



PLC

Engineering Software

MELSOFT GX Works3 (Ladder)

This course explains about basic functions of GX Works3 for those who will use GX Works3 for the first time. Operation methods of GX Works3 are introduced as the learner configures an example programmable controller system through this course. In this course, programs are configured using the programming language called Ladder.

Introduction Purpose of the Course

This course explains about basic functions of GX Works3 for those who will use GX Works3 for the first time. Operation methods of GX Works3 are introduced as the learner configures an example programmable controller system through this course.

In this course, programs are configured using the programming language called Ladder.

This course requires the basic knowledge of programmable controllers and the MELSEC Series programmable controllers.

The following courses are prerequisites prior to taking this course:

- FA Equipment for Beginners (PLCs)
- MELSEC iQ-R Series Basic

Introduction Course structure



The contents of this course are as follows.
It is recommended that you start from Chapter 1.

Chapter 1 - GX Works3 overview

Learn the fundamentals of GX Works3

Chapter 2 - System design

Learn about designing a programmable controller system

Chapter 3 - Program editing

Learn about creating control programs

Chapter 4 - Operation check

Learn about operation checks on created programs

Chapter 5 - Maintenance

Learn about maintenance after the system is in operation

Final Test

Pass grade: 60% or higher

Introduction**How to Use This e-Learning Tool**

Go to the next page		Go to the next page.
Back to the previous page		Back to the previous page.
Move to the desired page		"Table of Contents" will be displayed, enabling you to navigate to the desired page.
Exit the learning		Exit the learning.

Safety precautions

When you learn based on using actual products, please carefully read the safety precautions in the corresponding manuals.

Precautions in this course

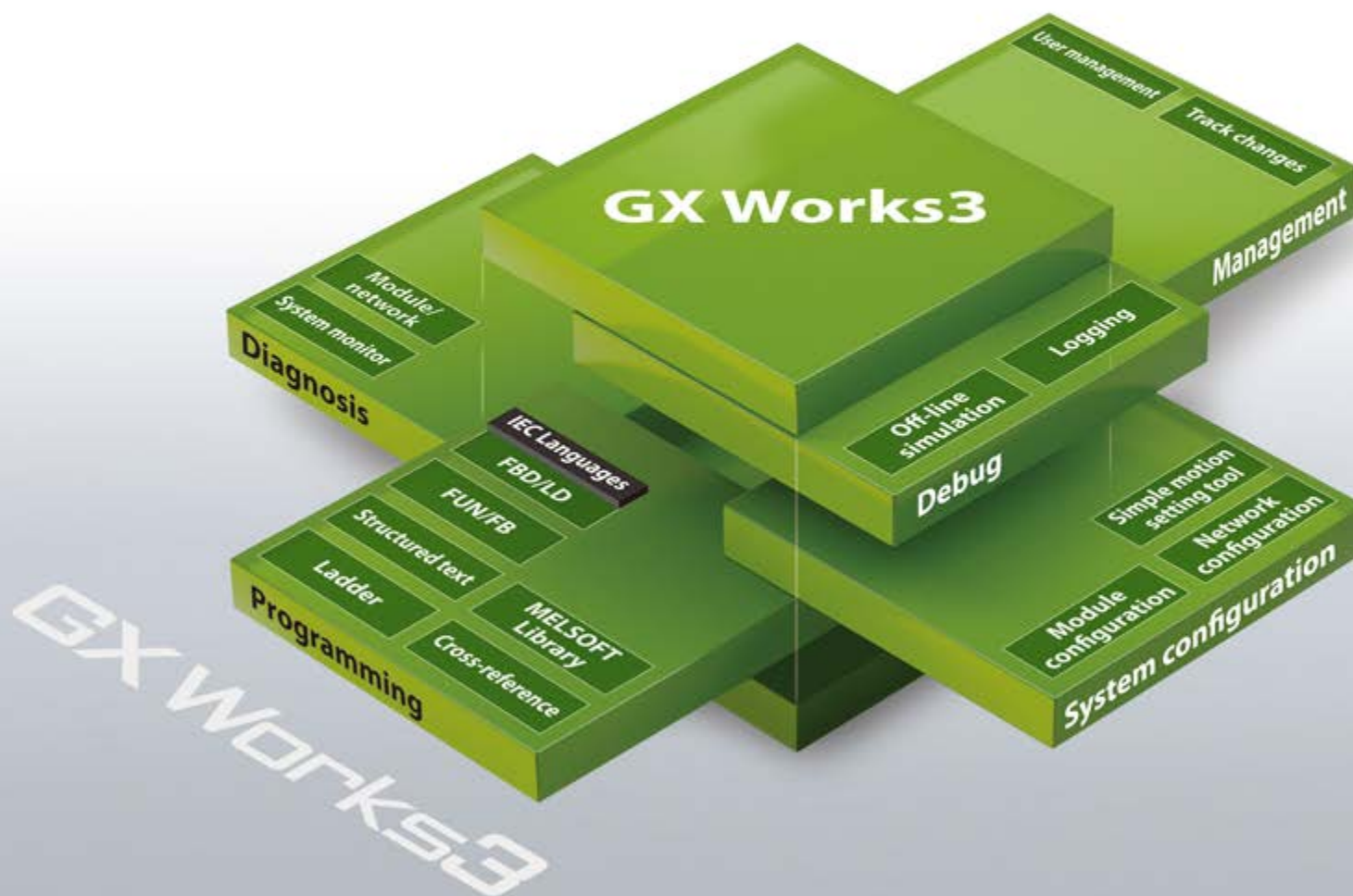
The displayed screens of the software version that you use may differ from those in this course.

This course uses the following software version:

- GX Works3 Version 1.007H

Chapter 1 GX Works3 overview

GX Works3 is the programming and maintenance software specifically designed for the MELSEC iQ-R Series control system. GX Works3 consists of various different components that help to simplify project creation and maintenance tasks.



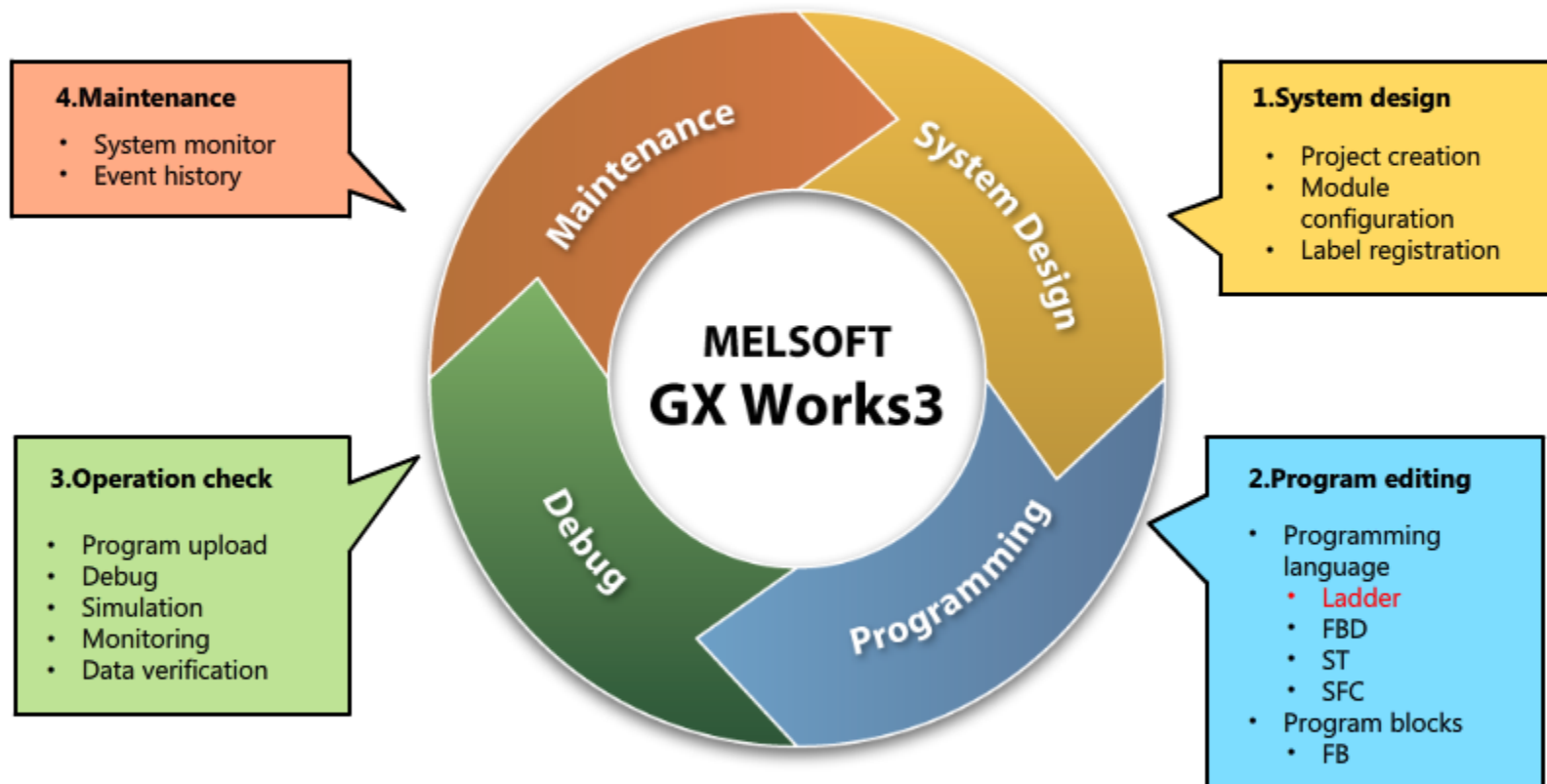
1.1

Scenes for GX Works3

The figure below shows typical lifecycle of a programmable controller system. GX Works3 can be used in all the scenes as shown below.

This course will introduce GX Works3 features in this order.

In this course, programs are configured using the programming language called Ladder.



1.2**Summary**

In this chapter, you have learned:

- GX Works3 overview

Important points to consider:

GX Works3 overview

GX Works3 consists of various different components that help to simplify project creation and maintenance tasks.

Chapter 2 System design

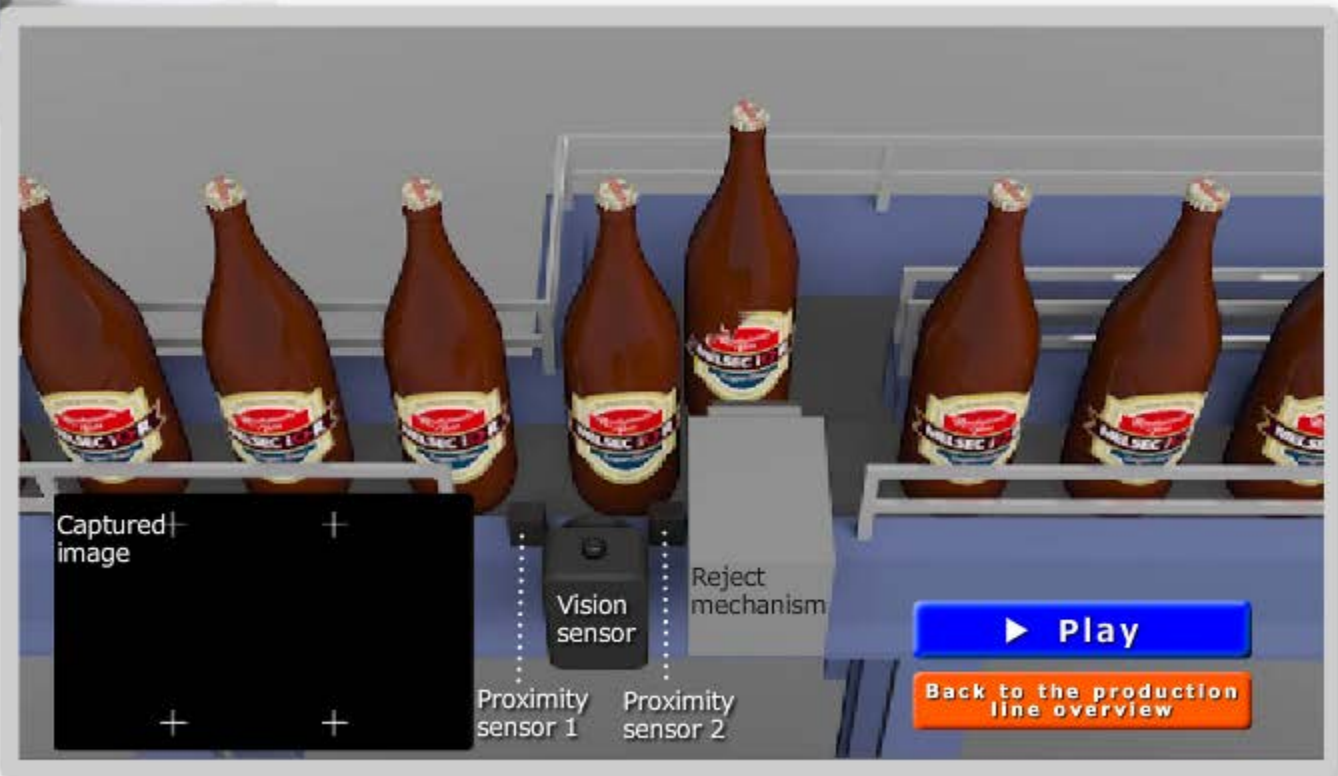


This chapter explains about designing a programmable controller system.



- 2.1 Programmable controller system example
- 2.2 Components for the example system
- 2.3 Main features of GX Works3
- 2.4 Creating a project
- 2.5 Module configuration according to the system
- 2.6 Setting module operations
- 2.7 Giving names to devices
- 2.8 Saving the created content
- 2.9 Summary

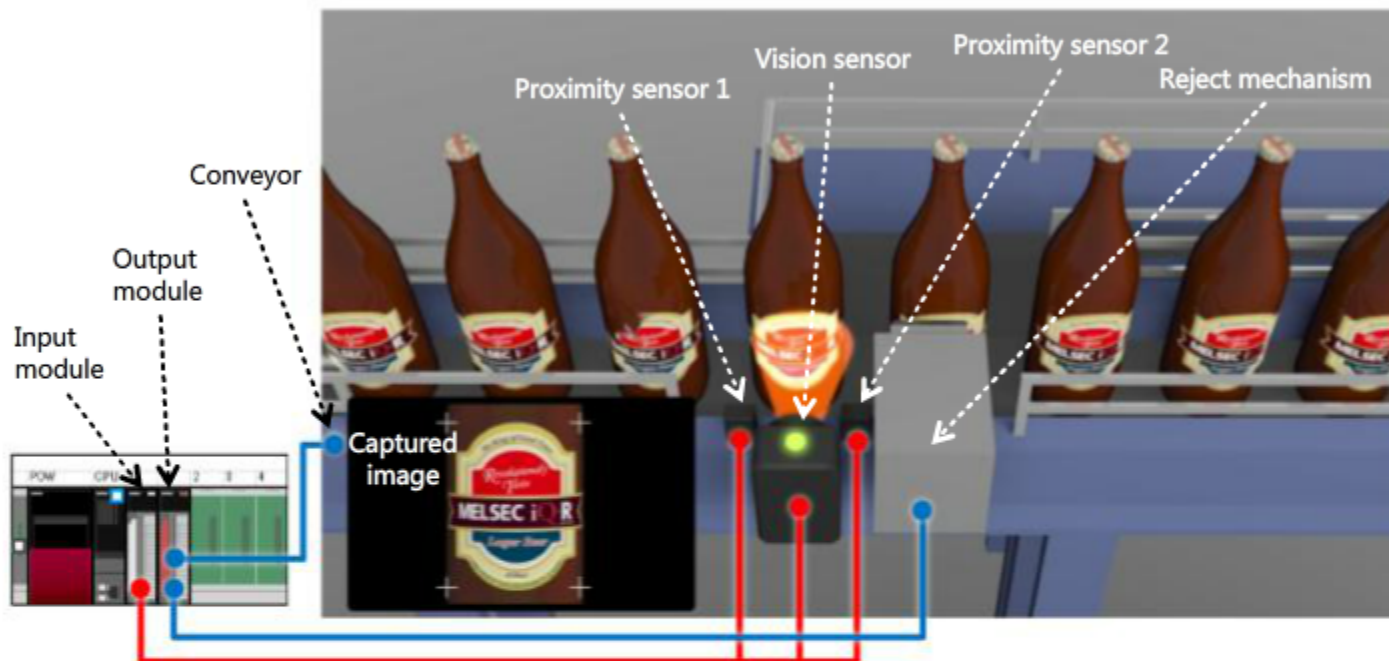
2.1 Programmable controller system example



2.2

Components for the example system

The example label inspection system requires the following components.



Programmable controller	Model
CPU module	R04CPU
Base unit	R35B
Power supply module	R61P
Input module	RX40C7
Output module	RY10R2

External equipment	Details
Proximity sensors 1, 2	Detects bottle positions.
Vision sensor	Checks if a label is correctly pasted on a bottle.
Reject mechanism	Pushes out a bottle with a defective label.
Conveyor	Conveys bottles to the sensors and the reject mechanism.

2.3

Main features of GX Works3

Screen layout of GX Works3 should be understood before designing a system.
Place the mouse cursor over a window or an area to learn about its functions.

Element Selection window

Elements available for programming are listed. Elements can be dragged and dropped onto the Work window. A required element can be searched, and often used elements can be added to Favorites.

The screenshot shows the MELSOFT GX Works3 interface. The main window displays a ladder logic editor with a table for writing instructions. The table has columns for 'Write', '1', '2', '3', '4', '5', '6', '7', '8', and '9'. The first row contains the instruction '(0)'. A green callout box highlights the 'Element Selection' window on the right, which lists various programming elements like Contact instructions, Association instructions, Output instructions, Shift instructions, Master Control instructions, Termination instructions, Stop instructions, Ignored instructions, BASIC INSTRUCTIONS, Comparison Operators, Arithmetic Operators, Data transfer instructions, Logical Operations, and Data shift instructions. The interface also includes a menu bar, a toolbar, a navigation pane on the left, and a status bar at the bottom.

2.3

Main features of GX Works3

MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation 0010:RY10R2 Module Para... x

Project

- Module Configuration
- Program
- FB/FUN
- Label
- Device
- Parameter
 - System Parameter
 - R04CPU
 - Module Information
 - 0000:RX40C7
 - 0010:RY10R2
 - Module Parameter
 - Module POU (Short)
 - Remote Password

Setting Item List

Input the Setting Item to Search

Setting of error-time output mode

Refresh Setting


Setting Item

Item	Setting Value
Setting of error-time output mode	
Y00	Clear
Y01	Clear
Y02	Clear
Y03	Clear
Y04	Clear
Y05	Clear
Y06	Clear
Y07	Clear
Y08	Clear
Y09	Clear
Y0A	Clear
Y0B	Clear
Y0C	Clear

Explanation

Item List Find Result

Check

Click  to proceed to the next section.
To view again, click on the "Replay" button.

Replay

R04 Host CAP NUM

2.4

Creating a project

MELSOFT GX Works3 (Untitled Project) - [ProgPou [PRG] [LD] 2Step]

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help


Navigation ProgPou [PRG] [Local Label ...] ProgPou [PRG] [LD] 2Step x

Write	1	2	3	4	5	6	7	8	9
1	(0)								

Output

Output Result of Power Supply Capacity and... Device Assign

R04 Host 0/2 Step Overwrite CPU NUM

Click  to proceed to the next section.
To view again, click on the "Replay" button.


Replay

2.5

Module configuration according to the system

The screenshot shows the MELSOFT GX Works3 interface for module configuration. The main workspace displays a rack configuration with slots labeled POW, CPU 0, 1, 2, 3, and 4. Slot 1 is highlighted with a green box. The right sidebar shows the 'Element Selection' panel with a list of modules, including RY10R2 (16 points) and RY40NT5I (16 points). A green callout box is overlaid on the bottom right, containing text and a 'Replay' button.

The next section explains how to automatically read an existing system configuration directly from the hardware.

Click  to proceed to the next section.

To view again, click on the "Replay" button.

Replay

2.5.1

Reading the actual module configuration



MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation: ProgPou [PRG] [Local Label ...] ProgPou [PRG] [LD] 2Step Module Configuration * x

Element Selection: (Find POU) Display Target: All iQ-R Series Main Base Extension Base RQ Extension Base

POU... Fav... His... Mo...

Input the Configuration ...

Output: Read Module Configuration from PLC Error: 0 Warning: 0

No.	Result	Occurrence Site	Occurrence Position	Explanation

Output: Result of Power Supply Capacity and... Device Assignment

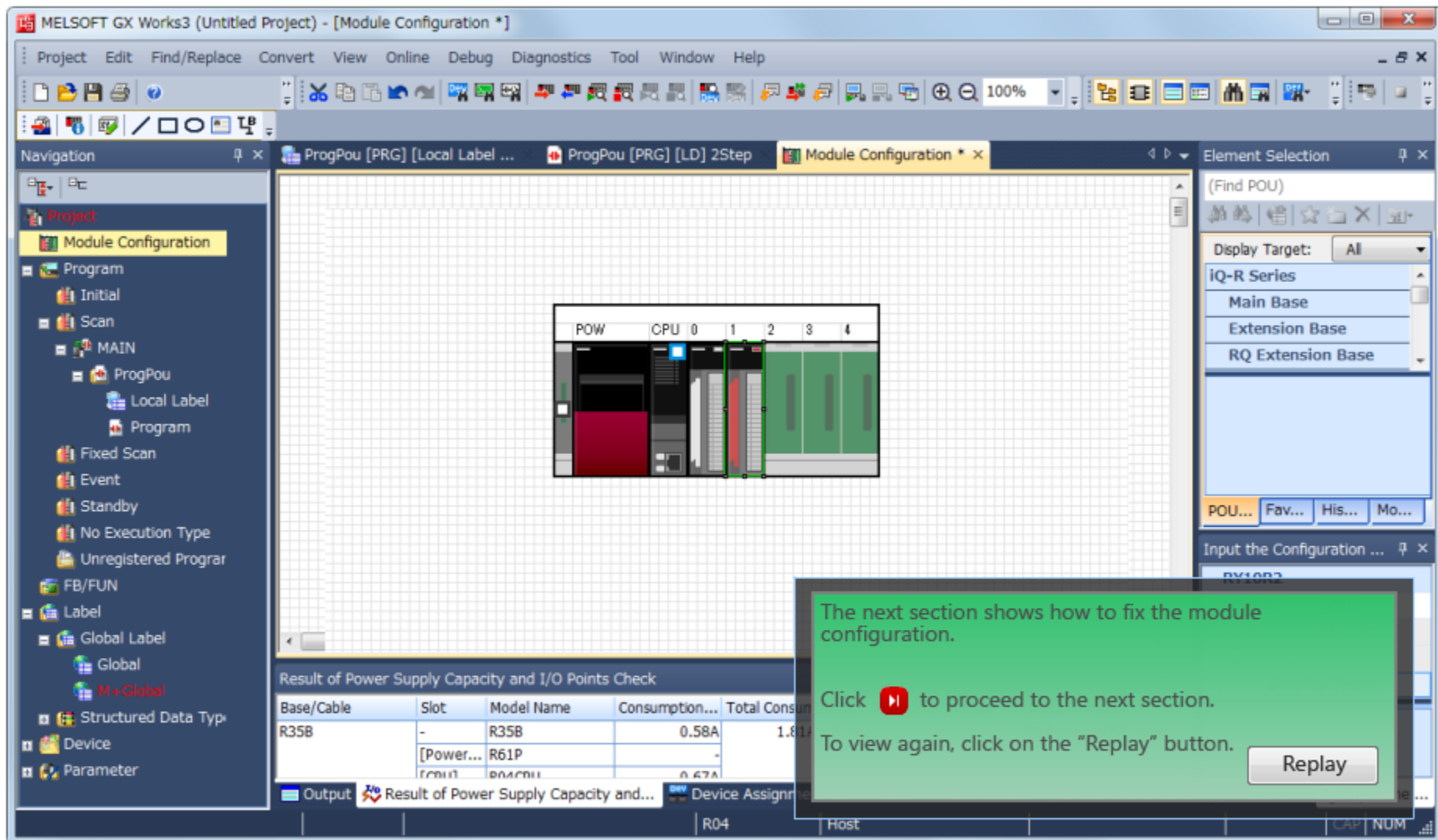
R04 Host CAP NUM

Click to proceed to the next section. To view again, click on the "Replay" button.

Replay

2.5.2


Checks after module configuration



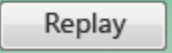
The screenshot shows the MELSOFT GX Works3 software interface. The main window displays a rack configuration with slots labeled POW, CPU 0, 1, 2, 3, and 4. A table at the bottom shows the 'Result of Power Supply Capacity and I/O Points Check'.

Base/Cable	Slot	Model Name	Consumption...	Total Consum
R35B	-	R35B	0.58A	1.81
[Power...	R61P		-	
[CPU]	[Power...		0.67A	

The next section shows how to fix the module configuration.

Click  to proceed to the next section.

To view again, click on the "Replay" button.



2.5.3

Fixing the module configuration



MELSOFT GX Works3 (Untitled Project) - [Module Configuration]

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation ProgPou [PRG] [Local Label ...] ProgPou [PRG] [LD] 2Step Module Configuration x

Element Selection (Find POU)

Display Target: All

iQ-R Series

- Main Base
- Extension Base
- RQ Extension Base
- PLC CPU
- Process CPU
- C Controller
- Motion CPU
- Power Supply
- Input
- Output
- RY10R2 16 points (Ci)
- RY40NT5I 16 points (Si)
- RY40P13F 16 points (Si)

Click to proceed to the next section.
To view again, click on the "Replay" button.

Replay

R04 F05C NUM

2.6

Setting module operations



MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation ProgPou [PRG] [Local Label ...] ProgPou [PRG] [LD] 2Step Module Configuration x

Element Selection (Find POU)

Module Label

- 3E00:R04CPU
- 0010:RY10R2
- RY10R2_1**

RY10R2_1

PO... Fav... His... Mod...

Input the Configuration ...

R04CPU

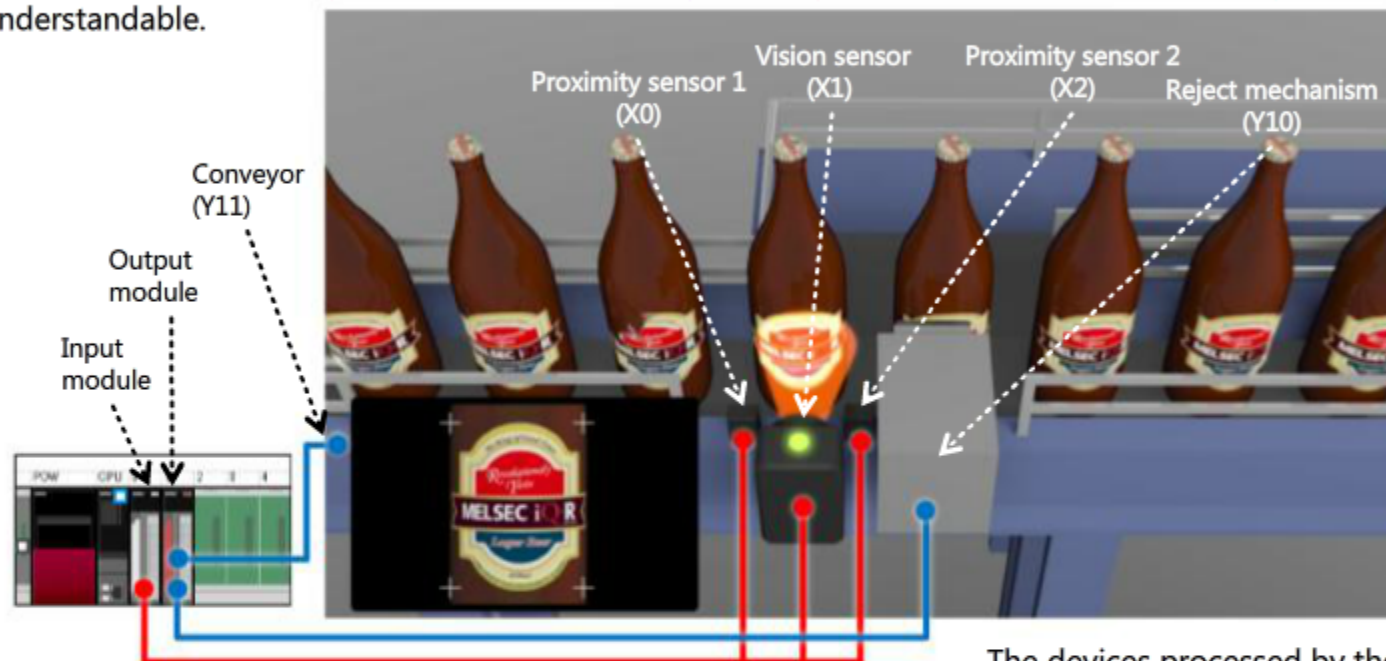
Points of Emp. 16 Points

Click to proceed to the next section.
To view again, click on the "Replay" button.

Replay

2.7 Giving names to devices

Device names, which are processed by programmable controllers, can have labels for easier understanding. A label name can be information such as device usage or connected device. By showing such information as labels, program contents become more easily understandable.



I/O devices corresponding to external equipment are assigned with the following labels.

External equipment	Device	Input or output	Label
Proximity sensor 1	X0	Input	ProximitySensor_1
Vision sensor	X1	Input	VisionSensorResult
Proximity sensor 2	X2	Input	ProximitySensor_2
Reject mechanism	Y10	Output	PusherStart
Conveyor	Y11	Output	ConveyorStart

The devices processed by the programmable controller are assigned with the following labels. Details are given in Chapter 3.

Label
ProximitySensor1_Count
ProximitySensor2_Count
DefectiveLabelCount
LastDefectiveLabelCount
PushCompleteTime
PushCompleteTimer
PushTrigger

2.7.1 Label types

This section explains different types of labels before proceeding to the explanation of label registration procedure. The main two types are Global Label and Local Label, and they differ by their applicable ranges.

Global Label


Global Label can be used for different programs within a project.

Beverage production line project

	Label Name	Data Type	Class
1	Amount	Double Word [Unsigned]/Bit String [32-bit]	VAR_GLOBAL
2	Year	Double Word [Unsigned]/Bit String [32-bit]	VAR_GLOBAL

Inspect
program

The next section shows how to actually register a Global Label.

Click  to proceed to the next.

To view again, click on the "Replay" button.

Replay

Sorting
program

	Label Name	Data Type	Class
1	Amount	Double Word [Unsigned]/Bit String [32-bit]	VAR_GLOBAL
2	Year	Double Word [Unsigned]/Bit String [32-bit]	VAR_GLOBAL

2.7.2

Global Label registration

MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation: .label ... Global [Global Label Setting] x ProgPou [PRG] [LD] 536Ste... Module Configuration COMMENT [Device Comme...]

Module Configuration

- Program
 - Initial
 - Scan
 - MAIN
 - ProgPou
 - Local Label
 - Program
 - Fixed Scan
 - Event
 - Standby
 - No Execution Type
 - Unregistered Program
 - FB/FUN
 - Label
 - Global Label
 - Global
 - M+Global
 - Structured Data Types
 - Device
 - Device Comment
 - Each Program Device Cor
 - Common Device Comme

Global Label Setting


<Filter> Easy Display << Display Setting Check

	Label Name	Data Type		Class	Assign (Device/Label)	Initial Value
1	ProximitySensor_1	Bit	...	VAR_GLOBAL	X0	
2	ProximitySensor_2	Bit	...	VAR_GLOBAL	X2	
3	VisionSensorResult	Bit	...	VAR_GLOBAL	X0	
4	PusherStart	Bit	...	VAR_GLOBAL	Y10	
5	ConveyorStart	Bit	...	VAR_GLOBAL	Y11	
6			...			

Extended Display: Automatic

System label is reserved to be registered. System label is reserved to be released.

To execute the Reservation to Register/Release for the system label, reflection to the system label database is required. Please execute 'Reflect to System Label Database'. It is unnecessary to change reference side project when assigned device is changed in system label Ver.2.
 * Only IQ-R series/GOT 2000 series is available for system label Ver.2.
 * To execute Online Program Change, execute Online Program Change and save.

Click  to proceed to the next section.
 To view again, click on the "Replay" button.

Replay

R04 Host Row 5 Column 5 CAP NUM

2.7.3

Local Label registration

MELSOFT GX Works3 .

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation


- Module Configuration
 - Program
 - Initial
 - Scan
 - MAIN
 - ProgPou
 - Local Label
 - Program
 - Fixed Scan
 - Event
 - Standby
 - No Execution Type
 - Unregistered Program

- FB/FUN
- Label
- Global Label
 - Global
 - M+Global
- Structured Data Types
- Device
- Device Comment
 - Each Program Device Cor
 - Common Device Comme

ProgPou [PRG] [Local Label ... x Global [Global Label Setting] ProgPou [PRG] [LD] 536Ste... Module Configuration COMME 4

<Filter> Easy Display << Display Setting Check

	Label Name	Data Type	Class	Initial Value	Constant
1	ProximitySensor1_Count	Double Word [Unsigned]/Bit String [32-bit]	.. VAR		
2	DefectiveLabelCount	Double Word [Unsigned]/Bit String [32-bit]	.. VAR		
3	LastDefectiveLabelCount	Double Word [Unsigned]/Bit String [32-bit]	.. VAR		
4	ProximitySensor2_Count	Double Word [Unsigned]/Bit String [32-bit]	.. VAR		
5	PushCompleteTime	Word [Unsigned]/Bit String [16-bit]	.. VAR		
6	PushCompleteTimer	Timer	.. VAR		
7	PushTrigger	Bit	.. VAR		
8			..		

Click  to proceed to the next section.
To view again, click on the "Replay" button.

Replay

Extends

R04 Host Row 8Column 1 CAP NUM

2.7.4

Module Label overview

MELSOFT GX Works3 (Untitled Project) - [Module Configuration]

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation ProgPou [PRG] [Local Label ...] ProgPou [PRG] [LD] 2Step Module Configuration x


Element Selection (Find POU)

Module Label

- 3E00:R04CPU
- 0010:RY10R2
- RY10R2_1
- R1 Version: 00A
- ul

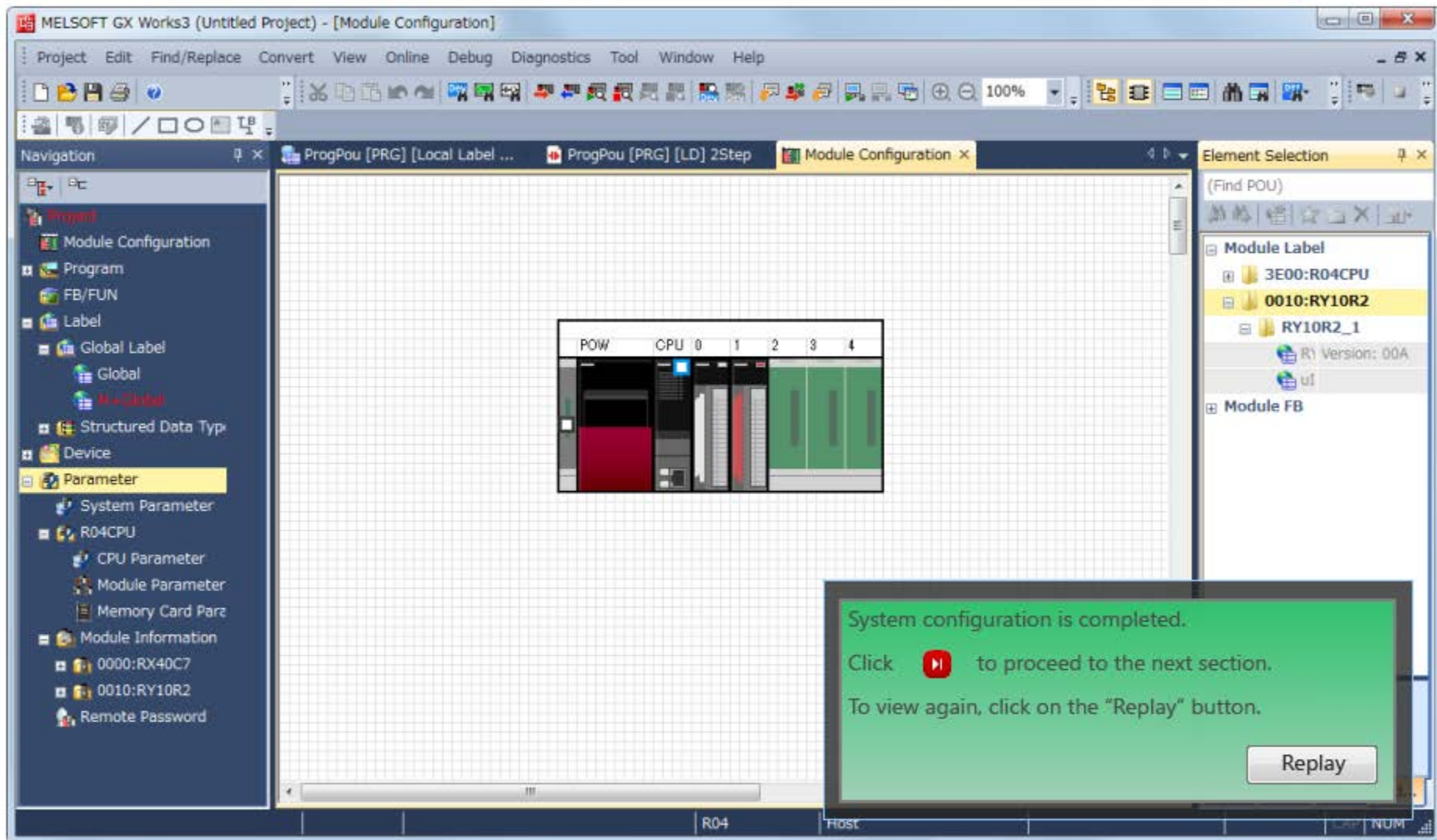
Module FB

POW CPU 0 1 2 3 4

System configuration is completed.
Click  to proceed to the next section.
To view again, click on the "Replay" button.

Replay

R04 FROST EXP NUM



The screenshot displays the MELSOFT GX Works3 software interface in the 'Module Configuration' mode. The main workspace shows a rack configuration diagram with columns labeled 'POW', 'CPU 0', '1', '2', '3', and '4'. A green callout box in the bottom right corner contains the text: 'System configuration is completed. Click [Next button] to proceed to the next section. To view again, click on the "Replay" button.' The 'Replay' button is located at the bottom right of the callout box. The left sidebar shows a tree view of the project structure, with 'Parameter' selected. The right sidebar shows the 'Element Selection' panel with a tree view of the module configuration, including '3E00:R04CPU', '0010:RY10R2', and 'RY10R2_1'. The status bar at the bottom shows 'R04', 'FROST', 'EXP', and 'NUM'.

2.8

Saving the created content

MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation ProgPou [PRG] [Local Label ...] Global [Global Label Setting] ProgPou [PRG] [LD] 417Step x Module Cor

Project occ

- Module Configuration
- Program
 - Initial
 - Scan
 - MAIN
 - ProgPou
 - Local Label
 - Program
 - Fixed Scan
 - Event
 - Standby
 - No Execution Type
 - Unregistered Program
- FB/FUN
- Label
 - Global Label
 - Global
 - M+Global
 - Structured Data Types
 - Device
 - Device Comment

Write

	1	2	3	4	5
1	(0) SM402				
2					
3	(72) ProximitySensor_1				
4		VisionSensorResult			
5	(204) ProximitySensor_2				
6		D<	DefectiveLabelID...	LastDefectiveLabelID...	D=
7					
8	(353) PushTrigger	PushCompleteTimer			
9	PusherStart				
10	(415)				

Element Selection


(Find POU)

Display Target: All

SEQUENCE INSTRUCTIONS

POU... Fav... Hist... Mo...

Input the Configuration D...

Click  to proceed to the next section.
To view again, click on the "Replay" button.

Replay

R04 Host 163/417 Step Overwrite CAP NUM

In this chapter, you have learned:

- Programmable controller system example
- Components for the example system
- Main features of GX Works3
- Creating a project
- Module configuration according to the system
- Setting module operations
- Giving names to devices
- Saving the created content

Important points to consider:

Module configuration	Module configuration of GX Works3 is a graphical diagram showing a physical module configuration. Basic parameters can also be set from this diagram.
Label	Easily recognizable names can be assigned as labels to make a program more understandable.
Global Label	Global Labels can be used in multiple programs within a project.
Local Label	Local Labels can be used in a program.
Module Label	Module Label is a label already assigned to an I/O address or a buffer memory address of a specific module.

Chapter 3 Program editing

This chapter explains about creating control programs.

- 3.1 Programming languages and their characteristics
- 3.2 System specifications
- 3.3 Program contents
- 3.4 Editing a program
- 3.5 Using grouped instructions
- 3.6 Making a program understandable
- 3.7 Creating comments in multiple languages
- 3.8 Checking the program for mistakes
- 3.9 Converting a program to the executable format
- 3.10 Summary



Programming

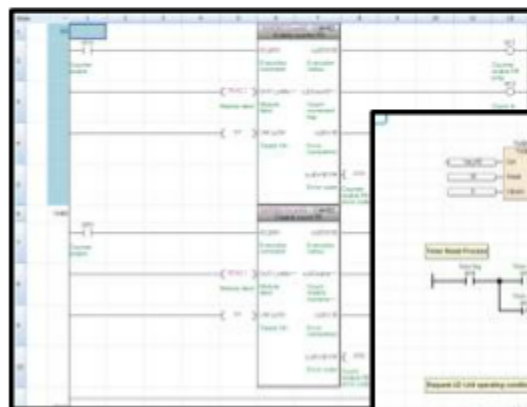
3.1

Programming languages and their characteristics

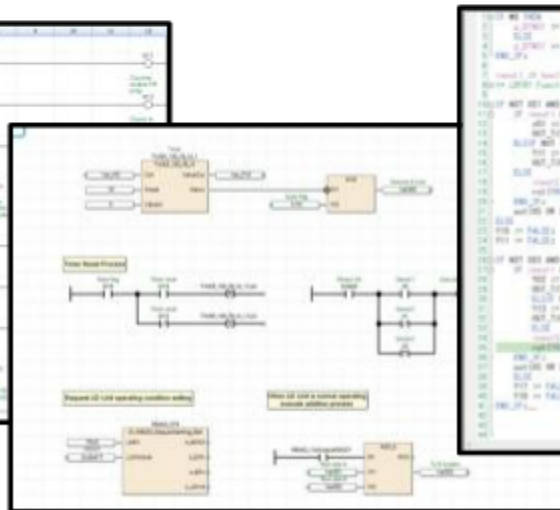
Operations of a programmable controller must be written out as a control program. GX Works3 supports the following programming languages. Various different programming languages can be used within the same project.

Programming language	Features
Ladder	<ul style="list-style-type: none"> In ladder programming, contacts and coils are used to create a program resembling an electrical circuit. Instruction processes are easy to follow even for a user with little experience.
FBD (Function Block Diagram)	<ul style="list-style-type: none"> In FBD, a program consists of function blocks. Program contents are easily seen and are easily reproduced.
ST (Structured Text)	<ul style="list-style-type: none"> ST program is described using texts. ST may be familiar for programmers who have experience in C programming.
SFC (Sequential Function Chart) * Coming soon	<ul style="list-style-type: none"> Conditions and processes are described in a flow chart. Program steps are easy to follow.

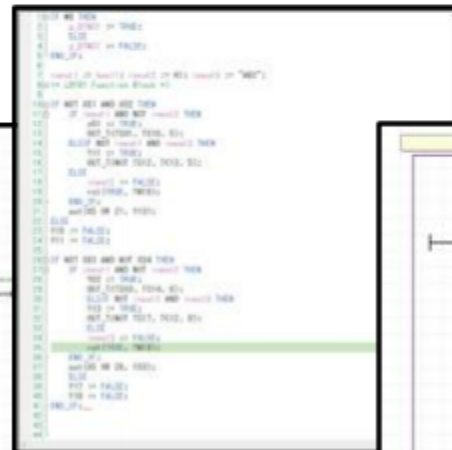
This course will use Ladders in creating the example inspection system program.



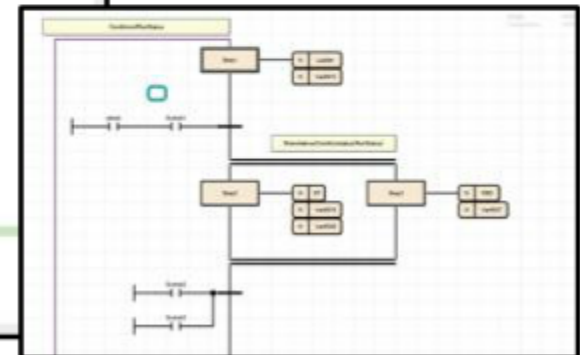
Ladder



FBD



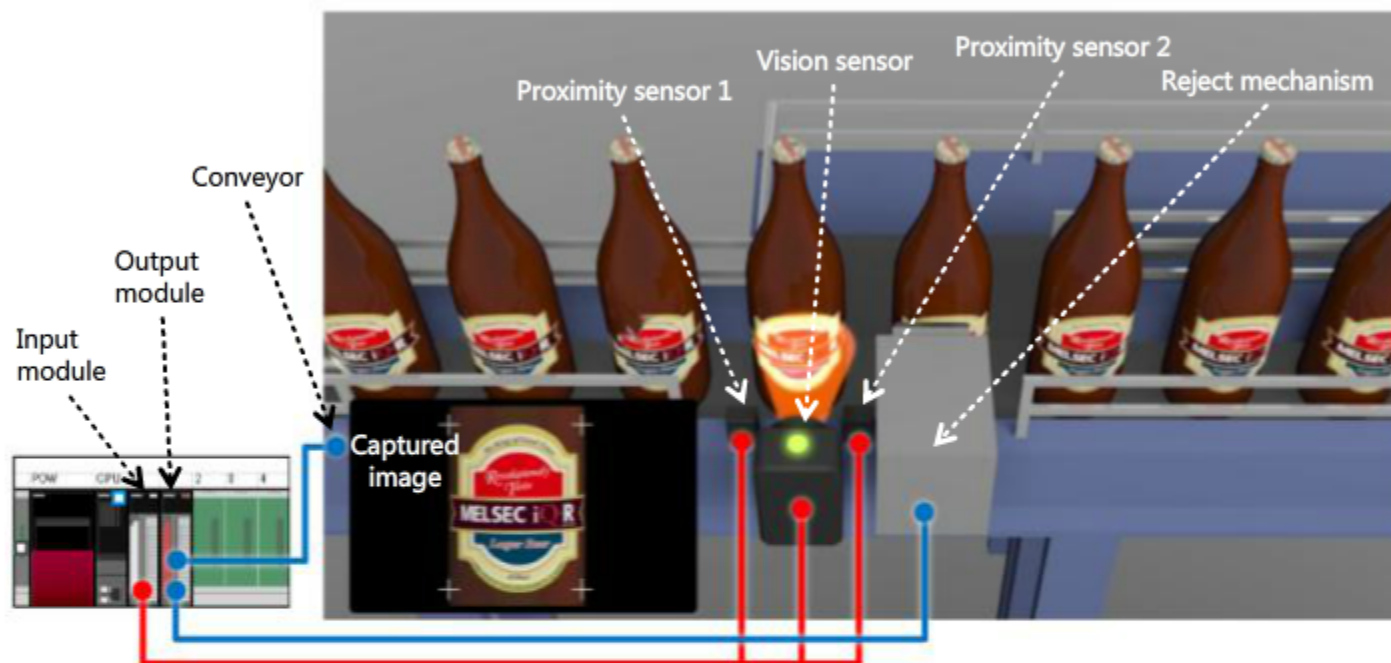
ST



SFC

3.2 System specifications

Before proceeding to program editing, please confirm the specification of the example system.



I/O devices

External equipment	Input or output	Global Label
Proximity sensor 1	Input	ProximitySensor_1
Vision sensor	Input	VisionSensorResult
Proximity sensor 2	Input	ProximitySensor_2
Reject mechanism	Output	PusherStart
Conveyor	Output	ConveyorStart

Internal devices

Label name (Local Label)
ProximitySensor1_Count
ProximitySensor2_Count
DefectiveLabelCount
LastDefectiveLabelCount
PushCompleteTime
PushCompleteTimer
PushTrigger

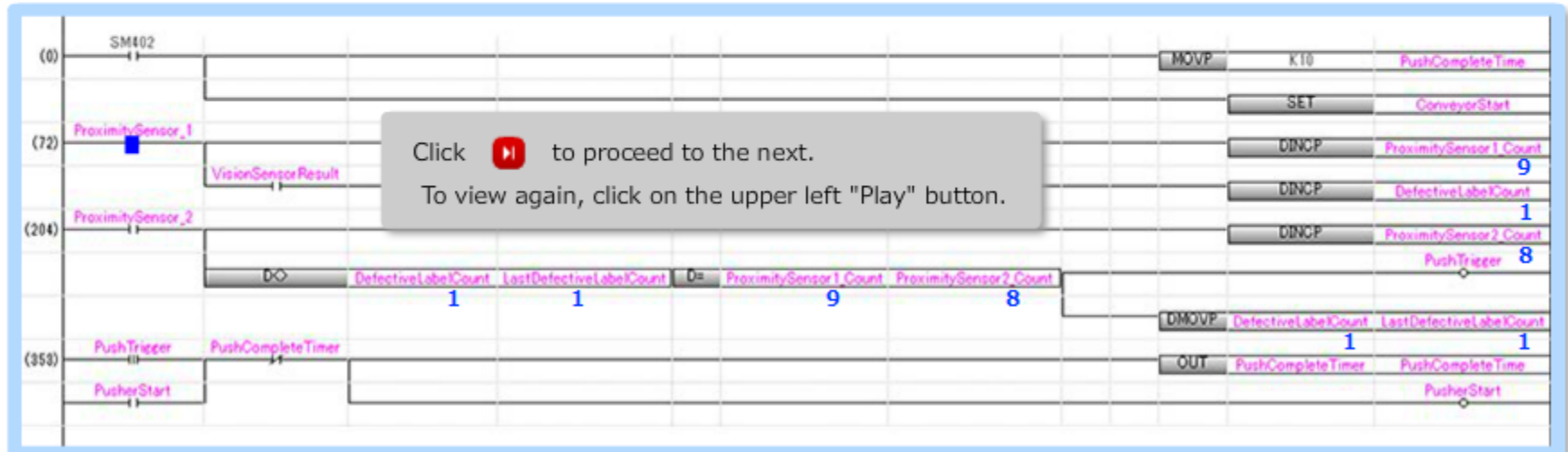
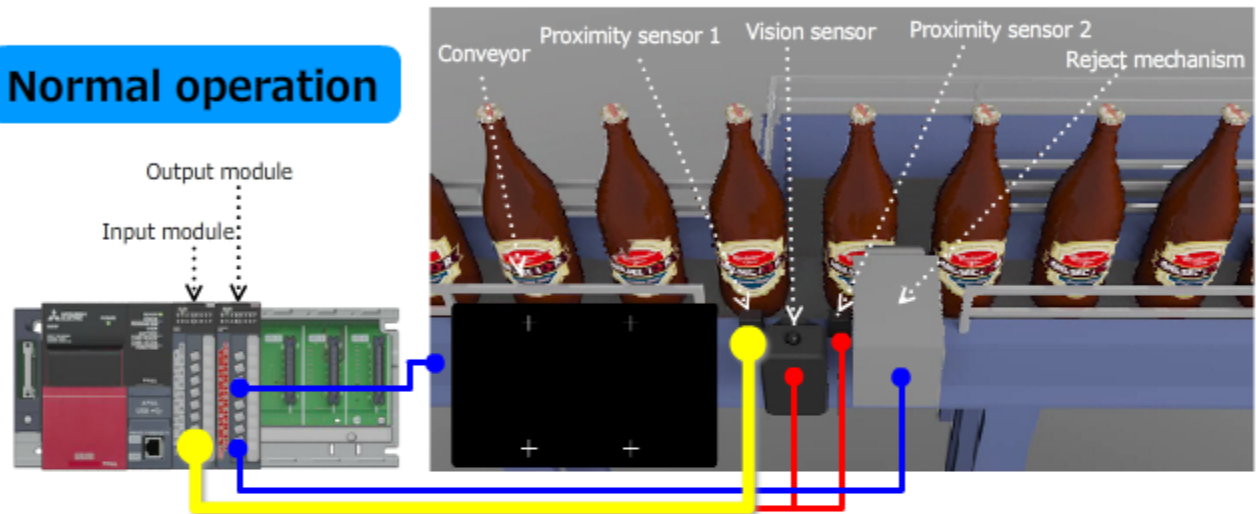
3.3 Program contents


This section explains about the program required for the example inspection system.
Here is the example inspection system and the control program linked with the system operation.

Normal operation

Please click the button below to start the animation.

Play



Click  to proceed to the next.
To view again, click on the upper left "Play" button.

3.4

Editing a program

MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Global [Global Label Setting] ProgPou [PRG] [Local Label ...] ProgPou [PRG] [LD] 2Step * x

Write 1 2 3 4 5 6 7 8 9 10 11 12

1 SM402 MOV K10 PushCompleteTime

2 SET ConveyorStart

3 ProximitySensor_1 DINCP ProximitySensor1...

4 VisionSensor... DINCP DefectiveLabelCount

5 ProximitySensor_2 DINCP ProximitySensor2...

6 D<> Defective... LastDefect... D= ProximityS... ProximityS... PushTrigger

7 DMOV Defect... LastDefectiveLabe...


8 PushTrigger PushComple... OUT PushC... PushCompleteTime

9 PusherStart PusherStart

10 (0) .END

Increment

DINCP[Incrementing]

Click  to proceed to the next section.
To view again, click on the "Replay" button.

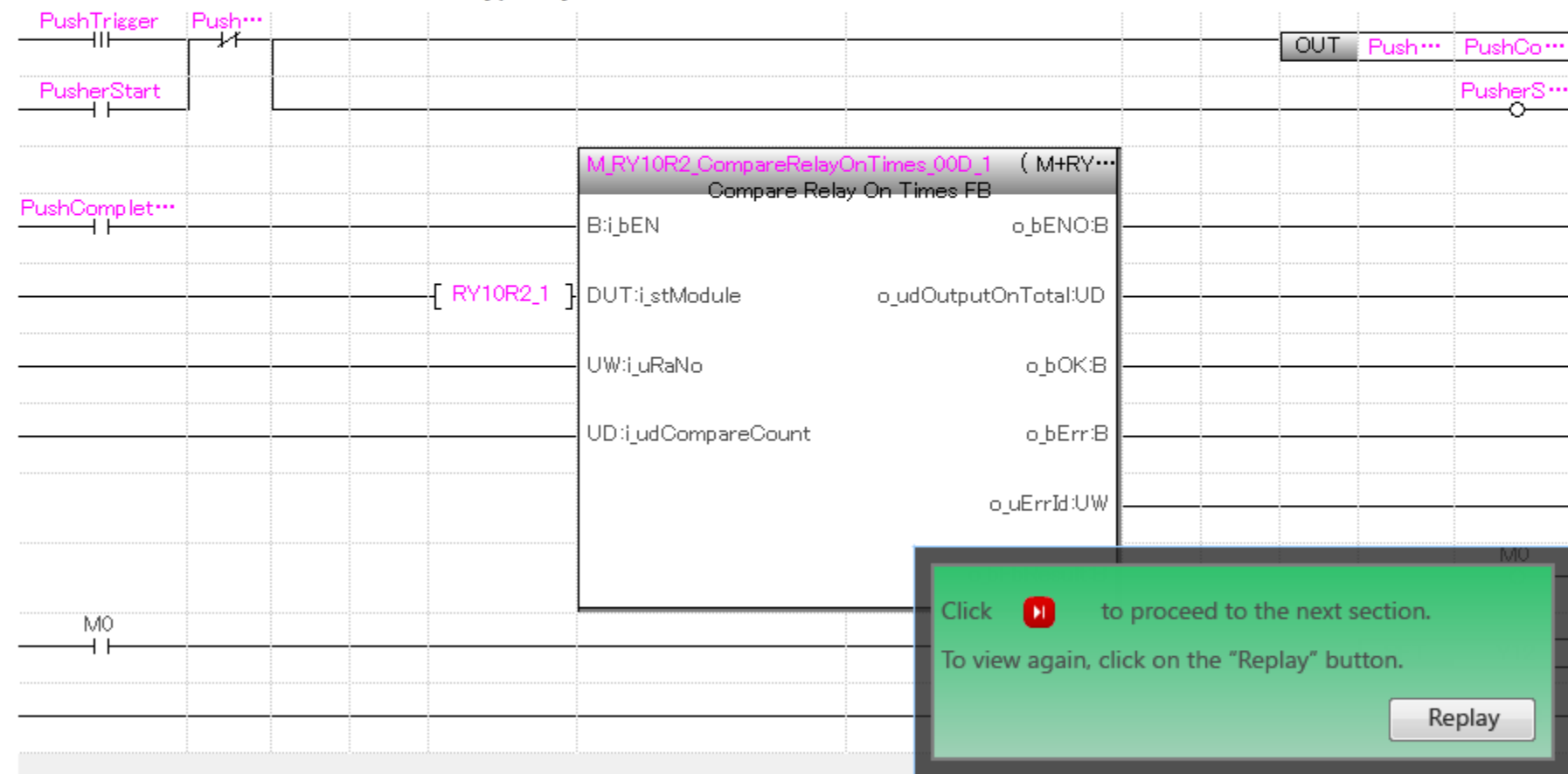
Replay

R04 FROST 0/2 Step Overwrite CAP NUM

3.5 Using grouped instructions

In a program, often-used instructions can be grouped together as a function block (FB). FB can simplify a lengthy program and shorten the programming time.

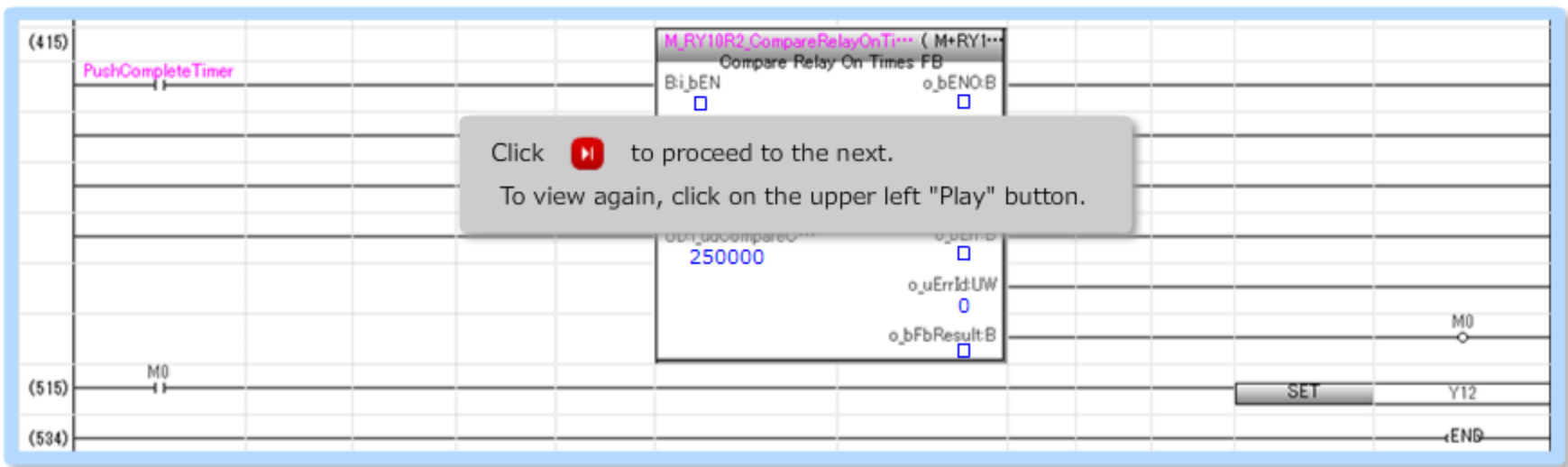
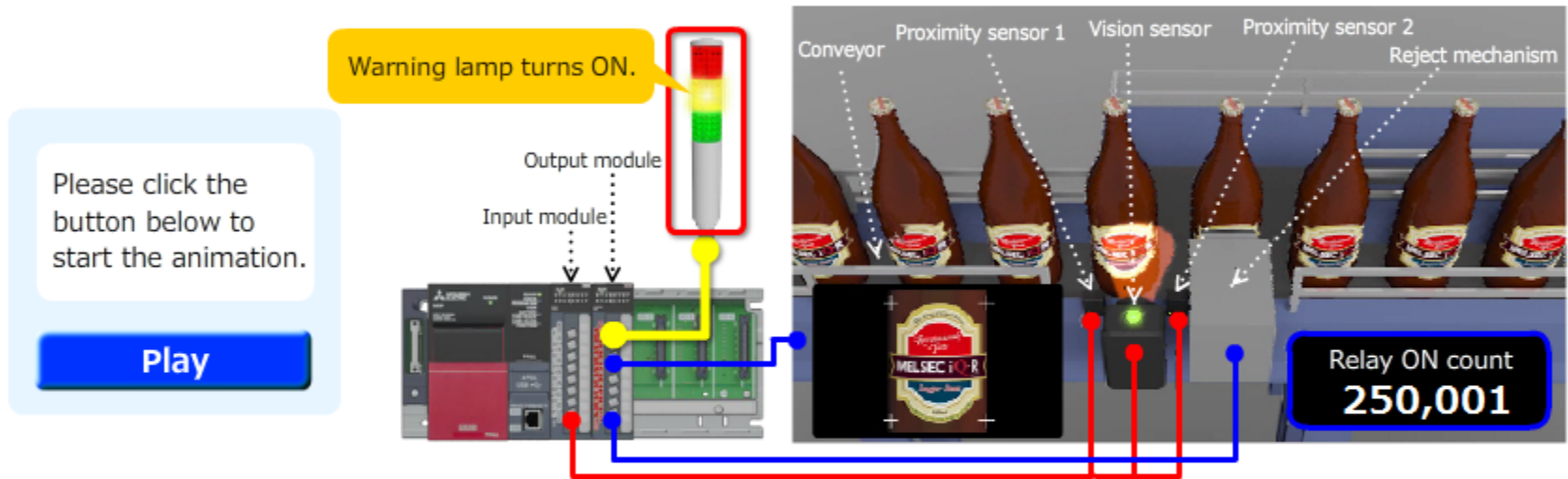
An FB can be created by the user or a selection of FBs can be obtained from your local Mitsubishi Electric representative. GX Works3 also has pre-made FBs called Module FB. The Module FB is specific to a module and contains a set of instructions that are typically used.



3.5.1

Creating a program containing Module FB

The bottle label inspection system uses a relay output module to control the reject mechanism. Although the relay type output module can handle large load current as it uses mechanical contacts, (which can have a limited service life) the internal relay contact would require servicing. To allow for this maintenance, a program that notifies the service life is required and can be done easily using a Module FB.



3.5.2

Placing a Module FB

MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

ProgPou [PRG] [Local Label ...] Global [Global Label Setting] ProgPou [PRG] [LD] 536Step x Module Configuration COMMENT [Device Comme...]

Write 1 2 3 4 5 6 7 8 9 10 11 12

6 D◊ DefectiveL... LastDefectiv... D= ProximitySensor1_C... ProximitySensor2_Co... PushTrizzer

7 DMOVP DefectiveL... LastDefectiveLabelC

8 (353) PushTrizzer PushComp... OUT PushComp... PushCompleteTi

9 Push排出ボタン PusherStart

10 (415) M_RY10R2_CompareRelayOnTim... (M+RY...
Compare Relay On Times FB

11 PushComp... Bi_bEN o_bENOB

12 [RY10R2_1] DUT:i_stModule o_udOutputOnTotal...

13 UW:i_uRaNo o_bOKB


14 UD:i_udCompareCo... o_bErrB

15

16

17 (515) M0

18 (534)

Click  to proceed to the next section.
To view again, click on the "Replay" button.

Replay

R04 Host -/536 Step Overwrite CAP NUM

3.6


Making a program understandable

MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

ProgPou [PRG] [Local Label ...] Global [Global Label Setting] ProgPou [PRG] [LD] 536Step x Module Configuration COMMENT [Device Comme...]

Write	1	2	3	4	5	6	7	8	9	10	11	12	
1	(1) Initial settings												
2												Set the operation timing of the reject arm	
3	(0) ON once after CPU RUN	SM402								MOVP	K10	PushCompleteTime	
4											SET	ConveyorStart	
5	(2) Defective label processing												
6												Count the bottles inspected by the visi	
7	(72) Detects that a bottle reached the vision sensor	ProximityS...										DINCP	ProximitySensor1_Count
8												Count the bottles with defective label	
9													
10	(3) Reject arm processing												
11													

Click  to proceed to the next.
To view again, click on the "Replay" button.

Replay

R04 Host -/536 Step Overwrite CAP NUM

3.7

Creating comments in multiple languages


MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation .local Label ... Global [Global Label Setting] ProgPou [PRG] [LD] 536Step Module Configuration COMMENT [Device Comme... x

Device Name M0 Detailed Conditions

Device Name	Japanese/日本語	English(Display Target)	Chinese/中文
+ M0	リレー寿命設定値に到達でON	On when relay life limit reached	到达继电器寿命设定值时ON
M1			
M2			
M3			
M4			
M5			
M6			
M7			
M8			
M9			
M10			
M11			
M12			
M13			
M14			
M15			
M16			
M17			
M18			
M19			
M20			
M21			
M22			
M23			
M24			
M25			
M26			
M27			
M28			
M29			
M30			

Click  to proceed to the next.
To view again, click on the "Replay" button.

Replay

R04 Host Row 1Column 1 CAP NUM

3.8

Checking the program for mistakes



MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation ProgPou [PRG] [Local Label ...] Global [Global Label Setting] ProgPou [PRG] [LD] 536Step x Module < >

Element Selection (Find POU)

Display Target: All


SEQUENCE INSTRUCTION

- Contact instructions
- Association instruction
- Output instructions
- Shift instructions
- Master Control instruction
- Termination instruction

POU... Fav... His... Mo...

Find and Replace Find Device/Label+

(Entire Projects)

Click  to proceed to the next.
To view again, click on the "Replay" button.

Replay

Write 1 2 3 4 5 6

1 ((1) Initial settings)

2 SM402

3 (0) ON once after CPU RUN

4

5 ((2) Defective label processing)

6 ProximityS***

7 (72) Detects that a bottle reached the vision sensor

8 VisionSens***

9 ON when defective bottle label is detected

10 ((3) Reject arm processing)

11

R04 Host Overwrite CAP NUM

3.9

Converting a program to the executable format

MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

ProgPou [PRG] [Local Label ...] Global [Global Label Setting] ProgPou [PRG] [LD] 536Step x Module Configuration COMMENT [Device Comme...]

Write 1 2 3 4 5 6 7 8 9 10 11

15 DMOV

16 Measure the operation time of the

17 (353) PushTrigger PushCompl... OUT PushComp...


18 PusherStart

19 (4) Relay life limit warning processing

20 (415) M.RY10R2.CompareRelayOnTimes.00... (M+RY... Compare Relay On Times FB B:I_bEN

21 PushCompl...

22 [RY10R2_1] DUT:i_stModule Module label for the output

Click  to proceed to the next section. To view again, click on the "Replay" button.

Replay

R04 Host Overwrite CAP NUM

3.10 Summary

In this chapter, you have learned:

- Programming languages and their characteristics
- System specifications
- Program contents
- Editing a program
- Using grouped instructions
- Making a program understandable
- Creating comments in multiple languages
- Checking the program for mistakes
- Converting a program to an executable format

Important points to consider:

FB	<ul style="list-style-type: none"> • Various instructions that are used multiple times are grouped together in a function block (FB). • FB can simplify a lengthy program and shorten the overall programming time. • An FB can be created by the user, or utilize one of the FB pre-installed in GX Works3.
Module FB	<ul style="list-style-type: none"> • Module FB is an FB specific to a module and contains a set of instructions typically used for the module.
Comment	<ul style="list-style-type: none"> • Make a program more understandable to the programmer and the others. • Reduces chances of programming mistakes. • Entered in multiple languages.
Program conversion	<ul style="list-style-type: none"> • Required to be converted to a format executable by the programmable controller CPU module.

Chapter 4 Operation check

This chapter explains how to check the operation of created programs.

- 4.1 Confirming the example inspection system
- 4.2 Debugging using the simulation function
- 4.3 Debugging on the actual system
- 4.4 Preparing for the system operation
- 4.5 Summary



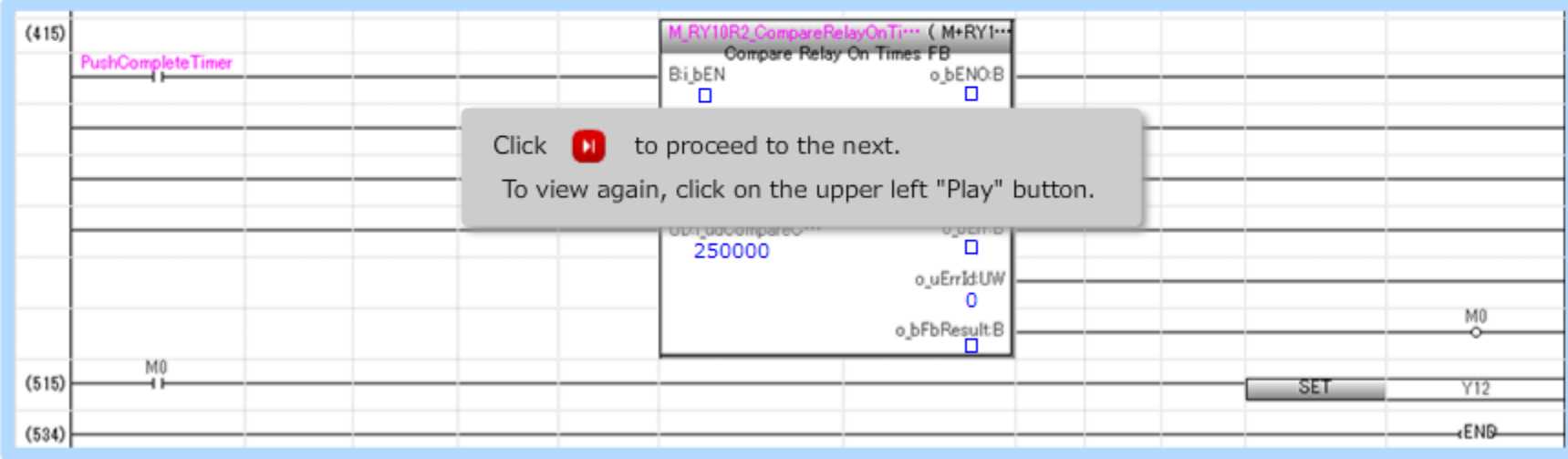
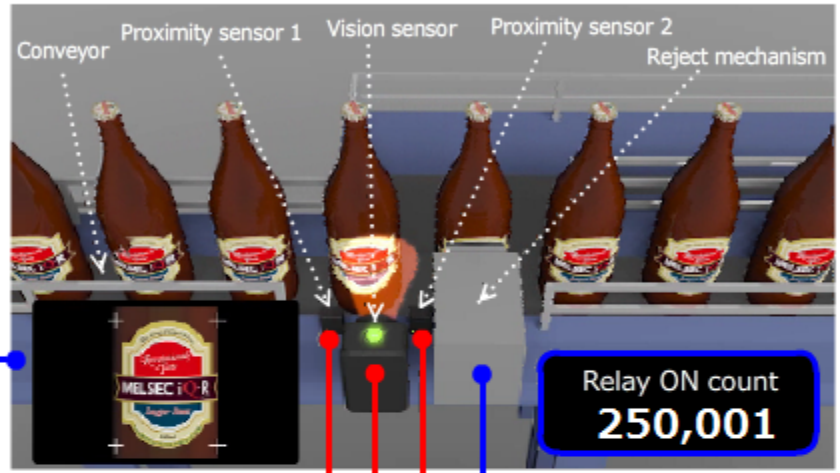
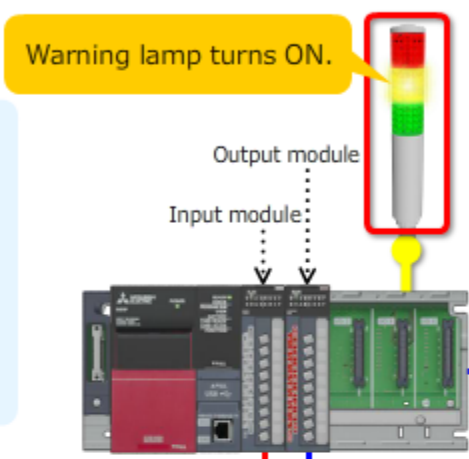
4.1

Confirming the example inspection system

The program to inspect labels and the program to detect the relay limit life are shown here.

Please click the button below to start the animation.

Play



Click to proceed to the next.
To view again, click on the upper left "Play" button.

4.2

Debugging using the simulation function

MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Max.: 1.922ms

Navigation

ProgPou [PRG] [LD] Monitor... ProgPou [PRG] [Local Label ... Global [Global Label Setting] Module

Element Selection

(Find POU)

Display Target: All

SEQUENCE INSTRUCTIONS

- Contact instructions
- Association instructions
- Output instructions
- Shift instructions
- Master Control instructions
- Termination instructions
- Stop instruction
- Ignored instructions

BASIC INSTRUCTIONS

- Comparison Operation instructi
- Arithmetic Operation instructio
- Data transfer instructions
- Logical Operation instructions
- Data shift instructions

GX Simulator3

1.1 R04CPU

LED

READY ERROR P. RUN USER

SWITCH

RUN STOP

RESET

ProximitySen... VisionSensor... ON when defective bottle label is detected

(2) Defective label processing

(72) Detects that a bottle reached the vision sensor

(3) Reject arm processing


(204) Detects that a bottle reached the reject mechanism

DefectiveLa... LastDefectiveL... ProximitySensor_Count Proximity...

PushTrizeer PushComple...

(353)

R04 Simulation (1.1) 456/536 Step Overwrite CAP NOM

Click  to proceed to the next section.
To view again, click on the "Replay" button.

Replay

4.2

Debugging using the simulation function



MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

ProgPou [PRG] [LD] Monitor... x ProgPou [PRG] [Local Label ... Global [Global Label Setting] Module Configuration COMMENT [Device Comme...

1 2 3 4 5 6 7 8 9 10 11 12

1 (1) Initial settings

2 Set the operation timing of the reject arm

3 (0) ON once after CPU RUN

4 MOV K10 PushCompleteTime 10

5 SET ConveyorStart

5 (2) Defective label processing

6 Count the bottles inspected by the vision ...

7 (72) Detects that a bottle reached the vision sensor

8 ProximitySensor1_Count

9 VisionSensorResult

ON when defective bottle label is detected

10 Count the bottle with defective labels

11 DefectiveLabelCount

1 (3) Reject arm processing

1 ProximitySensor2

2 (204) Detects that a bottle reached the reject mechanism

Click to proceed to the next section.
To view again, click on the "Replay" button.

Replay

R04 Host 104/536 Step Overwrite CAP NUM

4.3

Debugging on the actual system

Resetting the CPU module




Executing control programs



Executing control programs

P RUN LED turns on, and the control program is executed.



Click  to proceed to the next section.
To view again, click on the "Replay" button.

Replay

4.3

Debugging on the actual system



4.4

Preparing for the system operation



MELSOFT GX Works3

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

ProgPou [PRG] [LD] 536Step ProgPou [PRG] [Local Label ... Global [Global Label Setting] Module Configuration COMMENT [Device Comme... Verify Result [Verify With P...

Result List


Verify Source: Editing Data Verify Destination: PLC
 Source Project: LIS_en Destination Project: R04n CPU
 Verify Source Data Name: Verify Destination Data Name:

Result List

No.	Type	Data Name(Verify Source)	Data Name(Verify Destination)	Verify Result
1	Program File	MAIN	MAIN	Match
2	Program	ProgPou	ProgPou	Match
3	FB/FUN	M+RY10R2_CompareRelayOnTimes_00D	M+RY10R2_CompareRelayOnTimes_00D	Match
4	Parameter	System Parameter	System Parameter	Match
5	Parameter	CPU Parameter	CPU Parameter	Match

0 differences

R04 HOST CPU R04

Click  to proceed to the next.
 To view again, click on the "Replay" button.

4.5**Summary**

In this chapter, you have learned:

- Confirming the bottling label inspection system
- Debugging using the simulation function
- Testing on the actual system
- Preparing for the system operation

Important points to consider:

Simulation feature	The simulation function checks the program operation without physical modules.
Monitoring function	The program being executed can be monitored by using the monitoring function.

Chapter 5 Maintenance

This chapter explains how to maintain a system by using GX Works3.

- 5.1 Checking for any abnormalities
- 5.2 Investigating the error cause
- 5.3 Maintenance at overseas locations
- 5.4 Course summary

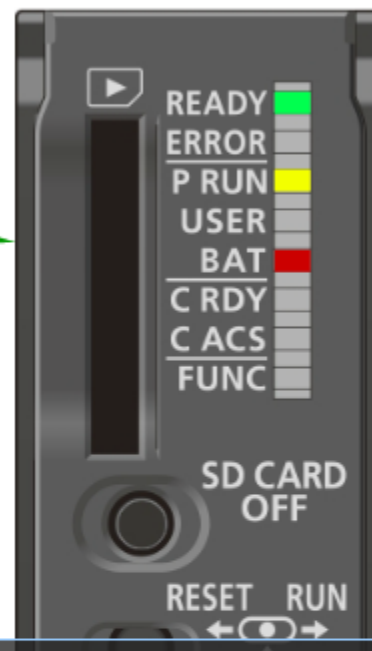


5.1


Checking for any abnormalities



Preliminary diagnosis can be performed by looking at CPU module LED lamps. Flashing "BAT LED" indicates an error related to the battery.



The next section explains more how to do this.

Click  to proceed to the next section.

To view again, click on the "Replay" button.

Replay

5.2

Investigating the error cause



MELSOFT GX Works3 (Untitled Project)

Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

System Monitor Main Base(R35B)

Module Find Target Find

Main Base(R35B) ⚠

Extension Base Uninstall

Extension Base Uninstall

Extension Base Uninstall

Extension Base Uninstall

Extension Base Uninstall

Extension Base Uninstall

Extension Base Uninstall

Extension Base Uninstall

Operation Status

	No. 1	No. 2	No. 3	No. 4
Operation Status	RUN	-	-	-

Display Setting... Monitoring Stop Monitoring

	Power	CPU	I/O0	I/O1	I/O2	I/O3	I/O4
Start I/O No.	-	3E00	0000	0010	0020	0030	0040
Points	-	-	16 Point	16 Point	16 Point	16 Point	16 Point
Module Name	R61P	R04CPU	RX40C7	RY10R2	-	-	-
Error Status	-	⚠ 1090	-	-	-	-	-
Module Configuration							
Control CPU	-	-	-	-	-	-	-
Network Information (Port 1)	-	-	-	-	-	-	-
IP Address (Port 1 IPv4)	-	192.168.3.39	-	-	-	-	-
Module Synchronous Status	-	-	-	-	-	-	-

Product Information List... Event History... Create File...

Click to proceed to the next section. To view again, click on the "Replay" button.

Replay

R04 HOST CAP NUM

5.3

Maintenance at overseas locations



MELSOFT GX Works3


Project Edit Find/Replace Convert View Online Debug Diagnostics Tool Window Help

Navigation

- Project
 - Module Configuration
 - Program
 - Initial
 - Scan
 - MAIN
 - ProgPou
 - Local Label
 - Program**
 - Fixed Scan
 - Event
 - Standby
 - No Execution Type
 - Unregistered Program
- FB/FUN
- Label
 - Global Label
 - Global
 - M+Global
 - Structured Data Types
- Device
 - Device Comment
 - Each Program Device Corr

ProgPou [PRG] [LD] 536Step x ProgPou [PRG] [Local Label ...] Global [Global Label Setting] Module Configuration COMME

Wr	5	6	7	8	9	10	11	12
2		UW:i_uRaNo		o_bOK:B				
3								
2		UD:i_udCompareCount		o_bErr:B				
4								
2				o_uErrId:UW				
5								
2				o_bFbResult:B				M0
6								到达继电器寿命设定值时ON
2								Warning lamp ON
2								
8								
2								
9								

Click  to proceed to the next section.
To view again, click on the "Replay" button.

Replay

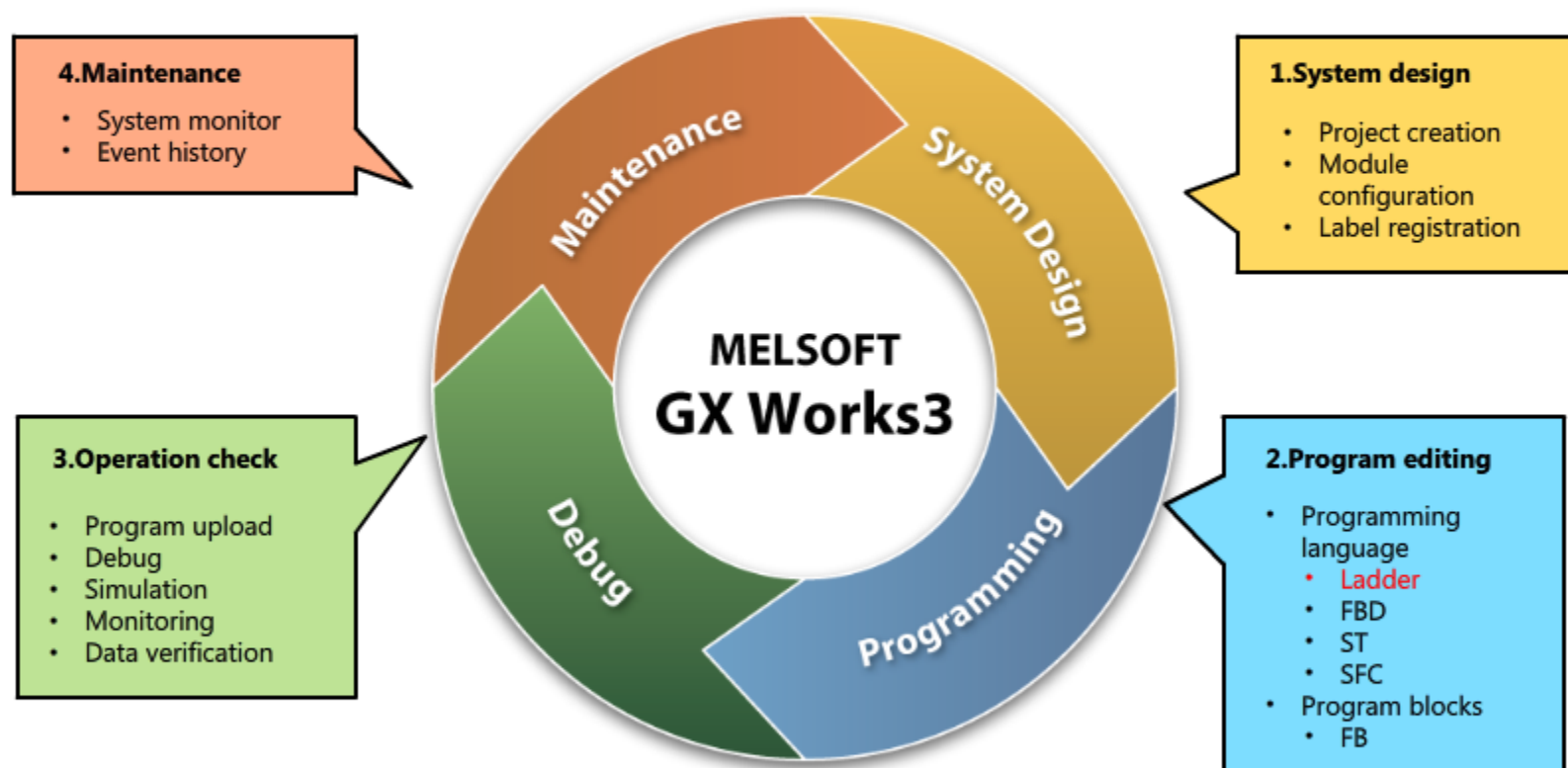
R04 Host Row 1Column 1 Overwrite CAP NUM

5.4

Course summary

The program for the bottling label inspection system has been successfully completed, and the system is confirmed to be operating normally. This brings us to the end of this e-Learning course.

GX Works3 is the essential software in configuring control programs for MELSEC programmable controller systems.



5.5

Summary

In this chapter, you have learned:

- Checking for any abnormalities
- Investigating the error cause
- Maintenance at overseas locations
- Course summary

Important points to consider:

Comments in multiple languages	When using the created program at oversea sites, the comment language can be switched according to the language spoken by the local maintenance engineer.
Diagnosis function	When the system operates abnormally, connecting a computer where GX Works3 is installed to the programmable controller will start up automatic diagnosis.

Now that you have completed all of the lessons of the **Engineering Software MELSOFT GX Works3 (Ladder)** course, you are ready to take the final test. If you are unclear on any of the topics covered, please take this opportunity to review those topics.

There are a total of 7 questions (7 items) in this Final Test.

You can take the final test as many times as you like.

How to score the test

After selecting the answer, make sure to click the **Answer** button. Your answer will be lost if you proceed without clicking the Answer button. (Regarded as unanswered question.)

Score results

The number of correct answers, the number of questions, the percentage of correct answers, and the pass/fail result will appear on the score page.

Correct Answers : 2

Total Questions : 9

Percentage : 22%

To pass the test, you have to answer **60%** of the questions correct.

Proceed

Review

Retry

- Click the **Proceed** button to exit the test.
- Click the **Review** button to review the test. (Correct answer check)
- Click the **Retry** button to retake the test again.

✓ Overview of GX Works3
Please select the correct description about GX Works3. (Multiple answers)

- Software must be switched according to the usage such as system design, startup, and maintenance.
- ✓ GX Works3 can be used in various stages of the product development lifecycle such as system design and maintenance.
- Different programming languages cannot be used within the same project.
- ✓ The simulation function enables the program operation to be checked without requiring physical modules.
- ✓ On a program, comments can be added in different languages, and the displayed language is switchable.

Back

✓ Label types
Please select the correct description about labels. (Multiple answers)

- ✓ A Global Label can be used in multiple programs.
- A Local Label can be used in multiple programs.
- ✓ Easily recognizable names can be assigned as "labels" to make a program more understandable.
- Labels Increase the processing speed of programs.

Back

✓ Overview of FB
Please select the correct description about FB. (Multiple answers)

- ✓ Often-used instructions can be grouped as an FB.
- Custom FBs cannot be created.
- ✓ FB will simplify a large program.
- ✓ Programming time is reduced by grouping often-used instructions as an FB.
- FB stands for Function Bank.

Back

**Overview of Module FB and Module Label**

Please select the correct description about Module FB and Module Label. (Multiple answers)

- Module FB contains a set of instructions typically used for a specific module.
- Every Module FB must be created and is not available pre-installed.
- Module Labels can be used without considering I/O and buffer memory addresses.

Back

✓ Overview of comments
Please select the correct description about comments. (Multiple answers)

- By having comments, the program becomes more understandable.
- Comments make a program more understandable and also reduce any mistakes.
- If the program is used overseas, comments can be added in the local language to make the program content understandable in the local language.
- Comments are automatically translated into the selected language.
- Comments are used to show the version of the program.

Back



Comment types

Which type of comments is added to a ladder rung? Please select an answer.

- Device/label comment
- Statement
- Note

Back

**Automatic diagnosis**

When a system error occurs, the diagnosis feature of GX Works3 is automatically started just by connecting to a computer. Please select the correct connection method between the computer and the CPU module.

Ethernet connection

USB connection

Back

Test**Test Score**

You have completed the Final Test. Your results are as follows.
To end the Final Test, proceed to the next page.

Correct answers : 7

Total questions : 7

Percentage : 100%

Proceed

Review

Congratulations. You passed the test.

You have completed the **Engineering Software MELSOFT GX Works3 (Ladder)** course.

Thank you for taking this course.

We hope you enjoyed the lessons and the information you acquired in this course will be useful in the future.

You can review the course as many times as you want.

Review

Close