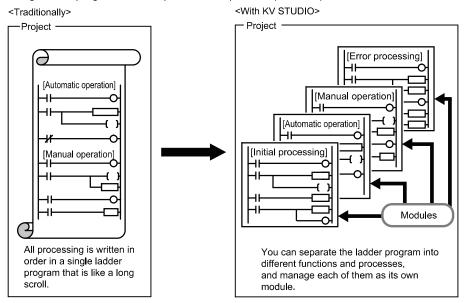
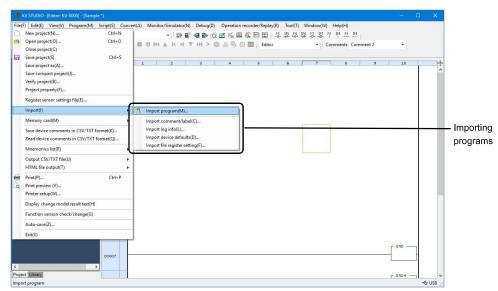
Overview

A normal ladder program that is used to control a device involves multiple processes such as initialization, automatic operation, manual operation, and error handling. Traditionally, these multiple processes have been managed consecutively in a single ladder program. With KV STUDIO, you can create programs within a project for the different aspects such as processes and functions, and then manage these programs as independent components (modules).



■ Module management

In KV STUDIO, modules are managed from the workspace of each project. You can easily reuse programs by importing programs.



To display the workspace, click the corresponding item on the "View" menu. You can also use the icons on the toolbar to show and hide this element.

You can change the display position of this element.

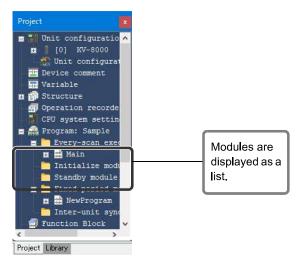
Reference

You can also reuse modules by opening multiple instances of KV STUDIO and copying the modules between applications.

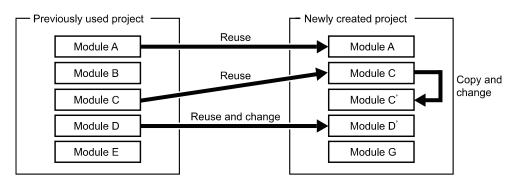
■ Module features

• It's easy to understand the structure of an entire program

When you use the workspace of KV STUDIO, each module is displayed as a tree, so even individuals who did not create the program can understand its entire structure at a glance.



It's easy to reuse just the necessary parts of a program



• Program creation, editing, and debugging can be done efficiently

By splitting modules up according to functions and processes, you can reduce the scope that must be considered during program creation and debugging. Additionally, by using local devices, you do not have to assign devices to each module, which makes it easier to have multiple people create programs.

"4-12 Local Devices" (page 4-200)

4 (1.3 /

Password protection is available

To prevent the contents of modules from being changed easily, you can set a password. When you set a password, the module program cannot be displayed or edited without first entering the password.

Use the "Program properties" dialog box to set the password.

- "Module Passwords" (page 4-114)
- "Setting a Function Block Password" (page 4-141)
- "Macro Passwords" (page 4-191)

Module Type

There are "every-scan execution modules", "initialization modules", "standby modules", "fixed period modules" and "inter-unit synchronization modules" as modules.

Use the KV STUDIO workspace to manage these different types of modules that you create. You can also change the module type after you create modules.

Every-scan execution type

These modules are executed from the first scan when operations start (when power is turned on and when the mode is switched from PRG to RUN).

You can use the module instructions, module stop (MDSTOP) and module start (MDSTRT), to control the execution status of the module during operation.

Operations when you use the module instructions are the same as the standby module operations.

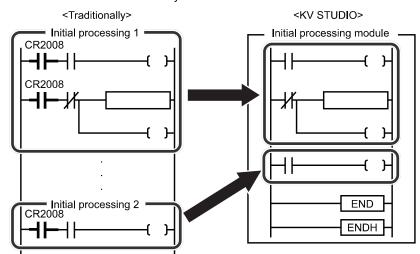
"Standby module operation" (page 4-101)

Reference

- If you use a module instruction but do not control the execution status, the module will be executed during each scan.
- When you create a new project in KV STUDIO, one every-scan execution type module with the same name as the project will be created automatically.

Initialize module

These modules are executed only during the first scan when operations start. Use these modules to perform initializations such as the initial processing of devices and restoring the default settings of KV-8000 Series functions. Even if there are no initialize modules in the project, the KV-8000 Series can operate. Create these modules as necessary.



CR2008: This is turned ON for only one scan when operation starts.

By registering modules as initialize modules, you can reduce the scan time by the time that is required to process the initialize module programs because these modules will not be executed from the second scan.



- Some instructions cannot be used in initialize modules.
 - "List of Unusable Instructions" (page A-21)
- You cannot use the module instructions (MDSTRT and MDSTOP) to control the execution status of initialize modules.
- You cannot operate the KV-8000 Series with only initialize modules. Be sure to use these modules together with at least one every-scan execution type module or standby module.

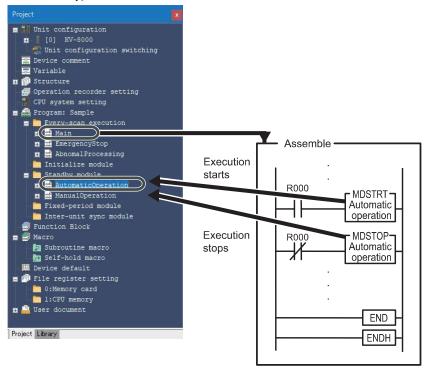
Standby module

These modules are stopped when operations start. You can use the module instructions (MDSTRT and MDSTOP) to control the execution status of these modules during operation.

"Module instructions" (page 4-115)

Example of calling standby modules

In this example, the execution status of the standby module "AutomaticOperation" is controlled from the every-scan execution type module "Main."



Point

You cannot operate the KV-8000 Series with only standby modules. Be sure to use these modules together with at least one every-scan execution type module or initialize module.



You can control the execution status of every-scan execution type modules in the same way.

■ Standby module operation

Starting the execution of standby modules

In the scan in which the module execution start instruction is called, standby modules are not executed. They will be executed from the next scan.



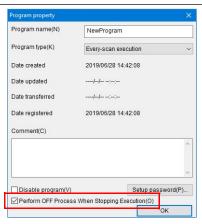
The initial status of the differential execution instruction when the execution of a standby module starts is ON when a rising edge is detected and OFF when a falling edge is detected. For example, if this is turned ON from the first scan when execution starts, rising edge differentials will not be detected.

Stopping the execution of standby modules

The operation when standby modules stop executing (when an MDSTP instruction is executed) differs according to the "Program property" -> "Perform OFF Process When Stopping Execution" setting of each program.

Point

In the case of KV-8000A, the setting is not displayed, and the operation is the same as when "OFF processing when execution stopped" item is always checked.



If "Perform OFF Process When Stopping Execution" is not checked

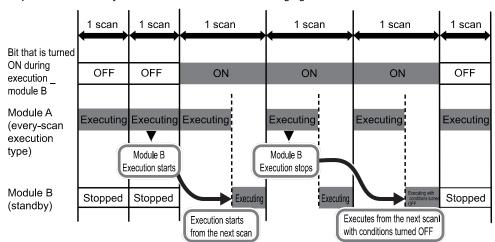
The standby module stops in the next scan after the stop module execution instruction (MDSTP) is executed.

If "Perform OFF Process When Stopping Execution" is checked

When the stop instruction is called during program execution, these modules are executed in the scan that the stop instruction was called.

In the subsequent scan, these modules are executed when all execution conditions are turned OFF. Operations are stopped in the following scan (the second scan after the stop instruction was called).

The operation of standby modules is shown in the following figure.



The following instructions are exceptions.

Instruction	Status		
OUB	Turned OFF.		
OFDL	Reset.		
SHOT	- Neset.		
С	The status is held.		
CJ, NCJ, and GOTO	Execution does not jump.		

Reference -

The ladder program within the module operates the same as when the execution condition of the MC instruction is turned OFF for just one scan before operations are stopped.

For details, see the MC and MCR instructions in the KV-8000/7000/5000/3000/1000 Series and KV Nano Series "Instruction Reference Manuals."

Fixed-Period Module

Routines in Fixed-Period Module can be separated from ordinary scanning and executed according to pre-defined period. Each project can use 4 fixed period modules.

■ Fixed-Period Module Execution

To execute Fixed-Period Module, in addition to writing the Fixed-Period Module, the following also needs to be done.

• Interrupt enable (EI) instruction

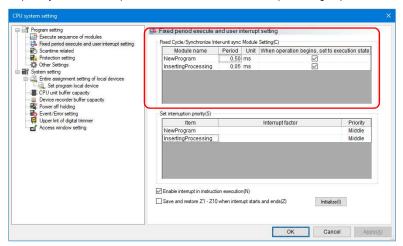
Like the interrupt routine, Enable/Inhibited of the fixed period execution is done through the EI/DI instruction. The EI instruction is written in each scanned module and executed.

Example

"EI/DI Instruction" (page 4-218)

CPU System settings

Open the "CPU system settings" window from the KV STUDIO workspace, and set up the execution period and priority in the "Fixed-period execution and user interrupt" settings option of the "Program settings"



"Fixed-Period Module settings"

The period range is 0.01 s to 60.00 s (in 0.01 s increments) or 0.05 ms to 6000.00 ms (in 0.01 ms increments). If you want to execute Fixed-Period Module right from the program is started, select the "Execute right from the start" check box. If you want to start/stop it during operation, use the module instruction (MDSTRT/MDSTOP).

"interrupt priority settings"

When the priority of a fixed period module is set to "medium," only interrupt programs and fixed period modules with a "High" priority setting can perform multiple interrupts.

Interrupt programs set to "medium" or "low" cannot interrupt the execution of fixed period module programs. Interrupt programs set to "medium" priority can interrupt the processing of interrupt programs set to "low priority. However, during the processing of interrupt programs with "high" or "medium" priority and fixed period modules, they are in standby mode.

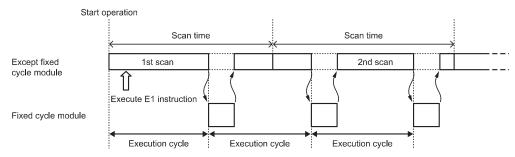


The execution cycle of fixed period modules can be set only from KV STUDIO. It can't be changed from the ladder program.

■ Fixed-Period Module Operation

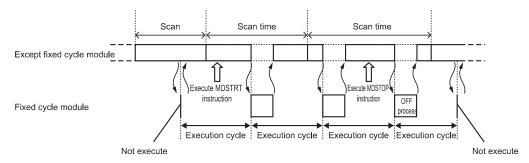
When the "When operation begins, set to execution state" check box is selected (default)

If EI instruction is executed, begin to execute Fixed-Period Module.



When the "When operation begins, set to execution state" check box is not selected

In EI instruction execution state, the module instruction is used to control execution state of Fixed Period Module.



For OFF processing, please see III "Standby module operation" (page 4-101).



Execution period is irrelevant with execution state of Fixed-Period Module, counting is started when starting to run. In case of executing El command while driving in fixed period module mode, it might performed immediately after El command execution without waiting for the cycle of fixed period execution.



- When all of the following conditions are met, the period fluctuation is within 15 μ s.
- When selecting the "Enable interrupt during instruction execution" checkbox in "Fixed period execution and user interrupt" setting of "CPU system setting"
- User interrupts with a priority higher than Fixed Period Module are not used
- High priority interrupt conditions have not been executed.
- Low priority interrupt conditions have not been executed beforehand (4 μs maximum when they have been executed)
- The KV-5000/3000 Series expansion units are not used (14 µs maximum when they have been used)
- DI instructions/DIC instructions are not being used to inhibit interrupts.
- 1 ID is not registered with a device of 513 words or more during the recording/ tracking setting.
- · When fixed-period execution timing occurs in interrupt inhibited state (when executing DI instruction) within the same execution period, Fixed-Period Module is executed in the interrupt enable state (executing EI instruction), .

Fixed-Period Module is separated with the ordinary scanning in interrupt processing. For details, please refer to III "Interrupts".

■ Device processing in the start/end

Device processing at start/end is the same as ordinary interrupt processing, at start/end of Fixed-Period Module execution, the following projects are saved/reset automatically.

- · Internal register
- Calculation flag (CR2009 to CR2012)
- · Index register(Z11 to Z23)
- TM0 to TM3

Relay To programmed in just executed STG instruction operand



Index register Z01 to Z10 can be set to execute save/reset processing in "CPU system setting" → "fixed period execution and user interrupt" settings.

Inter-unit Synchronous Module

The program within the inter-unit synchronous module is run in a preset cycle, separate from regular scans. One Inter-unit Synchronous Module project can be used per project.

■ Inter-unit Synchronous Module Operation Method

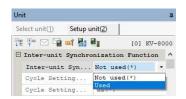
The requirements for running the inter-unit synchronous module other than creating the inter-unit synchronous module are as follows.

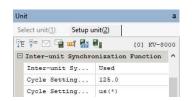
Settings for inter-unit synchronization

The contents below must be configured using the unit editor.

Settings for the CPU unit

The inter-unit synchronization of the unit setting for the CPU unit should be changed to "use". After that, the parameter of the cycle must be changed.







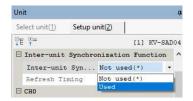
The value that can be configured for the cycle will differ depending on the expansion unit performing the inter-unit synchronization. Please check the manual for each expansion unit.

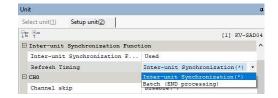
Settings for the expansion unit

The inter-unit synchronization function of the expansion unit to be run should be set to use.

The setting for refresh timing can be changed as necessary.

"Inter-Unit Synchronization Refresh Devices" (page 5-169)





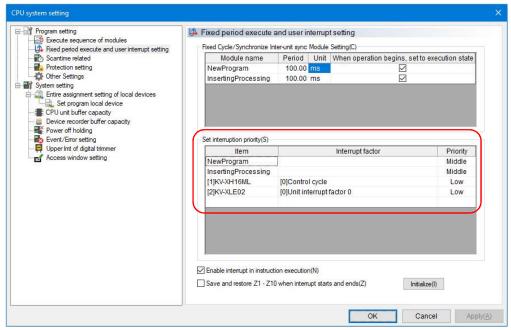
Settings for inter-unit synchronization authorization

Authorization/prohibition of inter-unit synchronization will be conducted by EI/DI command, in the same way as interruption programs. It should be performed after interrupting every scan module or standby module and stating an authorization (EI) command. Example)

```
CR2002
   4 6
 Always ON
"EI/DI Instruction" (page 4-218)
```

Settings for CPU system

The priority should be configured from [Fixed cycle operation and user interruption setting] in [Program settings] after opening "CPU system settings" from the (KV STUDIO) workspace.



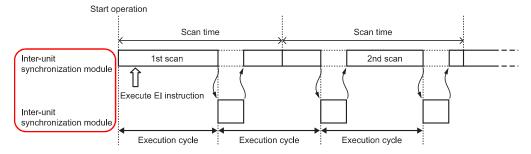
[Settings for interruption priority]

The inter-unit synchronous module with the priority set to "high" cannot interrupt other interruption programs or fixed cycle modules. Operation will have to wait while running the inter-unit synchronous module.

■ Inter-unit Synchronous Module Performance

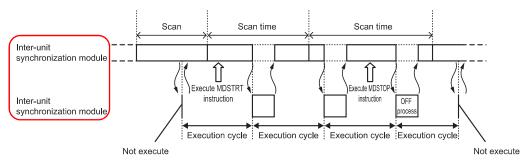
• If "Be in a performing state when starting operation" is checked (default)

If an EI command is performed, operation of the inter-unit synchronous module will begin.



• If "Be in a performing state when starting operation" is not checked

If an EI command is in a state of performance, the operational state of the inter-unit synchronous module should be regulated.



Please see "Stopping the execution of standby modules" (page 4-102) for information about OFF processing.



The performance cycle will be counted from when operation begins, regardless of the performance state or the inter-unit synchronous module. In case of executing El command while driving in inter-unit synchronization module mode, it might performed immediately after El command execution without waiting for the cycle of inter-unit synchronization execution.

Reference _

- If all the conditions below are to be met, the variation between cycles will be within 15µs.
 - If "Authorize interruption while performing a command" is being checked with the [Fixed cycle performance and user interruption settings] of the [CPU settings]
 - A user interruption with a higher priority than the inter-unit synchronous module is not being used
 - A factor with a high interruption priority is not being performed
 - A factor with a low interruption priority is not being performed immediately before (4µs at most will be added when being performed)
 - An expansion unit for the KV-5000/3000 series is not being used (14µs at most will be added when being performed)
 - · A command from DI command/DIC command is not being prohibited
- If the inter-unit synchronizing execution timing has come while it is in interruption disable mode (DI command execution), the inter-unit synchronization module execution will be started immediately after turns to interruption enabled mode (EI command execution).

The inter-unit synchronous module should be run as an interruption process, separate from regular scans. See ["4-13 Interrupts" for further details on the interruption process.

■ Regarding processing of the device when beginning and finishing

Saving/restoring of the items below will be conducted automatically when beginning/finishing the interunit synchronous module performance, in the same way as a regular interruption process.

- · Internal register
- Operation flag (CR2009-CR2012)
- Index register (Z11-Z23)
- TM0-TM3
- Relay stated in the operand of the STG command performed directly before



The index register Z01-Z10 can be configured to conduct a saving/restoring process with "Fixed cycle performance and user interruption settings" \rightarrow "CPU system settings".

Module Execution Sequence

You can set the execution sequence within a scan of the modules registered in a project. Normally, this is the order in which the modules are created. Only when the execution sequence within a scan of the ladder programs has an effect on operations should you change the sequence. From the workspace, open the "CPU system setting" dialog box, and then click "Program setting" and then "Execute sequence of modules."

Order	Module Name	Module Type	
1	Poslnit	Initialize	Range within which
2	HSPInit	Initialize	the order can be changed
3	Pos	Every-scan execution type	
4	LogInfoUpdate	Standby	
5	HSP	Every-scan execution type	Range within which
6	CommProc	Standby	the order can be changed
7	MainOpeProc	Every-scan execution type	

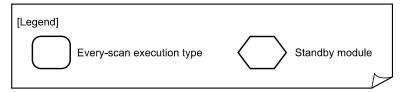
The following rules apply to setting the execution sequence of modules within a scan.

- You can change the order of initialize modules, but you cannot switch the order of initialize modules with those of other types of modules.
- You can change the order of every-scan execution type and standby modules, but you cannot move these modules ahead of the initialize modules in the sequence.

Module Configuration Examples

This section shows module configuration examples.

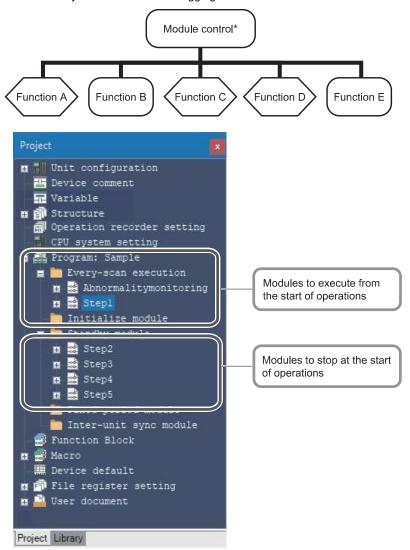
To simplify the explanations, the initialize modules are omitted.



Configuration by functions

Modules are created according to process details, connected devices, and expansion units. Use the every-scan execution type and standby modules differently depending on the execution conditions when operations start.

Stopping the unnecessary modules makes debugging easier and reduces scan time.

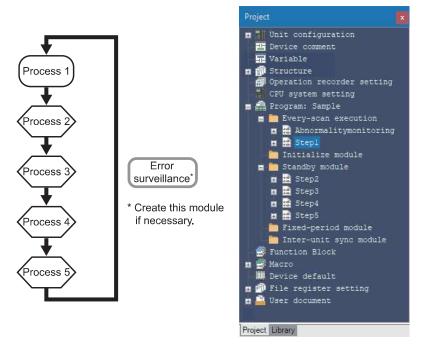


The execution status of each function is controlled by the "Project" module. Create the "Project" module if necessary.

Configuration by processes

Modules are created according to processes and operations.

In this example, the modules to be executed when operations start are created as every-scan execution type modules and all other modules are created as standby modules.



Reference

Because the module execution sequence is the execution sequence within a scan, there is also no need to consider the module execution sequence when creating programs by processes.

Programming without considering modules

When you create a new project, one every-scan execution type module with the name "Main" is created automatically, so you can create programs without considering modules.



Module System Devices

A module system device is a device that is used to capture the operation status of a module. Only the bit that is turned ON during execution can be referenced by other programs.



You cannot attach device comments to module system devices.



KV STUDIO automatically assigns module system devices to the CPU unit work area.

Bit that is turned ON for only one scan when module execution starts, "@CR2008"

This bit is turned ON for only the first scan after module execution starts.

From the second scan, it is turned OFF.

This bit can only be referenced from its own module. It cannot be referenced from the macros executed in its own module.

In KV STUDIO, this device is represented by "@CR2008."

• Bit that is turned OFF for only one scan when module execution starts,

"@CR2007"

This bit is turned OFF for only the first scan after module execution starts.

From the second scan, it is turned ON.

This bit can only be referenced from its own module. It cannot be referenced from the macros executed in its own module.

In KV STUDIO, this device is represented by "@CR2007."

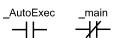
Bit that is turned ON during execution, "_(module name)"

This bit is turned ON when the module is being executed.

Because this bit can be referenced not only from its own module but from other programs as well, it is useful in situations such as when you want to control the execution of standby modules.

In KV STUDIO, modules are indicated by "_(module name)" (underscore + module name).

Example



Module Passwords

■ Module passwords

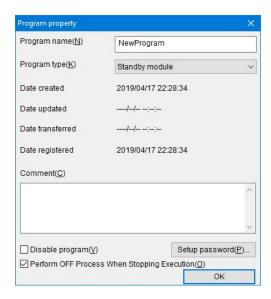
You can limit who can display modules in KV STUDIO by setting passwords on the modules. Follow the procedure below to set a password.

- Select the module to set a password for in the workspace.
- 2 Right-click, and then select "Property" in the menu that is displayed.

The "Program property" dialog box appears.



- Alternative procedure Select the module in the workspace, and select "Program" -> "Properties" from the menu.
 - · Select and right click the module in the workspace and select "Setup password" from the menu.



Click "Set password."

The "Set password" dialog box is displayed.



4 Enter the password in both text boxes (the second is for confirmation), and then click "OK."

After you set the password, when an attempt is made to display the module in KV STUDIO, a dialog box will be displayed asking the user to enter the password.

If the user does not enter the password, they will not be able to display the module in KV STUDIO.

LADDER PROGRAMMING

Module execution **MDSTRT** Module execution MDSTOP

stop

Stops the execution of the specified module

specified module

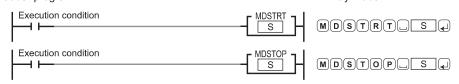
Starts the execution of the

Ladder program

MDSTRT

MDSTOP

Entry mode



Operation	Explanation	Occupied size
S	Specifies the module name.* 1	_

*1 Use of " ", #, \$ is not allowed. Directly enter the name of the module set on KV STUDIO.

The number of characters that can be entered is as follows.

- KV-8000(A) \ KV-7000 (CPU function version 2.0 or later) Up to 64 characters regardless of half-width or full-width
- KV-7000 (CPU function version lower than 2.0) Up to 24 characters regardless of half-width or full-width
- KV-5500/5000/3000/1000 KV Nano Series Up to 24 half-width alphanumerics, or up to 12 full-width alphanumerics

Operation Description

MDSTRT When the execution condition is ON, the module specified by S is started.

The specified module starts operating from the next scan.

Nothing happens on modules that are already started up.

MDSTOP When the execution condition is ON, the module specified by S is started.

Nothing happens on modules that are already stopped.

This instruction can also stop the self-module.

The OFF process when stopped differs according to the CPU unit. Refer to the instruction manual foreach CPU unit for details.



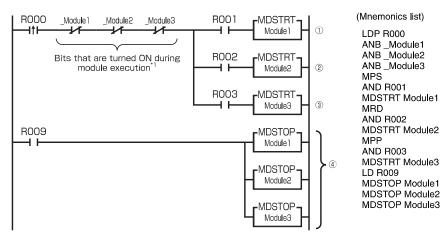
- The initial state of differential execution type instructions at start of module execution is ON when the rising edge is detected, and OFF when the falling edge is detected. Differentiation is never detected immediately when execution is started.
- When the MDSTRT and MDSTOP instructions have been executed multiple times in the samescan on the same module, execution follows the module instruction that was last executed.
- Do not stop all modules. All programs will not be executed when all modules have stopped. So, restore this state by one of the following methods. (Execution cannot be restored unless program execution is temporarily stopped.)
 - ◆Turn the CPU unit OFF then back ON again.
 - ♦Switch the RUN-PROG selector switch on the CPU unit to PROG then backto RUN again.
 - **◆**Turn the RUN mode of the program OFF then back ON again in KV STUDIO.
- Execution of the initialization module cannot be started or stopped.

Operation flags

CR2009	No change in status
CR2010	No change in status
CR2011	No change in status
CR2012	No change in status

Sample Program

When the rising edge of input relay R000 is detected, the module currently selected by mode switching (R001 to R003) is executed, and execution of all modules is stopped when input relay R009 turns ON. During execution of one of the modules, execution of other modules cannot be started.



- ① "Module 1" is executed when R001 is ON.
- (2) "Module 2" is executed when R002 is ON.
- (3) "Module 3" is executed when R003 is ON.
- (4) Module 1 to 3 is stopped when R009 is ON.

For details on the module execution ON bit, see "Modules" in the T "Module System Devices" (page 4-113).