ul style="list-style-type: none"> It is not registered if the device part on the same row is blank or the device is illegal. (2) You can describe a comment of up to 32 characters.

[Note] Describe data in any of the following combinations.
 (1) Device, equipment name, comment
 (2) Device, comment
 (3) Device, equipment name

Save the above data in the CSV format. The following example shows the above data saved in the CSV format.

X0, SAFETY, Safety unit run
 X1, COVER, Safety cover close
 X2, READY, Operation ready complete
 X3, OIL-M, Oil pressure motor
 X4, PARTS-A, Parts A ready complete

5. DEVICE COMMENT CREATION

5.3 Converting Device Comments

5.3 Converting Device Comments

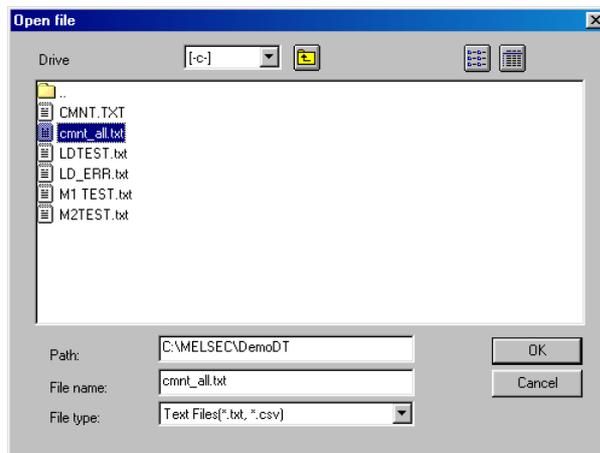
Convert the comment data (CSV format), which was created using a spreadsheet tool or like, into GX Developer data in the following method. Use "GX Converter (data conversion software package)" for conversion. GX Converter can be started from the GX Developer menu.

5.3.1 Starting GX Converter and specifying the file to be converted

Perform the following operation from GX Developer to start GX Converter (read).

[Project] → [Import file] → [Import from TEXT ,CSV format file]

On the following screen, specify the file to be converted (cmnt_all.txt) and click [OK].

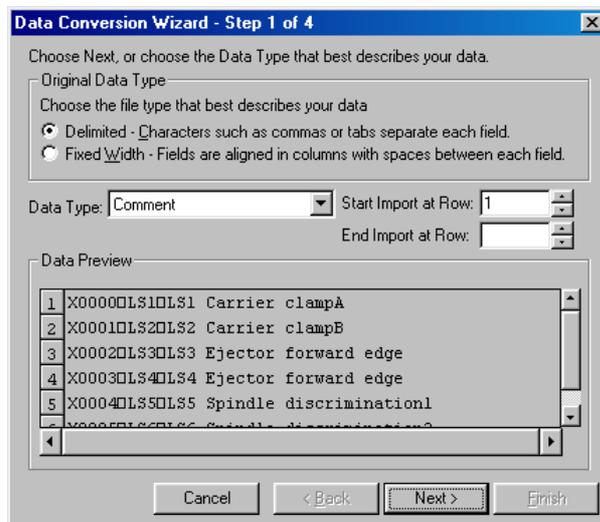


5.3.2 Conversion format setting

Set the conversion format on the following data conversion wizard screen.

(1) Data conversion wizard 1/4

Choose [Original Data Type]-[Delimited] and [Data Type]-[Comment], and click [Next>].

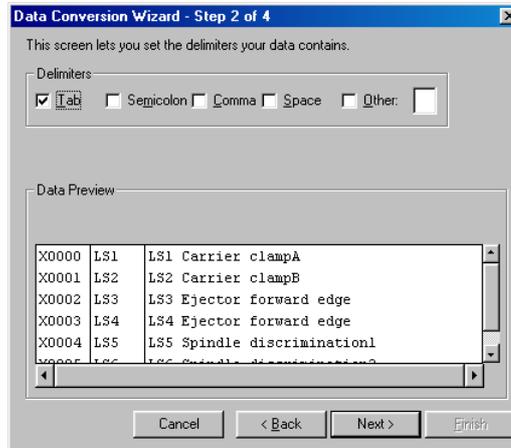


5. DEVICE COMMENT CREATION

5.3 Converting Device Comments

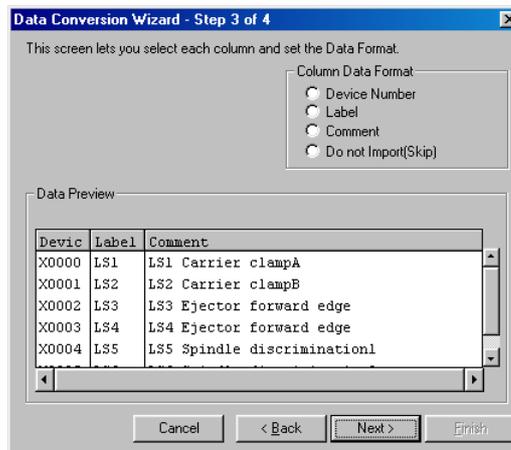
(2) Data conversion wizard 2/4

Choose [Delimiters]-[Tab] and click [Next>].



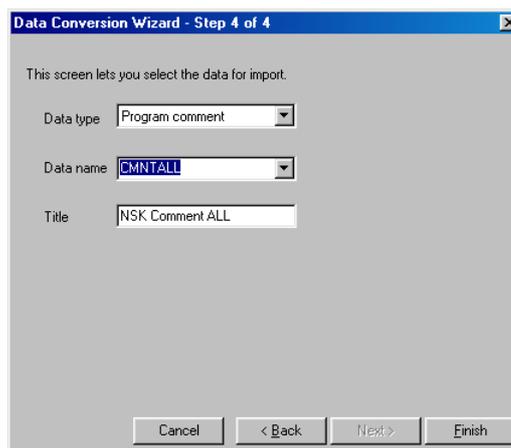
(3) Data conversion wizard 3/4

Make sure that the column parts in the [Data Preview] list are in order of [Device Number], [Label] and [Comment], and click [Next>].



(4) Data conversion wizard 4/4

Choose [Data type]-[Common comment] or [Program comment], set the comment file name used on GX Developer in [Data name] and a comment annotation in [Title], and click [Finish].



5. DEVICE COMMENT CREATION

5.3 Converting Device Comments

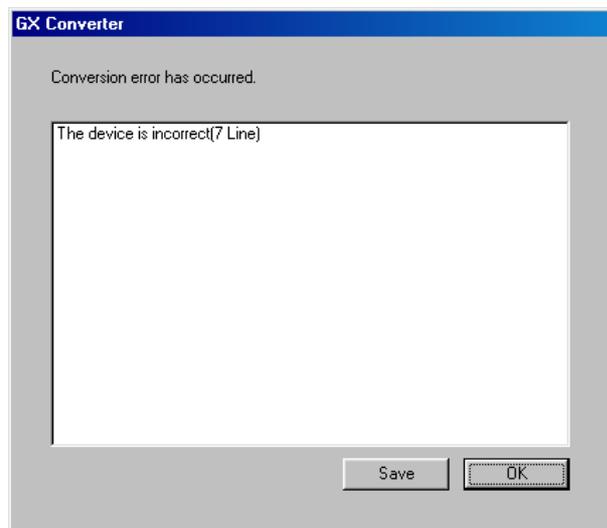
(5) Completion

The setting is complete when the following dialog appears. Click [OK].



(6) Error status

If an error occurred during conversion, its status and the line where it occurred are displayed.



6. RELATIONSHIPS BETWEEN GX DEVELOPER AND ONBOARD FUNCTIONS

6.1 Function Differences

6. RELATIONSHIPS BETWEEN GX DEVELOPER AND ONBOARD FUNCTIONS

The CNC controller has onboard functions as the PLC development environment that does not use GX Developer. The onboard functions have monitoring and editing functions for ladder program testing and adjustment.

This chapter explains the items related to GX Developer and onboard functions.

6.1 Function Differences

The onboard functions do not have the following functions among the main functions as the PLC development environment.

- **Function to create a new ladder program**
- **All PLC message-related functions**
- **All device comment-related functions**

6.2 Specification Differences

There are the following differences in specifications between GX Developer and onboard functions.

(1) Restriction specifications related to circuit display/editing

The restriction specifications related to circuit display/editing differ between GX Developer and onboard functions. Since the restriction specifications of the onboard functions are narrower than those of GX Developer, care should be taken when creating a circuit on the GX Developer side.

(a) Limits according to circuit handling specifications

		Onboard	GX Developer
Display specifications of single screen	Number of contacts	8 contacts, 1 coil	11 contacts, 1 coil
	Number of lines	9 lines	Depending on window size and screen reduction ratio
Restriction specifications of single circuit		18 lines (return count: 0) (Note 1)	24 lines of 200 or more series contacts

(Note 1) The relationships between the return count and the maximum value of the vertical width of the circuit that can be created at that time are as follows.

Return count (times)	0	1	2	3	4	5	6	7	8
Vertical width of circuit (lines)	18	11	7	5	4	3	3	2	2

Excess over the above will result in "LADDER ERROR".

(b) Handling a circuit exceeding the 18-line restriction specifications

If the circuit created on the GX Developer side exceeds the circuit restriction specifications of the onboard functions, the following message is displayed.

Operation	Message	Handling
When "read" or "monitor" function is used to display circuit	"DISPLAY OVER FLOW"	Circuit is not displayed. (Only bus is displayed)
When "write", "insertion", "deletion" or "conversion" function is selected	"NOT WRITE"	Editing operation is inhibited.

6. RELATIONSHIPS BETWEEN GX DEVELOPER AND ONBOARD FUNCTIONS

6.2 Specification Differences

(2) Step count calculation specifications

The step count specifications in the individual instructions of the MELSEC and MELDAS differ in some instructions. Therefore, steps may be different when the same circuit is displayed by GX Developer and onboard functions.

This will give rise to a problem especially when the circuit diagram printed from GX Developer is used for maintenance with the onboard functions. To resolve this problem, a tool (PCNV6LM) to print a circuit diagram in the MELDAS step specifications is available.

(3) NC instructions which cannot be used with GX Developer

MELDAS original specification instructions that cannot be handled with GX Developer are used after being replaced with alternate instructions which can be handled by GX Developer. The specifications when these instructions are displayed with the onboard are shown below.

Table of alternate instruction correspondence

Classification	Instruction symbol	Onboard display	GX Developer display
Bit	DEFR	—[DEFR D]—	D — ↑ —
Average value	AVE	—[AVE S D n]	—[S.AVE S D n]
Carry flag set	STC	—[STC]	—[S.STC]
Carry flag reset	CLC	—[CLC]	—[S.CLC]
ATC	ATC	—[ATC Kn Rn Rm Mm]	—[S.ATC Kn Rn Rm Mm]
ROT	ROT	—[ROT Kn Rn Rm Mm]	—[S.ROT Kn Rn Rm Mm]
TSRH	TSRH	—[TSRH Rm Rn Mn]	—[S.TSRH Rm Rn Mn]
DDBA	DDBA	—[DDBA Rn/Dn]	—[S. DDBA Rn/Dn]
DDBS	DDBS	—[DDBS Rn]	—[S. DDBS Rn]
CAL1	CAL1	—[CAL1 Pn]	—[S. CAL1 Pn]
BIT	LDBIT	[<= S1 n]—	[<= S1 n]—
	ANDBIT	—[<= S1 n]—	—[<= S1 n]—
	ORBIT	└[<= S1 n]┘	└[<= S1 n]┘
	LDBII	[<> S1 n]—	[<> S1 n]—
	ANDBII	—[<> S1 n]—	—[<> S1 n]—
	ORBII	└[<> S1 n]┘	└[<> S1 n]┘

(4) Circuit representation specifications

The circuit of the END instruction is not displayed by the onboard functions.

7. PROCEDURE FOR WRITING PLC PROGRAMS, ETC. TO ROM
7.1 Data That Can Be Written to ROM

7. PROCEDURE FOR WRITING PLC PROGRAMS, ETC. TO ROM

This chapter explains the procedure for ROM writing for the ladder programs and PLC-related data that have been developed. For the M60/60S series, this operation is not required since data is written to the ROM as soon as it is written from GX Developer.

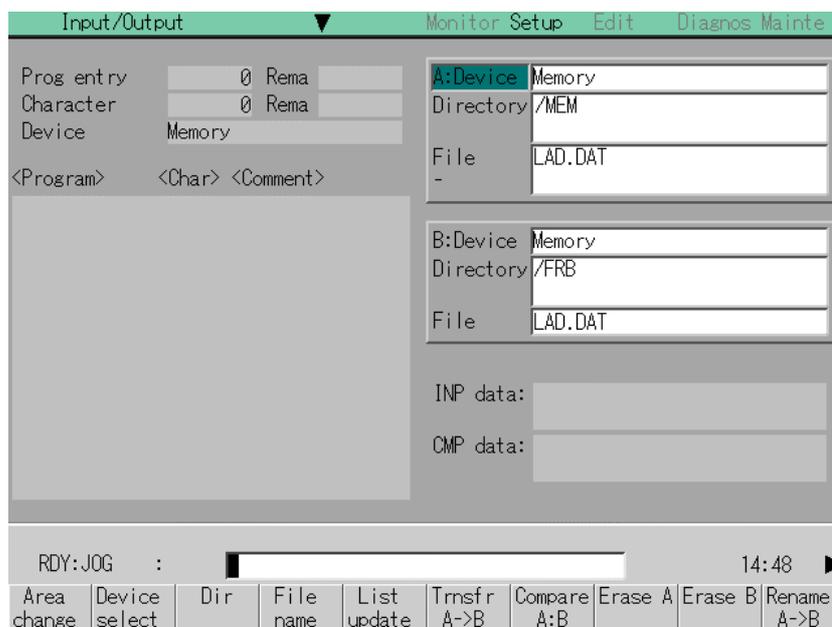
7.1 Data That Can Be Written to ROM

The following four data can be written to ROM. Note that multiple data of the same type cannot be written to ROM.

	Related data classification	Remarks
1	PLC program (ladder)	PLC program code
2	PLC program comment	Comment data for GPPQ/W
3	Message 1st language	1st language data of message data such as alarm messages, operator messages, and PLC switches
4	Message 2nd language	Same as above (2nd language data)

7.2 ROM Writing Operation Procedure

GX Developer does not have the ROM writing function for ladder programs, etc. from GX Developer by remote operation. Perform ROM writing operation from the I/O screen of the CNC controller. By specifying the transfer source information (internal RAM) and transfer destination information (F-ROM) on the I/O screen and starting transfer, the ladder program, etc. stored in the RAM are written to the ROM. The contents of ROM is erased automatically before write is performed. The following shows the I/O screen example of the M600M. (Write from internal RAM to external cassette F-ROM)



7. PROCEDURE FOR ROMING PLC PROGRAMS, ETC.

7.2 ROM Writing Operation Procedure

The following gives the common operations and individual operations of the M600M and M600L operation procedures. For details of the I/O screen, refer to the instruction manual of that model.

(1) Common operation

Set the [Device], [Directory] and [File] of [A:Device] and [B:Device] respectively. Selecting any of the following functions on the menu keys starts the corresponding operation.

(a) [Trnsfr A->B]

Data is transferred from Device A to Device B. If Device B is an F-ROM, the F-ROM contents is erased and the data of Device A is written to it. Devices A, B may be other than the internal RAM or F-ROM (storage such as the hard disk).

(b) [Compare A:B]

Data of Device A and Device B are compared. Devices A, B may be other than the internal RAM or F-ROM.

(c) [Erase A]

The data of Device A is erased. Note that it is not erased when the internal RAM is specified.

(d) [Erase B]

The data of Device B is erased. Note that it is not erased when the internal RAM is specified.

(2) Individual operations (How to set the [Device], [Directory] and [File])

(a) M600M

(i) Device selection

Specify [Memory] from the menu key.

(ii) Directory designation

Type the storage region type in the setting box.

Internal RAM : "/MEM"

Internal F-ROM: "/FRA"

External cassette F-ROM: "/FRB"

(iii) File name designation

Type the file name in the setting box.

Fixed to "LAD.DAT"

(b) M600L

(i) Device selection

Specify [Memory] from the menu key.

(ii) Directory designation

Specify [PLC prog] from the menu key.

(iii) File name designation

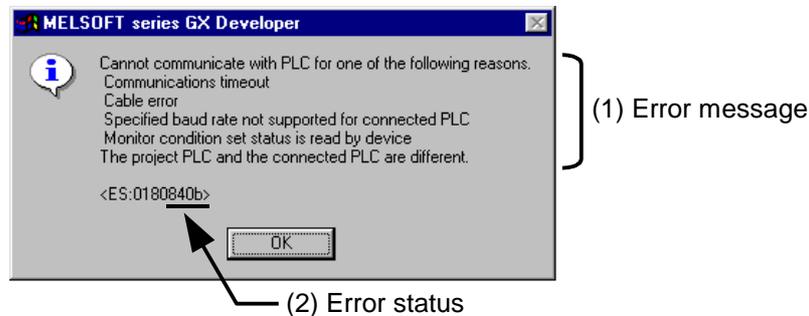
Specify the storage region type from the menu key.

[Inside RAM], [Inside F-ROM] or [Outside F-ROM]

8. LIST OF ERROR STATES

8. LIST OF ERROR STATES

If an error has occurred in GX Developer, the following dialog appears. The error message and error status are displayed in the dialog.



[Note] When an error occurred at GX Developer On-line function, the error message may not explain exactly the state in the CNC controller side. Always refer to the error list.

The following table indicates the causes and remedies of the errors that can occur during online operation with the CNC controller. For other errors, refer to the GX Developer Operating Manual.

[Note] Read "PLC" in the messages as "CNC controller".

Status	Message	Cause	Remedy
2056	The executed function is not supported. Please check the manual and other documentation.	The GX Developer version and NC do not match.	This occurs if the GX Developer version is 7.10L to 7.14Q. Upgrade the version to 7.17T or above. The version upgrade can be downloaded from MELFANS web.
4002	The executed function is not supported. Please check the manual and other documentation.	An operation not supported by the specifications was attempted.	Check the operation procedures.
4010	Cannot write because the PLC is executing a RUN command. Stop the PLC, then execute again.	The PLC of the NC is running.	After stopping the PLC of the NC, start execution again.
4021	The applicable drive is not ready. Check the applicable drive, then execute again.	The specified target memory does not exist or is not in a usable status.	Change the target memory.
4029	Insufficient file capacity. Execute again after deleting unnecessary files.	An attempt was made to write a file that exceeds the storage capacity.	Examine the file structure so that the data falls within the limited capacity.
402b	The file cannot be accessed. Carry out formatting, then execute again.	An attempt was made to write the same type of file.	After deleting the same type of file from the NC side, start execution again.
4031	The specified device No. exceeds the permissible range. Specify a device No. that is within the range set in the parameter.	The access request given is outside the accessible device range.	Check the number range of each device.
4052	The file is write protected. Change the file attributes to enable writing to the file.	The specified target memory is a write-disabled device (F-ROM).	Specify "internal RAM" as the target memory.
4053	Writing to the flash ROM failed.	An error occurred in the process to erase or to write data into the flash ROM. (Only MELDAS 60/60S Series)	The hardware may be faulty or deteriorated. Contact a Mitsubishi Sales Office.

8. LIST OF ERROR STATES

Status	Message	Cause	Remedy
4070	The program before correction differs from the registered program.	A ladder instruction outside the specification is included.	Perform verification to identify the instruction that is the cause of the problem.
4080	Incorrect abnormal.	When executing "Read from PLC" or "Verify PLC" function: Data not included in the specifications was found in the designated file.	The sequence program or message data in the CNC controller may be damaged. Delete the corresponding file and start again, or initialize the PLC data storage area. If the problem cannot be resolved, contact the Mitsubishi System Department.
		When executing "Write to PLC": Multiple END instructions were found in the designated sequence program file.	Edit the sequence program in the list mode to delete END instructions except only one at the last line.
8201	Cannot communicate with the PLC. Execute again after checking the connections with the PLC.	The serial communication cable is faulty. <ul style="list-style-type: none"> • Not connected • DTR signal off 	Check the serial port setting and cable connection.
840b	Cannot communicate with PLC for one of the following reasons. Communications timeout Cable error Specified baud rate not supported for connected PLC Monitor condition set status is read by device The project PLC and the connected PLC are different.	There is no response from the NC. <ul style="list-style-type: none"> • The CNC has not started properly. • The connection channel of the CNC side serial port is different. • The serial cable outside the specifications is used for signal connection. • An incorrect Ethernet address is set on the NC side 	Check the following. <ul style="list-style-type: none"> • CNC side status • Cable connection • Bit selection: GPP communication valid • Set the Ethernet address

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

APPENDIX 1. OPERATION METHODS USING GPPQ

This section explains the GPPQ proper operation method of the function explanation and the PLC program development procedure when "MELSEC QnA Series GPP Function Software Package (GPPQ)" is used as development tool.

Appendix 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

The GPPQ functions explained here are those supported by the M600, M60/60S Series in the "off-line functions" operated with the GPPQ independently and "on-line functions" carried out connected to the CNC controller.

Refer to the enclosed Operating Manual (off-line section and on-line section) for function details.

Appendix 1.1.1 Function support conditions (general section)

The following shows a list of GPPQ outline functions supported by the M600, M60/60S Series. A © mark indicates functions that can be used by the M600, M60/60S Series. An ✕ mark indicates that the function cannot be used because it is related to "MELSEC Series" characteristic functions. The function details during on-line are described in the next section.

List of general section functions (1) © : Possible, ✕ : Support not possible/not determined

Mode	Function	Support	Remarks
Initialization	New creation	©	
	File new read	©	
	PLC new read	→	Refer to the List of on-line section functions
	File quit	©	
	PLC type change	✕	Q4ACPU only
Option	Environment setting	©	
	Display and operation option	©	
	Startup setting	©	
Circuit	Write	→	Refer to the List of on-line section functions
	Read	→	Refer to the List of on-line section functions
	Monitor	→	Refer to the List of on-line section functions
	Test	→	Refer to the List of on-line section functions
	Debug	→	Refer to the List of on-line section functions
	File access	→	Refer to the List of on-line section functions
	PLC access	→	Refer to the List of on-line section functions
	Program search	→	Refer to the List of on-line section functions
	Comment display	→	Refer to the List of on-line section functions
	Program edit	→	Refer to the List of on-line section functions
	Monitor and test	→	Refer to the List of on-line section functions
	Window changeover	→	Refer to the List of on-line section functions
	Option	→	Refer to the List of on-line section functions
List	Write	©	
	Read	©	
	File access	©	
	PLC access	→	Refer to the List of on-line section functions
	Program search	©	
	Comment display	©	
	Program edit	©	
	Window changeover	©	
	Option	©	

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

List of general section functions (2) ◎ : Possible, ✕ : Support not possible/not determined

Mode	Function	Support	Remarks
Parameter	PLC name setting	✕	
	PLC system setting	✕	
	PLC file setting	✕	
	Device setting	✕	
	PLC RAS setting	✕	
	I/O assignment	✕	
	MELSECNET (II,10)/Ethernet setting	✕	
	MELSECNET/MINI setting	✕	
	MELSECNET/J setting	✕	
	Miscellaneous setting	✕	
	Duplex PLC setting	✕	
	SFC	✕	
	X/Y assignment confirmation	✕	
	File access	✕	
	PLC access	✕	
	Parameter search	✕	
	Parameter edit	✕	
	Miscellaneous setting	✕	
Window changeover	✕		
Option	✕		
Device	Device value input	→	Refer to the List of on-line section functions
	Device initialization	→	Refer to the List of on-line section functions
	Buffer memory simulation	→	Refer to the List of on-line section functions
	File access	◎	
	PLC access	→	Refer to the List of on-line section functions
	Device search	◎	
	Device display	◎	
	Device edit	→	Refer to the List of on-line section functions
	Window changeover	◎	
	Option	◎	
On-line	Drive name selection	→	Refer to the List of on-line section functions
	File name selection	→	Refer to the List of on-line section functions
	File access	→	Refer to the List of on-line section functions
	PLC access	→	Refer to the List of on-line section functions
	Data search	→	Refer to the List of on-line section functions
	Trace	→	Refer to the List of on-line section functions
	Trace device edit	→	Refer to the List of on-line section functions
	Device test	→	Refer to the List of on-line section functions
	Window changeover	→	Refer to the List of on-line section functions
PLC diagnosis	Diagnosis target selection	✕	
	Current error display	✕	
	Fault history display	✕	
	CPU message	✕	
	Unit detailed display	✕	
	File access	✕	
	PLC access	→	Refer to the List of on-line section functions
	Data search	✕	
	Error display	✕	
	Network monitor and test	✕	
	Window changeover	✕	
Option	✕		

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

List of general section functions (3) ◎ : Possible, ✕ : Support not possible/not determined

Mode	Function	Support	Remarks
Text creation	Device comment	◎	
	Pointer statement	◎	
	Interlinear statement	◎	
	Note	◎	
	Print statement	◎	
	File access	◎	
	PLC access	→	Refer to the List of on-line section functions
	Comment search	◎	
	Comment edit	◎	
	Window changeover	◎	
	Option	◎	
Printer	Printer data setting	◎	
	Print execution	◎	
	Printer setting	◎	
	Paper size change	◎	
	Edit	◎	
	Window changeover	◎	
	Option	◎	
File maintenance	Machine name/File name selection	◎	
	File access	◎	
	PLC access	→	Refer to the List of on-line section functions
	File search	◎	
	Display	◎	
	IC memory card	✕	
	Window changeover	◎	
Program generation	Label name definition	◎	
	File access	◎	
	Program generation	◎	
	Label display	◎	
	Display	◎	
	Label edit	◎	
	Window changeover	◎	
	Option	◎	
SFC		✕	
Quit		◎	

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

Appendix 1.1.2 Function support conditions (on-line section)

The following shows a list of GPPQ on-line functions supported by the M600, M60/60S Series. A © mark indicates functions that can currently be used by the M600, M60/60S Series. An ✕ mark indicates that the function cannot be used because it is related to "MELSEC Series" characteristic functions.

List of on-line section functions (1) ©: Possible, ✕: Support not possible/not determined

Mode	Function	Menu	Detailed function	Support	Remarks		
Circuit mode	6.2 Circuit write	6.2.1 Write during run		✕			
	6.3 Circuit read	6.3.1 Step No. read		©			
		6.3.2 Instruction read		©			
		6.3.3 Device read		©			
		6.3.4 Contact and coil designation read		©			
		6.3.5 Statement and note designation read		©			
		6.3.6 Last circuit designation read		©			
	6.4 Monitor	6.4.1 Monitor screen common display	Monitor destination CPU name		©		
			No. of Monitor program steps		©		
			Monitor program name		©		
			Scan time		✕		
			Monitor interval		©		
			CPU operation status		©		
			Monitor destination		©		
			Flicker during communication		©		
		6.4.2 Circuit monitor	Bit device monitor display		©		
			Word device monitor display		©		
			Digit designation monitor display		©		
			Timer/counter monitor display		©		
			Double word monitor display		©		
			Index modification monitor display		©		
			Monitor step changeover		©		
			Changeover to next circuit block		©		
			Changeover to previous circuit block		©		
			Monitor stop		©		
			Monitor column numeric value/No. of display levels changeover		©		
			6.4.3 Device registration monitor	Bit device monitor display		©	
		Word device monitor display			©		
		Digit designation monitor display			©		
		Timer/counter monitor display			©		
		Double word monitor display			©		
		Index modification monitor display			©		
		16-bit integer designation			©		
		32-bit integer designation			©		
		Real number designation			✕		
		Character designation			✕		
		Registration device erase		©			
		6.4.4 ON/OFF cause automatic search			©		
		6.5 Test	6.5.1 Monitor trigger stop	Stop at bit device rising edge		✕	
				Stop at bit device falling edge		✕	
Stop at word device designation value				✕			
Monitor restart				✕			

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

List of on-line section functions (2) ◎ : Possible, ✕ : Support not possible/not determined

Mode	Function	Menu	Detailed function	Support	Remarks
Circuit mode	6.5 Test	6.5.2 Forced ON/OFF	Forced ON	✕	
			Forced OFF	✕	
		6.5.3 Current value change	16-bit change	✕	
			32-bit change	✕	
		6.5.4 Setting value change	Timer	✕	
			Counter	✕	
	6.6 Debug	6.6.1 Step execution		✕	
		6.6.2 Partial execution		✕	
	6.7 File menu	6.7.1 Read		◎	
		6.7.2 Write		◎	
		6.7.3 Compare		◎	
	6.8 PLC menu	6.8.1 Read		◎	
		6.8.2 Write	Parameter	✕	
			Sequence program	◎	
			Entire range	◎	
			Range designation	✕	
			Device comment	◎	
			Device default value	✕	
			Simulation data	✕	
			File register	✕	
			Device memory	✕	
			Entire range	✕	
			Range designation	✕	
			6.8.3 Compare		◎
		6.8.4 New edit target read		✕	
		6.8.5 List of files	File name	◎	
			Classification	◎	
			Size	◎	
			Date	◎	
			Time	◎	
			Header statement	◎	
			No. of files	◎	
			Continuous max. open capacity	◎	
			Entire open capacity	◎	
6.8.6 Connection designation		Peripheral side interface	◎		
		PLC side interface	✕	Fixed	
		Target CPU	✕	Fixed	
	Target memory	✕	Fixed		
6.8.7 Remote operation	CPU status display	◎			
	RUN	◎			
	STOP	◎			
	PAUSE	✕			
	STEP-RUN	✕			
	Latch clear	✕			
	Reset	✕			
	Operation during RUN, STEP-RUN	✕			
	Execution destination designation	✕			
	Execution unit designation	✕			
6.8.8 Write option		✕	For write during run		

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

List of on-line section functions (3) Ⓞ : Possible, ✕ : Support not possible/not determined

Mode	Function	Menu	Detailed function	Support	Remarks	
Circuit mode	6.9 Display menu	6.9.1 Circuit display changeover	Device name display	Ⓞ		
			Comment display	Ⓞ		
			Statement display	Ⓞ		
			Display with note	Ⓞ		
	6.10 Monitor and test menu	6.10.1 Device batch monitor	Bit device monitor display	Ⓞ		
			Word device monitor display	Ⓞ		
			Bit and word monitor	Ⓞ		
			Bit multipoint monitor	Ⓞ		
			Word multipoint monitor	Ⓞ		
			Timer/counter multipoint monitor	Ⓞ		
			Target device change	Ⓞ		
			Decimal/hexadecimal display changeover	Ⓞ		
			Integer/real number display changeover	Ⓞ		
			0 → F/F → 0 display changeover	Ⓞ		
			Forced ON	Ⓞ		
			Forced OFF	Ⓞ		
			6.10.2 Multiple device batch monitor	Bit device monitor display	Ⓞ	
				Word device monitor display	Ⓞ	
		Decimal/hexadecimal display changeover		Ⓞ		
		Integer/real number display changeover		Ⓞ		
		Registration device erase		Ⓞ		
		Forced ON		Ⓞ		
		Forced OFF		Ⓞ		
		6.10.3 ON/OFF cause automatic search		Ⓞ		
		6.10.4 Scan time measure		✕		
		6.10.5 Device registration monitor	Bit device monitor display	Ⓞ		
			Word device monitor display	Ⓞ		
			Digit designation monitor display	Ⓞ		
			Timer/counter monitor display	Ⓞ		
			Double word monitor display	Ⓞ		
			Index modification monitor display	Ⓞ		
			16-bit integer designation	Ⓞ		
			32-bit integer designation	Ⓞ		
			Real number designation	Ⓞ		
			Character designation	Ⓞ		
			Registration device erase	Ⓞ		
		6.10.6 Monitor and test menu	Constant monitoring	✕		
			Word device condition	Ⓞ		
			Bit device condition	Ⓞ		
			Step No.	✕		
		6.10.7 Monitor stop condition setting	Monitoring does not stop	Ⓞ		
			Word device condition	Ⓞ		
Bit device condition			Ⓞ			
Operation status			✕			
6.10.8 Monitor data registration/application			Ⓞ			
6.10.9 Device test		Forced ON	Ⓞ			
	Forced OFF	Ⓞ				
	Forced ON/OFF reverse	Ⓞ				
	Current value change	Ⓞ				

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

List of on-line section functions (4) ◎ : Possible, ✕ : Support not possible/not determined

Mode	Function	Menu	Detailed function	Support	Remarks	
Circuit mode	6.10 Monitor and test menu	6.10.10 Sampling/monitoring trace		✕		
		6.10.11 Step execution		✕		
		6.10.12 Partial execution		✕		
		6.10.13 Skip execution		✕		
		6.10.14 Simulation		✕		
		6.10.15 Monitor column numeric value display changeover	Numeric value display format	◎		
			No. of display level	◎		
6.10.16 Program execution status monitor		✕				
Parameter mode	8.2 PLC menu	6.11.1 Monitor destination setting		✕		
				→	Refer to 6.8 PLC menu	
Device mode	9.2 Device value input			✕		
	9.3 Default value range setting			✕		
	9.4 Buffer memory simulation			✕		
	9.5 PLC menu			→	Refer to 6.8 PLC menu	
	9.6 Edit menu	9.6.1 Simulation range		✕		
On-line mode	10.2 Drive name selection		Internal RAM	◎	Fixed	
			IC memory card	✕	Fixed	
	10.3 File selection			◎		
	10.4 File menu	10.4.1 Read			◎	
		10.4.2 Write			◎	
		10.4.3 Compare			◎	
		10.4.4 New edit target read			◎	
		10.4.5 List of files			◎	
		10.4.6 Delete			◎	
		10.4.7 File quit			◎	
	10.5 PLC menu	10.5.1 Read			◎	
			10.5.2 Write			
			Parameter	✕		
			Sequence program	◎		
			Entire range	◎		
			Range designation	✕		
			Device comment	◎		
			Device default value	✕		
			Simulation data	✕		
			File register	✕		
			Device memory	✕		
			Entire range	✕		
			Range designation	✕		
		10.5.3 Compare			◎	
		10.5.4 New edit target read			◎	
	10.5.5 List of files		File name	◎		
			Classification	◎		
		Size	◎			
		Date	◎			
		Time	◎			
		Header statement	◎			
		No. of files	◎			
		Continuous max. open capacity	◎			
		Entire open capacity	◎			
10.5.6 Connection designation			Peripheral side interface	◎		
		PLC side interface	✕	Fixed		
		Target CPU	✕	Fixed		
		Target memory	✕	Fixed		

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.1 GPPQ Function Outline and Functions Supported by the M600, M60/60S Series

List of on-line section functions (5) ◎ : Possible, ✕ : Support not possible/not determined

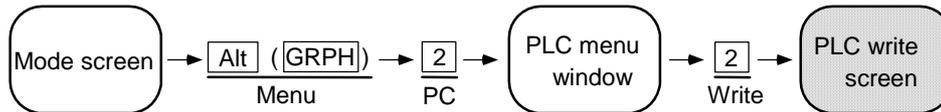
Mode	Function	Menu	Detailed function	Support	Remarks	
On-line mode	10.5 PLC menu	10.5.7 Remote operation	CPU status display	◎		
			RUN	◎		
			STOP	◎		
			PAUSE	✕		
			STEP-RUN	✕		
			Latch clear	✕		
			Reset	✕		
			Operation during RUN, STEP-RUN	✕		
			Execution destination designation	✕		
			Execution unit designation	✕		
			10.5.8 Write option		✕	For write during run
			10.5.9 Key word registration		✕	
			10.5.10 Device memory operation		✕	
			10.5.11 PLC memory batch operation		✕	
	10.5.12 Delete		◎			
	10.5.13 Header statement creation		✕			
	10.6 Search menu	10.6.1 Head/end search		✕		
			10.6.2 File search	✕		
			10.6.3 Frequency search	✕		
			10.6.4 Time search	✕		
			10.6.5 Data search	✕		
	10.7 Trace menu	10.7.1 Sampling trace		✕		
			10.7.2 Monitoring trace	✕		
10.7.3 Status latch			✕			
10.7.4 Program trace			✕			
10.8 Edit menu	10.8.1 Cut/copy/paste		✕			
10.9 Test menu	10.9.1 Device test		✕			
		Forced ON	◎			
		Forced OFF	◎			
		Forced ON/OFF reverse	◎			
	Current value change		◎			
10.10 Window menu	10.10.1 Display window change			◎		
PLC diagnosis mode	11.2 Diagnosis target selection			✕		
	11.3 Current error display			✕		
	11.4 Fault history display			✕		
	11.5 CPU message			✕		
	11.6 Unit detailed display			✕		
	11.7 File menu			✕		
	11.8 PLC menu			✕		
	11.9 Search menu			✕		
	11.10 Display menu			✕		
	11.11 Network menu			✕		
	11.12 Window menu			✕		
11.13 Option menu			✕			
File maintenance mode	13.2 PLC menu			→	Refer to 6.8 PLC menu	
	13.3 IC card menu			✕		

Appendix 1.2 Ladder Transfer to the CNC Controller

The method of transferring a ladder to the CNC controller with the GPPQ (especially the restrictions and M600, M60/60S characteristic operations) is explained below.

Appendix 1.2.1 Menu selection

The screen is selected from the menu using the following GPPQ basic operation.



Appendix 1.2.2 Screen operation

The required options are selected and executed on the following screen.

[Write to PLC]			
Interface	RS232C <-->	QnACPU	
Target PLC	Network : 0	Station : FF	PLC Type : Q4A
Target Mem	Internal RAM	Title []
1. File	1. File Name [Title []
	2. Tgt	1. [*] Parameter	
		2. [*] Seq/SFC Prog	1. (*) Whole Range ←
		3. [] Device Comment	2. () Step Range []-[]
		4. [] Dev Init Value	3. () Step Range P []-[]
		5. [] Simulation Data	4. () Block Range []-[]
		6. [] File Register	1. (*) Whole Range
			2. () Specify ZR[]-[]
2. Device Mem	1. [] Internal		1. (*) Whole Range
			2. () Specify Detail Range
Execute(Y)		Cancel(N)	
Ctrl+L : filelist Ctrl+D : directo Space : sele Esc : close			

[Restrictions]

The following options of [2. Tgt] under [1. File] can be selected.

- [1. Parameter]
- [2. Seq/SFC Prog] and [1. Whole Range]
- [3. Device Comment]

Note that nothing will be transferred even if [1. Parameter] is selected.

APPENDIX 1. OPERATION METHODS USING GPPQ
APPENDIX 1.2 Ladder Transfer to the CNC Controller

Appendix 1.2.3 Operation during a transfer error

The ladder data is converted to the ladder machine code characteristic to the CNC controller simultaneously with the ladder transfer from the GPPQ to the CNC controller. A conversion error will occur if devices or instruction formats are used that are not supported by the M600, M60/60S Series. During a conversion error, the following message appears at the bottom of the GPPQ screen after the M600 ladder machine code is converted to an "NOP code" and transferred to the last step.

Cannot communicate with PLC. Error No.=4070

The following warning statement appears in the header statement item when the file having the conversion error is displayed with the PLC file list function.

[List]						
:						
File	Type	Size	Data	Time	Title	
PLCTEST	QnA Seq	12345	97-11-18	10:34	[**WARNING: Checksum ERROR!**]	
:						

If an attempt is made to run the PLC in this status, the following alarm message will appear on the NC side, and the PLC will not run.

SY0002 User ladder code error

Appendix 1.2.4 Error step No. confirmation method

The error step can be confirmed using the PLC compare function. The mismatched contents appear as in the following example when the PLC comparison is executed. Refer to "Appendix 1.4 Ladder comparison with the CNC controller" for details.

[Sequence Program Diff List]

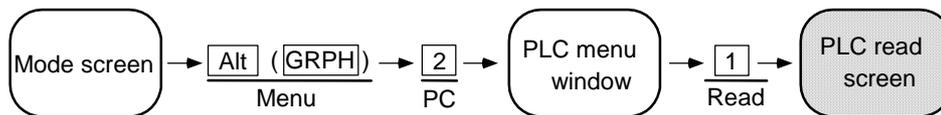
[Sequence Program Diff List]						
<Memory>			<PLC>			
Step	Ins	I/O	Step	Ins	I/O	
15	AND=	R4918	15	NOP		
		K106				
18	+	R4916	16	NOP		
		K10000				
		D87				
22	END	17		NOP		
	No Data exists.	18		+	R4916	
		22		END	K10000	
		22		END	D87	
	No Data exists.	22		END		
PgDn : Next page				Esc: Close		

Appendix 1.3 Ladder Read from the CNC Controller

The method of reading a ladder from the CNC controller to the GPPQ is explained below.

Appendix 1.3.1 Menu selection

The screen is selected from the menu using the following GPPQ basic operation.



Appendix 1.3.2 Screen operation

The file names are designated and executed on the following screen.

File name designation method

Input the file name stored in the read target memory.

Display the read target memory list screen by pressing **Ctrl** + **D**.

Display the file list screen by pressing **Ctrl** + **L**.

After selecting the file using **↑** and **↓**, display the file name selected in the file name designation column by pressing **Enter**.

[Read from PLC]			
Interface	RS232C <--->	QnACPU	
Target PLC	Network : 0	Station : FF	PC Type : Q4A
Target Mem	Internal RAM Title []		
1. File	1. File Name []	Title []	
	2. Tgt		
	1. [*] Parameter	1. (*) Whole Range	
	2. [*] Seq/SFC Prog	2. () Step Range []-[]	
	3. [] Device Comment	3. () Step Range P [] - []	
	4. [] Dev Init Value	4. () Block Range []-[]	
	5. [] Simulation Data	1. (*) Whole Range	
	6. [] File Register	2. () Specify ZR[]-[]	
2. Device Mem	1. [] Internal		
	1. (*) Whole Range		
	2. () Specify Detail Range		
Execute(Y)		Cancel(N)	
Ctrl+L : filelist Ctrl+D : directo Space : sele Esc : close			

[Restrictions]

The following options of [2. Tgt] under [1. File] can be selected.

[2. Seq/SFC Prog] and [1. Whole Range]

[3. Device Comment]

Appendix 1.5 PLC-Related Data Development Procedure

This section explains a procedure for developing PLC-related data such as alarm messages, operator messages, PLC switches. The following items are the same as those of GX Developer. Refer to the GX Developer side.

File name, development procedure, message data description method

Appendix 1.5.1 Conversion to GPPQ data

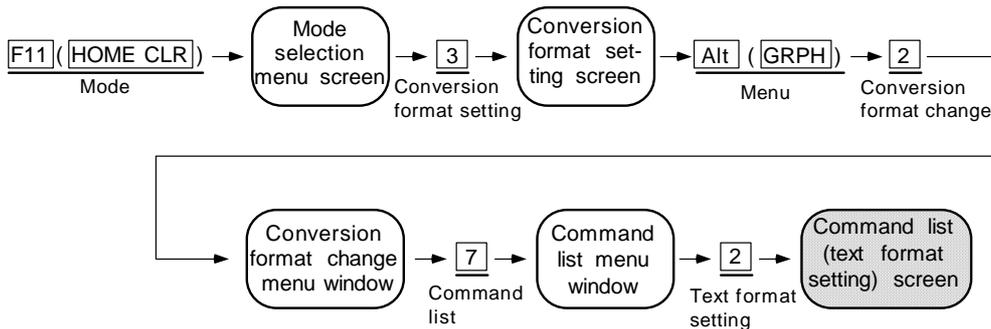
Use the "CNVQ (data conversion software package)" for conversion from text data to GPPQ data. Instead of text data, spreadsheet software data can also be converted to GPPQ data. Refer to the Operating Manual for details.

(1) Initialization

After CNVQ is started, the project file (file for setting the drive/path name, system name, machine name, file name, etc.) is read or newly created. Refer to the Operating Manual for the operation method.

(2) Conversion Format Setting

The screen is selected from the menu using the following CNVQ basic operation.



The conversion format is set on the following screen. Select the "Statement" item shown by the arrow (1) when converting message data. Refer to the CNVQ Operating Manual for details.

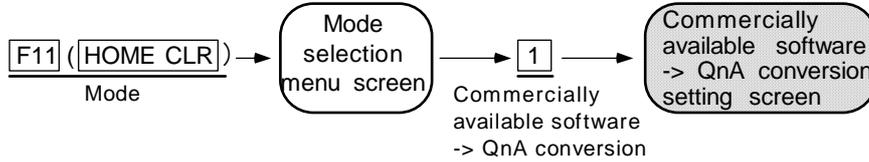
[Instruction list (text format setting)]	
1. Delimiter	1. (*) Tab 1. (*) Auto Adjust (1Tab [8]Column)
1. (*) Space	2. () Left Align
2. (*) Space	3. () Specify [1]Tab
3. (*) Statement	1. (*) Auto Adjust
4. () Note	2. () Left Align
5. () Device Label	3. () Specify [1]Column
6. () Without NOP	
Execute(Y)	Cancel(N)
Space : Select Esc : Close	

(1)

APPENDIX 1. OPERATION METHODS USING GPPQ
Appendix 1.5 PLC-Related Data Development Procedure

Appendix 1.5.2 Commercially available software QnA conversion setting

The screen is selected from the menu using the following CNVQ basic operation.

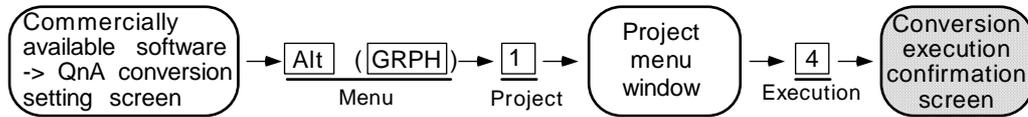


The conversion source/destination directory path and conversion file name, etc., are set on the following screen. Refer to the Operating Manual for details.

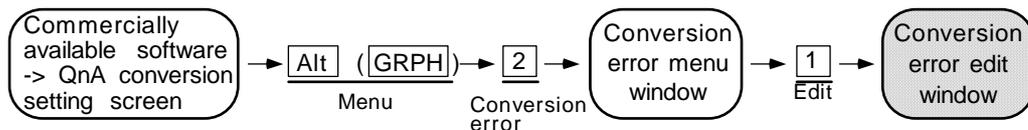
Other --> QnA Convert		C:M600\TEST\SETUPM1		<Ins> F11:Mode	
Conversion Destination File Drive/Path[C:\GPPQ\USR					
Conversion Source File Drive/Path[C:\CNVQ\USR\M600\TEST					
#	Conv	SourceFile	DestFile	Data #	FileType ConvType Another
000	[]	[M1TEST]	[M600\TEST\M1TEST] [1]	<Text> < List > []
001	[]	[]	[]] []	< > < > []
002	[]	[]	[]] []	< > < > []
003	[]	[]	[]] []	< > < > []
004	[]	[]	[]] []	< > < > []
005	[]	[]	[]] []	< > < > []
006	[]	[]	[]] []	< > < > []
007	[]	[]	[]] []	< > < > []
008	[]	[]	[]] []	< > < > []
009	[]	[]	[]] []	< > < > []
010	[]	[]	[]] []	< > < > []
PgUp : Prev Page			PgDn : Next Page		Ctrl+P :Drive/Path Enter : Decide

Appendix 1.5.3 Conversion execution

The screen is selected from the menu using the following CNVQ basic operation.



The conversion status appears at the bottom of the screen. If "There were n conversion errors" appears when the conversion is finished, display the conversion error edit screen from the menu using the following CNVQ basic operation to confirm the error contents.



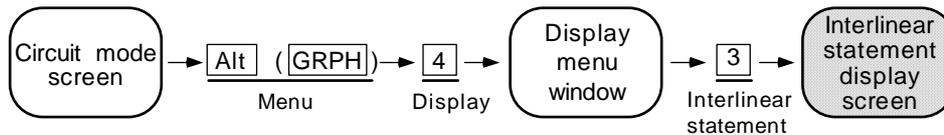
Appendix 1.5.4 Operation with the GPPQ

Message data in the GPPQ is handled as "Integrated type interlinear statements" in the PLC program. "Integrated type interlinear statements" are interlinear comments to help in understanding the PLC program. Interlinear comments transferred with the PLC program to the controller are called "integrated type".

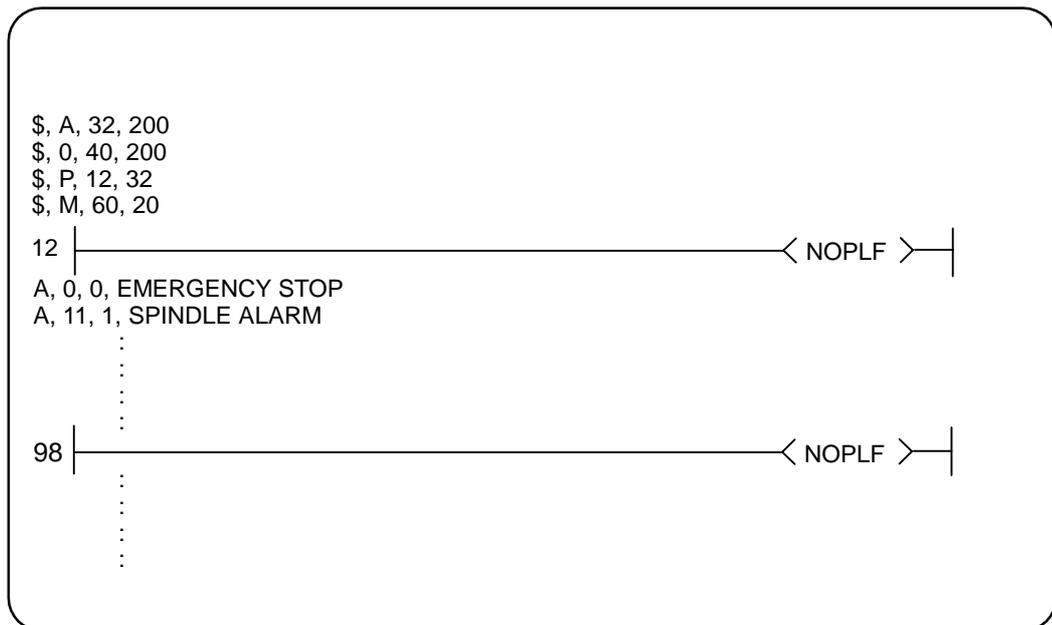
"Interlinear statements" can be displayed in the "Circuit mode", and edited on the "Interlinear statement edit screen".

(1) Interlinear Statement Display in the Circuit Mode

The screen is selected from the menu using the following GPPQ basic operation.



The following appears. At that time, the page break code and end code appear as normal ladder codes.



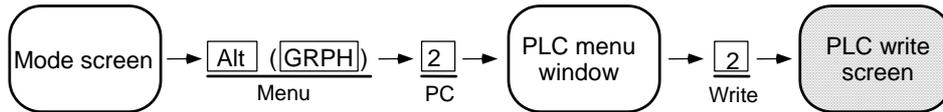
APPENDIX 1. OPERATION METHODS USING GPPQ
Appendix 1.5 PLC-Related Data Development Procedure

Appendix 1.5.5 Transfer to the controller

The following shows the method of transferring a message from the GPPQ to the CNC controller. The transfer method is the same as that of the ladder code. Ladder codes and message data are distinguished by their file names only.

(1) Menu Selection

The screen is selected from the menu using the following basic operation.



(2) Screen Operation

Transfer is executed in the following screen.

An example of transferring a message 1st language file "M1TEST.GPG" is shown here.

[Write to PLC]			
Interface	RS232C <--->	QnACPU	
Target PLC	Network : 0	Station : FF	PC Type : Q4A
Target Mem	Internal RAM	Title []
1. File	1. File Name [M1TEST]	Title [message 1st language test]	
	2. Tgt	1. [] Parameter	
		2. [*] Seq/SFC Prog	1. (*) Whole Range
			2. () Step Range []-[]
	3. [] Device Comment	3. () Step Range P [] - []	
	4. [] Dev Init Value	4. () Block Range []-[]	
	5. [] Simulation Data		
	6. [] File Register	1. (*) Whole Range	
		2. () Specify ZR[]-[]	
2. Device Mem	1. [] Internal	1. (*) Whole Range	
		2. () Specify Detail Range	
Execute(Y)		Cancel(N)	
Ctrl+L : filelist Ctrl+D : directo Space : sele Esc:close			

Appendix 1.5.6 Reading and comparing from the controller

The following describes the method of reading and comparing a message from the CNC controller to the GPPQ. The method of reading and comparing is the same as that of ladder codes. Ladder codes and message data are distinguished by their file names only.

(1) Menu Selection/Screen Operation

Refer to the following sections for operation methods.

For reading : "Appendix 1.3 Ladder Read from the CNC Controller"

For comparing : "Appendix 1.4 Ladder Comparison with CNC Controller"

(2) Message Read Format

The message description format was shown in "5.3.1 Description format", but there are no special rules concerning provision of descriptions in the setting area or the order of message description in the message area. For that reason, the description format may differ between transfer and reading of the message data.

The following shows the format during reading as the "Standard description format".

Standard description format of message data

Alarm message setting	... (a)
Operator message setting	
PLC switch setting	
Comment message setting	
Alarm messages	... (b)
Operator messages	... (c)
PLC switches	... (d)
Comment messages	... (e)
NOPLF	
END	

(a) Setting area

The settings are described in order of alarm, operator, PLC switch and comment. The maximum value is described if the setting is abbreviated.

(b) Alarm messages

Each message data is described in order of the index Nos.

(c) Operator messages

The same as the alarm messages.

(d) PLC switches

Each message data is described in order of the switch Nos.

(e) Comment messages

These messages are described in the same order as described before transfer.

(f) Others

- Spaces and tabs are not included before and after the comma(,) separating the message data factors.
- The message character string is handled the same as normal data even when blank.
- The NOPLF code between messages is described to the position to which the message data following the NOPLF code during transfer moved.

(3) Message Data Comparison

Message data comparison can be executed in the same manner as the ladder program, however, note that the target compared is the ladder code (NOPLF, END) only. Therefore, message data described as "Integrated type interlinear statements" are not compared. When comparing, read the data to the GPPQ side using the read function, then compare using the master file and file comparison function.

APPENDIX 1. OPERATION METHODS USING GPPQ
Appendix 1.6 Procedure for ROM Writing

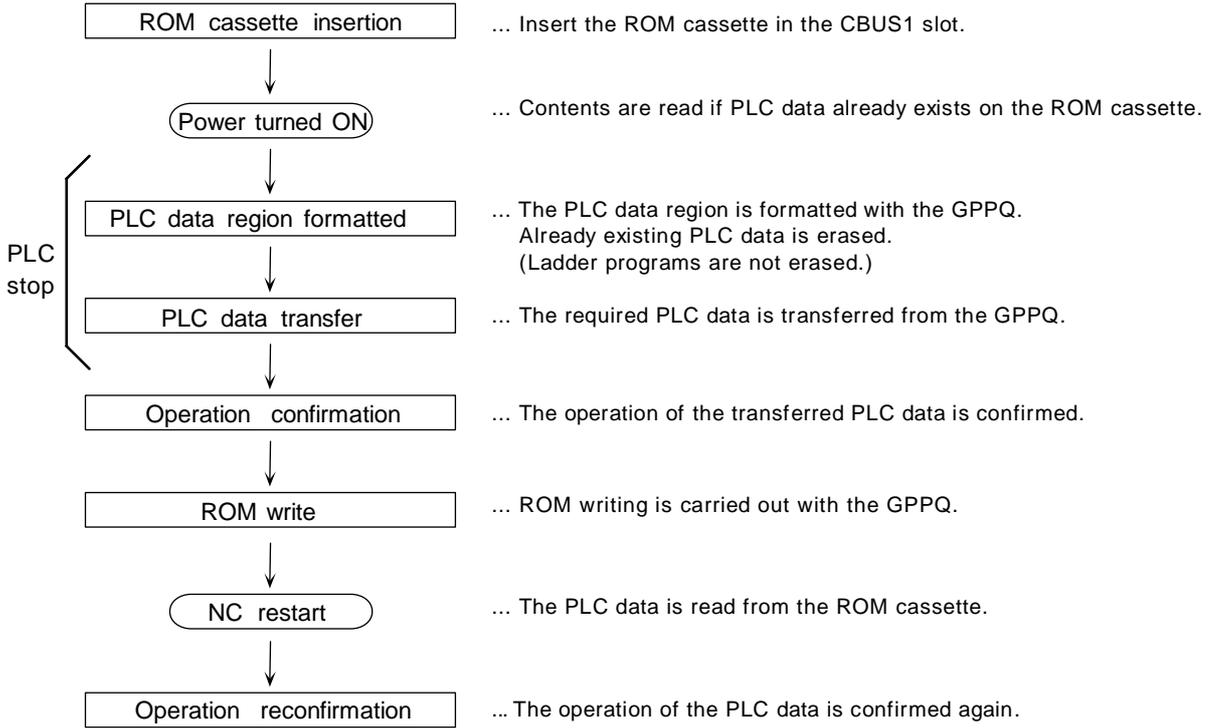
Appendix 1.6 Procedure for ROM Writing

The procedure to write the developed PLC program and PLC related data to ROM is explained here. The following items are the same as that of GX Developer. Refer to the GX Developer side.

- Target Data to be written to ROM

Appendix 1.6.1 Procedure to write to ROM

The following shows the procedure to write to ROM.



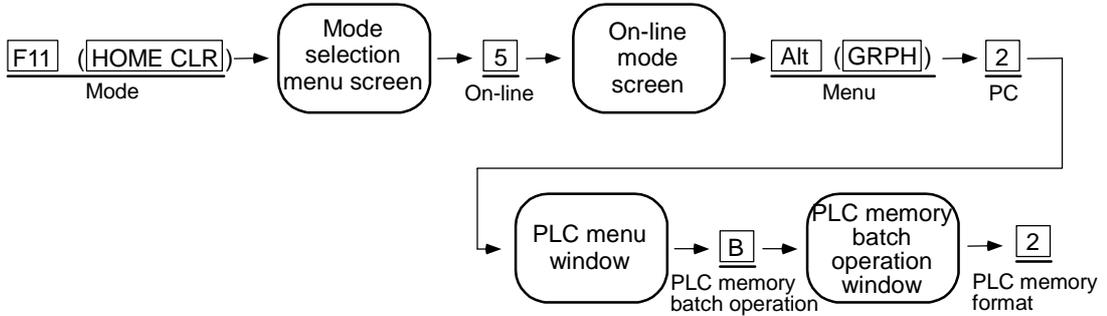
APPENDIX 1. OPERATION METHODS USING GPPQ
Appendix 1.6 Procedure for ROM Writing

Appendix 1.6.2 PLC data region formatting

It is recommended to initialize the data region before ROM writing. (This is not absolutely required, however.)

The difference from file deletion is that the file control region is initialized.

The screen is selected from the menu using the following basic operation.



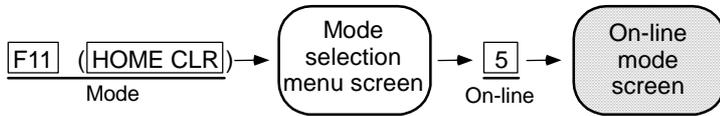
The PLC related data (excluding ladder programs) is completely erased by the above operation.

Appendix 1.6.3 ROM writing from the GPPQ

(1) On-line mode

ROM writing operations are carried out on the On-line mode screen.

The mode is selected from the menu using the following basic operation.



The following drive name selection screen or file name selection screen appears. If the file name selection screen appears, it can be returned to the drive name selection screen with the Escape [esc] key.

On-Line	CPU:Q4A	NW:Local	C:TESTM1TEST
Interface	RS232Cport	<---> QnACPU	
Target PLC	Network : 0	Station : FF	PLC Type : Q4A
Target Memory	Contents	Title	
<Drive>			
0. Internal RAM		[ON BOARD S-RAM + (D-RAM)]
1. IC Card A(RAM)		[ON BOARD D-RAM]
2. IC Card A(ROM)		[]
3. IC Card B(RAM)		[]
4. IC Card B(ROM)		[]

APPENDIX 1. OPERATION METHODS USING GPPQ
Appendix 1.6 Procedure for ROM Writing

(2) Drive name selection screen

The drive mounted on the CNC controller can be selected, and the following operations carried out, on the drive name selection screen.

- Copying between drives (Including flash ROM writing)
- Displaying the drive details (date of creation, size)
- Confirmation of the drive open capacity, etc.

The mounting and PLC data availability is checked for each drive. The header statement appears when PLC data exists. There are sections in which the drive target memory and the actual CNC controller drive meaning differ, so confirm with the header statement.

The following shows the meanings in the CNC controller of each drive.

Drive	Screen display	Meaning in the M600	Header statement display
0	Internal RAM	Internal S-RAM (ladder programs)	ON BOARD S-RAM + (D-RAM)
1	IC memory card A (RAM)	Internal D-RAM (other than ladder programs)	ON BOARD D-RAM
2	IC memory card A (ROM)	Internal F-ROM	ON BOARD F-ROM
3	IC memory card B (RAM)	Expansion S-RAM (for future use)	
4	IC memory card B (ROM)	Expansion F-ROM	ADD ON F-ROM

• Drive 0

This refers to the file groups stored in the internal S-RAM (ladder programs) and internal D-RAM (other than ladder programs).

File transfer with GPPQ is carried out in this drive 0.

Only internal S-RAM information is displayed for the drive capacity.

• Drive 1

This refers to the file groups stored in the internal D-RAM (other than ladder programs).

Only internal D-RAM information is displayed for the drive capacity.

• Drive 2

This refers to the file groups stored in the internal F-ROM. The ladder related data storage availability to the internal F-ROM differs depending on the system.

Approx. 30 seconds are required until the finish of the writing (copying to drive 2) process to the internal F-ROM.

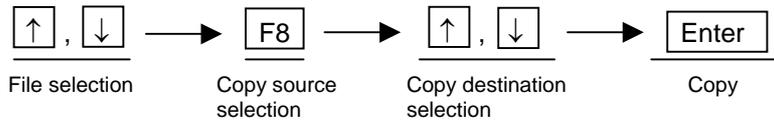
• Drive 4

This refers to the file group stored in the cassette F-ROM inserted in the expansion bus. The expansion bus insertion position is automatically judged.

Approx. 10 seconds are required until the finish of the writing (copying to drive 4) process to the cassette F-ROM.

APPENDIX 1. OPERATION METHODS USING GPPQ
Appendix 1.6 Procedure for ROM Writing

(3) Copying operation between drives (flash ROM writing operations)



Select the copy source target memory using and , and press . Set the band cursor to the copy destination target memory drive. The following dialog box appears when is pressed.

Do you want to copy from Internal RAM to IC Card B(ROM)?
Yes(Y) No(N)

Press to execute.

The copy process between drives is executed in the background. The write process starting message is displayed immediately after execution, and the next operation is not accepted during the copy process. The copy process time differs depending on the copy target.

A message will appear at the first operation after the copy process is finished if an error occurred in the copy process. No message will appear if the copy process finished normally.

The following shows the meanings of the messages during and after execution.

Cannot communicate with PLC.
Error No. = 4068

The drive background writing process has started.

Cannot communicate with PLC.
Error No. = 4008

Other processes cannot be accepted during drive writing process. Wait a few seconds and execute again.

Cannot write correctly to the ROM

The flash ROM writing was unsuccessful.

**APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B
AND M600, M60/60S**

Appendix 2.1 Development Tools, etc.

**APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT
BETWEEN PLC4B AND M600, M60/60S**

This section explains differences between the PLC4B development environment and M600, M60/60S series PLC development environment.

Appendix 2.1 Development Tools, etc.

In the M600, M60/60S, a user PLC development environment that used the MELSEC PLC development tool was constructed. Consequently, the tools used at each development process differ. A comparison of each process is shown in "Table 2.1 List of development tool comparisons". Refer to the respective Instruction Manuals for details on each tool.

Table 2.1 List of development tool comparisons

Development process		PLC4B development environment	MELSEC PLC development environment
Application from the old model	Tool	Ladder and message conversion tool (CHG4PB)	Ladder list converter (CLST6L)
	Hardware	PC9801/PC-AT	PC-AT
List -> ladder conversion	Tool	PLC development software (list section) (LIST4B)	GX Converter
	Hardware	PC9801/PC-AT	PC-AT
Ladder creation	Tool	PLC development software (ladder section) (PLC4B)	GX Developer
	Hardware	PC9801/PC-AT	PC-AT
Message creation	Tool	PLC development software (ladder section) (PLC4B)	Text editor -> GX Converter -> GX Developer
	Hardware	PC9801/PC-AT	PC-AT
Transfer to the CNC	Tool	PLC4B <-> FLD <-> M500 controller	GX Developer <-> RS232C <-> CNC controller
	Hardware	Via FLD	Via RS-232C
Monitor	Tool (1)		GX Developer
	Hardware		PC-AT <-> CNC controller
ROM writing	Tool (2)	PLC onboard (ONBD)	PLC onboard
	Hardware	M500 controller	CNC controller
ROM writing	Tool	PLC onboard -> F-ROM	I/O screen -> F-ROM
	Hardware	M500 controller	CNC controller
Print output	Tool (1)	PLC development software (ladder section) (PLC4B)	GX Developer
	Hardware	PC9801/PC-AT	PC-AT
Print output	Tool (2)	PLC onboard (ONBD)	
	Hardware	M500 controller	

**APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B
AND M600, M60/60S**

Appendix 2.2 Devices and Device Assignments

Appendix 2.2 Devices and Device Assignments

In the M600 M60/60S, the device types and assignments have been reconsidered by supporting the user PLC development environment that used the MELSEC PLC development tool.

A comparison of the differences of each device is shown in the following tables "List of device differences".

Refer to the "List of M600 PLC I/F" for details on assignments.

Table 2.2.1 List of device differences (M600M series)

Device	M500 device No.	M600 device No.	Remarks
X	X0 to X4FF	X0 to XAFF	Integrated to X and assignment changed.
U	U0 to U17F		
I	I0 to I4BF		
Y	Y0 to Y57F	Y0 to YDFF	Integrated to Y and assignment changed.
W	W0 to W1FF		
J	J0 to J63F		
S	S0 to S1FF	_____	Divided to X and Y and assignment changed.
M	M0 to N5119	M0 to N5119	Integrated to M and assignment changed.
G	G0 to G3071	M5120 to M8191	
F	F0 to F127	F0 to F255	
L	L0 to L255	←	
E	E0 to E127	SM0 to SM127	
T/Q	T0 to T15	T0 to T15	Integrated to T and assignment changed.
	Q0 to Q39	T16 to T55	
	T16 to T95	T56 to T135	
	Q40 to Q135	T136 to T231	
	T96 to T103	T232 to T239	
	Q136 to Q151	T240 to T255	
C	C0 to C23	C0 to C23	Integrated to C and assignment changed.
B	B0 to B103	C24 to C127	
D	D0 to D1023	←	
R	R0 to R8191	←	Assignment changed.
A	A0, A1	_____	Discontinued (replaced by D and R registers).
Z	Z	Z0	Discontinued (replaced by Z1).
V	V	Z1	
N	N0 to N17	←	
P	P0 to P255	←	
K	K-32768 to K32767 K-2147483648 to K2147483647	←	
H	H0 to HFFFF H0 to HFFFFFFF	←	

**APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B
AND M600, M60/60S**

Appendix 2.2 Devices and Device Assignments

Table 2.2.2 List of device differences (M600L series)

M500 device No.			M600 device No.		Remarks
X	X0 to X4BF	->	X0 to X4BF	X	Input for 2nd part system Input for 3rd to 8th part system Input for 3rd and 4th spindle Input for 5th and 6th spindle
U	U0 to U17F	->	X4C0 to X63F		
I	I0 to I3FF	->	X640 to XA3F		
S	S0 to S3F	->	XA40 to XA7F		
S	S80 to SBF	->	XA80 to XABF		
Y	Y0 to Y53F	->	Y0 to Y53F	Y	Output for 2nd part system Output for 3rd to 8th part system Output for 3rd and 4th spindle Output for 5th and 6th spindle
W	W0 to W1FF	->	Y540 to Y73F		
J	J0 to J63F	->	Y740 to YD7F		
S	S40 to S7F	->	YD80 to YDBF		
S	SC0 to SFF	->	YDC0 to YDFF		
M	M0 to M5119	->	M0 to M5119	M	
G	G0 to G3071	->	M5120 to M8191		
F	F0 to F127	->	F0 to F127	F	
L	L0 to L255	->	L0 to L255	L	
E	E0 to E127	->	SM0 to SM127	SM	
T	T0 to T15	->	T0 to T15	T	10ms timer
Q	Q0 to Q39	->	T16 to T55		10ms timer (fixed timer)
T	T16 to T95	->	T56 to T135		100ms timer
Q	Q40 to Q135	->	T136 to T231		100ms timer (fixed timer)
T	T96 to T103	->	T232 to T239		100ms integrated timer
Q	Q136 to Q151	->	T240 to T255		100ms integrated timer (fixed timer)
C	C0 to C23	->	C0 to C23	C	
B	B0 to B103	->	C24 to C127		
D	D0 to D1023	->	D0 to D1023	D	
R	R0 to R8191	->	R0 to R8191	R	
A	A0, A1	->			Discontinued (replaced by D and R registers).
Z	Z	->	Z0	Z	
V	V	->	Z1		
N	N0 to N7	->	N0 to N7	N	
P	P0 to P255	->	P0 to P255	P	
K	K-32768 to K32767 K-2147483648 to K2147483647	->	K-32768 to K32767 K-2147483648 to K2147483647	K	
H	H0 to HFFFF H0 to HFFFFFFFF	->	H0 to HFFFF H0 to HFFFFFFFF	H	

**APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B
AND M600, M60/60S**

Appendix 2.2 Devices and Device Assignments

Table 2.2.3 List of device differences (M60 series)

PLC4B device No.			GX Developer device No.		Remarks
X	X0 to X4BF	->	X0 to X4BF	X	Input for 2nd part system
U	U0 to U17F	->	X4C0 to X63F		
Y	Y0 to Y53F	->	Y0 to Y53F	Y	Output for 2nd part system
W	W0 to W1FF	->	Y540 to Y73F		
M	M0 to M5119	->	M0 to M5119	M	
G	G0 to G3071	->	M5120 to M8191		
F	F0 to F127	->	F0 to F127	F	
L	L0 to L255	->	L0 to L255	L	
E	E0 to E127	->	SM0 to SM127	SM	
T	T0 to T15	->	T0 to T15	T	10ms timer
Q	Q0 to Q39	->	T16 to T55		10ms timer (fixed timer)
T	T16 to T95	->	T56 to T135		100ms timer
Q	Q40 to Q135	->	T136 to T231		100ms timer (fixed timer)
T	T96 to T103	->	T232 to T239		100ms integrated timer
Q	Q136 to Q151	->	T240 to T255		100ms integrated timer (fixed timer)
C	C0 to C23	->	C0 to C23	C	
B	B0 to B103	->	C24 to C127		
D	D0 to D1023	->	D0 to D1023	D	
R	R0 to R8191	->	R0 to R8191	R	
A	A0, A1	->			Discontinued (replaced by D and R registers).
Z	Z	->	Z0	Z	
V	V	->	Z1		
N	N0 to N7	->	N0 to N7	N	
P	P0 to P255	->	P0 to P255	P	
K	K-32768 to K32767 K-2147483648 to K2147483647	->	K-32768 to K32767 K-2147483648 to K2147483647	K	
H	H0 to HFFFF H0 to HFFFFFFFF	->	H0 to HFFFF H0 to HFFFFFFFF	H	

**APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B
AND M600, M60/60S**

Appendix 2.2 Devices and Device Assignments

Table 2.2.4 List of device differences (M60S series)

PLC4B device No.			GX Developer device No.		Remarks
X	X0 to X4BF	->	X0 to X4BF	X	Input for 2nd part system Input for 3rd to 8th part system Input for 3rd spindle Input for 4th spindle Input for 5th spindle Input for 6th spindle
U	U0 to U17F	->	X4C0 to X63F		
I	I0 to I3FF	->	X640 to XA3F		
S	S0 to S1F	->	XA40 to XA5F		
S	S40 to S5F	->	XA60 to XA7F		
S	S80 to S9F	->	XA80 to XA9F		
S	SC0 to SDF	->	XAA0 to XABF		
Y	Y0 to Y53F	->	Y0 to Y53F	Y	Output for 2nd part system Output for 3rd to 8th part system Output for 3rd spindle Output for 4th spindle Output for 5th spindle Output for 6th spindle
W	W0 to W1FF	->	Y540 to Y73F		
J	J0 to J63F	->	Y740 to YD7F		
S	S20 to S3F	->	YD80 to YD9F		
S	S60 to S7F	->	YDA0 to YDBF		
S	SA0 to SBF	->	YDC0 to YDDF		
S	SE0 to SFF	->	YDE0 to YDFF		
M	M0 to M5119	->	M0 to M5119	M	
G	G0 to G3071	->	M5120 to M8191		
F	F0 to F127	->	F0 to F127	F	
L	L0 to L255	->	L0 to L255	L	
E	E0 to E127	->	SM0 to SM127	SM	
T	T0 to T15	->	T0 to T15	T	10ms timer
Q	Q0 to Q39	->	T16 to T55		10ms timer (fixed timer)
T	T16 to T95	->	T56 to T135		100ms timer
Q	Q40 to Q135	->	T136 to T231		100ms timer (fixed timer)
T	T96 to T103	->	T232 to T239		100ms integrated timer
Q	Q136 to Q151	->	T240 to T255		100ms integrated timer (fixed timer)
C	C0 to C23	->	C0 to C23	C	
B	B0 to B103	->	C24 to C127		
D	D0 to D1023	->	D0 to D1023	D	
R	R0 to R8191	->	R0 to R8191	R	
A	A0, A1	->			Discontinued (replaced by D, R registers)
Z	Z	->	Z0	Z	
V	V	->	Z1		
N	N0 to N7	->	N0 to N7	N	
P	P0 to P255	->	P0 to P255	P	
K	K-32768 to K32767 K-2147483648 to K2147483647	->	K-32768 to K32767 K-2147483648 to K2147483647	K	
H	H0 to HFFFF H0 to HFFFFFFFF	->	H0 to HFFFF H0 to HFFFFFFFF	H	

**APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B
AND M600, M60/60S**

Appendix 2.3 PLC Instructions

Appendix 2.3 PLC Instructions

As for types and numbers of the PLC instructions, those of the PLC4B development environment and of the MELSEC PLC development environment are not different basically. However, some instructions have been changed because the user PLC development environment using the MELSEC PLC development tool has been supported in M600, M60/60S.

The instruction range that can be used in the MELSEC-QnA Series PLC program differs from the instruction range that can be used by the MELDAS Series. Because of this, some instructions that can be used by the M600, M60/60S cannot be handled with the GX Developer. There are also instructions that can be used by the GX Developer but cannot be used by the M600, M60/60S. When these are arranged, they are classified into the three following types.

- M600, M60/60S instructions that cannot be handled with the GX Developer
- M600, M60/60S instructions that the format differs from that of the GX Developer
- Instructions that can be used by the GX Developer, but cannot be used by the M600, M60/60S

Appendix 2.3.1 Instructions that cannot be handled with the GX Developer

Instructions that cannot be handled with the GX Developer are substituted with alternate instructions that can be handled with the GX Developer. Instructions that can be alternated are shown in "Table 2.3.1 Table of alternate instruction correspondence".

When some instructions described in "Table 2.3.1 Table of alternate instruction correspondence" are created with the GX Developer with the M600, M60/60S instruction sign left as is, an error results and creation cannot be carried out. Create the instructions using the GX Developer instruction sign described in the correspondence table.

When PLC programs containing alternate instructions are written from the GX Developer to the M600, M60/60S, they are rewritten to the original MELDAS Series instructions.

Table 2.3.1 Table of alternate instruction correspondence

Classification	Original MELDAS Series instruction		GX Developer instruction	
	Instruction sign	Symbol	Instruction sign	Symbol
Bit	DEFR	—[DEFR D]—	ANDP	$\begin{array}{c} D \\ \\ \uparrow \\ \end{array}$
Average value	AVE	—[AVE S D n]—	S.AVE	—[S.AVE S D n]—
Carry flag set	STC	—[STC]—	S.STC	—[S.STC]—
Carry flag reset	CLC	—[CLC]—	S.CLC	—[S.CLC]—
ATC	ATC	—[ATC Kn Rn Rm]-< Mm >—	S.ATC	—[S.ATC Kn Rn Rm Mm]—
ROT	ROT	—[ROT Kn Rn Rm]-< Mm >—	S.ROT	—[S.ROT Kn Rn Rm Mm]—
TSRH	TSRH	—[TSRH Rm Rn]-< Mn >—	S.TSRH	—[S.TSRH Rm Rn Mn]—
DDBA	DDBA	—[DDBA Rn / Dn]—	S.DDBA	—[S.DDBA Rn / Dn]—
DDBS	DDBS	—[DDBS Rn]—	S.DDBS	—[S.DDBS Rn]—
CAL1	CAL1	—[CAL1 Pn]—	S.CAL1	—[S.CAL1 Pn]—
BIT	LDBIT	[BIT S1 n] —	LD<=	[<= S1 n] —
	ANDBIT	— [BIT S1 n] —	AND<=	— [<= S1 n] —
	ORBIT	└ [BIT S1 n] ┘	OR<=	└ [<= S1 n] ┘
	LDBII	[BII S1 n] —	LD<>	[<> S1 n] —
	ANDBII	— [BII S1 n] —	AND<>	— [<> S1 n] —
	ORBII	└ [BII S1 n] ┘	OR<>	└ [<> S1 n] ┘

[Note] Among the above instructions, the ranges of the usable instructions depends on the model.

**APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B
AND M600, M60/60S**

Appendix 2.3 PLC Instructions

Appendix 2.3.2 Instructions with the format that differs from that of the GX Developer

The device types and assignments have been reconsidered as "Appendix 2.2 Devices and Device Assignments".

The format of conventional "instructions used by inputting or outputting an accumulator (A0, A1)" is changed so that the general word register can be designated with arguments without fixing by A0 or A1 input or output because of the discontinuance of accumulator (A0, A1).

Format-changed instructions are shown in "Table 2.3.2 Table of format-changed instruction correspondence".

When some instructions described in "Table 2.3.2 Table of format-changed instruction correspondence" are created with the GX Developer with the conventional instruction sign left as is, an error results and creation cannot be carried out. Create the instructions using the GX Developer instruction sign described in the correspondence table.

Table 2.3.2 Table of format-changed instruction correspondence

Classification	Original MELDAS Series instruction		Format-changed instruction		Remarks
	Instruction sign	Symbol	Instruction sign	Symbol	
Right rotation	ROR	—[ROR n]—	ROR	—[ROR D n]—	*1
	RCR	—[RCR n]—	RCR	—[RCR D n]—	
	DROR	—[DROR n]—	DROR	—[DROR D n]—	
	DRCR	—[DRCR n]—	DRCR	—[DRCR D n]—	
Left rotation	ROL	—[ROL n]—	ROL	—[ROL D n]—	
	RCL	—[RCL n]—	RCL	—[RCL D n]—	
	DROL	—[DROL n]—	DROL	—[DROL D n]—	
	DRCL	—[DRCL n]—	DRCL	—[DRCL D n]—	
Search	SER	—[SER S1 S2 n]—	SER	—[SER S1 S2 D n]—	*2
Quantity of 1	SUM	—[SUM S]—	SUM	—[SUM S D]—	*3

*1 : D is the head No. of the rotation device (word 16-bit device)

*2 : D is the head No. of the device that stores the search results (word bit device)

*3 : D is the head No. of the device that stores the total No. of bits (word bit device)

APPENDIX 2. DIFFERENCES IN DEVELOPMENT ENVIRONMENT BETWEEN PLC4B AND M600, M60/60S

Appendix 2.3 PLC Instructions

Appendix 2.3.3 Instructions that can be used with the GX Developer, but cannot be used by the M600, M60/60S

When instructions that cannot be used by the M600, M60/60S are written from the GX Developer to the M600, M60/60S, they are rewritten to "NOP" instructions, and if they are run, an alarm will occur. (Note that instructions described in "Table 2.3.1 Table of alternate instruction correspondence" are rewritten to the corresponding M600, M60/60S instructions.)

"Instructions that can be used by the GX Developer, but cannot be used in the M600, M60/60S " are defined by the following expression.

"Instructions that can be used by the GX Developer, but cannot be used in the M600, M60/60S"

= "All instructions described in the QnA Programming Instruction Manual"

- ("All instructions described in the M600, M60/60S PLC Programming Instruction Manual"

+ "Table 2.3.1 Table of alternate instruction correspondence"

+ "Table 2.3.2 Table of format-changed instruction correspondence")

Refer to "Appendix 3. MELSEC QnA Series Instructions Lists."

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.1 Sequence Instructions

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER

The following instruction lists are excerpts from the "QnACPU Programming Manual (Common Instructions)" (model name: QNACPU-P (KYOUTU) 13J522).

In these lists, the instructions "marked **×**" are unavailable for the M600, M60/60S series. (When written from GX Developer to the M600, M60/60S series, they are replaced by "NOP" instructions.)

Appendix 3.1 Sequence Instructions

Contact instructions

Classification	Instruction sign	Symbol
Contact	LD	
	LDI	
	AND	
	ANI	
	OR	
	ORI	
	LDP	
	LDF	
	ANDP	
	ANDF	
	ORP	
	ORF	

Output instructions

Classification	Instruction sign	Symbol
Output	OUT	
	SET	
	RST	
	PLS	
	PLF	
	FF	
	DELTA	
	DELTAP	

Coupling instructions

Classification	Instruction sign	Symbol
Coupling	ANB	
	ORB	
	MPS	
	MRD	
	MPP	
	INV	
	MEP	
	MEF	
	EGP	
	EGF	

Shift instructions

Classification	Instruction sign	Symbol
Shift	SFT	
	SFTP	

Master control instructions

Classification	Instruction sign	Symbol
Master control	MC	
	MCR	

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.2 Comparison Operation Instructions

End instructions

Classification	Instruction sign	Symbol
Program end	FEND	
	END	

Other instructions

Classification	Instruction sign	Symbol
Stop	STOP	
No operation	NOP	
	NOPLF	
	PAGE	

Appendix 3.2 Comparison Operation Instructions

Comparison operation instructions

Classification	Instruction sign	Symbol
16-bit data comparison	LD=	
	AND=	
	OR=	
	LD<>	
	AND<>	
	OR<>	
	LD>	
	AND>	
	OR>	
	LD<=	
	AND<=	
	OR<=	
	LD<	
	AND<	
	OR<	
	LD>=	
	AND>=	
	OR>=	

Comparison operation instructions (Continued)

Classification	Instruction sign	Symbol
32-bit data comparison	LDD=	
	ANDD=	
	ORD=	
	LDD<>	
	ANDD<>	
	ORD<>	
	LDD>	
	ANDD>	
	ORD>	
	LDD<=	
	ANDD<=	
	ORD<=	
	LDD<	
	ANDD<	
	ORD<	
	LDD>=	
	ANDD>=	
	ORD>=	

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.2 Comparison Operation Instructions

Comparison operation instructions (Continued)

Classification	Instruction sign	Symbol
Real number data comparison	LDE =	$\boxed{E = S1 S2}$
	ANDE =	$\boxed{E = S1 S2}$
	ORE =	$\boxed{E = S1 S2}$
	LDE <>	$\boxed{E <> S1 S2}$
	ANDE <>	$\boxed{E <> S1 S2}$
	ORE <>	$\boxed{E <> S1 S2}$
	LDE >	$\boxed{E > S1 S2}$
	ANDE >	$\boxed{E > S1 S2}$
	ORE >	$\boxed{E > S1 S2}$
	LDE <=	$\boxed{E <= S1 S2}$
	ANDE <=	$\boxed{E <= S1 S2}$
	ORE <=	$\boxed{E <= S1 S2}$
	LDE <	$\boxed{E < S1 S2}$
	ANDE <	$\boxed{E < S1 S2}$
	ORE <	$\boxed{E < S1 S2}$
	LDE >=	$\boxed{E >= S1 S2}$
ANDE >=	$\boxed{E >= S1 S2}$	
ORE >=	$\boxed{E >= S1 S2}$	

Comparison operation instructions (Continued)

Classification	Instruction sign	Symbol
Character string data comparison	LD\$ =	$\boxed{\$ = S1 S2}$
	AND\$ =	$\boxed{\$ = S1 S2}$
	OR\$ =	$\boxed{\$ = S1 S2}$
	LD\$ <>	$\boxed{\$ <> S1 S2}$
	AND\$ <>	$\boxed{\$ <> S1 S2}$
	OR\$ <>	$\boxed{\$ <> S1 S2}$
	LD\$ >	$\boxed{\$ > S1 S2}$
	AND\$ >	$\boxed{\$ > S1 S2}$
	OR\$ >	$\boxed{\$ > S1 S2}$
	LD\$ <=	$\boxed{\$ <= S1 S2}$
	AND\$ <=	$\boxed{\$ <= S1 S2}$
	OR\$ <=	$\boxed{\$ <= S1 S2}$
	LD\$ <	$\boxed{\$ < S1 S2}$
	AND\$ <	$\boxed{\$ < S1 S2}$
	OR\$ <	$\boxed{\$ < S1 S2}$
	LD\$ >=	$\boxed{\$ >= S1 S2}$
AND\$ >=	$\boxed{\$ >= S1 S2}$	
OR\$ >=	$\boxed{\$ >= S1 S2}$	
Block data comparison	BKCMP =	$\boxed{BKCMP = S1 S2 D n}$
	BKCMP <>	$\boxed{BKCMP <> S1 S2 D n}$
	BKCMP >	$\boxed{BKCMP > S1 S2 D n}$
	BKCMP <=	$\boxed{BKCMP <= S1 S2 D n}$
	BKCMP <	$\boxed{BKCMP < S1 S2 D n}$
	BKCMP >=	$\boxed{BKCMP >= S1 S2 D n}$
	BKCMP =P	$\boxed{BKCMP =P S1 S2 D n}$
	BKCMP <>P	$\boxed{BKCMP <>P S1 S2 D n}$
	BKCMP >P	$\boxed{BKCMP >P S1 S2 D n}$
	BKCMP <=P	$\boxed{BKCMP <=P S1 S2 D n}$
	BKCMP <P	$\boxed{BKCMP <P S1 S2 D n}$
	BKCMP >=P	$\boxed{BKCMP >=P S1 S2 D n}$

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.2 Comparison Operation Instructions

Arithmetic operation instructions

Classification	Instruction sign	Symbol
BIN 16-bit addition/ subtraction	+	— + S D
	+P	— +P S D
	+	— + S1 S2 D
	+P	— +P S1 S2 D
	—	— — S D
	-P	— -P S D
	—	— — S1 S2 D
	-P	— -P S1 S2 D
BIN 32-bit addition/ subtraction	D+	— D+ S D
	D+P	— D+P S D
	D+	— D+ S1 S2 D
	D+P	— D+P S1 S2 D
	D—	— D— S D
	D—P	— D—P S D
	D—	— D— S1 S2 D
	D—P	— D—P S1 S2 D
BIN 16-bit multipli- cation/ division	*	— * S1 S2 D
	*P	— *P S1 S2 D
	/	— / S1 S2 D
	/P	— /P S1 S2 D
BIN 32-bit multipli- cation/ division	D*	— D* S1 S2 D
	D*P	— D*P S1 S2 D
	D/	— D/ S1 S2 D
	D/P	— D/P S1 S2 D

Arithmetic operation instructions (Continued)

Classification	Instruction sign	Symbol
BCD 4-digit addition/ subtraction	B+	— B+ S D
	B+P	— B+P S D
	B+	— B+ S1 S2 D
	B+P	— B+P S1 S2 D
	B—	— B— S D
	B—P	— B—P S D
	B—	— B— S1 S2 D
	B—P	— B—P S1 S2 D
BCD 8-digit addition/ subtraction	DB+	— DB+ S D
	DB+P	— DB+P S D
	DB+	— DB+ S1 S2 D
	DB+P	— DB+P S1 S2 D
	DB—	— DB— S D
	DB—P	— DB—P S D
	DB—	— DB— S1 S2 D
	DB—P	— DB—P S1 S2 D
BCD 4-digit multipli- cation/ division	B*	— B* S1 S2 D
	B*P	— B*P S1 S2 D
	B/	— B/ S1 S2 D
	B/P	— B/P S1 S2 D
BCD 8-digit multipli- cation/ division	DB*	— DB* S1 S2 D
	DB*P	— DB*P S1 S2 D
	DB/	— DB/ S1 S2 D
	DB/P	— DB/P S1 S2 D

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.2 Comparison Operation Instructions

Arithmetic operation instructions (Continued)

Classification	Instruction sign	Symbol
Floating-point data addition/subtraction	E+	$\overline{\text{E+}} \quad \text{S} \quad \text{D}$
	E+P	$\overline{\text{E+P}} \quad \text{S} \quad \text{D}$
	E+	$\overline{\text{E+}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	E+P	$\overline{\text{E+P}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	E-	$\overline{\text{E-}} \quad \text{S} \quad \text{D}$
	E-P	$\overline{\text{E-P}} \quad \text{S} \quad \text{D}$
	E-	$\overline{\text{E-}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	E-P	$\overline{\text{E-P}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
Floating-point data multiplication/division	E*	$\overline{\text{E*}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	E*P	$\overline{\text{E*P}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	E/	$\overline{\text{E/}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	E/P	$\overline{\text{E/P}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
BIN block addition/subtraction	BK+	$\overline{\text{BK+}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
	BK+P	$\overline{\text{BK+P}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
	BK-	$\overline{\text{BK-}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
	BK-P	$\overline{\text{BK-P}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
Character string data coupling	\$+	$\overline{\text{\$+}} \quad \text{S} \quad \text{D}$
	\$+P	$\overline{\text{\$+P}} \quad \text{S} \quad \text{D}$
	\$+	$\overline{\text{\$+}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	\$+P	$\overline{\text{\$+P}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
BIN data increment	INC	$\overline{\text{INC}} \quad \text{D}$
	INCP	$\overline{\text{INCP}} \quad \text{D}$
	DINC	$\overline{\text{DINC}} \quad \text{D}$
	DINCP	$\overline{\text{DINCP}} \quad \text{D}$
	DEC	$\overline{\text{DEC}} \quad \text{D}$
	DECP	$\overline{\text{DECP}} \quad \text{D}$
	DDEC	$\overline{\text{DDEC}} \quad \text{D}$
	DDECP	$\overline{\text{DDECP}} \quad \text{D}$

Data conversion instructions

Classification	Instruction sign	Symbol
BCD conversion	BCD	$\overline{\text{BCD}} \quad \text{S} \quad \text{D}$
	BCDP	$\overline{\text{BCDP}} \quad \text{S} \quad \text{D}$
	DBCDC	$\overline{\text{DBCDC}} \quad \text{S} \quad \text{D}$
	DBCDCP	$\overline{\text{DBCDCP}} \quad \text{S} \quad \text{D}$

Data conversion instructions (Continued)

Classification	Instruction sign	Symbol
BIN conversion	BIN	$\overline{\text{BIN}} \quad \text{S} \quad \text{D}$
	BINP	$\overline{\text{BINP}} \quad \text{S} \quad \text{D}$
	DBIN	$\overline{\text{DBIN}} \quad \text{S} \quad \text{D}$
	DBINP	$\overline{\text{DBINP}} \quad \text{S} \quad \text{D}$
BIN -> floating-point conversion	FLT	$\overline{\text{FLT}} \quad \text{S} \quad \text{D}$
	FLTP	$\overline{\text{FLTP}} \quad \text{S} \quad \text{D}$
	DFLT	$\overline{\text{DFLT}} \quad \text{S} \quad \text{D}$
	DFLTP	$\overline{\text{DFLTP}} \quad \text{S} \quad \text{D}$
Floating-point -> BIN conversion	INT	$\overline{\text{INT}} \quad \text{S} \quad \text{D}$
	INTP	$\overline{\text{INTP}} \quad \text{S} \quad \text{D}$
	DINT	$\overline{\text{DINT}} \quad \text{S} \quad \text{D}$
	DINTP	$\overline{\text{DINTP}} \quad \text{S} \quad \text{D}$
BIN 16-bit <-> 32-bit conversion	DBL	$\overline{\text{DBL}} \quad \text{S} \quad \text{D}$
	DBLP	$\overline{\text{DBLP}} \quad \text{S} \quad \text{D}$
	WORD	$\overline{\text{WORD}} \quad \text{S} \quad \text{D}$
	WORDP	$\overline{\text{WORDP}} \quad \text{S} \quad \text{D}$
BIN -> gray code conversion	GRY	$\overline{\text{GRY}} \quad \text{S} \quad \text{D}$
	GRYP	$\overline{\text{GRYP}} \quad \text{S} \quad \text{D}$
	DGRY	$\overline{\text{DGRY}} \quad \text{S} \quad \text{D}$
	DGRYP	$\overline{\text{DGRYP}} \quad \text{S} \quad \text{D}$
Gray code -> BIN conversion	GBIN	$\overline{\text{GBIN}} \quad \text{S} \quad \text{D}$
	GBINP	$\overline{\text{GBINP}} \quad \text{S} \quad \text{D}$
	DGBIN	$\overline{\text{DGBIN}} \quad \text{S} \quad \text{D}$
	DGBINP	$\overline{\text{DGBINP}} \quad \text{S} \quad \text{D}$
2's complement	NEG	$\overline{\text{NEG}} \quad \text{D}$
	NEGP	$\overline{\text{NEGP}} \quad \text{D}$
	DNEG	$\overline{\text{DNEG}} \quad \text{D}$
	DNEGP	$\overline{\text{DNEGP}} \quad \text{D}$
	ENEG	$\overline{\text{ENEG}} \quad \text{D}$
	ENECP	$\overline{\text{ENECP}} \quad \text{D}$
Block conversion	BKBCD	$\overline{\text{BKBCD}} \quad \text{S} \quad \text{D} \quad \text{n}$
	BKBCDP	$\overline{\text{BKBCDP}} \quad \text{S} \quad \text{D} \quad \text{n}$
	BKBIN	$\overline{\text{BKBIN}} \quad \text{S} \quad \text{D} \quad \text{n}$
	BKBINP	$\overline{\text{BKBINP}} \quad \text{S} \quad \text{D} \quad \text{n}$

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.2 Comparison Operation Instructions

Data transfer instructions

Classification	Instruction sign	Symbol
16-bit data transfer	MOV	— MOV S D
	MOV	— MOV S D
32-bit data transfer	DMOV	— DMOV S D
	DMOV	— DMOV S D
Floating-point data transfer	EMOV	— EMOV S D
	EMOV	— EMOV S D
Character string data transfer	\$MOV	— \$MOV S D
	\$MOV	— \$MOV S D
16-bit data NOT transfer	CML	— CML S D
	CML	— CML S D
32-bit data NOT transfer	DCML	— DCML S D
	DCML	— DCML S D
Block transfer	BMOV	— BMOV S D n
	BMOV	— BMOV S D n
Same data block transfer	FMOV	— FMOV S D n
	FMOV	— FMOV S D n
16-bit data change	XCH	— XCH S D
	XCH	— XCH S D
32-bit data change	DXCH	— DXCH S D
	DXCH	— DXCH S D
Block data change	BXCH	— BXCH S D n
	BXCH	— BXCH S D n
Upper/lower byte change	SWAP	— SWAP D
	SWAP	— SWAP D

Program branch instructions

Classification	Instruction sign	Symbol
Jump	CJ	— CJ Pn
	CJ	— CJ Pn
	SCJ	— SCJ Pn
	SCJ	— SCJ Pn
	JMP	— JMP Pn
	JMP	— JMP Pn
	GOEND	— GOEND

Program execution control instructions

Classification	Instruction sign	Symbol
Interrupt disable	DI	— DI
Interrupt enable	EI	— EI
Interrupt disable/enable setting	IMASK	— IMASK S
Return	IRET	— IRET

I/O refresh instructions

Classification	Instruction sign	Symbol
I/O refresh	RFS	— RFS D n

Other useful instructions

Classification	Instruction sign	Symbol
Up/down counter	UDCNT1	— UDCNT1 S D n
	UDCNT1	— UDCNT1 S D n
	UDCNT2	— UDCNT2 S D n
	UDCNT2	— UDCNT2 S D n
Teaching timer	TTMR	— TTMR D n
Special timer	STMR	— STMR S n D
Nearest access control	ROTC	— ROTC S n1 n2 D
Ramp signal	RAMP	— RAMP n1 n2 D1 n3 D2
Pulse density	SPD	— SPD S n D
Pulse output	PLSY	— PLSY n1 n2 D
Pulse width modulation	PWM	— PWM n1 n2 D
Matrix input	MTR	— MTR S D1 D2 n

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.3 Application Instructions

Appendix 3.3 Application Instructions

Logical operation instructions

Classification	Instruction sign	Symbol
AND	WAND	— WAND S D —
	WANDP	— WANDP S D —
	WAND	— WAND S1 S2 D —
	WANDP	— WANDP S1 S2 D —
	DAND	— DAND S D —
	DANDP	— DANDP S D —
	DAND	— DAND S1 S2 D —
	DANDP	— DANDP S1 S2 D —
	BKAND	— BKAND S1 S2 D n —
	BKANDP	— BKANDP S1 S2 D n —
OR	WOR	— WOR S D —
	WORP	— WORP S D —
	WOR	— WOR S1 S2 D —
	WORP	— WORP S1 S2 D —
	DOR	— DOR S D —
	DORP	— DORP S D —
	DOR	— DOR S1 S2 D —
	DORP	— DORP S1 S2 D —
	BKOR	— BKOR S1 S2 D n —
	BKORP	— BKORP S1 S2 D n —
Exclusive OR	WXOR	— WXOR S D —
	WXORP	— WXORP S D —
	WXOR	— WXOR S1 S2 D —
	WXORP	— WXORP S1 S2 D —
	DXOR	— DXOR S D —
	DXORP	— DXORP S D —
	DXOR	— DXOR S1 S2 D —
	DXORP	— DXORP S1 S2 D —
	BKXOR	— BKXOR S1 S2 D n —
	BKXORP	— BKXORP S1 S2 D n —

Logical operation instructions (Continued)

Classification	Instruction sign	Symbol
NOT-exclusive OR	WXNR	— WXNR S D —
	WXNRP	— WXNRP S D —
	WXNR	— WXNR S1 S2 D —
	WXNRP	— WXNRP S1 S2 D —
	DXNR	— DXNR S D —
	DXNRP	— DXNRP S D —
	DXNR	— DXNR S1 S2 D —
	DXNRP	— DXNRP S1 S2 D —
	BKXNOR	— BKXNOR S1 S2 D n —
	BKXNORP	— BKXNORP S1 S2 D n —

Rotation instructions

Classification	Instruction sign	Symbol
Right rotation	ROR	— ROR D n —
	RORP	— RORP D n —
	RCR	— RCR D n —
	RCRP	— RCRP D n —
Left rotation	ROL	— ROL D n —
	ROLP	— ROLP D n —
	RCL	— RCL D n —
	RCLP	— RCLP D n —
Right rotation	DROR	— DROR D n —
	DRORP	— DRORP D n —
	DRCR	— DRCR D n —
	DRCRP	— DRCRP D n —
Left rotation	DROL	— DROL D n —
	DROLP	— DROLP D n —
	DRCL	— DRCL D n —
	DRCLP	— DRCLP D n —

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.3 Application Instructions

Shift instructions

Classification	Instruction sign	Symbol
n-bit shift	SFR	$\overline{\text{SFR}} \quad \text{D} \quad \text{n}$
	SFRP	$\overline{\text{SFRP}} \quad \text{D} \quad \text{n}$
	SFL	$\overline{\text{SFL}} \quad \text{D} \quad \text{n}$
	SFLP	$\overline{\text{SFLP}} \quad \text{D} \quad \text{n}$
1-bit shift	BSFR	$\overline{\text{BSFR}} \quad \text{D} \quad \text{n}$
	BSFRP	$\overline{\text{BSFRP}} \quad \text{D} \quad \text{n}$
	BSFL	$\overline{\text{BSFL}} \quad \text{D} \quad \text{n}$
	BSFLP	$\overline{\text{BSFLP}} \quad \text{D} \quad \text{n}$
1-word shift	DSFR	$\overline{\text{DSFR}} \quad \text{D} \quad \text{n}$
	DSFRP	$\overline{\text{DSFRP}} \quad \text{D} \quad \text{n}$
	DSFL	$\overline{\text{DSFL}} \quad \text{D} \quad \text{n}$
	DSFLP	$\overline{\text{DSFLP}} \quad \text{D} \quad \text{n}$

Bit processing instructions

Classification	Instruction sign	Symbol
Bit set/reset	BSET	$\overline{\text{BSET}} \quad \text{D} \quad \text{n}$
	BSETP	$\overline{\text{BSETP}} \quad \text{D} \quad \text{n}$
	BRST	$\overline{\text{BRST}} \quad \text{D} \quad \text{n}$
	BRSTP	$\overline{\text{BRSTP}} \quad \text{D} \quad \text{n}$
Bit test	TEST	$\overline{\text{TEST}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	TESTP	$\overline{\text{TESTP}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	DTEST	$\overline{\text{DTEST}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
	DTESTP	$\overline{\text{DTESTP}} \quad \text{S1} \quad \text{S2} \quad \text{D}$
Bit device Batch reset	BKRST	$\overline{\text{BKRST}} \quad \text{S} \quad \text{n}$
	BKRSTP	$\overline{\text{BKRSTP}} \quad \text{S} \quad \text{n}$

Data processing instructions

Classification	Instruction sign	Symbol
Data search	SER	$\overline{\text{SER}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
	SERP	$\overline{\text{SERP}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
	DSER	$\overline{\text{DSER}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
	DSERP	$\overline{\text{DSERP}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
	DSERP	$\overline{\text{DSERP}} \quad \text{S1} \quad \text{S2} \quad \text{D} \quad \text{n}$
Bit check	SUM	$\overline{\text{SUM}} \quad \text{S} \quad \text{D}$
	SUMP	$\overline{\text{SUMP}} \quad \text{S} \quad \text{D}$
	DSUM	$\overline{\text{DSUM}} \quad \text{S} \quad \text{D}$
	DSUMP	$\overline{\text{DSUMP}} \quad \text{S} \quad \text{D}$
Decode	DECO	$\overline{\text{DECO}} \quad \text{S} \quad \text{D} \quad \text{n}$
	DECOP	$\overline{\text{DECOP}} \quad \text{S} \quad \text{D} \quad \text{n}$
Encode	ENCO	$\overline{\text{ENCO}} \quad \text{S} \quad \text{D} \quad \text{n}$
	ENCOP	$\overline{\text{ENCOP}} \quad \text{S} \quad \text{D} \quad \text{n}$
7-segment decode	SEG	$\overline{\text{SEG}} \quad \text{S} \quad \text{D}$
	SEGP	$\overline{\text{SEGP}} \quad \text{S} \quad \text{D}$
Dissociation • Association	DIS	$\overline{\text{DIS}} \quad \text{S} \quad \text{D} \quad \text{n}$
	DISP	$\overline{\text{DISP}} \quad \text{S} \quad \text{D} \quad \text{n}$
	UNI	$\overline{\text{UNI}} \quad \text{S} \quad \text{D} \quad \text{n}$
	UNIP	$\overline{\text{UNIP}} \quad \text{S} \quad \text{D} \quad \text{n}$
	NDIS	$\overline{\text{NDIS}} \quad \text{S1} \quad \text{D} \quad \text{S2}$
	NDISP	$\overline{\text{NDISP}} \quad \text{S1} \quad \text{D} \quad \text{S2}$
	NUNI	$\overline{\text{NUNI}} \quad \text{S1} \quad \text{D} \quad \text{S2}$
	NUNIP	$\overline{\text{NUNIP}} \quad \text{S1} \quad \text{D} \quad \text{S2}$
	WTOB	$\overline{\text{WTOB}} \quad \text{S} \quad \text{D} \quad \text{n}$
	WTOBP	$\overline{\text{WTOBP}} \quad \text{S} \quad \text{D} \quad \text{n}$
	BTOW	$\overline{\text{BTOW}} \quad \text{S} \quad \text{D} \quad \text{n}$
	BTOWP	$\overline{\text{BTOWP}} \quad \text{S} \quad \text{D} \quad \text{n}$

APPENDIX 3. LIST OF INSTRUCTIONS USABLE WITH GX DEVELOPER
Appendix 3.3 Application Instructions

Data processing instructions (Continued)

Classification	Instruction sign	Symbol
Retrieval	MAX	— MAX S D n —
	MAXP	— MAXP S D n —
	MIN	— MIN S D n —
	MINP	— MINP S D n —
	DMAX	— DMAX S D n —
	DMAXP	— DMAXP S D n —
	DMIN	— DMIN S D n —
	DMINP	— DMINP S D n —
Sort	SORT	— SORT S1 n S2 D1 D2 — S2: Number of data blocks to be compared at a time. D1: Device to be forced ON at sort completion D2: Used by system
	DSORT	— DSORT S1 n S2 D1 D2 — S2: Number of data blocks to be compared at a time. D1: Device to be forced ON at sort completion D2: Used by system
Total value calculation	WSUM	— WSUM S D n —
	WSUMP	— WSUMP S D n —
	DWSUM	— DWSUM S D n —
	DWSUMP	— DWSUMP S D n —

Table operation instructions

Classification	Instruction sign	Symbol
Table processing	FIFW	— FIFW S D —
	FIFWP	— FIFWP S D —
	FIFR	— FIFR S D —
	FIFRP	— FIFRP S D —
	FPOP	— FPOP S D —
	FPOPP	— FPOPP S D —
	FINS	— FINS S D n —
	FINSP	— FINSP S D n —
	FDEL	— FDEL S D n —
	FDELP	— FDELP S D n —

Structuring instructions

Classification	Instruction sign	Symbol
Repeat	FOR	— FOR n —
	NEXT	— NEXT —
	BREAK	— BREAK D Pn —
	BREAKP	— BREAKP D Pn —
	Sub-routine program call	CALL
CALLP		— CALLP Pn S1~Sn —
RET		— RET —
FCALL		— FCALL Pn S1~Sn —
FCALLP		— FCALLP Pn S1~Sn —
ECALL		— SORT * Pn S1~Sn — * : Program name
ECALLP		— ECALLP * Pn S1~Sn — * : Program name
EFCALL		— EFCALL * Pn S1~Sn — * : Program name
EFCALLP		— EFCALLP * Pn S1~Sn — * : Program name
COM		— COM —
Fixed index qualification	IX	— IX S — Device qualification circuit
	IXEND	— IXEND —
	IXDEV	— IXDEV —
	IXSET	— IXSET Pn D — Designation of qualification value

Revision History

Date of revision	Manual No.	Revision details
Aug. 2002	BNP-B2252B	First edition created.
Sept. 2004	BMP-B2252C	<ul style="list-style-type: none">• Order of item configuration changed.• Matters related to program development added to "Safety Precautions".• Section "1. PLC Development Environment" totally reviewed.• Mistakes, etc., corrected.

Notice

Every effort has been made to keep up with software and hardware revisions in the contents described in this manual. However, please understand that in some unavoidable cases simultaneous revision is not possible.

Please contact your Mitsubishi Electric dealer with any questions or comments regarding the use of this product.

Duplication Prohibited

This manual may not be reproduced in any form, in part or in whole, without written permission from Mitsubishi Electric Corporation.

©2002-2004 MITSUBISHI ELECTRIC CORPORATION
ALL RIGHTS RESERVED



MODEL	M600 60/60S Series
MODEL CODE	008-244
Manual No.	BNP-B2252C(ENG)