MITSUBISHI ELECTRIC

SERVO SYSTEM CONTROLLER TECHNICAL BULLETIN [1 / 79]

[Issue No.] SSC-A-0001-B
[Title] Migration Guide of Motion Controller from Q17nDSCPU to RnMTCPU
[Date of Issue] November 2018 (Revised November 2026 Version B)
[Relevant Models] R64MTCPU, R32MTCPU, R16MTCPU, Q173DSCPU, Q172DSCPU

Thank you for your continued support of Mitsubishi Electric Servo System Controllers.
This bulletin describes guidance and precautions for migrating an existing system using Q173DSCPU/Q172DSCPU (hereinafter called Q17nDSCPU) to a system using R64MTCPU/R32MTCPU/R16MTCPU (hereinafter called RnMTCPU).

Point

When the MELSEC-Q series modules are replaced with the MELSEC iQ-R series, terminals and connectors may also need to be replaced. If no replacement device is available in the MELSEC iQ-R series, MELSEC-Q series modules can still be used with an RQ extension base. For details, refer to the "MELSEC iQ-R Module Configuration Manual (SH-081262ENG)" and user's manual of the module.
RnMTCPU does not support the virtual mode. For details, refer to "Replacement Virtual mode with Advanced synchronous control (L-03123)".
The information in this document is based on the product lineup as of June 2026, as well as the specifications of the Motion controllers (operating software Ver. AC for QnDSCPU and Ver. 32 for RnMTCPU) and engineering software MELSOFT MT Works2 (Ver. 1.210U).
The contents are subject to change without notice due to new product addition and specification change. Please refer to the latest product lineup and specifications when considering a migration.

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE : TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
NAGOYA WORKS : 1-14 , YADA-MINAMI 5-CHOME , HIGASHI-KU, NAGOYA , JAPAN

[Contents]

[Relevant Documents] 4
1. PREPARATIONS FOR MIGRATION 5
1.1. Controller Replacement 5
1.1.1. Correspondence table for components 5
1.1.2. External dimensions, mass, and installation 8
1.1.3. Operating system software 9
1.1.4. Engineering software 9
1.2. Replacement of Servo Amplifier, Servo Motor, and Servo System Network 10
1.2.1. Replacing MR-J4 series with MR-J5 series 10
1.2.2. Replacing MR-J3 series with MR-J5 series 10
1.2.3. Comparison of servo system networks 11
2. MIGRATION OF A SERVO SYSTEM 12
2.1. Case Study for Migration 12
2.2. Replacing MR-J4 Series with MR-J5 Series 13
2.2.1. Servo system with Q17nDSCPU and MR-J4-B (Before migration) 13
2.2.2. Servo system with RnMTCPU and MR-J5-B (After migration) 14
2.2.3. Whole system migration (Recommended) 15
2.2.4. Phased migration 16
2.2.5. Individual replacement 17
2.3. Replacing MR-J3 Series with MR-J5 Series 18
2.3.1. Servo system with Q17nDSCPU and MR-J3-B (Before migration) 18
2.3.2. Servo system with RnMTCPU and MR-J5-B (After migration) 19
2.3.3. Whole system migration (Recommended) 20
2.3.4. Phased migration 21
2.3.5. Individual replacement 23
3. DIFFERENCES BETWEEN Q17nDSCPU AND RnMTCPU 24
3.1. Table of the Differences 24
3.2. Error Codes System 31
4. PROJECT DIVERSION 34
4.1. List of Divertible/Not Divertible Data (SV13/SV22) 34
4.2. Project Diversion Procedures 36
4.2.1. Procedures for project diversion by MT Developer2 36
5. MIGRATION PROCEDURES 43
5.1. System Settings 43
5.1.1. External signal input module 46
5.1.2. Synchronous encoder interface module 48
5.1.3. Manual pulse generator interface module 50
5.1.4. Interrupt module 52
5.2. Programming 53
5.2.1. Acceleration/deceleration time/command torque time constant 53
5.2.2. Torque limit value settings 55
5.2.3. Torque limit value change request, Torque limit value change request instruction from PLC CPU to Motion CPU 56
5.2.4. Speed switching control 57
5.2.5. Write device data to CPU shared memory/Read device data from CPU shared memory 58

5.2.6. Synchronous control dedicated functions 59
5.2.7. Cam conversion/editing data 60
5.2.8. Motion SFC program (Y/N transition) 61
5.2.9. Input device (X) 62
5.2.10. SM/SD device automatic refresh 64
5.2.11. RUN/STOP 64
5.2.12. Common bit device SET/RST request register 65
5.2.13. CPU buffer memory (fixed scan communication area) 66
5.3. High-Speed Input Request Signal, Mark Detection Signal 68
5.3.1. High-speed input request signal 68
5.3.2. Mark detection function 69
6. DEVICE COMPARISON 70
6.1. Internal Relays 70
6.2. Data Register 70
6.3. Motion Register 71
6.4. Special Relay 71
6.5. Special Register 74
6.6. CPU Buffer Memory (CPU Shared Memory) 78
7. REVISIONS 79

[Relevant Documents]

Refer to the following relevant documents for the replacement.

(1) Motion controller

Manual title	Manual No.
MELSEC iQ-R Motion Controller User's Manual	IB-0300235
MELSEC iQ-R Motion Controller Programming Manual (Common)	IB-0300237
MELSEC iQ-R Motion Controller Programming Manual (Program Design)	IB-0300239
MELSEC iQ-R Motion Controller Programming Manual (Positioning Control)	IB-0300241
MELSEC iQ-R Motion Controller Programming Manual (Advanced Synchronous Control)	IB-0300243
<Motion controller> Replacement Virtual mode with Advanced synchronous control	L-03123
R□MTCPU Before Using the Product	BCN-B62008-342

(2) PLC

Manual title	Manual No.
MELSEC iQ-R Module Configuration Manual	SH-081262ENG
MELSEC iQ-R CPU Module User's Manual (Startup)	SH-081263ENG
MELSEC iQ-R CPU Module User's Manual (Application)	SH-081264ENG
MELSEC-Q Series to MELSEC iQ-R Series Migration Guide	L-08510ENG

(3) Servo amplifier

Manual title	Manual No.
Transition from MELSERVO-J4 Series to J5 Series Handbook	L-03201ENG
Transition from MELSERVO-J3/J3W Series to J4 Series Handbook	L-03127ENG
MR-J5-B/MR-J5W-B User's Manual (Introduction)	IB-0300578ENG
MR-J4-_B_(-RJ) Servo Amplifier Instruction Manual	SH-030106ENG
MR-J4W2-_B/MR-J4W3-_B/MR-J4W2-0303B6 Servo Amplifier Instruction Manual	SH-030105ENG
MR-J5 Safety Instructions and Precautions for AC Servos	IB-0300391
MR-J4 Instructions and Cautions for Safe Use of AC Servos	IB-0300175

(4) Engineering software

Manual title	Manual No.
GX Works3 Operating Manual	SH-081215ENG
GX Works2 Version 1 Operating Manual (Common)	SH-080779ENG

1. PREPARATIONS FOR MIGRATION

Prepare the modules, operating system software, and engineering software based on the tables in this chapter.

1.1. Controller Replacement

1.1.1. Correspondence table for components

Item			Before migration Q17nDSCPU	After migration RnMTCPU
			Model	Model
Main base unit			Q3_DB	R3_B
Power supply module			Q6_P	R6_P
Extension base unit			Q6_B	R6_B
Extension cable			QC_B	RC_B
PLC CPU module	Universal model QCPU		Q_U_CPU	R_CPU
	Universal model QnUDVCPU (High-speed type CPU)		Q_UDVCPU	
Motion controller			Q172DSCPU	R16MTCPU
			Q173DSCPU	R32MTCPU R64MTCPU
Input module	AC input	100 to 120 V AC	QX10(-TS)	RX10(-TS)
		100 to 240 V AC	QX28	RX28
	DC input (Positive common)	24 V DC	QX40(-S1)(-TS) QX41(-S1) QX42(-S1)	RX40C7(-TS) *1 RX41C4 *1 RX42C4 *1
			QX41-S2	RX41C6HS *1
	DC input (Negative common)	24 V DC	QX80(-TS) QX81 QX82(-S1)	RX40C7(-TS) *1 RX41C4 *1 RX42C4 *1
			QX81-S2	RX41C6HS *1
	DC input (Positive common/ negative common shared)	5 to 12 V DC	QX70 QX71 QX72	-
	DC high-speed input (Positive common)	24 V DC	QX40H	RX40PC6H
		5 V DC	QX70H	RX61C6HS *1
	DC high-speed input (Negative common)	24 V DC	QX80H	RX40NC6H
		5 V DC	QX90H	RX61C6HS *1
	DC/AC input	48 V DC/ 48 V AC	QX50	-

[Continued]

Item			Before migration Q17nDSCPU	After migration RnMTCPU
			Model	Model
Output module	Relay output	24 V DC, 240 V AC	QY10(-TS) QY18A	RY10R2(-TS) RY18R2A
	Triac output	100 to 240 V AC	QY22	RY20S6
	Transistor output (Sink type)	12 to 24 V DC	QY40P(-TS) QY41P QY42P QY50	RY40NT5P(-TS) RY41NT2P(-TS) RY42NT2P RY40NT5P(-TS)
		5 to 12 V DC	QY70	-
			QY71	RY41NT2H
	Transistor output (Source type)	12 to 24 V DC	QY80(-TS) QY81P QY82P	RY40PT5P(-TS) RY41PT1P RY42PT1P
	Transistor high-speed output (Sink type)	5 to 24 V DC	QY41H	RY41NT2H
	Transistor output (All points independent)	5 to 24 V DC	QY68A	-
I/O combined module	DC input/ transistor output	Input: 24 V DC Output: 12 to 24 V DC	QH42P	RH42C4NT2P
			QX41Y41P QX48Y57	-
Analog input module		Voltage/current input	Q64AD(H)	R60AD(H)4
		Voltage input	Q68ADV	R60ADV8
		Current input	Q68ADI	R60ADI8
Channel isolated analog input module		Voltage/current input	Q64AD-GH	-
			Q64ADH	R60AD8-G
		Current input	Q62AD-DGH	-
			Q66AD-DG	-
Analog output module		Voltage/current output	Q62DA(N) Q64DA(N)	R60DA4
			Q64DAH	R60DAH4
		Voltage output	Q68DAV(N)	R60DAV8
		Current output	Q68DAI(N)	R60DAI8
Channel isolated analog output module		Voltage/current output	Q62DA-FG	-
			Q66DA-G	R60DA8-G
Analog I/O module		Voltage/current input and output	Q64AD2DA	-
Interrupt module			QI60	RX40C7 *1
Servo external signals interface module			Q172DLX	Input module such as RX41C4
Synchronous encoder input module			Q172DEX	- (via MR-J5-B-RJ)

[Continued]

Item		Before migration Q17nDSCPU	After migration RnMTCPU
		Model	Model
Manual pulse generator input module		Q173DPX	RD62D2 (Differential-input type, 2ch) *2 RD62P2 (DC input, 2ch) *2 RD62P2E (DC input, 2ch) *2
Serial absolute synchronous encoder		Q170ENC Q171ENC-W8	HK series rotary servo motor (Connected with MR-J5-B-RJ) *3
Cable for serial absolute synchronous encoder		Q170ENCCBL_M	
Manual pulse generator		MR-HDP01	MR-HDP01 *4
Safety signal module		Q173DSXY	-
Cable for forced stop input		Q170DEMICBL_M	Not required
Connector for forced stop input cable		Q170DEMICON	
Battery unit		Motion controller built-in battery	Not required
Battery	For Motion controller	Q6BAT	Not required
	For synchronous encoder	A6BAT	
Optical hub unit		MR-MV200	MR-MV200
SSCNET III cable		MR-J3BUS_M MR-J3BUS_M-A MR-J3BUS_M-B	MR-J3BUS_M MR-J3BUS_M-A MR-J3BUS_M-B

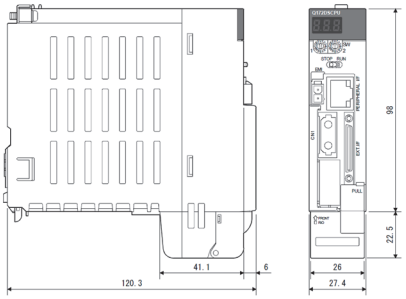
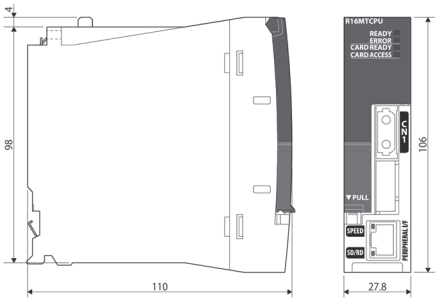
*1 This module is positive/negative common shared type.
*2 Connect this module to an external power supply separately.
*3 HK series rotary servo motors can operate as a synchronous encoder when connected to an MR-J5-B-RJ servo amplifier.
*4 When using the manual pulse generator with RnMTCPU, be sure to provide a separate power supply.
Mitsubishi Electric has confirmed the operation of the following manual pulse generator.

Product name	Model name	Description	Manufacturer
Manual type rotary encoder	UFO-M2-0025-2Z1-B00E	Number of pulses per revolution: 25 pulse/rev (100 pulse/rev per after magnification by 4)	Nemicon Corporation

Point

- The RnMTCPU only controls MELSEC iQ-R series modules. It cannot control MELSEC-Q series modules.
- The RnMTCPU does not have an EMI terminal for the forced stop input. When the existing model executes forced stop by using the forced stop cable, input the forced stop signal to an input module, and set the input module device No. in the forced stop input settings ([Motion CPU Common Parameter] → [Basic Setting]).
- RnMTCPU is battery-less.
- Select the power supply module after estimating the system current consumption. The current consumption can be estimated on the "FA Integrated Selection Tool" on the Mitsubishi Electric FA global website.
- RnMTCPU consumes more current than Q17nDSCPU, and therefore fewer RnMTCPU controllers can be connected to each power supply module.
- If the current capacity of the power supply module is insufficient after migration, separate the system using the extension base unit (R6_B).
- For details on the alternative devices, refer to "MELSEC-Q Series to MELSEC iQ-R Series Migration Guide (L-08510ENG)".

1.1.2. External dimensions, mass, and installation

Item	Q17nDSCPU	RnMTCPU
External dimensions		
Exterior dimensions [mm]	120.5 [H] × 27.4 [W] × 120.3 [D]	106.0 [H] × 27.8 [W] × 110.0 [D]
Mass [kg]	0.38	0.28
Consumption current 5 V DC [A]	1.44 (Q172DSCPU), 1.75 (Q173DSCPU)	1.20

Standard base unit dimensions (H × W × D [mm])

Series	_33B	_35B	_38B	_312B
MELSEC-Q	98.0 × 189 × 44.1	98.0 × 245 × 44.1	98.0 × 328 × 44.1	98.0 × 439 × 44.1
MELSEC iQ-R	101 × 189 × 32.5	101 × 245 × 32.5	101 × 328 × 32.5	101 × 439 × 32.5

Point

- When the number of slots is the same, the mounting hole positions of the base unit are compatible.
  For details on the installation method, refer to "MELSEC iQ-R Module Configuration Manual (SH-081262ENG)".
  For compliance with the EMC and Low Voltage Directives, refer to "EMC and Low Voltage Directives" in the above manual.

1.1.3. Operating system software

Use the operating system software (OS) for RnMTCPU.
For Q17nDSCPU, the OS software is available by application. For RnMTCPU, as the OS software is integrated, use SW10DNC-RMTFW.

Q17nDSCPU				RnMTCPU	
OS Type	Motion CPU model	Operation system software model		Motion CPU model	Operation system software model
SV13	Q173DSCPU	SW8DNC-SV13QJ		R64MTCPU R32MTCPU R16MTCPU	SW10DNC-RMTFW (installed before shipment)
	Q172DSCPU	SW8DNC-SV13QL			
SV22	Q173DSCPU	SW8DNC-SV22QJ			
	Q172DSCPU	SW8DNC-SV22QL			

1.1.4. Engineering software

Product name	Model	Version	Remark
MELSOFT MT Works2	SW1DND-MTW2-EC	Ver. 1.210U or later	(Note-1), (Note-2)
MELSOFT MR Configurator2	SW1DND-MRC2-EC	Ver. 1.165X or later	(Note-3)

(Note-1): The model name has been changed because it has become available as DVD.
(Note-2): Use MELSOFT GX Works3 to set the PLC CPU and the R series CPU common parameters.
(Note-3): Supported servo amplifier functions and compatible servo motors vary depending on the version of MELSOFT MR Configurator2. Use the latest version.

1.2. Replacement of Servo Amplifier, Servo Motor, and Servo System Network

1.2.1. Replacing MR-J4 series with MR-J5 series

Select an MR-J5 series servo amplifier and a servo motor connectable to the selected servo amplifier. The servo system network does not need to be changed. Use SSCNET III/H. For details on the transition, refer to "Transition from MELSERVO-J4 Series to J5 Series Handbook (L-03201ENG)".

Q17nDSCPU			RnMTCPU	
Servo amplifier MR-J4 series	MR-J4-_B1	→	Servo amplifier MR-J5 series	MR-J5-_B + MR-CM08K1
	MR-J4-_B1-RJ			MR-J5-_B-RJ + MR-CM08K1
	MR-J4-_B			MR-J5-_B
	MR-J4-_B-RJ			MR-J5-_B-RJ
Rotary servo motor	HG series		Rotary servo motor	HK series

Point

The HK series rotary servo motors have a built-in battery-less absolute position encoder and therefore do not require a battery.

1.2.2. Replacing MR-J3 series with MR-J5 series

Select an MR-J5 series servo amplifier and a servo motor connectable to the selected servo amplifier. Change the servo system network from SSCNET III to SSCNET III/H.
For details on the transition, refer to "Transition from MELSERVO-J3/J3W Series to J4 Series Handbook (L-03127ENG)" and "Transition from MELSERVO-J4 Series to J5 Series Handbook (L-03201ENG)".

Q17nDSCPU			RnMTCPU	
Servo amplifier MR-J3 series	MR-J3-_B1	→	Servo amplifier MR-J5 series	MR-J5-_B + MR-CM08K1
	MR-J3-_B1-RJ			MR-J5-_B-RJ + MR-CM08K1
	MR-J3-_B			MR-J5-_B
	MR-J3-_B-RJ			MR-J5-_B-RJ
Rotary servo motor	HF/HC/HA series		Rotary servo motor	HK series
Servo system network	SSCNET III		Servo system network	SSCNET III/H

Point

The HK series rotary servo motors have a built-in battery-less absolute position encoder and therefore do not require a battery.

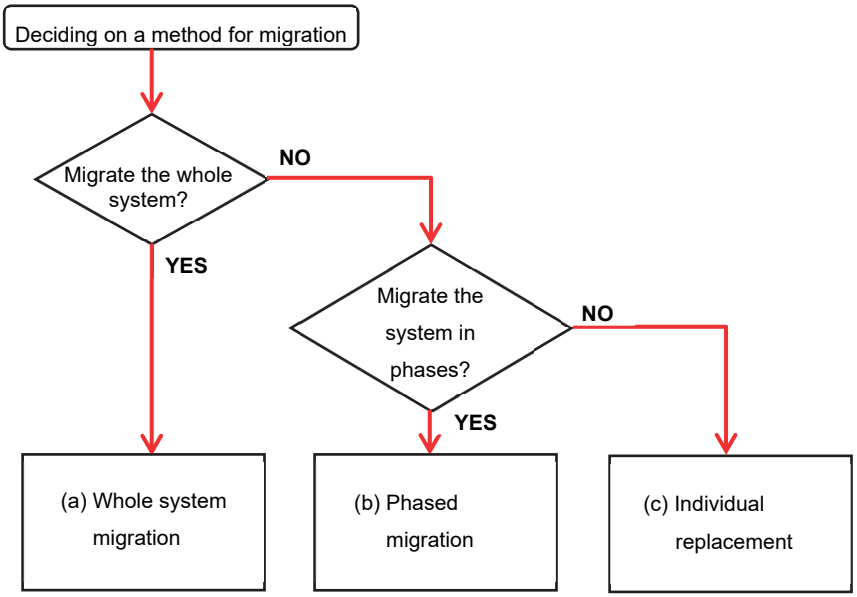
1.2.3. Comparison of servo system networks

Item		SSCNET III	SSCNET III/H
Motion controller		Q17nDSCPU (MR-J3 series connected) RnMTCPU (MR-J3 series connected)	Q17nDSCPU (MR-J4/J5 series connected) RnMTCPU (MR-J4/J5 series connected)
Communication medium		Optical fiber cable	Optical fiber cable
Communication speed		50 Mbps	150 Mbps
Communication cycle	Send	0.44 ms / 0.88 ms	0.22 ms / 0.44 ms / 0.88 ms
	Receive	0.44 ms / 0.88 ms	0.22 ms / 0.44 ms / 0.88 ms
Max. number of control axes per line		Communication cycle 0.44 ms: 8 axes/line Communication cycle 0.88 ms: 16 axes/line	Communication cycle 0.22 ms: 4 axes/line Communication cycle 0.44 ms: 8 axes/line Communication cycle 0.88 ms: 16 axes/line
Transmission distance	Standard code for inside panel	Up to 3 m between stations Maximum overall distance: 48 m (3 m × 16 axes)	Up to 3 m between stations Maximum overall distance: 48 m (3 m × 16 axes)
	Standard cable for outside panel	Up to 20 m between stations Maximum overall distance: 320 m (20 m × 16 axes)	Up to 20 m between stations Maximum overall distance: 320 m (20 m × 16 axes)
	Long-distance cable	Up to 50 m between stations Maximum overall distance: 800 m (50 m × 16 axes)	Up to 100 m between stations Maximum overall distance: 1600 m (100 m × 16 axes)

2. MIGRATION OF A SERVO SYSTEM

2.1. Case Study for Migration

The following is a case study that describes the methods for migrating a system.



(a) Whole system migration (Recommended)

This method involves replacing the controller, servo amplifiers, and servo motors simultaneously.
When an MR-J3 series is replaced with an MR-J5 series, the network will also be updated.
When using this method, refer to section 2.2.3 for replacing MR-J4 with MR-J5, and section 2.3.3 for replacing MR-J3 with MR-J5.

(b) Phased migration

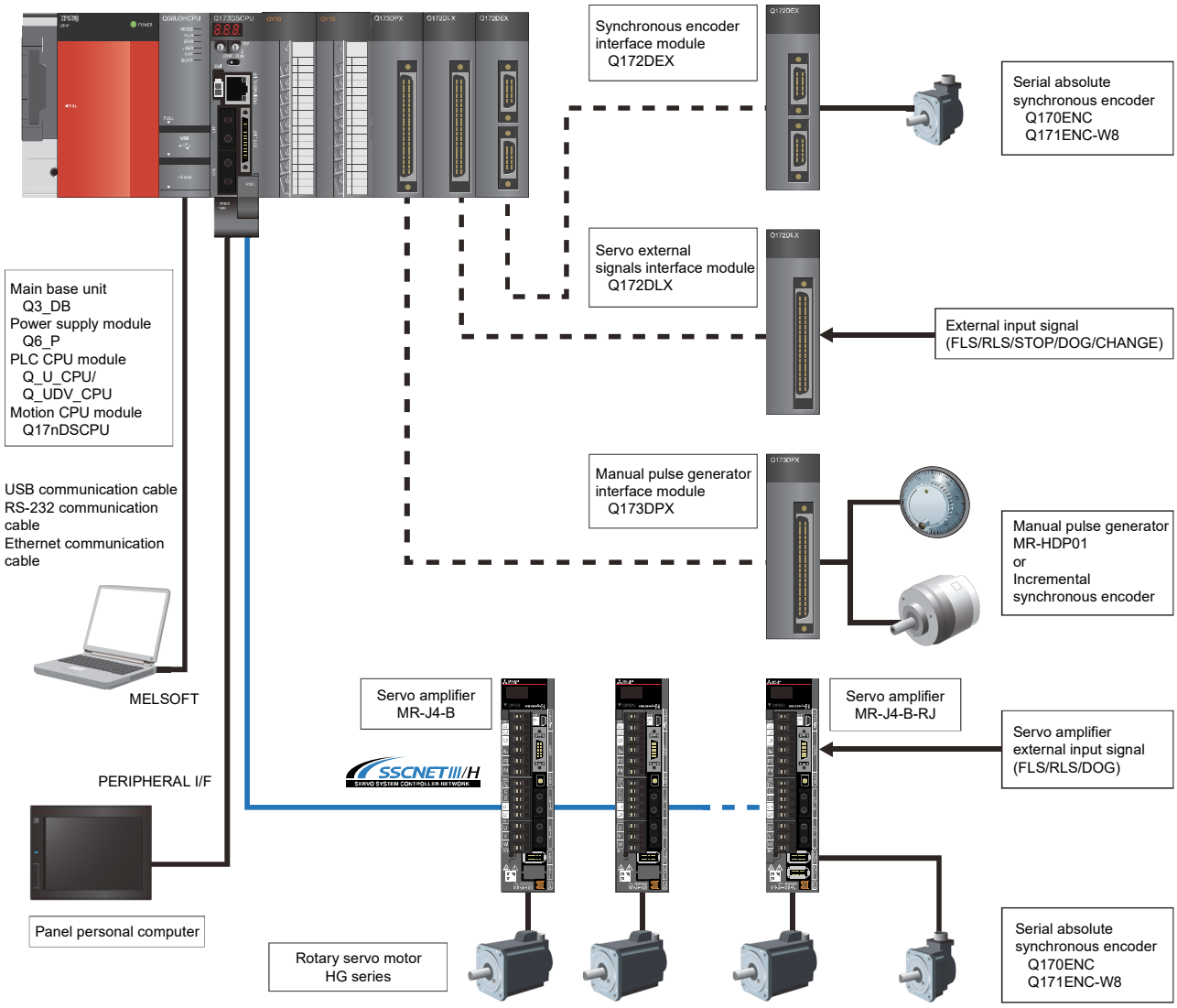
This method involves replacing the controller, servo amplifiers, and servo motors in phases.
Phased migration can be useful when a whole system migration is difficult due to installation timing or cost constraints.
When using this method, refer to section 2.2.4 for replacing MR-J4 with MR-J5, and section 2.3.4 for replacing MR-J3 with MR-J5.

(c) Individual replacement

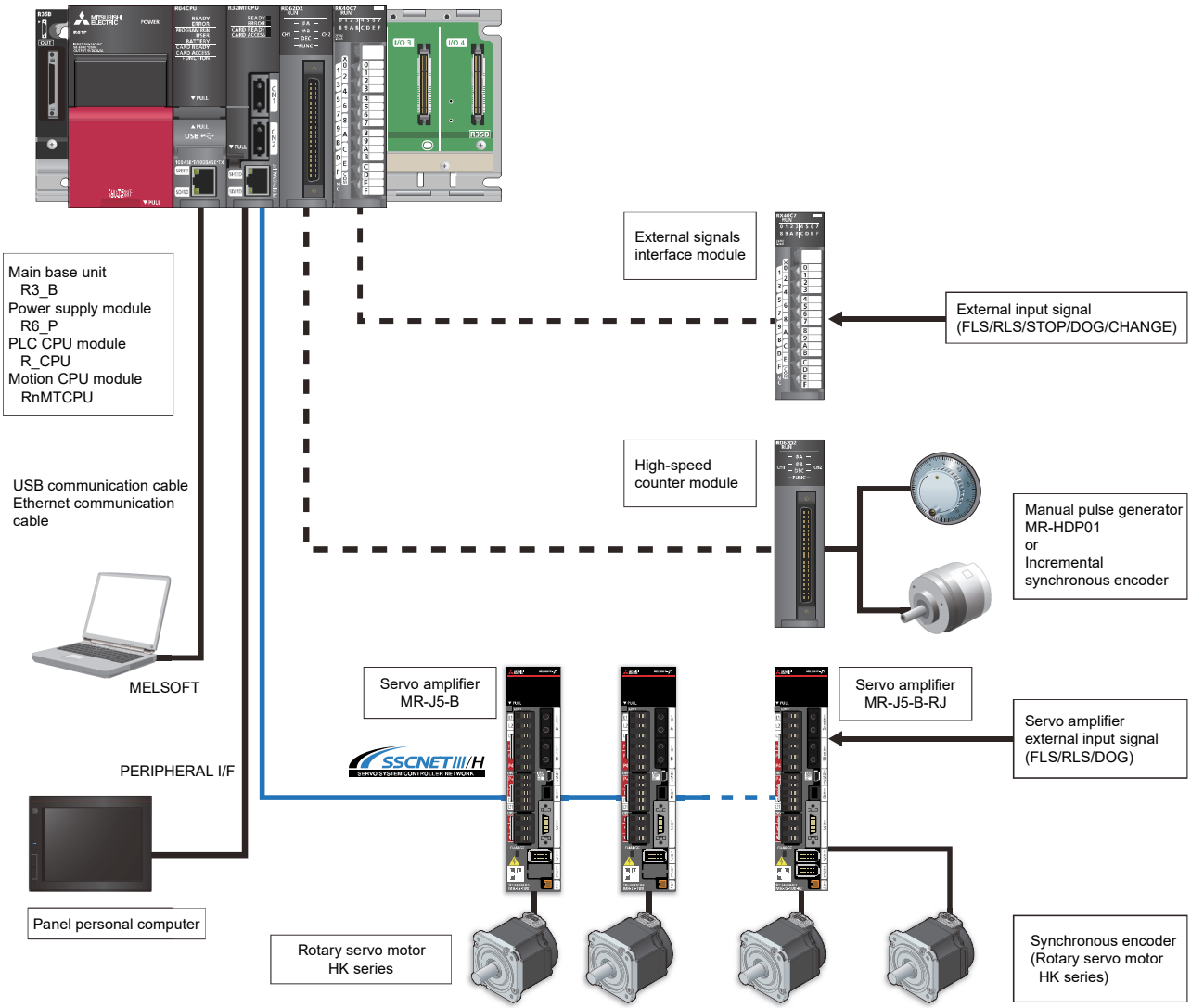
This method involves only replacing the devices in the servo system that have failed.
When using this method, refer to section 2.2.5 for replacing MR-J4 with MR-J5, and section 2.3.5 for replacing MR-J3 with MR-J5.

2.2. Replacing MR-J4 Series with MR-J5 Series

2.2.1. Servo system with Q17nDSCPU and MR-J4-B (Before migration)

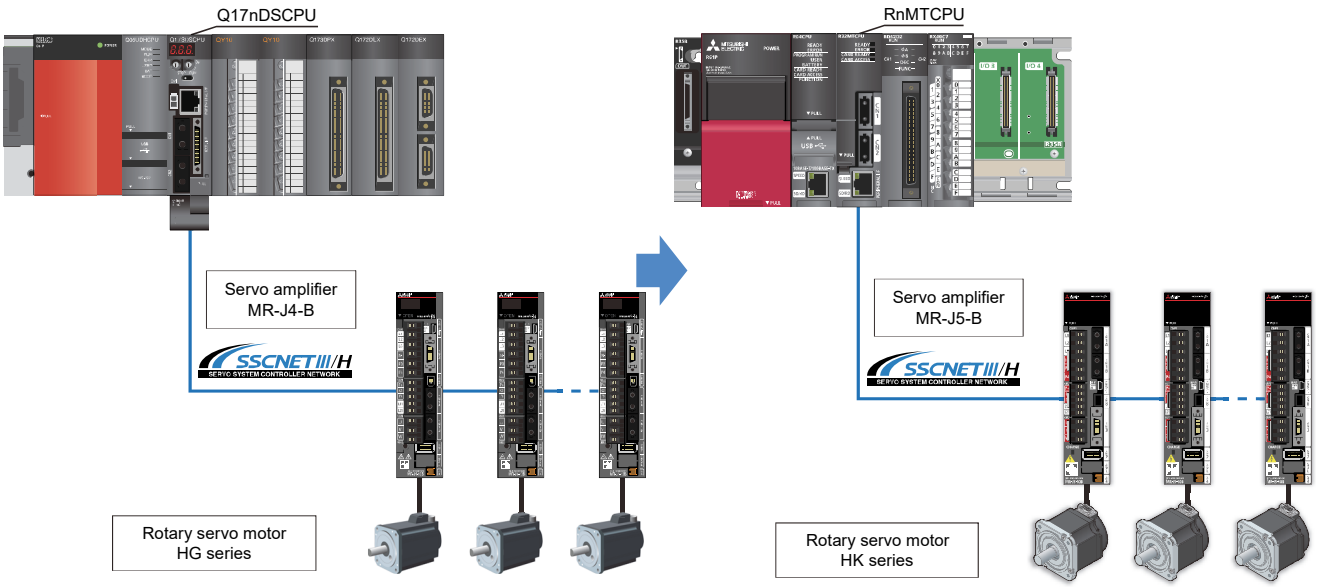


2.2.2. Servo system with RnMTCPU and MR-J5-B (After migration)



2.2.3. Whole system migration (Recommended)

The following shows the procedure for migrating a whole system.



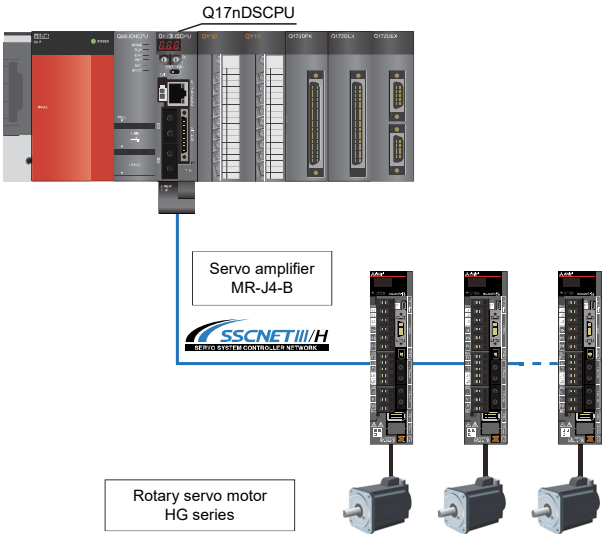
[System changes]

Product name	Model before migration		Model after migration
Main base unit	Q3_DB		R3_B
Power supply module	Q6_P		R6_P
PLC CPU module	Q_U_CPU/QnUDVCPU		R_CPU
Motion CPU module	Q172DSCPU Q173DSCPU		R16MTCPU R32MTCPU/R64MTCPU
Servo external signals interface module	Q172DLX		RX41C4, etc.
Synchronous encoder interface module	Q172DEX		- (via MR-J5-B-RJ)
Manual pulse generator interface module	Q173DPX		RD62D2/RD62P2/RD62P2E
Servo amplifier	MR-J4-B		MR-J5-B
Rotary servo motor	HG series		HK series

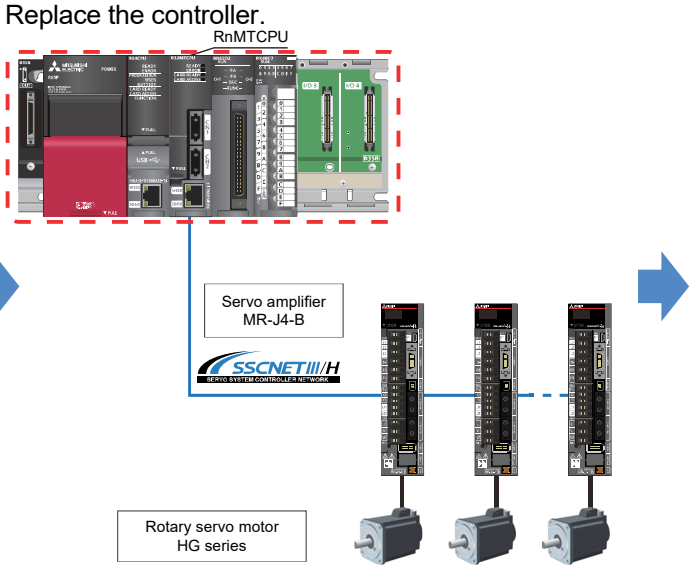
2.2.4. Phased migration

The following shows the procedure for a phased migration, where the controller is replaced with RnMTCPU in the first phase, and the MR-J4-B servo amplifiers are gradually replaced with MR-J5-B in the subsequent phases.

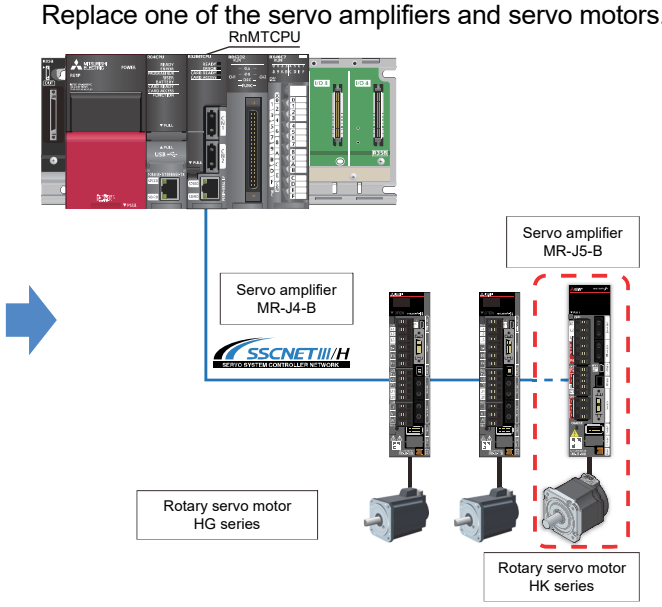
[Before migration]



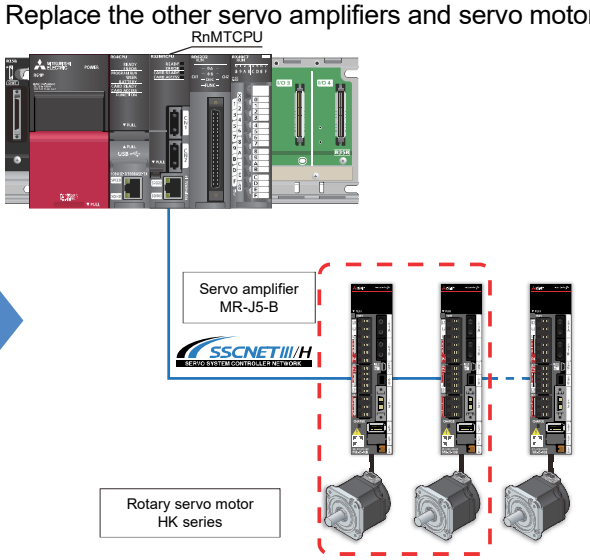
[Step 1]
Replace the controller.



[Step 2]
Replace one of the servo amplifiers and servo motors.



[Step 3]
Replace the other servo amplifiers and servo motors.



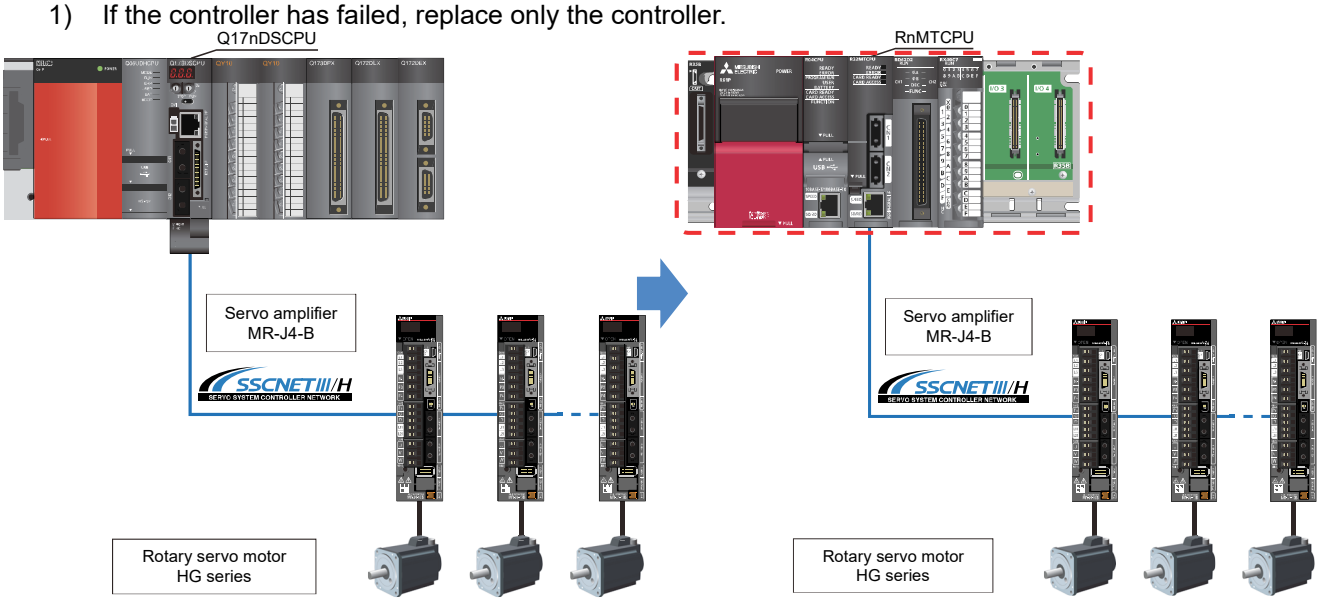
Point

- Q17nDSCPU/RnMTCPU can be connected to MR-J4-B and MR-J5-B. The servo amplifiers can also be replaced before replacing the controller.
- MR-J4-B and MR-J5-B can be used on the same network. However, use servo amplifiers from the same series for axes that require high-precision synchronization at the machine side across multiple axes, such as the axes used for interpolation control and synchronous control.
- Only HG series rotary servo motors can be connected to MR-J4-B.
- Both HG series and HK series rotary servo motors can be connected to MR-J5-B.

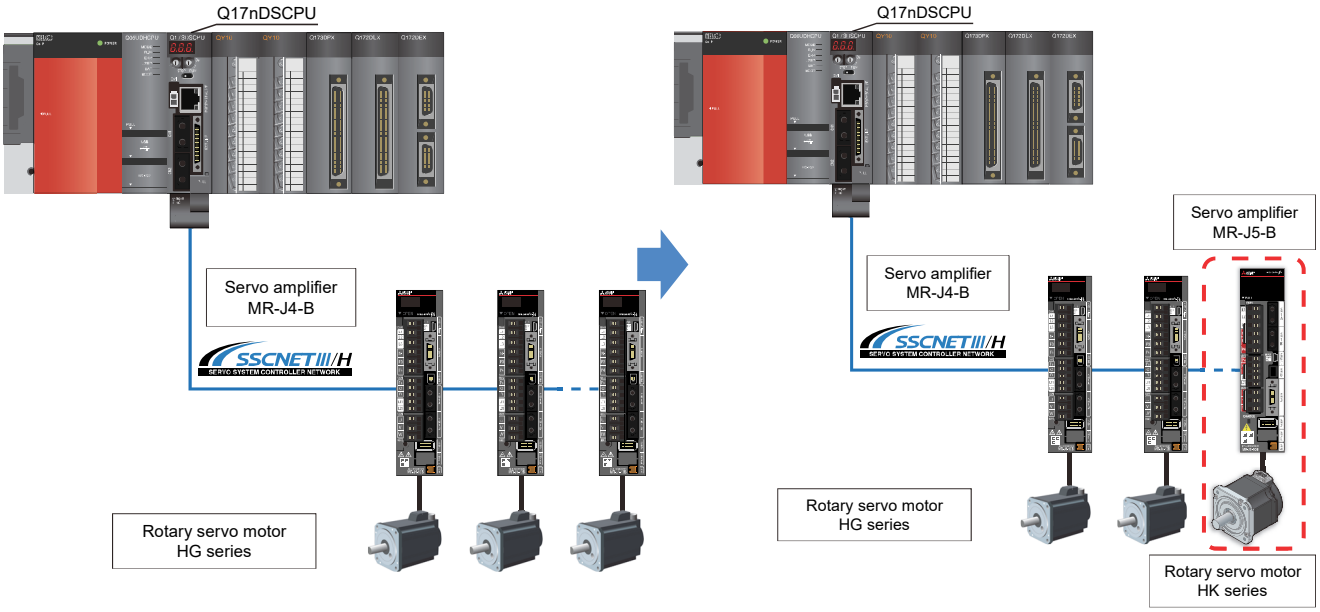
2.2.5. Individual replacement

The following shows the procedure for replacing individual devices when they have failed.

1) If the controller has failed, replace only the controller.



2) If any of the servo amplifiers or servo motors have failed, replace the failed servo amplifiers and servo motors.

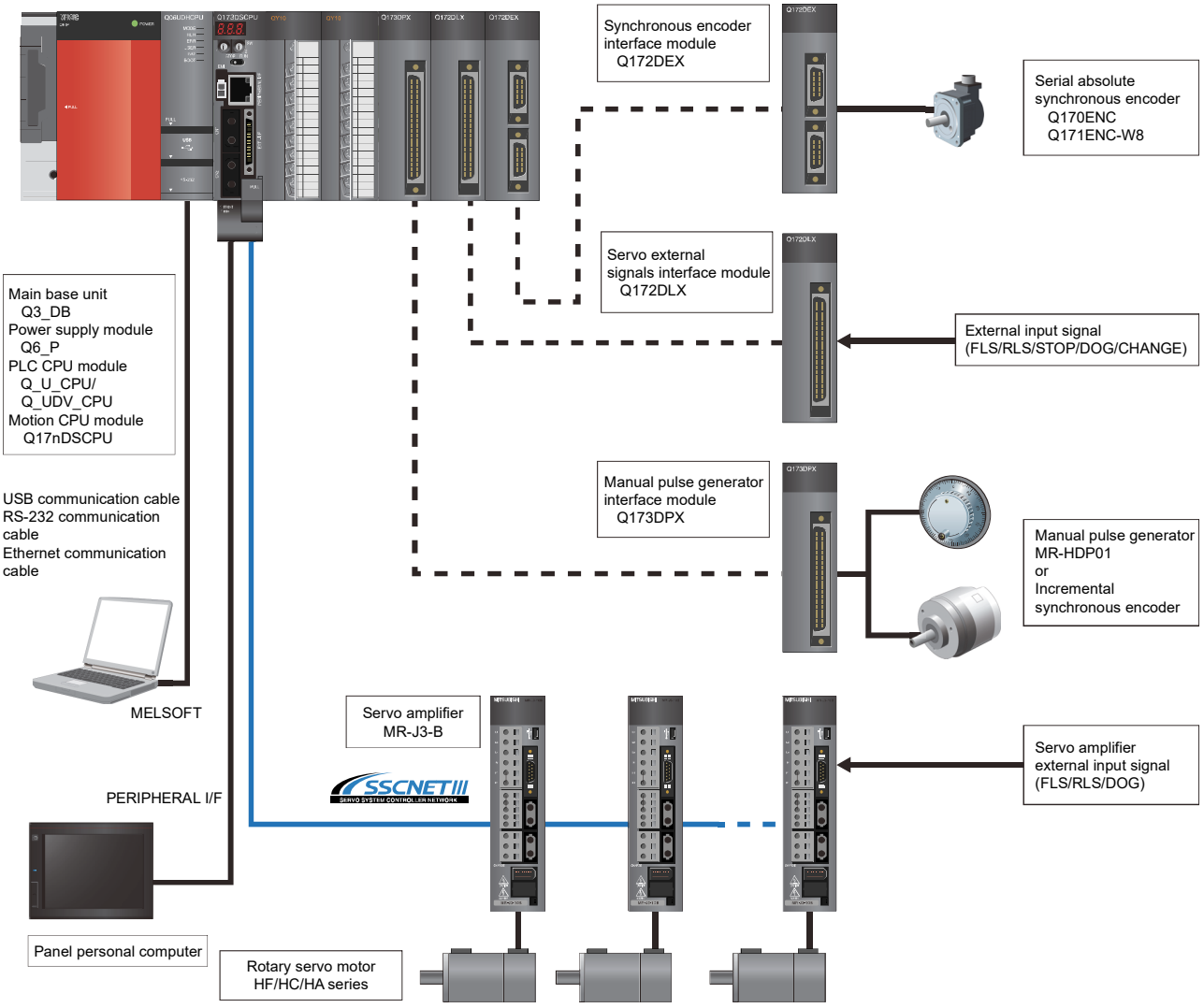


Point

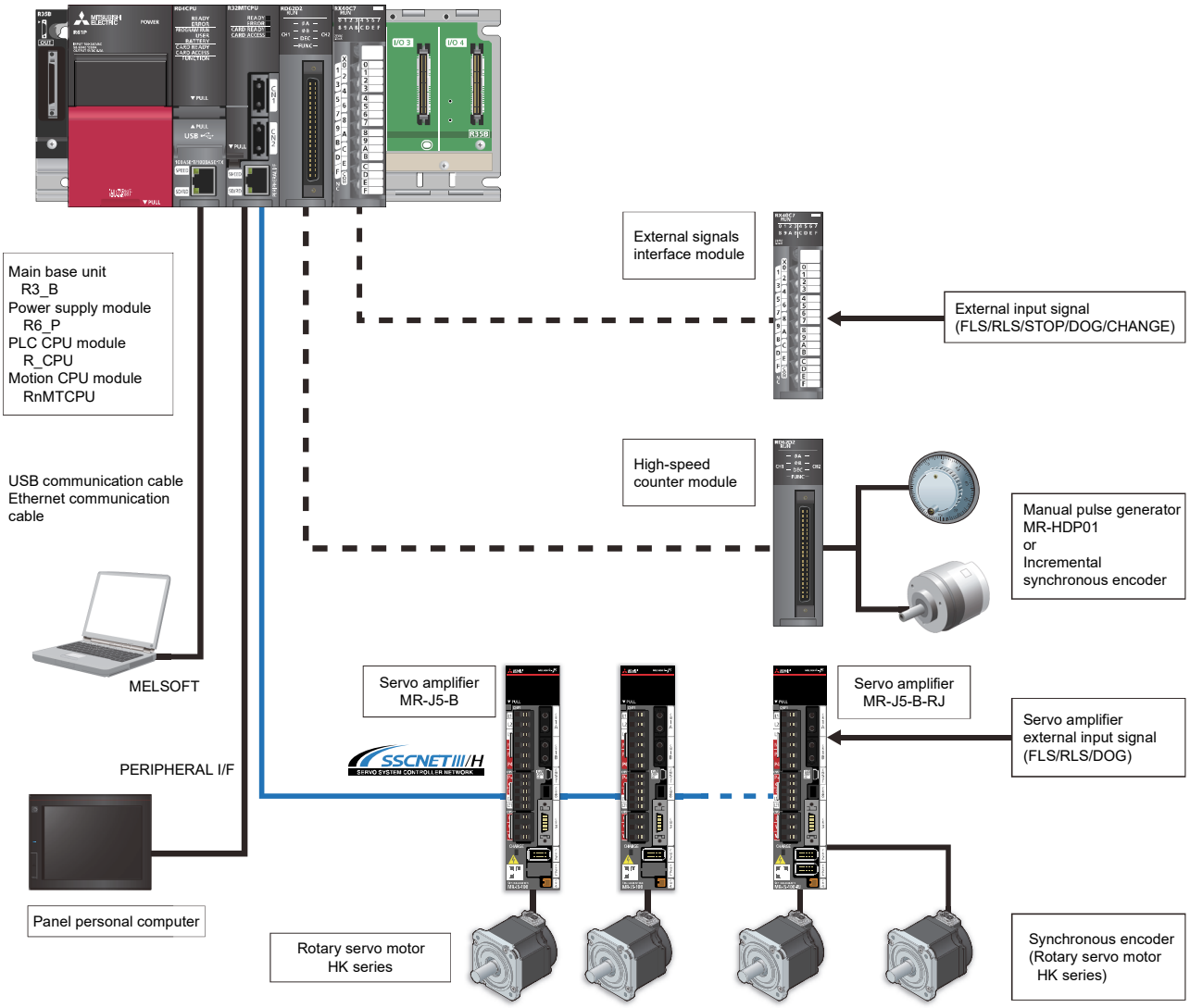
- Q17nDSCPU/RnMTCPU can be connected to MR-J4-B and MR-J5-B.
- MR-J4-B and MR-J5-B can be used on the same network. However, use servo amplifiers from the same series for axes that require high-precision synchronization at the machine side across multiple axes, such as the axes used for interpolation control and synchronous control.
- Only HG series rotary servo motors can be connected to MR-J4-B.
- Both HG series and HK series rotary servo motors can be connected to MR-J5-B.

2.3. Replacing MR-J3 Series with MR-J5 Series

2.3.1. Servo system with Q17nDSCPU and MR-J3-B (Before migration)

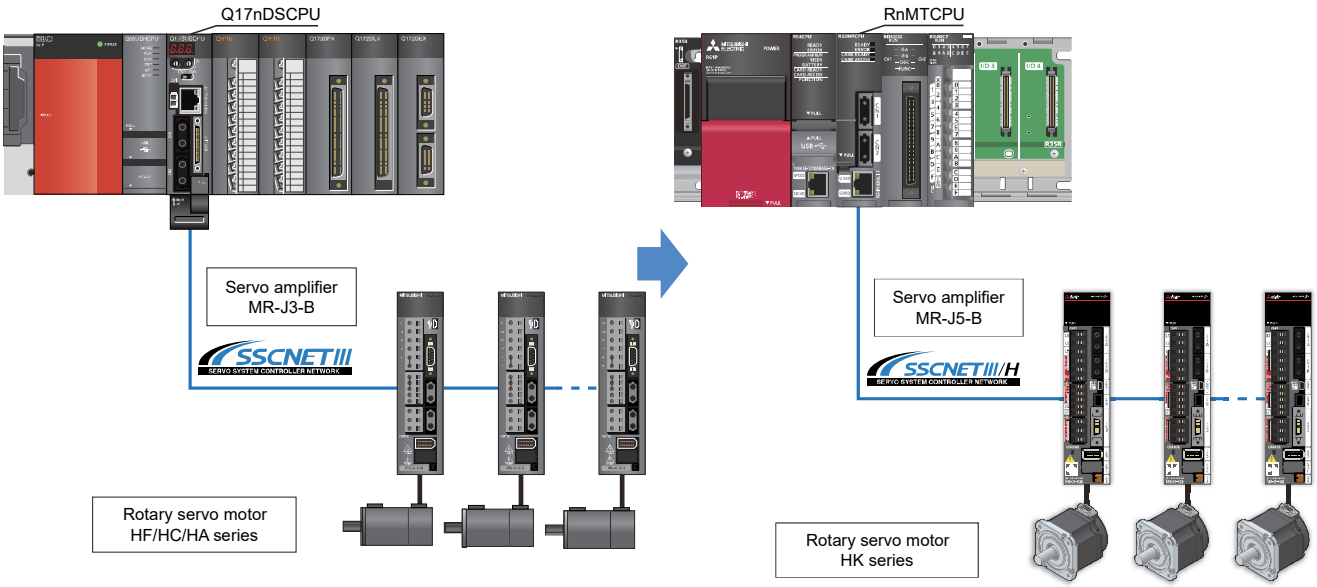


2.3.2. Servo system with RnMTCPU and MR-J5-B (After migration)



2.3.3. Whole system migration (Recommended)

The following shows the procedure for migrating a whole system.



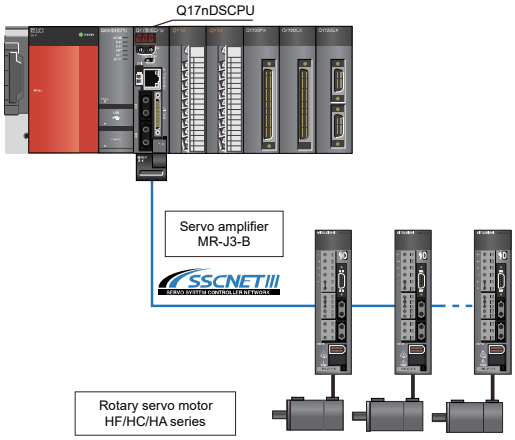
[System changes]

Product name	Model before migration		Model after migration
Main base unit	Q3_DB		R3_B
Power supply module	Q6_P		R6_P
PLC CPU module	Q_U_CPU/QnUDVCPU		R_CPU
Motion CPU module	Q172DSCPU Q173DSCPU		R16MTCPU R32MTCPU/R64MTCPU
Servo external signals interface module	Q172DLX		RX41C4, etc.
Synchronous encoder interface module	Q172DEX		- (via MR-J5-B-RJ)
Manual pulse generator interface module	Q173DPX		RD62D2/RD62P2/RD62P2E
Servo amplifier	MR-J3-B		MR-J5-B
Rotary servo motor	HF/HC/HA series		HK series

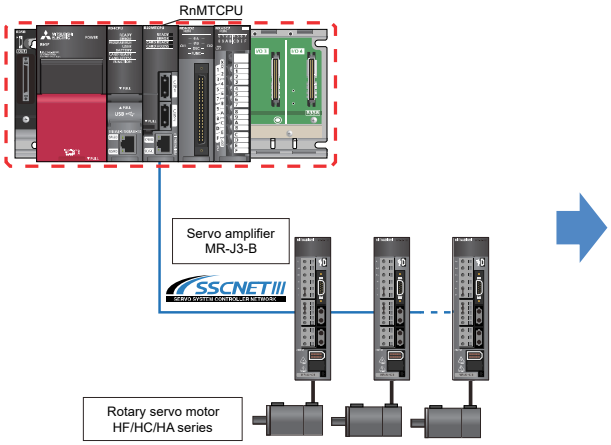
2.3.4. Phased migration

The following shows the procedure for a phased migration, where the controller is replaced with RnMTCPU in the first phase, and the MR-J3-B servo amplifiers are gradually replaced with MR-J5-B in the subsequent phases.

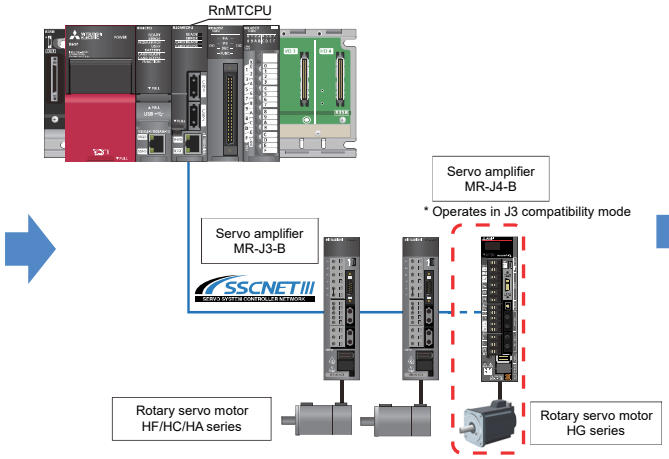
[Before replacement]



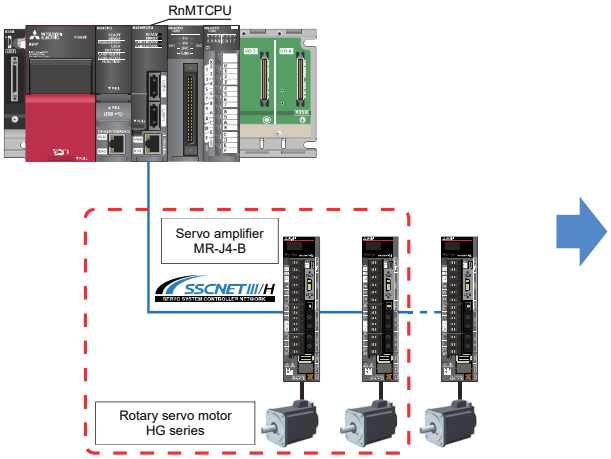
[Step 1]
Replace the controller.



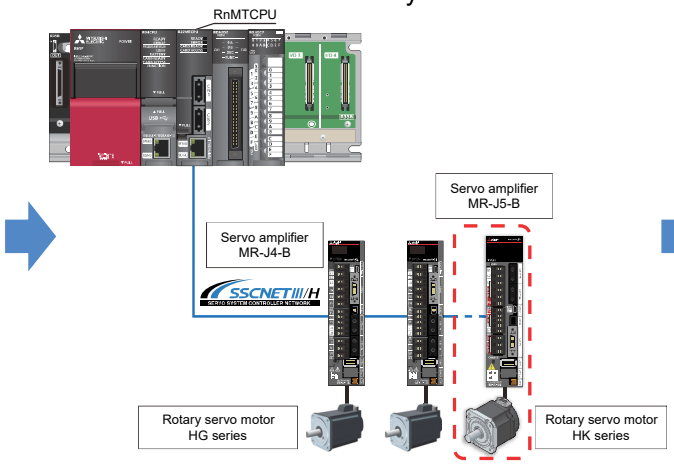
[Step 2]
Replace one of the servo amplifiers with an MR-J4-B servo amplifier and one of the rotary servo motors with an HG series rotary servo motor. Operate the MR-J4-B in J3 compatibility mode.



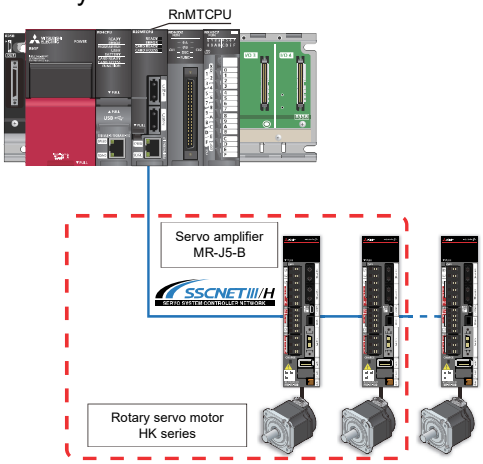
[Step 3]
Replace the other servo amplifiers with MR-J4-B servo amplifiers and the servo motors with HG series rotary servo motors. Set the servo system network to SSCNET III/H and operate all the servo amplifiers in J4 mode.



[Step 4]
Replace one of the MR-J4-B servo amplifiers with an MR-J5-B servo amplifier, and one of the servo motors with an HK series rotary servo motor.



[Step 5]
Replace the other servo amplifiers with MR-J5-B servo amplifiers, and the other servo motors with HK series rotary servo motors.



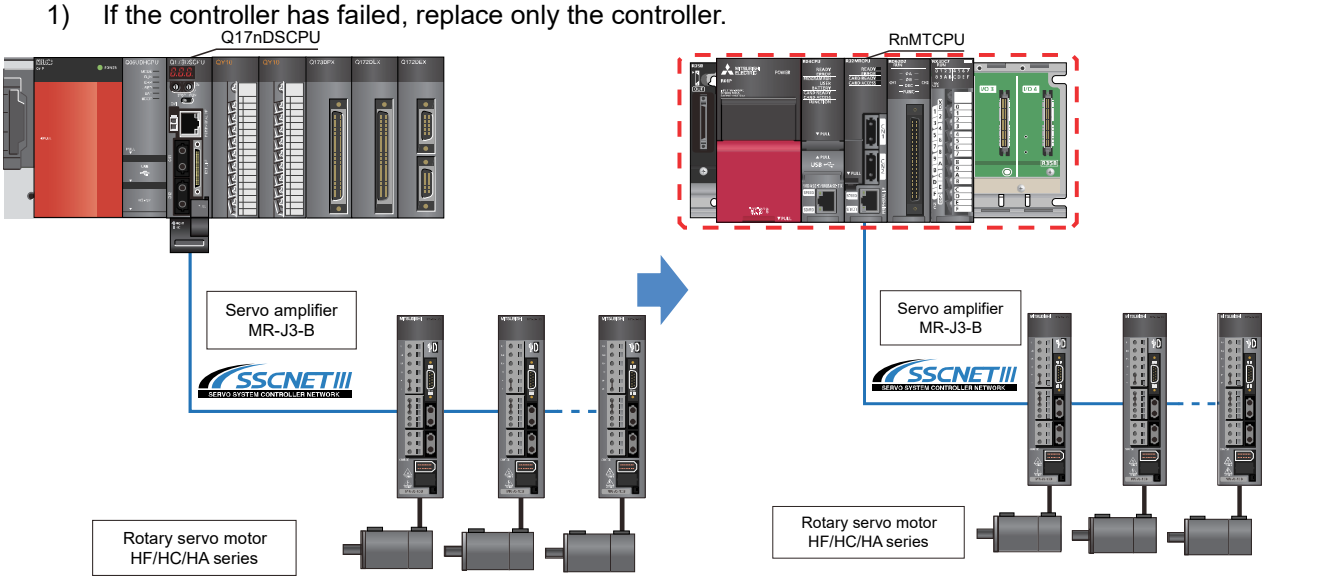
Point

- Q17nDSCPU/RnMTCPU can be connected to MR-J3-B, MR-J4-B, and MR-J5-B. The servo amplifiers can also be replaced before replacing the controller.
- MR-J3-B and MR-J4-B can both be used on the same network. To use them both on the same network, use SSCNET III for the servo system network and operate all MR-J4-B servo amplifiers in J3 compatibility mode.
- MR-J4-B and MR-J5-B can be used on the same network. However, use servo amplifiers from the same series for axes that require high-precision synchronization at the machine side across multiple axes, such as the axes used for interpolation control and synchronous control.
- MR-J3-B and MR-J5-B cannot be used on the same network.
- Only HG series rotary servo motors can be connected to MR-J4-B.
- Both HG series and HK series rotary servo motors can be connected to MR-J5-B.

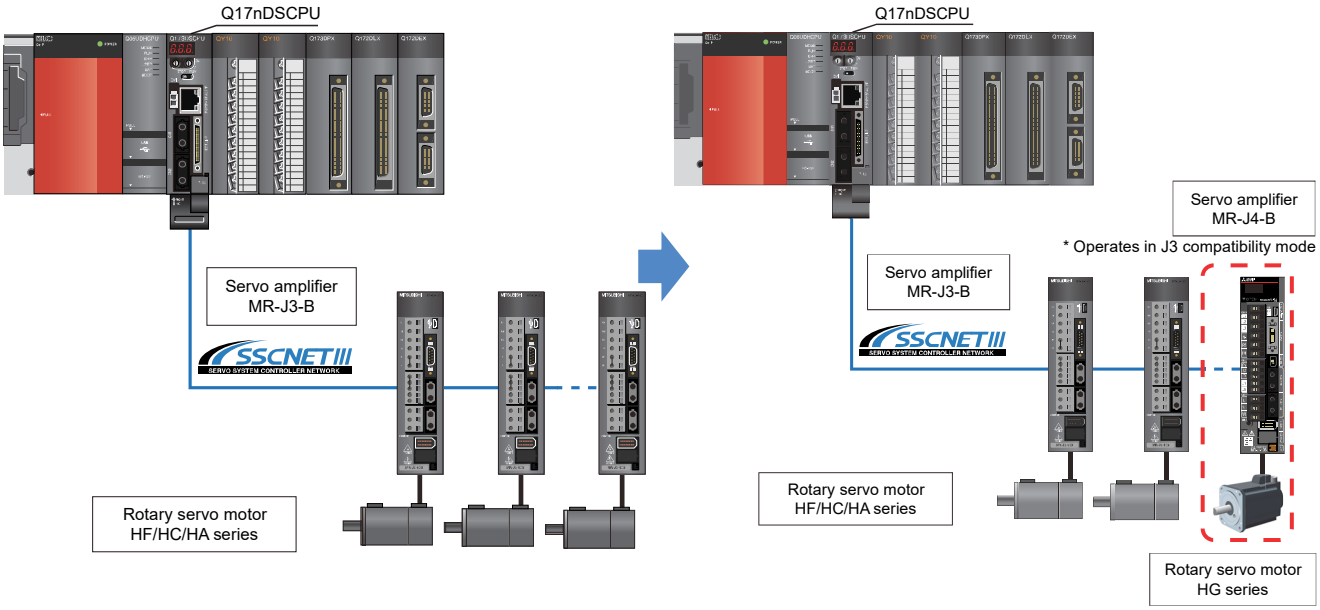
2.3.5. Individual replacement

The following shows the update procedure for replacing failed devices in the servo system.

1) If the controller has failed, replace only the controller.



2) If any of the servo amplifiers or servo motors have failed, replace the failed servo amplifiers and servo motors.



Point

- Q17nDSCPU/RnMTCPU can be connected to MR-J3-B and MR-J4-B.
- MR-J3-B and MR-J4-B can both be used on the same network. To use them both on the same network, use SSCNET III for the servo system network and operate all MR-J4-B servo amplifiers in J3 compatibility mode.
- Only HG series rotary servo motors can be connected to MR-J4-B.

3. DIFFERENCES BETWEEN Q17nDSCPU AND RnMTCPU

3.1. Table of the Differences

<table>
<tr><th colspan="2">Items</th><th>Q17nDSCPU</th><th>RnMTCPU</th><th>Points for migration</th></tr>
<tr><td rowspan="2">Operation cycle (Default value)</td><td>SV13</td><td>0.222 ms / 1 to 4 axes
0.444 ms / 5 to 10 axes
0.888 ms /11 to 24 axes
1.777 ms / 25 to 32 axes</td><td rowspan="2">0.222 ms / 1 to 2 axes
0.444 ms / 3 to 8 axes
0.888 ms / 9 to 20 axes
1.777 ms / 21 to 32 axes</td><td rowspan="2">If the operation cycle is set as default (automatic), the operation cycle will be changed.
Set a fixed operation cycle where necessary because the change in the operation cycle may change program execution timing.</td></tr>
<tr><td>SV22</td><td>0.444 ms / 1 to 6 axes
0.888 ms / 7 to 16 axes
1.777 ms / 17 to 32 axes</td></tr>
<tr><td colspan="2">Control method</td><td>PTP (Point to Point) control, Speed control, Speed/position switching control, Fixed-pitch feed, Constant speed control, Position follow-up control, Speed control with fixed position stop, Speed switching control, High-speed oscillation control, Speed-torque control, Synchronous control (SV22 (Virtual mode switching method/ Advanced synchronous control method))</td><td>PTP (Point to Point) control, Speed control, Speed/position switching control, Fixed-pitch feed, Continuous trajectory control, Position follow-up control, Speed control with fixed position stop, High-speed oscillation control, Speed-torque control, Tightening & press-fit control, Advanced synchronous control</td><td>• The term "Constant-speed control" has been changed to "Continuous trajectory control". However, the program is divertible as it is.
• If "Speed-switching control" is used, replace it with "Continuous trajectory control". (Refer to section 5.2.4.)</td></tr>
<tr><td rowspan="2">Motion dedicated PLC instruction</td><td>M(P).□</td><td>−</td><td>M(P).DDRD, M(P).DDWR, M(P).SFCS, M(P).SVST, M(P).CHGT, M(P).CHGV, M(P).CHGVS, M(P).CHGA, M(P).CHGAS, M(P).GINT, M(P).SVSTD, M(P).MCNST, M(P).BITWR</td><td>The D(P) instructions are executed at CPU communication cycle, while M(P) instructions are executed immediately.
Refer to "MELSEC iQ-R Motion Controller Programming Manual (Program Design)".</td></tr>
<tr><td>D(P).□</td><td>D(P).DDRD , D(P).DDWR, D(P).SFCS, D(P).SVST, D(P).CHGT, D(P).CHGT2 D(P).CHGV, D(P).CHGVS, D(P).CHGA, D(P).CHGAS, D(P).GINT</td><td>D(P).DDRD, D(P).DDWR, D(P).SFCS, D(P).SVST, D(P).CHGT (Note), D(P).CHGV, D(P).CHGVS, D(P).CHGA, D(P).CHGAS, D(P).GINT, D(P).SVSTD D(P).MCNST, D(P).BITWR

(Note): D(P).CHGT is equivalent to D(P).CHGT2.</td><td>• If the D(P).CHGT or D(P).CHGT2 is used, modify the program. (Refer to section 5.2.3.)
• Arguments of D(P).DDRD and D(P).DDWR have been changed. However, program modification is not required because the change is automatically reflected by GX Works3 at conversion.</td></tr>
<tr><td colspan="2">Servo external signal</td><td>Q172DLX signal, External input signals of servo amplifier (FLS/RLS/DOG), DI signal, Bit device</td><td>Bit device (When "Inter-module synchronization" is valid, "High accuracy" setting of actual input signal is possible), External input signals of servo amplifier (FLS/RLS/DOG)</td><td>−</td></tr>
<tr><td colspan="2">Cancel signal of servo program</td><td>Available</td><td>Available (Note-1)</td><td>−</td></tr>
</table>

[Continued]

Items		Q17nDSCPU	RnMTCPU	Points for migration
Limit switch output function		Up to 64 points	Up to 64 points	• Change the "Motor current value" to the "Actual current value". • When diverting from the virtual mode switching method, some data are not diverted.
Number of I/O points		Total 256 points (PX/PY) (Built-in interface in Motion CPU (Input 4 points) + I/O module + Intelligent function module)	Total 4096 (I/O module + Intelligent function module)	• When the existing program uses PX/PY, modify the program so that the PX/PY devices are replaced with the X/Y devices assigned in the system setting. (Refer to section 5.2.9.)
Devices	Input (X)	8192 points	12288 points (Real input 4096 points)	
	Output (Y)	8192 points	12288 points (Real input 4096 points)	
	Real I/O (PX/PY)	256 points	PX/PY are integrated into X/Y device	
	Internal relays (M)	12288 points	12288 points (default)	• The point assignment can be flexibly changed among devices within the total of 128 k words. • Motion registers (#) are not latched as default in RnMTCPU. Set the latch setting as needed.
	Link relays (B)	8192 points	8192 points (default)	
	Annunciators (F)	2048 points	2048 points (default)	
	Data registers (D)	19824 points	20480 points (default)	
	Link registers (W)	8192 points	8192 points (default)	
	Motion registers (#)	12288 points	12288 points (default)	
	Free-running timer (FT)	1 point (888 μs)	SD718, SD719 (888 μs free-running timer)	–
	Special relays (SM)	2256 points	4096 points	–
	Special registers (SD)	2256 points	4096 points	–
	Multiple CPU area devices (Fixed scan communication area)	Multiple CPU high speed transmission area Up to 14436 points/CPU (From U3E□\G10000) (Total of all CPUs: 14 kW)	CPU buffer memory (Fixed scan communication area) Up to 12288 points/CPU (From U3E□\HG0) (Total of all CPUs: 24 kW)	Replace the Multiple CPU high speed transmission area (from U3E□\G10000) in Q17nDSCPU with the CPU buffer memory (Fixed scan communication area (from U3E□\HG0)) in RnMTCPU. (Refer to section 5.2.13.)
		Updating cycle: 0.888 ms	Updating cycle: 0.222 ms to 7.111 ms (Variable depending on the setting)	
	Multiple CPU area devices	CPU shared memory 4096 points (From U3E□\G0)	CPU buffer memory 2097152 points (From U3E□\G0) (PLC CPU: 524288 points)	The self CPU operation information area has been deleted. (Refer to section 6.6.)
	Module access device (U□\G)	–	Up to 268435456 points	–

[Continued]

Items		Q17nDSCPU	RnMTCPU	Points for migration
Word device bit specification		Bit specification is possible in Multiple CPU area device only. (U3E□\G□.0 to F)	Bit specification is possible in all word devices.	–
Automatic refresh	Memory	Multiple CPU high speed transmission area of the Multiple CPU shared memory	CPU buffer memory/ CPU buffer memory (Fixed scan communication area)	–
	Automatic refresh setting	32 latch settings (Refresh END)	END: 32 latch ranges [Multiple CPU area devices] I45 executing: 32 latch ranges [Multiple CPU area devices (Fixed scan communication area)]	
	Multiple CPU high speed refresh function	128 settings Refresh timing: PLC CPU: refresh END, Motion CPU: operation cycle		When the refresh (I45 executing) is used, refresh synchronized with the fixed scan communication is also possible in PLC CPU.
Cancelling errors of Multiple CPU		M2039 OFF	SM50 ON • All errors can be cancelled. • After cancelling errors, SM50 turns OFF automatically.	Refer to section 3.2.
Self-diagnostic errors		When a Motion CPU-specific error occurs, 10000 to 10999 are stored in diagnostic errors (SD0) (the stored value varies depending on the error.) At this time, the self-diagnostic error flag (SM1) and diagnostic error flag (SM0) also turn ON.	All errors are assigned to the self-diagnostic error codes. When an error occurs, an error code is set in the latest self-diagnostics error code (SD0), and then Latest self-diagnostics error (SM0) and Latest self-diagnostics error (SM1) turns ON.	
Motion SFC error detection flag (M2039)		Provided (M2039 turns ON for all the errors occurred in the Motion CPU module.)	None (Integrated into the self-diagnostic errors)	
Battery error check of Motion CPU		Invalid setting possible	None (Battery-less)	
Error setting when a servo warning occurs		Selectable whether to output an error or not	Outputs an error	–
Peripheral I/F		PERIPHERAL I/F (Motion CPU) / USB/RS-232/Ethernet (via PLC CPU)	PERIPHERAL I/F (Motion CPU) / USB/Ethernet (via PLC CPU)	Use a compatible I/F to communicate with peripheral devices. If RS-232 communication is used, replace it with USB communication. In that case, replace the existing cable with the A-miniB USB cable as well.

[Continued]

Items	Q17nDSCPU	RnMTCPU	Points for migration
Forced stop input	• Use the devices (X/M) specified in the forced stop setting in system setting. • Use the EMI connector of the Motion CPU.	Use the devices (optional bit devices) specified in the forced stop setting in system setting.	• If the EMI terminal is used for the forced stop/release, access to the device via the input module.
Internal I/F connector	Provided (DI x 4, Manual pulse generator/Incremental synchronous encoder input x1)	None	• Change the input from DI to an Input module. • Change the input from the manual pulse generator/ synchronous encoder to a high-speed counter module.
High-speed input request signal	Built-in interface in Motion CPU(DI)/ Q172DLX(DOG/CHANGE)/ Q172DEX(TREN)/ Q173DPX(TREN)	Optional bit device/servo amplifier input	Change the signal input to the input module or DI input of servo amplifier.
Mark detection signal	Device/DI1 to DI4/ Q172DLX(DOG)	High-speed input request	The setting method has been changed. Change the setting. (Refer to section 5.3.2.)
RUN/STOP	RUN/STOP switch, Remote RUN/STOP M2000 ON/OFF directly, M3072 ON/OFF directly, D704 ON/OFF directly	RUN/STOP switch, Remote RUN/STOP, RUN contact	If M2000, M3072, or D704 is directly operated in the program to switch the RUN/STOP status, modify the program. (Refer to section 5.2.11.)
Output mode setting of STOP to RUN	-	Output the output (Y) status before STOP/ Clear the output (Y)	The default setting is "Output the output (Y) status before STOP". Change the setting if necessary.
ROM operation	• ROM writing is executed in "Mode operated by RAM"/ "Mode operated by ROM". • ROM writing can be executed for the data of MT Developer2.	Always operate with standard ROM data (write data of MELSOFT MT Developer2 to the standard ROM/transmit data of MELSOFT MT Developer2 from the SD memory card to the standard ROM using the file transmission at boot.)	–

[Continued]

Items			Q17nDSCPU	RnMTCPU	Points for migration
LED display			7-segment LED	Dot matrix LED READY, ERROR, CARD READY, CARD ACCESS	More information can be indicated on the LED display, enabling troubleshooting to be conducted more easily. (Refer to chapter 6 in "MELSEC iQ-R Motion Controller User's Manual".)
Rotary switch			2 (Normal mode, mode operated by ROM, installation mode, SRAM clear, Ethernet IP address display mode)	1 (Normal mode, installation mode, internal memory clear, Ethernet information display mode)	–
Latch range Setting	Latch (1)		1 setting (M,B,F,D,W devices)	Up to 32 settings (M, B, F, D, W, # devices)	# devices are latched as default in Q17nDSCPU, however, not in RnMTCPU. Change the latch settings as needed.
	Latch (2)				
Latch clear	Latch (1)		It is possible to clear with latch clear (1) and latch clear (1)(2) of remote latch clear.	• Clearing the MT Developer2 Motion CPU memory. • Cleaning built-in memory with Motion CPU rotary switch "C".	
	Latch (2)		It is possible to clear with latch clear (1)(2) of remote latch clear.	• Cleaning built-in memory with Motion CPU rotary switch "C".	
All clear function			Execute in installation mode	• The standard ROM and the latch range are cleared with the rotary switch for all clear. • The standard ROM is cleared by formatting the Motion CPU.	–
SSCNETIII	Communication speed		50 Mbps		–
	Transmission distance	Standard cable	Up to 20 m between stations Maximum overall distance 320 m (20 m × 16 axes)		
		Long distance cable	Up to 50 m between stations Maximum overall distance 800 m (50 m × 16 axes)		
	Servo amplifier		MR-J4-B (J3 compatibility mode), MR-J4W-B (J3 compatibility mode), MR-J3-_B, MR-J3W-_B, MR-J3-_B-RJ004, MR-J3-_B-RJ006, MR-J3-_B-RJ080W, MR-J3-_B Safety, MR-MT1200, FR-A700, VCII (Nikki Denso Co., Ltd.), VPH		
SSCNETIIII/H	Communication speed		150 Mbps		
	Transmission distance	Standard cable	Up to 20 m between stations Maximum overall distance 320 m (20 m × 16 axes)		
		Long distance cable	Up to 100 m between stations Maximum overall distance 1600 m (100 m × 16 axes)		
	Servo amplifier		MR-J5-_B, MR-J5W-_B, MR-J4-_B, MR-J4W-_B, FR-A800, LJ72MS15, VCII, VPH, AlphaStep/5-phase, IAI electric actuator controller		
			–	MR-J5-_B-LL, MR-J4-_B-LL, MR-MT2010	

[Continued]

Items		Q17nDSCPU	RnMTCPU	Points for migration
Amplifier-less operation function		Setting required for "EMI valid/EMI invalid" (At the amplifier-less operation start)	No setting required for "EMI valid/EMI invalid" (At the amplifier-less operation start)	The setting "EMI valid/EMI invalid" at the amplifier-less operation start has become unnecessary. Revise the program.
MC protocol communication (PERIPHERAL I/F)		Provided	None	Use the Ethernet port of PLC CPU for communication. Set the CPU No. of the Motion CPU for the SLMP/MC protocol request destination module I/O No.
Acceleration/deceleration time		1 to 65535 ms (1 word)	1 to 8388608 ms (2 words) 1 to 65535 (Note-2) (1 word)	1 word and 2 words can be selected for acceleration/deceleration time/command torque time constant in parameters and servo programs. (Refer to section 5.2.1.)
Torque limit value		1 [%] unit (Some items are set by 0.1 [%] unit)	0.1 [%] unit	Modify the program. (Refer to section 5.2.2.)
Motor speed (#8002+20n, #8003+20n)		0.1 r/min unit (0.1 mm/s for linear servo motors)	0.01 r/min unit (0.01 mm/s for linear servo motors)	Modify the program.
Operation control program	Motion dedicated instructions	CHGV, CHGVS, CHGT, CHGT2, CHGP	CHGV, CHGVS, CHGT(Note), CHGP (Note): Equivalent to CHGT2	If CHGT/CHGT2 instruction is used, modify the program. (Refer to section 5.2.3.)
	Others instructions	EI, DI, NOP, BMOV, FMOV, MULTW, MULTR, TO, FROM, RTO, RFROM	EI, DI, NOP, BMOV, FMOV, TO, FROM, RTO, RFROM	If MULTR/MULTW instruction is used, modify the program. (Refer to section 5.2.5.)
	Synchronous control dedicated function	CAMRD, CAMMK, CAMPSCL, CAMWR, CAMWR2	CAMRD, CAMMK, CAMPSCL, CAMWR(Note) (Note): Executes the function equivalent to CAMWR2 with argument setting.	Arguments have been added to all synchronous control dedicated functions. Modify the program. (Refer to section 5.2.6.)
	Y/N transition	Provided	Provided	The description method of the program has been changed. Modify the program. (Refer to section 5.2.8.)
Digital oscilloscope function		• Word 16CH, Bit 16CH. • Real-time display. • Sampling points: Up to 8192.	• Word 16CH, Bit 16CH • Real-time display. • Sampling points: Up to 133120 • Offline sampling. • Saving sampling results to an SD memory card.	Sampling can be performed without a personal computer by turning ON the "Sampling settings RUN request device" (SM860) after the setting file of trigger condition, etc., is stored to the ROM area of the Motion CPU or an SD memory card.
Security function		• Protection by password • Software security key	• Protection by password (32 characters). • Software security key (Common specification among MELSEC iQ-R series).	The setting method has been changed. Refer to "MELSEC iQ-R Motion Controller Programming Manual (Common)".

[Continued]

Manual title	Q17nDSCPU	RnMTCPU	Points for migration
Operating system software installation method	• MT Developer2	• MT Developer2 • SD memory card	• The installation files have been consolidated into one, making management of the files easier. • Supported installation using an SD memory card. (Refer to "MELSEC iQ-R Motion Controller Programming Manual (Common)".)
Safety Observation function	Safety signal comparison, Safety communication, STO, SS1, SS2, SOS, SLS, SBC, SSM	None	Use the Safety CPU module (R□SFCPU-SET) and the safety input/output. (The safety sub-functions and the safety levels differ depending on the combination of servo motor and servo amplifier, as well as the firmware version of the servo amplifier.)

(Note-1): Compatible with operating system software version 20 or later

(Note-2): Be sure to use this function in combination with MT Developer2 version "1.185T" or later and operating system software version "26" or later.

3.2. Error Codes System

MELSEC iQ-R series error codes are expressed with 4 hexadecimal digits (integer without 16-bit sign).
There are errors detected with each module's self-diagnostic function, and common errors detected when communicating between modules.
The error detection types and error code ranges are shown below.

Error detection type	Error code range	Description
Detection with each module's self-diagnostic function	0001H to 3FFFH	These are errors such as module self-diagnostic errors that are different for each module.
Detection when communicating between modules	4000H to 4FFFH	CPU module error
	7000H to 7FFFH	Serial communication module error
	B000H to BFFFH	CC-Link module error
	C000H to CFBFH	Ethernet module error
	D000H to DFFFH	CC-Link IE Field Network module error
	E000H to EFFFH	CC-Link IE controller network module error
	F000H to FFFFH	MELSECNET/H network module, MELSECNET/10 network module error

Errors detected at the RnMTCPU are divided into warnings and errors.
The categories and error code range of errors detected at the RnMTCPU are shown below.

Error detection type		Error code	Description	Remarks
Warning		0800H to 0FFFH	Warnings which do not stop servo programs	• Equivalent to some of the Q17nDSCPU minor errors
Error	Minor	1000H to 1FFFH	Errors which stop servo programs The CPU continues to operate (in RUN status).	• Equivalent to some of the minor errors of Q17nDSCPU, and the major errors
	Minor (SFC)	3100H to 3BFFH	Motion SFC execution errors The CPU continues to operate (in RUN status).	• Equivalent to Motion SFC errors of Q17nDSCPU.
	Moderate	2000H to 30FFH	Errors that put the CPU operation status to "During stop error".	• If the system parameter is set to "All station stop by stop error of CPU No.1 to 4", all CPUs of the whole system will be in stop status with the specified CPU stop error. • Equivalent to system setting errors of Q17nDSCPU.
	Major	3C00H to 3FFFH		• If the system parameter is set to "All station stop by stop error of CPU No.1 to 4", all CPUs of the whole system will be in stop status with the specified CPU stop error. • Equivalent to some of the self-diagnostic errors of Q17nDSCPU.

Refer to "MELSEC iQ-R Motion Controller Programming Manual (Common)" for details of at error detected details.

When the RnMTCPU detects an error, the error is displayed on the Motion CPU LED display, and the error code is stored in the relevant device.

Use the relevant device in which the error code is stored in the program to enable a machine control interlock.
The following shows the methods for checking and cancelling errors.

(1) Check methods for when an error occurs
(a) LED display
• The ERROR LED is ON (or flickers).
• The dot matrix LED displays ""AL" (flickers 3 times) → "Error code" (4 digits shown 2 at a time)".
(b) Special relays/special register
[Special relays]
• Latest self-diagnostics error (SM0)
• Latest self-diagnostics error (SM1)
• Warning detection (SM4)
• Detailed information 1: flag in use (SM80)
• Detailed information 2: flag in use (SM112)
[Special registers]
• Latest self-diagnostics error code (SD0)
• Clock time for latest self-diagnostic error occurrence (SD1 to SD7)
• Self-diagnostic error code (SD10 to SD25)
• Detailed information 1 information category (SD80)
• Detailed information 1 (SD81 to SD111)
• Detailed information 2 information category (SD112)
• Detailed information 2 (SD113 to SD143)
(c) GX Works3 module diagnostics (error information list)
(d) MT Developer2 Motion CPU error batch monitor (Motion error history)
(e) Axis status signals, and axis monitor devices
(f) Check the operation and error contents with the standard ROM of Motion CPU or the event history file saved in the memory card.

(2) Cancelling errors

Among the RnMTCPU errors, continue errors (minor errors, or continue mode moderate errors) and warnings can be cancelled.

Use the following method to cancel errors after eliminating the cause.

• Cancel with MELSOFT GX Works3 "Module diagnostics"
• Cancel with MELSOFT MT Works2 "Motion Monitor"
• Cancel with "Error reset (SM50)" (Note-1)

Error type	Information required to cancel error
System common errors	• Self-diagnostic error information (SD0 to SD7, SD10 to SD25) • Diagnosis error detection (SM0, SM1) • Warning detection (SM4) • Detailed information 1 (SD80 to SD111) • Detailed information 2 (SD112 to SD143) • Detailed information 1: flag in use (SM80) • Detailed information 2: flag in use (SM112) • AC/DC DOWN counter (SD53) • AC/DC DOWN detected (SM53) • I/O module verify error module number (SD61)
Positioning/synchronous control output axis errors/warnings (Note-1)	• Warning code • Error code • Error detection signal
Servo alarms/warnings (Note-1)	• Servo error code • Servo error detection signal
Synchronous control input axis errors/warnings (Note-1)	• Command generation axis warning code • Command generation axis error code • Command generation axis error detection signal • Synchronous encoder axis warning No. • Synchronous encoder axis error No. • Synchronous encoder axis error detection signal

(Note-1): Clears errors for all axes at the same time

Refer to Appendix 1 Error Codes in "MELSEC iQ-R Motion Controller Programming Manual (Common)" for details.

4. PROJECT DIVERSION

4.1. List of Divertible/Not Divertible Data (SV13/SV22)

○: Divertible △: Partially divertible ×: Not divertible

Data name		Divertible/ not divertible	Remarks
R series common parameters	System parameter Multiple CPU setting	△	Communication setting between CPUs: "Refresh (END) Setting" only
	System parameter Inter-module synchronization setting	×	
	Motion CPU module CPU parameter	△	"Name setting" and "latch setting" only
	Motion CPU module Module parameter	△	"IP address setting" only
Motion CPU common parameters (Q17nDSCPU: System settings)	Basic setting	○	
	Servo network setting	○	
	Axis label	○	
	Limit output data	△	When diverting from the virtual mode switching method, some data is not diverted.
	High-speed input request signal	×	
	Mark detection	×	
	Manual pulse generator connection setting	×	
	Vision system parameter	○	
	Head module	○	
Motion control parameter (Q17nDSCPU: Servo data settings)	Axis setting parameter	△	If the external signal input module is set in the servo external signal parameter, this parameter is initialized.
	Servo parameter	○	
	Parameter block	○	
	Synchronous control parameter	△	When diverting the data from the virtual mode switching method, some data is not diverted.
Motion SFC programs	Motion SFC parameter	○	
	Motion SFC program	○	
	Operation control program	△	Refer to section 5.2.
	Transition program	△	Refer to section 5.2.
Servo program	Command generation axis program allocation	○	
	Servo program	△	Refer to section 5.2.
Cam data		△	When diverting the data from the virtual mode switching method, cam No. may be changed.

[Continued]

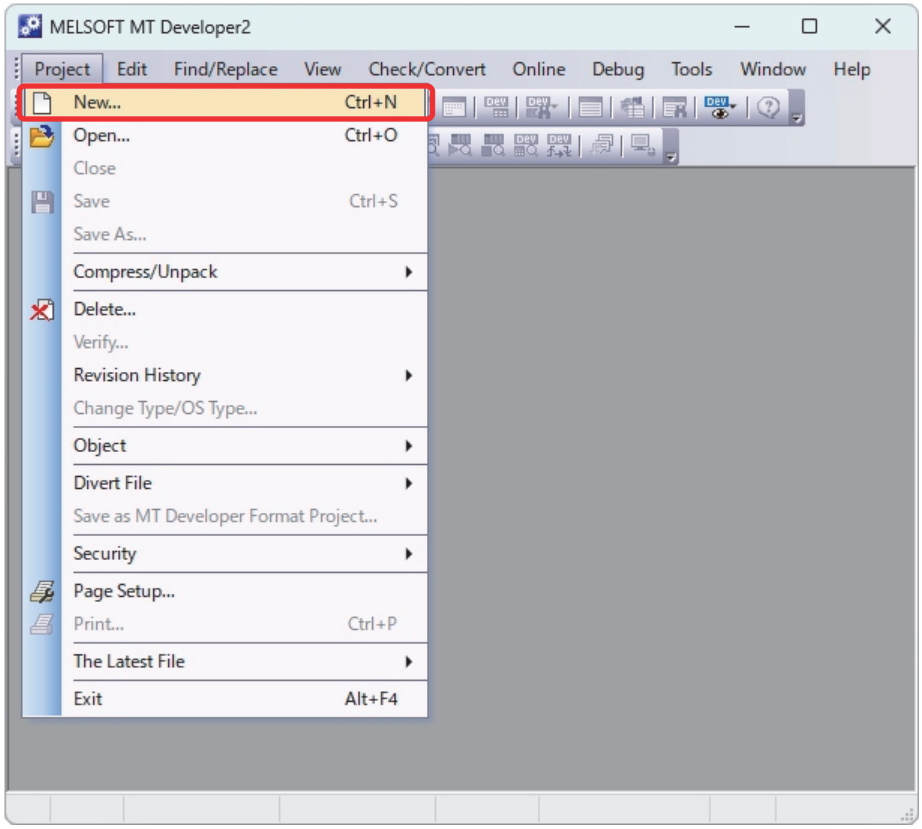
Data name	Divertible/ not divertible	Remarks
Label	○	
Structured data types	○	
Device memory	△	Special relays (SM) and special registers (SD) will not be diverted. Link relays (B), annunciators (F), and Link registers (W) are diverted within the range of the settable device points.
Device comment	○	
Backup data	×	
Communication setting	×	

4.2. Project Diversion Procedures

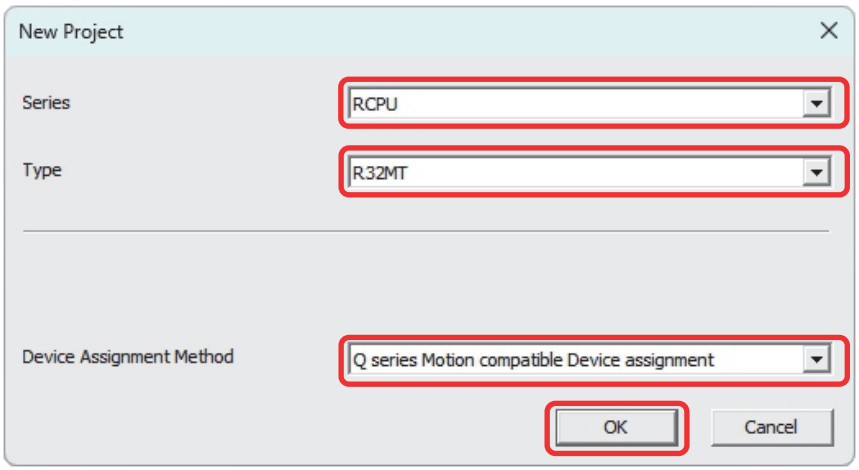
4.2.1. Procedures for project diversion by MT Developer2

The following shows the procedure for diverting Q17nDSCPU projects by MT Developer2.

1) Start MT Developer2. Select [New…] from [Project] menu.

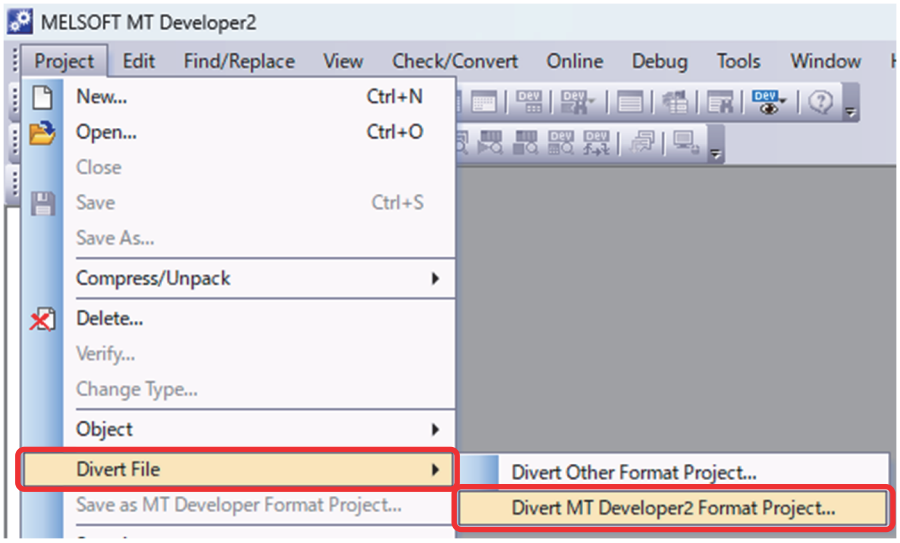


2) After the [New Project] screen appears, select RCPU for [Series], the replaced CPU model for [Type] (the setting example below: R32MT), and [Q series Motion compatible Device assignment] for Device Assignment Method, then click [OK].

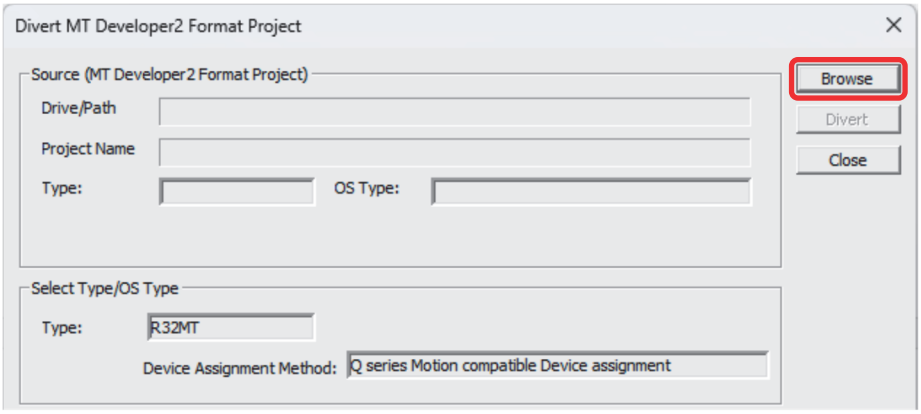


3) Divert the system parameters to the new project.
Refer to Section 5.1 for the setting method.
If the system parameters are not diverted, the basic settings of the Q17nDSCPU (such as auto refresh settings and latch range settings) will not be reflected in the R series common parameters of the RnMTCPU.

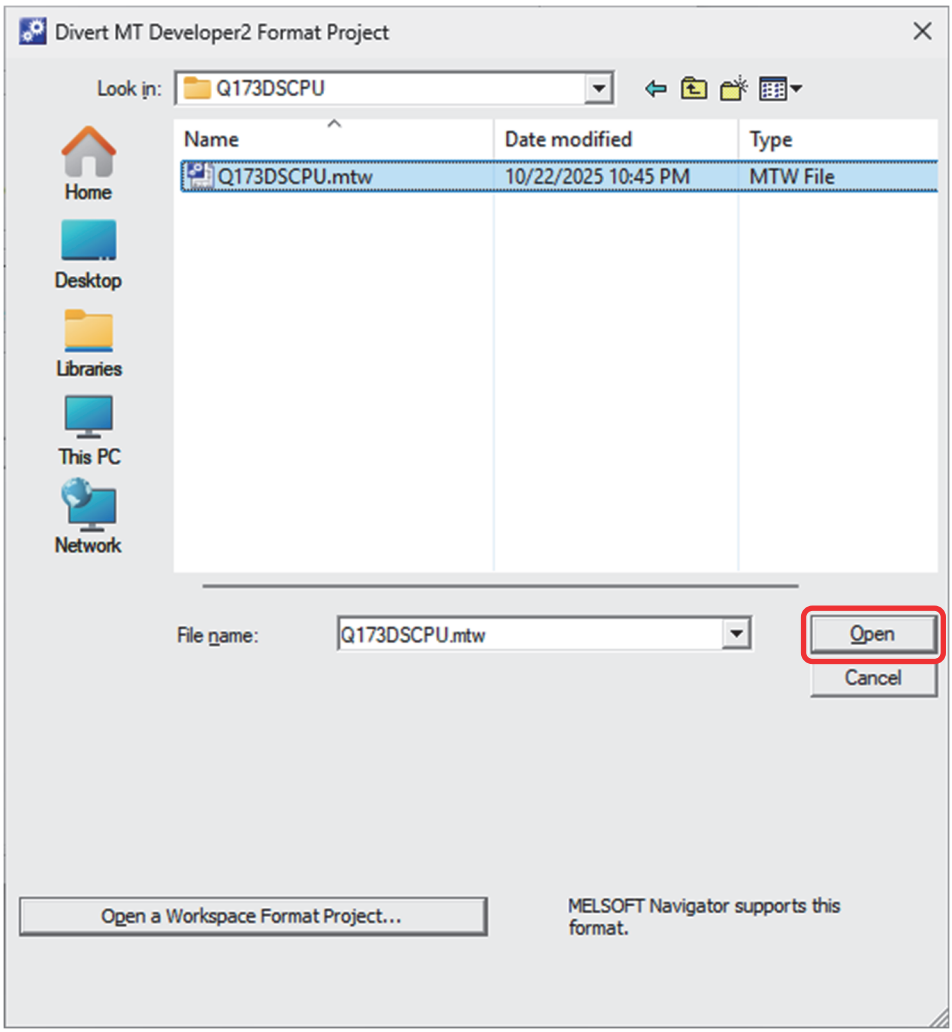
4) Start MT Developer2. Select [Divert file] - [Divert MT Developer2 Format Project…] from [Project] menu.



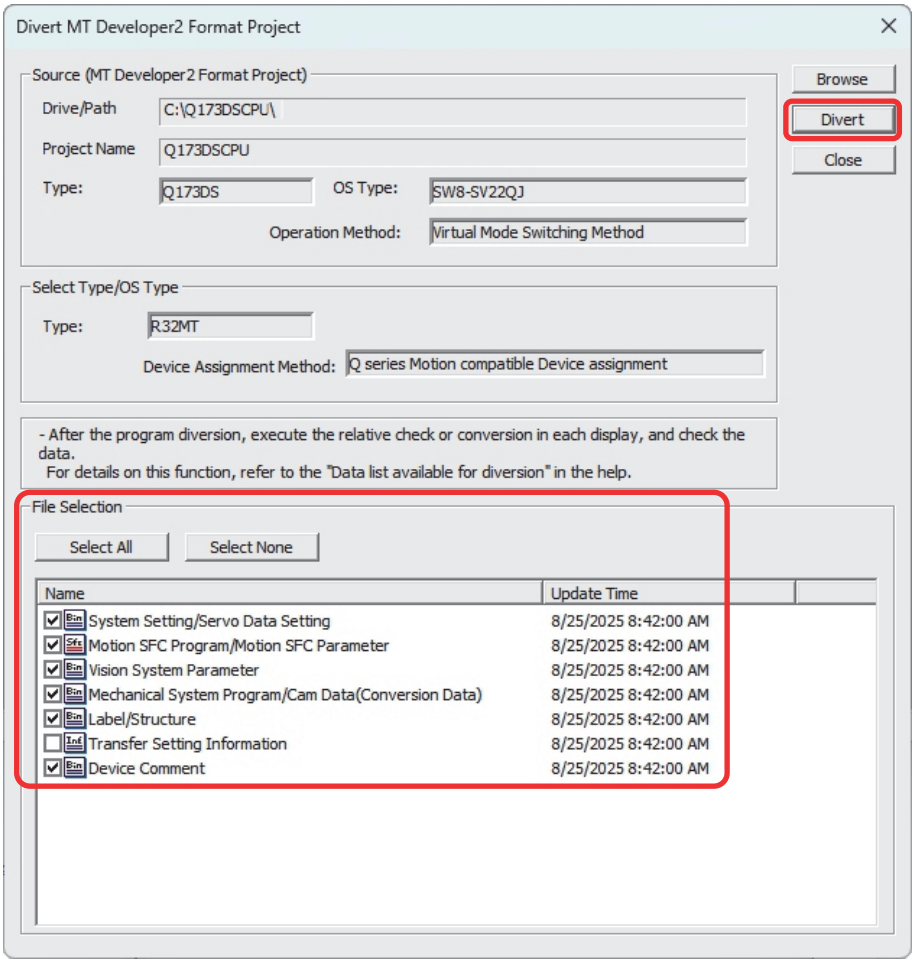
5) Click [Browse] on the [Divert MT Developer2 Format Project] screen.



6) Select the project to be diverted on the file selection window. Click [Open] to update the selected project (MT Developer2 Format Project).



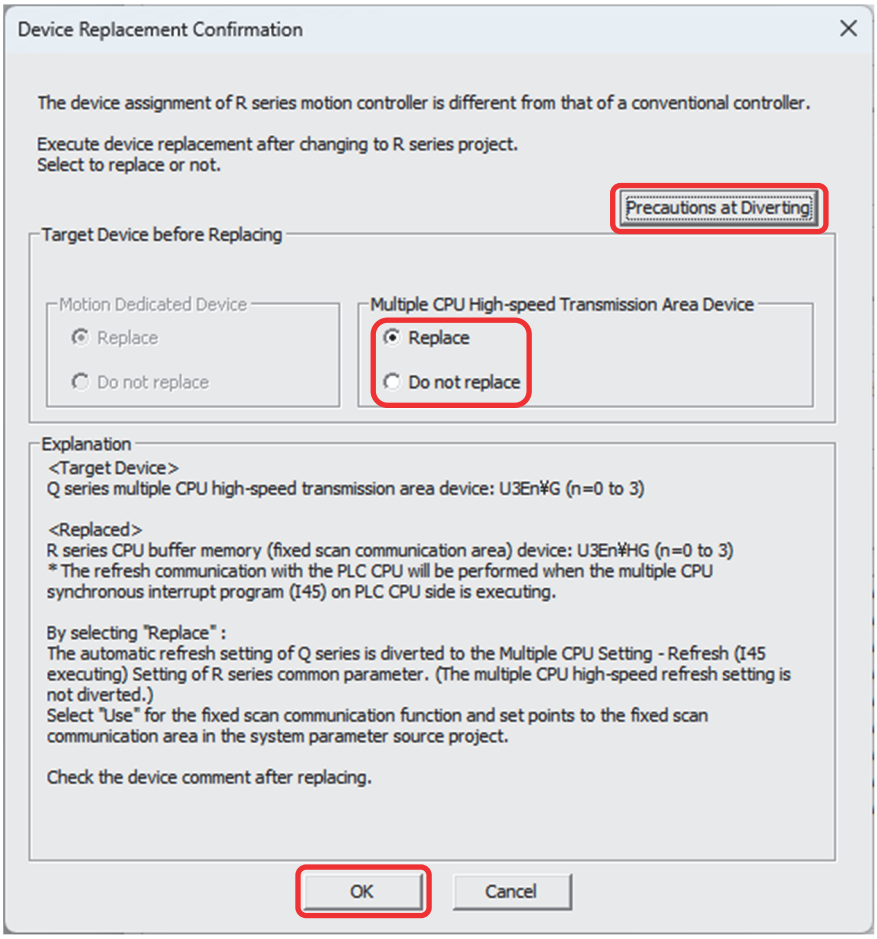
7) Check the box of the data to be diverted in the [File Selection]. Click [Divert].
When projects for Q17nDSCPU are diverted as those for RnMTCPU, clear the [Transfer Setting Information] checkbox since the [Transfer Setting Information] cannot be diverted.
If the [Transfer Setting Information] checkbox remains checked, an error message will be displayed.



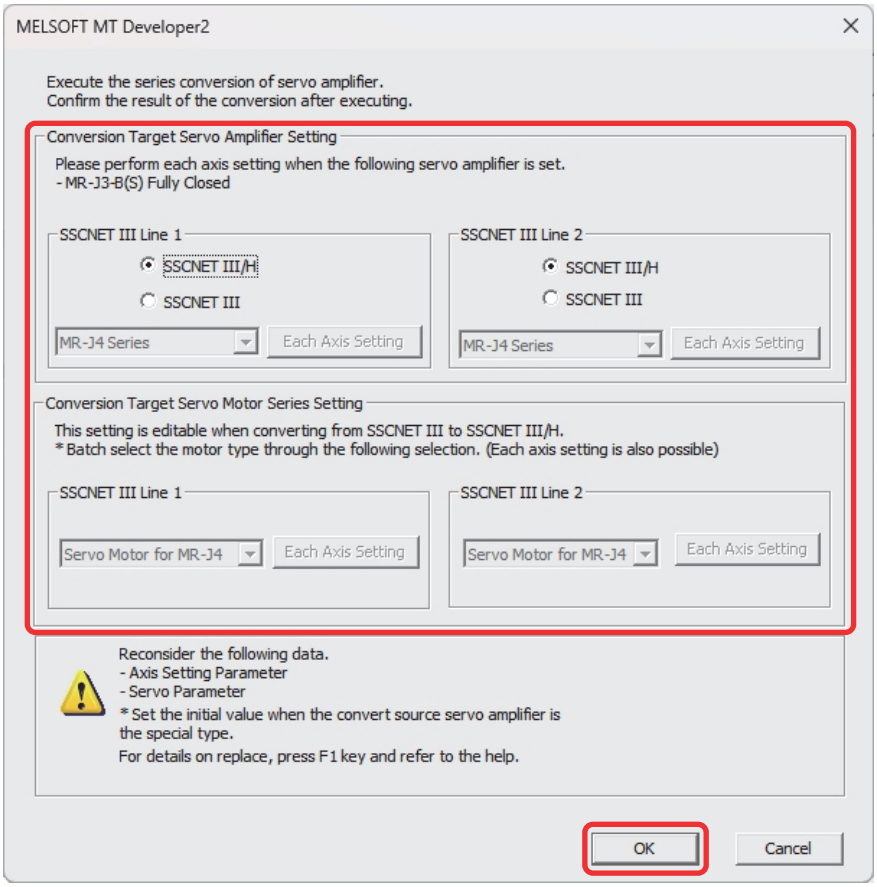
8) Review the [Precautions at Diverting] in the Device Replacement Confirmation. After that, select whether to replace the Q series automatic refresh setting in the [Multiple CPU High-speed Transmission Area Device], and click [OK].

Replace: Divert to [Refresh (I45 executing) setting] (Note-1)
Do not replace: Divert to [Refresh (END) setting]

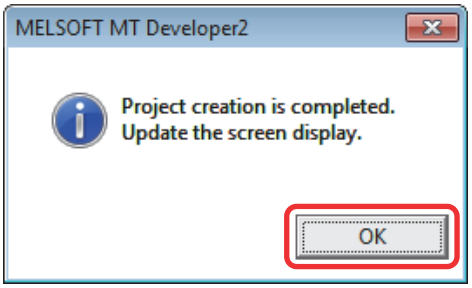
(Note-1): Add a sequence program that executes an I45 interrupt to execute a refresh.



9) Execute the series conversion of the servo amplifier and servo motor.
Select [SSCNETIII/H] in the [Conversion Target Servo Amplifier Setting] and select [Servo Motor of MR-J4] in the [Conversion Target Servo Amplifier Setting], and click [OK].

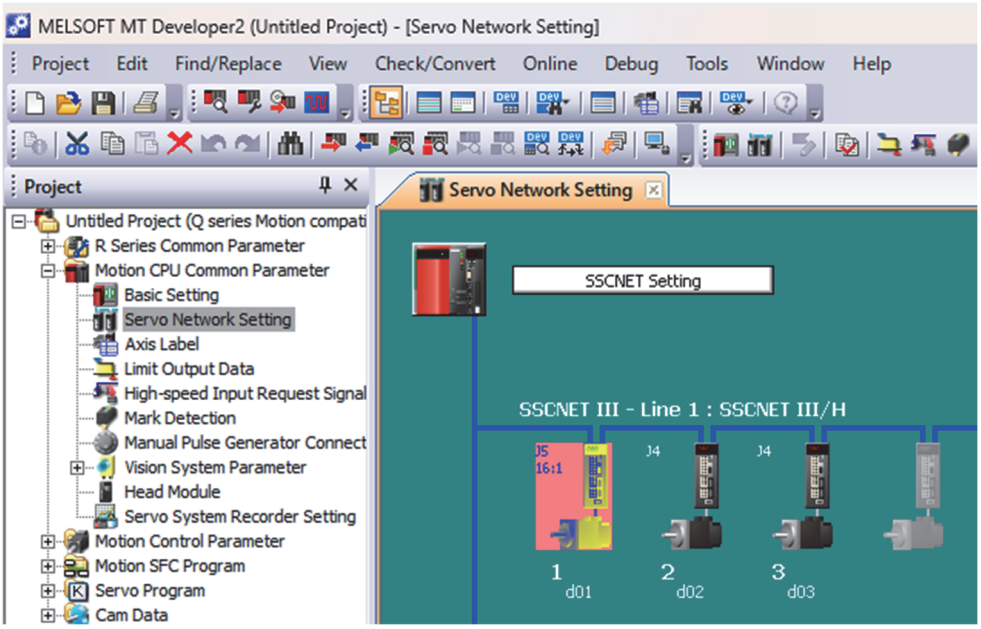


10) The diversion complete message will appear. After diversion, check each of the diverted data by executing a relative check or by converting the data.



(Note): Refer to MT Developer2 Help for details of the file diversion.

11) When using a MELSERVO-J5 series servo amplifier, change the servo amplifier set to each axis from the project window in [Servo Network Setting].
Refer to "Transition from MELSERVO-J4 Series to J5 Series Handbook (L(NA)03201)" for details.



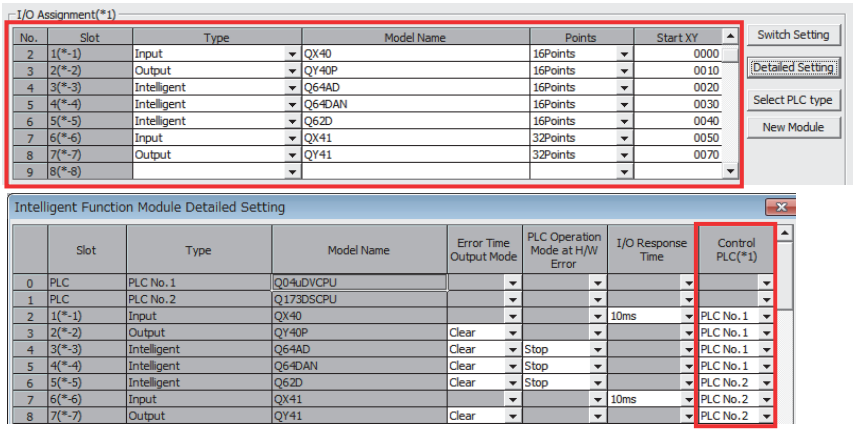
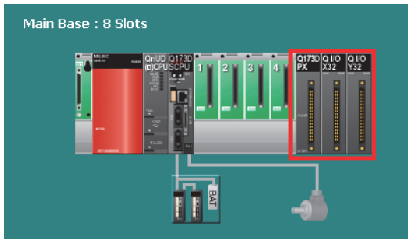
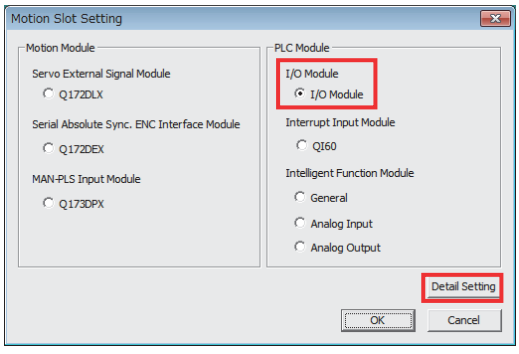
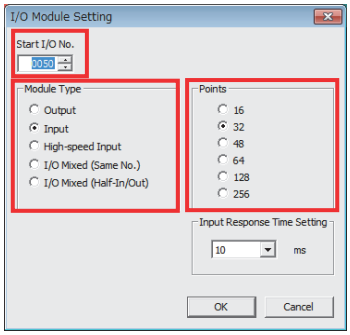
5. MIGRATION PROCEDURES

The following shows the procedures when migrating the existing system to a new one.

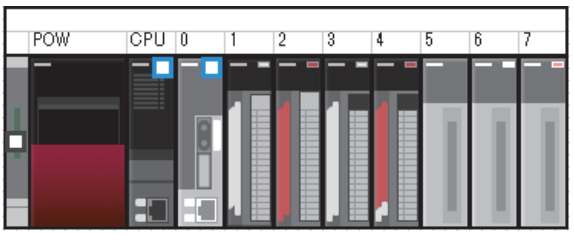
5.1. System Settings

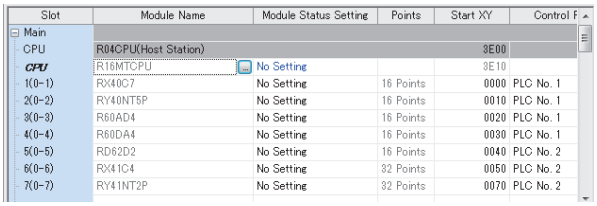
Refer to "MELSEC iQ-R Module Configuration Manual (SH-081262ENG)" for details of multiple CPU system settings (system settings including the installation position of CPU module, allocation of CPU No. and I/O No.)
The following shows the differences in system settings between Q17nDSCPU and RnMTCPU.

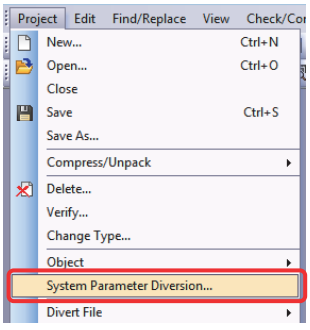
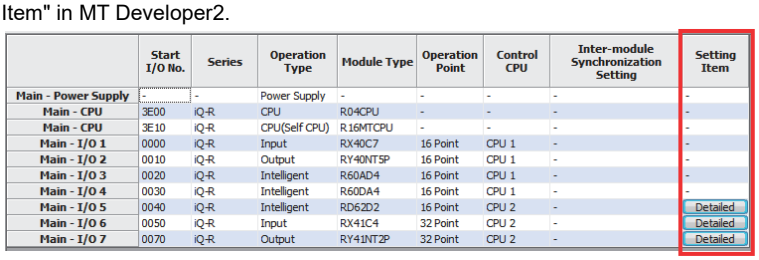
• System settings (Q17nDSCPU)
The system configuration is set separately in GX Works2 and MT Developer2, and their settings must be matched. The following shows an example of a system which uses various modules.

	Setting procedure	Description
1)	System configuration settings (GX Works2) (PLC CPU settings)	Set the multiple CPU system and I/O assignment in [Q parameter]. Specify the modules controlled by Motion Controller (CPU No.2) in [Detailed Setting]. 
2)	System configuration settings (MT Developer2) (Motion Controller settings)	Set the Multiple CPU system in [Multiple CPU setting] in [Basic setting], and set the modules controlled by the Motion CPU in [System Configuration]. 
3)	Settings for each module (MT Developer2)	Set each module on [Motion Slot Setting] screen from [System Configuration]. The example screen below sets QX41 input module in the I/O slot No. 6. • Start I/O No.: 0050H, Module type: Input, Points: 32 points  

• System settings (RnMTCPU)
The system configuration set in GX Works3 is read by MT Developer2, and therefore the setting is not set in MT Developer2. The following shows an example that is equivalent to the example system in the previous "System settings for Q17nDSCPU".

<table>
<tr><th colspan="2">Setting procedure</th><th>Description</th></tr>
<tr><td>1)</td><td>System configuration settings
(MELSOFT GX Works3)</td><td>Set the system configuration and the common parameters in GX Works3 (in [System configuration] and [System Parameter]).
Parameters
i. Module Configuration
ii. System Parameter
• I/O assignment setting
• Multiple CPU setting
• Synchronization setting
iii. Control PLC Settings (when using modules)
[Module Configuration] screen example


The modules in lighter color are the ones controlled by the Motion CPU. They are not controlled by the PLC CPU.
[System Parameter] screen example


Set "Control PLC" on the [System Parameter] screen.</td></tr>
<tr><td>2)</td><td>System configuration settings
(MT Developer2)</td><td>Select [System Parameter Diversion] from [Project] menu and divert the [System Parameter] from the GX Works3 project set in procedure 1).
(These common settings read from GX Works3 cannot be set in MT Developer2.)

</td></tr>
<tr><td>3)</td><td>Module parameter settings
(When using modules)</td><td>Set the parameters of the modules controlled by the Motion CPU (R32MTCPU). Select [R Series Common Parameters] → [Module Configuration List] → "Detailed" button in "Setting Item" in MT Developer2.

</td></tr>
</table>

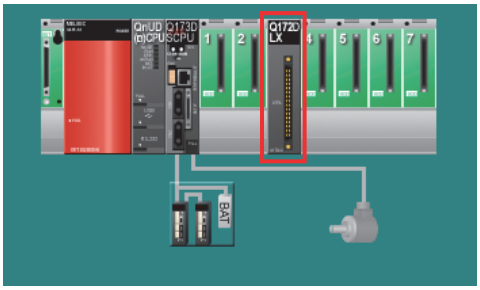
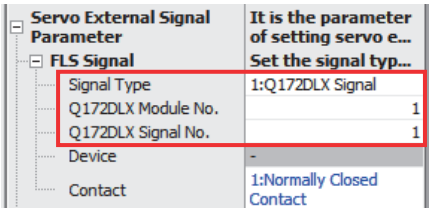
Point

- In the existing system with Q17nDSCPU, I/O No. is set separately by each CPU module. In the new system with RnMTCPU, common I/O No. is used between the PLC CPUs and the Motion CPUs.
- PX/PY devices are used when Q17nDSCPU accesses actual I/Os. However, the RnMTCPU uses X/Y devices instead of PX/PY devices to access actual I/Os. The PX/PY devices are not automatically converted at project diversion by MELSOFT MT Developer2. Be sure to modify the program. (An error will occur and writing operation cannot be performed.)

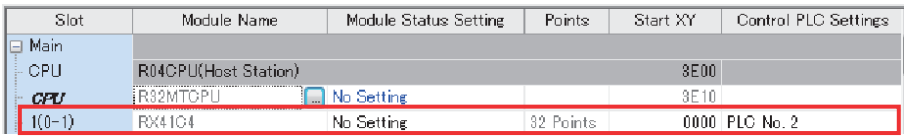
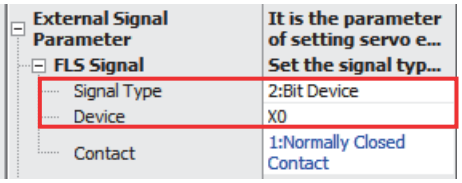
5.1.1. External signal input module

The following shows the differences in external signal settings between Q17nDSCPU and RnMTCPU.

• External signal settings (Q17nDSCPU)
Q17nDSCPU uses Q172DLX servo external signal interface module.
Set the module and external signals with MT Developer2.

	Setting item	Setting details
1)	[System Configuration] settings of MT Developer2	Set the Q172DLX servo external signal interface module on [System Configuration] screen. 
2)	[Servo Data] settings of MT Developer2	Set the FLS, RLS, STOP, and DOG signals as follows on the [Servo Data] setting screen.  [Signal Type] → "1: Q172DLX Signal" [Q172DLX Module No.] → 1 to 4 [Q172DLX Signal No.] → 1 to 8

• External input signal settings (RnMTCPU)
RnMTCPU uses a common input module with PLC CPU. The following shows an example in which the signal of RX41C4 input module is set in the external signal parameter for each axis.
The input module is set with GX Works3, and the external signal parameter for each axis is set with MT Developer2.

Setting item		Setting details
1)	[System Parameter] settings of GX Works3	Set RX41C4 input module on the [System Parameter] screen. Refer to Section 5.1 for setting method. 
2)	[Axis setting parameter] settings of MT Developer2	Set the external signal parameters (FLS, RLS, STOP, DOG) of the target axes as shown below on the [Axis setting parameter] setting screen. [Signal Type] → 2: Bit device [Device] → X0 (X device No. of the input module set in 1)) 

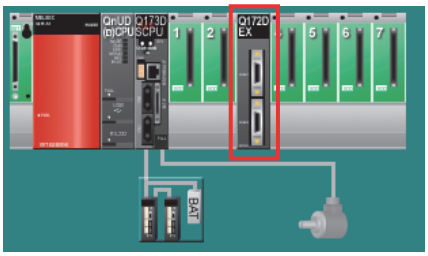
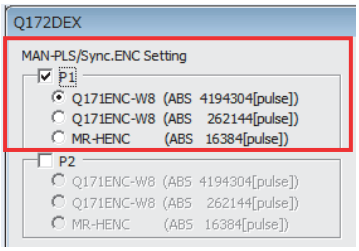
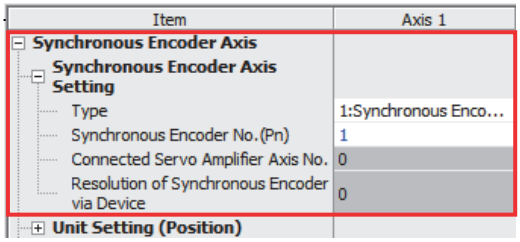
Point

When the MELSEC-Q series external signals interface module is replaced with the MELSEC iQ-R series input module, the detection accuracy depends on the operation cycle.
In order to detect signals at high accuracy, set the inter-module synchronization function to "Synchronize", and set the device of high-speed input request signal. And, change the accuracy of high-speed input request signals to "1: High-accuracy" before use.
Refer to "MELSEC iQ-R Motion Controller Programming Manual (Common)" for how to set the inter-module synchronization function.

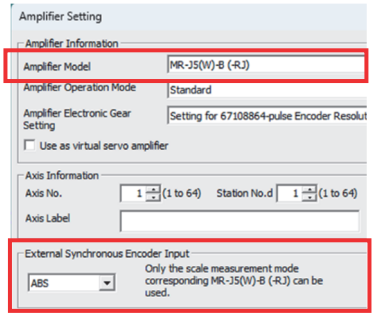
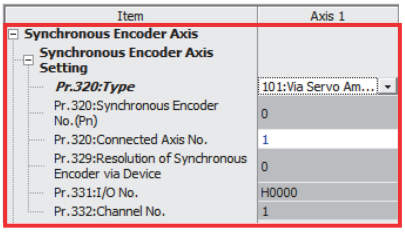
5.1.2. Synchronous encoder interface module

The following shows the differences in synchronous encoder settings between Q17nDSCPU and RnMTCPU.

• Synchronous encoder settings (Q17nDSCPU)
Q17nDSCPU uses Q172DEX synchronous encoder interface module. The interface module is set with MT Developer2.

	Setting item	Setting details
1)	[System Configuration] settings of MT Developer2	Set the Q172DEX synchronous encoder interface module on [System Configuration] screen.  
2)	[Servo Data] settings of MT Developer2	Set the following items for the target axes on [Synchronous Encoder Axis Parameter] screen. [Type] → 1: Synchronous Encoder Pn [Synchronous Encoder No.(Pn)] → 1 (the synchronous encoder No. set in [System Configuration] screen above.) 

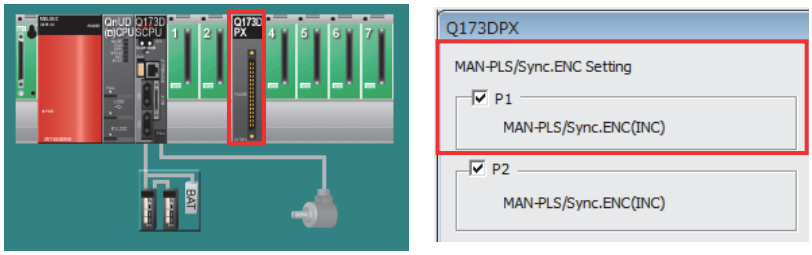
• Synchronous encoder settings (RnMTCPU)
RnMTCPU uses an HK series rotary servo motor for a synchronous encoder, and connects it to MR-J5-B-RJ. Use MT Developer2 for the settings of the synchronous encoder via a servo amplifier.

Setting procedure		Setting details
1)	[Servo Network] settings of MELSOFT MT Developer2	Set the servo amplifier and the synchronous encoder on [Amplifier Setting].  [Amplifier Model]: MR-J5(W)-B(-RJ) [External Synchronous Encoder Input]: ABS
2)	[Servo Parameter] settings of MELSOFT MT Developer2	Set "1: Four-wire type" in "Function selection C-8 (PC26) (Load-side encoder cable communication method selection)". 
3)	[Synchronous Encoder Axis Parameter] settings of MELSOFT MT Developer2	Set the following items for the target axes on [Synchronous Encoder Axis Parameter] screen. [Type] → 101: Via Servo Amplifier [Connected Axis No.] → The axis No. set in 1). [Servo Network] settings. 

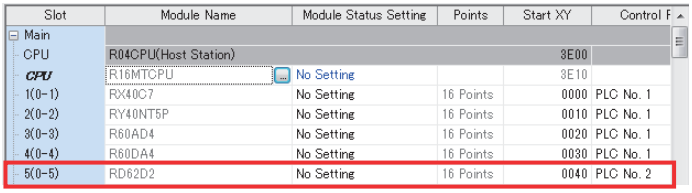
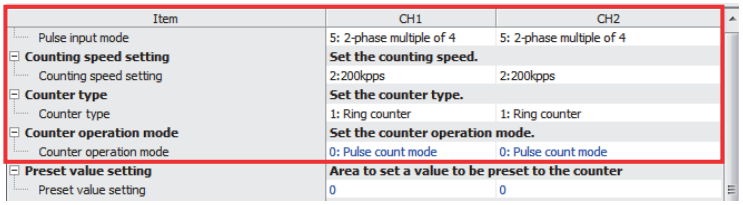
5.1.3. Manual pulse generator interface module

The following shows the differences in manual pulse generator settings between Q17nDSCPU and RnMTCPU.

• Manual pulse generator settings (Q17nDSCPU)
Q17nDSCPU uses Q173DPX manual pulse generator interface module.
The module is set with MELSOFT MT Developer2.

	Setting item	Setting details
1)	[System Configuration] settings of MT Developer2	Set the Q173DPX on [System Configuration] screen 

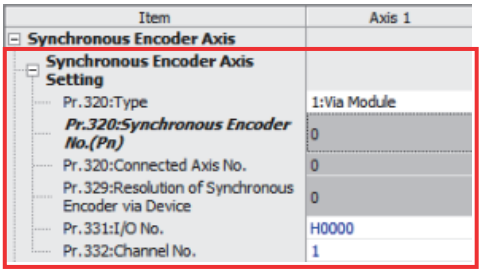
• Manual pulse generator settings (RnMTCPU)
RnMTCPU uses a common high-speed counter module with the PLC CPU.
The following shows a setting example when using RD62D2.
The high-speed counter module is set with GX Works3 and the external signals for each axis are set with MT Developer2.

	Setting item	Setting details
1)	[System Parameter] settings of GX Works3	Set the RD62D2 high-speed counter module in system parameter setting. Refer to section 5.1 for setting details. 
2)	[Module Detail Setting] settings of MT Developer2	Set the following items on [Module Detail Setting] screen. [Pulse input mode] → 2-phase multiple of 4 [Counting speed setting] → Select the maximum input pulse frequency (value after multiplied by 4) [Counter type] → 1: Ring counter [Counter operation mode] → Pulse count mode 

Slot	Module Name	Module Status Setting	Points	Start XY	Control F
Main					
CPU	R04CPU(Host Station)			3E00	
CPU	R16MTCPU	No Setting		3E10	
1(0-1)	RX40C7	No Setting	16 Points	0000	PLC No. 1
2(0-2)	RY40NT5P	No Setting	16 Points	0010	PLC No. 1
3(0-3)	R60AD4	No Setting	16 Points	0020	PLC No. 1
4(0-4)	R60DA4	No Setting	16 Points	0030	PLC No. 1
5(0-5)	RD62D2	No Setting	16 Points	0040	PLC No. 2

Item	CH1	CH2
Pulse input mode	5: 2-phase multiple of 4	5: 2-phase multiple of 4
Counting speed setting	**Set the counting speed.**	
Counting speed setting	2:200kpps	2:200kpps
Counter type	**Set the counter type.**	
Counter type	1: Ring counter	1: Ring counter
Counter operation mode	**Set the counter operation mode.**	
Counter operation mode	0: Pulse count mode	0: Pulse count mode
Preset value setting	**Area to set a value to be preset to the counter**	
Preset value setting	0	0

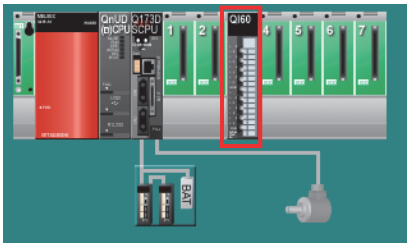
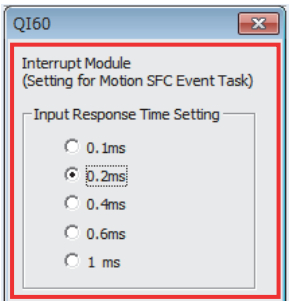
In addition, when the RnMTCPU uses an incremental synchronous encoder, the high-speed counter module can also be used. The high-speed counter module setting is the same as above, however, the axis setting is different. The following shows the setting details. Note that ring counter function, coincidence output function, counter function selection, and inter-module synchronization function are valid with direct operations on the RD62D2 buffer memory, however, the other functions are not valid.

Setting item		Setting details
3)	[Synchronous Encoder Axis Parameter] screen of MT Developer2	Set the high-speed counter module by the same procedure as in the previous page. Then, set the axes on [Synchronous Encoder Axis Parameter] of MT Developer2. [Type] → 1: Via Module [I/O No.] → H0000 (The I/O No. of the high-speed counter module set above) [Channel No.] → 1 to 2 (The channel No. of the high-speed counter module set above). 

5.1.4. Interrupt module

The following shows the differences in interrupt module settings between Q17nDSCPU and RnMTCPU.

• Interrupt module settings (Q17nDSCPU)
Q17nDSCPU uses QI60 interrupt module.
The interrupt module is set with MT Developer2.

	Setting item	Setting details
1)	[System Configuration] settings of MT Developer2	Set the QI60 interrupt module on [System Configuration] screen.  

• Interrupt module settings (RnMTCPU)
RnMTCPU uses a common input module with the PLC CPU. The signals to be used for interrupt need to be set as "interrupt" in the "input/interrupt setting". The following shows an example which uses RX40C7 input module.
The interrupt module is set with GX Works3, and the signals for each axis are set with MT Developer2.

	Setting item	Setting details
1)	[System Parameter] settings of GX Works3	Set the RX40C7 input module in the system parameter settings. Refer to Section 5.1 for setting details. Slot / Module Name / Module Status Setting / Points / Start XY / Control PLC Main CPU: R04CPU(Host Station) — 3E00 CPU: R16MTCPU — No Setting — 3E10 1(0-1): RX40C7 — No Setting — 16 Points — 0000 — PLC No. 2
2)	[Module Detail Setting] settings of MT Developer2	Set the following items on [Module Detail Setting] screen. [input/interrupt setting] → interrupt [interrupt condition setting] → leading edge [interrupt pointer] → I0 to I15 [Input response time setting]: 0.2 ms No. / input/interrupt setting / interrupt condition setting / interrupt pointer 1: interrupt — leading edge — I0 2: interrupt — leading edge — I1 Item / Setting Value input response time setting X00: 0.2ms X01: 0.2ms Point • The other points not used for interrupt can be used as input devices. • Change the setting items, [input/interrupt setting], [interrupt pointer], and [Input response time setting] according to your system.

5.2. Programming

5.2.1. Acceleration/deceleration time/command torque time constant

The setting range of the acceleration/deceleration time/command torque time constant of the parameters and servo programs is expanded from 1 word to 2 words in RnMTCPU. Therefore, when the acceleration/deceleration time/command torque time constant are indirectly designated, modify the programs according to the conditions below.
When diverting a program in 1 word, set it on the option screen of MT Developer2.
Refer to section 5.2.1 (2) for details.

[Items which need a program revision]

Function	Classification	Item
Motion control parameter (Axis setting parameter)	Expansion parameter	New acceleration time value device
		New deceleration time value device
	Speed-torque control data	Command speed acceleration time
		Command speed deceleration time
		Command torque time constant (positive direction)
		Command torque time constant (negative direction)
Motion control parameter (Synchronous control parameter)	Command generation axis parameter.	New acceleration time value device
		New deceleration time value device
Servo program	Each servo program	Acceleration time
		Deceleration time
		Rapid stop deceleration time
	Fixed position stop speed control	Fixed position stop acceleration/deceleration time

(1) When replacing a program with the data in 2 words

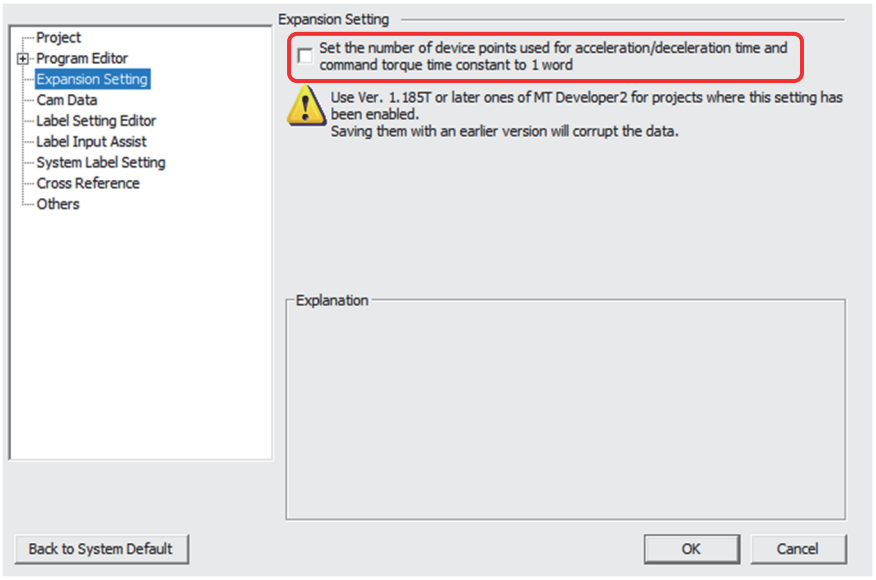
No.	Conditions		Migration procedure
1	Direct setting of the acceleration/ deceleration time/command torque time constant		• No need to modify the program.
2	Indirect setting of the acceleration /deceleration time/ command torque time constant	The start device No. is an even number	• Check whether the next device of the start device is usable or not. If it is unusable, secure another different 2 words of devices for the acceleration/deceleration time settings. • Note that no error occurs at program conversion.
3		The start device No. is an odd number	• An odd number cannot be set as the start device number since the acceleration/deceleration time setting requires 2 words. Secure 2 words of devices starting from even number. • If the device is an odd number, an error occurs at program conversion.

(2) When replacing a program with the data in 1 word

The setting range of the acceleration/deceleration time/command torque time constant is expanded from 1 word to 2 words in RnMTCPU. The number of words used must be changed to 1 word setting, if the program is used as it is.

The number of words used is set in the MT Developer2 options screen by selecting "Expansion Setting", and inserting a check for "Set the number of device points used for acceleration/deceleration time and command torque time constant to 1 word".

- No check: 2 words setting
- With check: 1 word setting



[Checking the number of words used]

(Note): The number of words used can be checked with the following monitor device.

Monitor item	Storage details	Monitor value	Refresh cycle	Device No.
Words used information	Stores information of the number of words used set with the acceleration/deceleration time and command torque time constant 1 word setting function.	0: 2 words setting 1: 1 word setting	At power ON	SD750

Point

- When changing the number of words used setting, rewrite the Motion control parameter (axis setting parameter, parameter block and synchronous control parameter) and servo program, and turn ON the Multiple CPU system power supply again. When changing the Motion CPU from STOP to RUN without turning ON the Multiple CPU system power supply again, a moderate error (error code: 30E6H) occurs.
- Used in combination with MT Works2 version "1.185T" or later and operating system software version "26" or later for the acceleration/deceleration time and command torque time constant 1 word setting function.

5.2.2. Torque limit value settings

All torque limit value is set by 0.1 [%] unit in RnMTCPU.
Refer to the following table for the program modification.

Function	Item	Unit		Points for migration
		Q17nDSCPU	RnMTCPU	
Motion control parameter (Parameter block)	Torque limit value	1 [%]	0.1 [%]	The unit is automatically converted to 0.1 [%] at project diversion
Axis setting parameter (Expansion parameter)	Torque limit value individual monitor device	0.1 [%]		No need to change.
Axis setting parameter (Speed/torque control data)	Torque limit value	0.1 [%]		No need to change.
	Torque command device	0.1 [%]		No need to change.
Axis setting parameter (Home position return data) (Note): Only when the stopper method is executed.	Torque limit value at creep speed	1 [%]		The unit is automatically converted to 0.1 [%] at project diversion. However, when the unit is indirectly designated, the unit is not automatically converted. Therefore, modify the program.
Servo program	Torque limit value (common)	1 [%]		The unit is not automatically converted regardless of direct or indirect designation. Therefore, modify the program.
	Torque limit value (parameter block)			
Data register (Monitor device)	Torque limit value (D14+20n)	1 [%]		Since the values stored in this monitor device will be changed due to the unit change, a revision is needed for programs which use "D14+20n".
Motion SFC instruction	Torque limit value change request (CHGT, CHGT2)	1 [%] 0.1 [%]		Since the instruction method has been changed, modify the program. Refer to 5.2.3 for details.
Motion dedicated PLC instruction	Torque limit value change request instruction from the PLC CPU to the Motion CPU (D(P).CHGT, D(P).CHGT2)	1 [%] 0.1 [%]		

5.2.3. Torque limit value change request, Torque limit value change request instruction from PLC CPU to Motion CPU

CHGT/D(P).CHGT instructions and CHGT2/D(P).CHGT2 instructions of Q17nDS can be replaced with CHGT/D(P).CHGT instructions of RnMTCPU.
However, since the instruction contents and setting data differ, modify the program.

(1) Replacement of CHGT/D(P).CHGT instructions

The CHGT instruction of Q17nDSCPU sets the same value for both positive and negative directions. However, for the CHGT instruction of RnMTCPU, set the value individually for positive and negative directions.
The torque limit value is set by [%] unit in CHGT instructions of Q17nDSCPU, and set by 0.1 [%] unit in CHGT instructions of RnMTCPU.

Ex.) A program which changes Axis 1 torque limit value by 10.0[%]

Q17nDSCPU		RnMTCPU
CHGT (K1 , K10)	→	CHGT (K1 , K100 , K100)
D(P).CHGT H3E1 "J1" K10		D(P).CHGT H3E1 "J1" K100 K100

Point

The changes above are not automatically reflected by MT Developer2 or GX Works3 at project diversion. Therefore, modify the project. (In MT Developer2, an error occurs during project conversion, and the program cannot be written.)
Since the instructions above are not supported by GX Works3, they will be replaced with instructions using SM4095.

(2) Replacement of CHGT2/D(P).CHGT2 instructions

Replace the CHGT2/D(P).CHGT2 instructions of Q17nDSCPU with the CHGT/D(P).CHGT instructions for RnMTCPU.
The instructions can be replaced simply by changing their names.

Ex.) A program which changes Axis 1 torque limit value by 20.0[%] in positive direction and by 10.0[%] in negative direction.

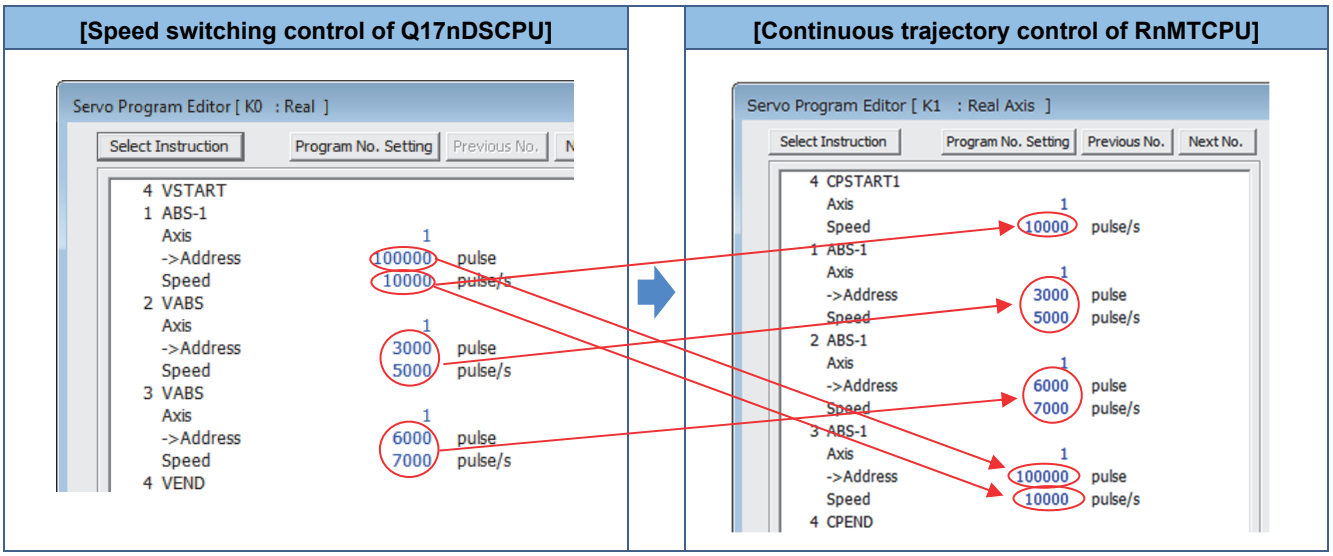
Q17nDSCPU		RnMTCPU
CHGT2 (K1 , K200 , K100)	→	CHGT (K1 , K200 , K100)
D(P).CHGT2 H3E1 "J1" K200 K100		D(P).CHGT H3E1 "J1" K200 K100

Point

The changes above are not automatically reflected by MT Developer2 or GX Works3 at project diversion. Therefore, modify the project. (An error occurs during project conversion, and the program cannot be written.)
Since the instructions above are not supported by GX Works3, they will be replaced with instructions using SM4095.

5.2.4. Speed switching control

The speed switching control is not available with RnMTCPU.
When the speed switching control is used, replace it with continuous trajectory control (name changed from constant-speed control).
The following shows a replacement example when changing the speed switching control to the continuous trajectory control.



Point

In speed switching control, the speed is set for each required point after the end point address/movement amount is set.
In continuous trajectory control, set the address/movement amount and the speed for each point.

5.2.5. Write device data to CPU shared memory/Read device data from CPU shared memory

(1) MULTW/MULTR instructions

MULTW/MULTR instructions were used when Q17nDSCPU accesses the CPU shared memory. In RnMTCPU, however, the MULTW/MULTR instructions have been eliminated because "CPU buffer memory access device (from U3E□\G0)" has become available to access the memory.
If those instructions are used before migration, replace them with one of the following.

- TO/FROM instruction
- BMOV instruction
- CPU buffer memory access device to directly access the memory

Ex. 1) Program which writes two words from D0 to the buffer memory (from A00H) of the self CPU (CPU No.2)

Q17nDSCPU		RnMTCPU
MULTW A00H, D0, K2, M0	→	TO 3E10H, A00H, D0, K2
		BMOV U3E1\G2560, D0, K2
		U3E1\G2560L = D0L

Ex. 2) Program which reads two words from the shared memory of CPU No.1 (C00H) to #0

Q17nDSCPU		RnMTCPU
MULTR #0, 3E0H, C00H, K2	→	FROM #0, 3E00H, C00H, K2
		BMOV #0, U3E0\G3072, K2
		#0L = U3E0\G3072L

Point

Make sure to modify the Motion SFC program since the MT Developer2 does not automatically convert Motion SFC programs at project diversion.
(An error occurs at program conversion and write operation cannot be performed.)

(2) Access to the other modules (MULTR/FROM/TO instructions)

In Q17nDSCPU, if the specified I/O number is invalid (for example, the specified module does not exist) when accessing another module using MULTR instructions or FROM/TO instructions, Q17nDSCPU outputs a Motion SFC error, but program execution continues. In RnMTCPU, CPU program execution can be stopped or continued with parameters.
([R series common parameter] - [CPU parameter] - [RAS setting] - [CPU module operation setting at error detected] - [Module I/O No. specification incorrect])

The default setting of the parameter is "Stop".
In order to make the setting equivalent to that of Q17nDSCPU (program execution does not stop) when the specified I/O number is invalid, change the parameter to "Continue".

5.2.6. Synchronous control dedicated functions

An SD memory card is newly selectable for a cam data saving area in RnMTCPU. To specify the cam data read/write area, an argument has been added to each of the read/write instructions.
Modify the programs by referring to the following revision examples.

(1) Cam data read (CAMRD), Cam auto-generation function (CAMMK)
The new argument is added to specify cam data read/write area, however, they can be omitted. When omitted, the cam data saved in the same area as Q17nDSCPU will be read. Therefore, in that case, the existing programs can be diverted without revision. Use the new argument only when the program uses the new cam data read/write area (SD memory card).

(2) Cam data write (CAMRW), Cam data write (for Cam open area) (CAMWR2)
Since the cam data write area can be specified by the new argument, CAMWR2 has been eliminated.

Ex 1.) A revision of a Q17nDSCPU program which uses CAMWR (A program which writes the data stored in #0 to #4099 to the specified area (from 1 to 2048 points) of No.256 cam data (stroke ratio data type))

Q17nDSCPU		RnMTCPU
CAMWR K256,K1,K2048,#0	→	CAMWR K256,K1,K2048,#0,H401

Ex. 2) A revision of a Q17nDSCPU program which uses CAMWR2 (A program which writes the data stored in #0 to #4099 to the specified area (from 1 to 2048 points) of No.256 cam data (stroke ratio data type))

Q17nDSCPU		RnMTCPU
CAMWR2 K256,K1,K2048,#0	→	CAMWR K256,K1,K2048,#0

Point

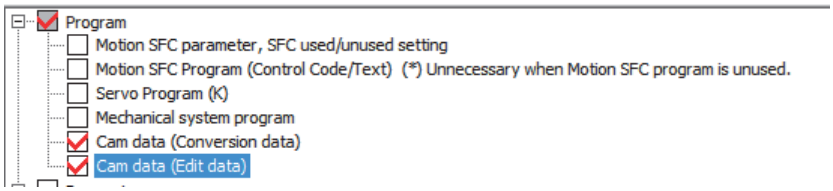
- If the write area argument is omitted, the cam data will be written to the same area as when Q17nDSCPU uses CAMWR2.
- Set "H401" for the cam read area when writing the cam data to the same area as when the Q17nDSCPU uses CAMWR.

(3) Cam position calculation (CAMPSCL)
There is no specification change in Cam position calculation.
The Q17nDSCPU programs can be diverted as it is.

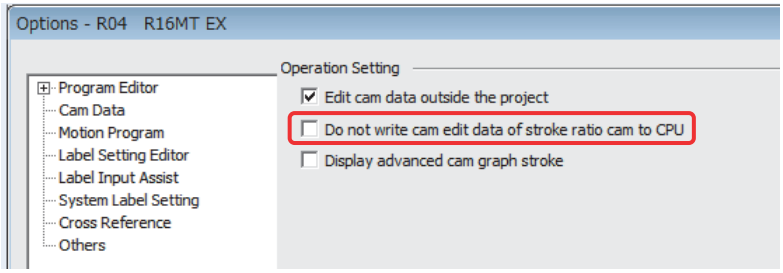
5.2.7. Cam conversion/editing data

When project data is written to the Q17nDSCPU, two types of cam data are written: "Cam data (Conversion data)" for cam control and "Cam data (Edit data)" for read and re-editing.
However, in RnMTCPU, since these two types of cam data are integrated, only "Cam data" is written. (The cam data (equivalent to edit data in Q17nDSCPU) is written to a CPU module and also used for cam control.)
In addition, a file password or an optional setting of MT Developer2 is available for protecting your cam data from unauthorized access. (For the optional setting, select [Tool] → [Option] → [Cam data] and check the box of "Do not write cam edit data of stroke ratio cam to CPU".)

Q17nDSCPU



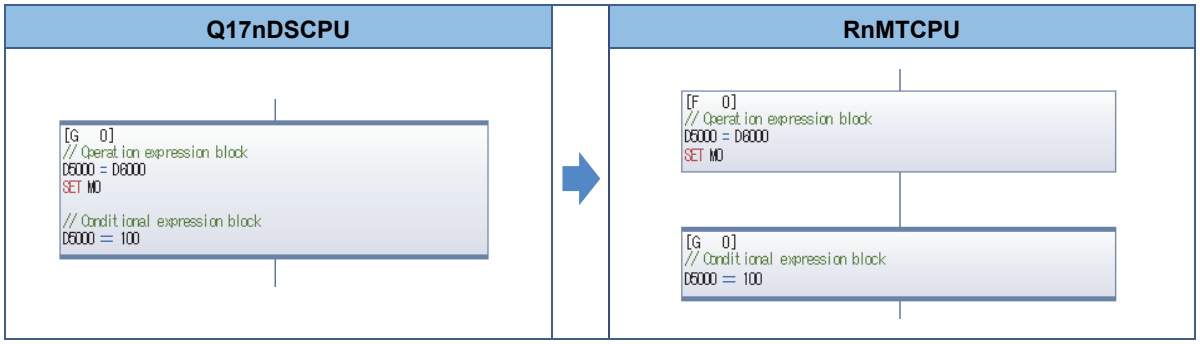
Optional settings



5.2.8. Motion SFC program (Y/N transition)

For Q17nDSCPU, conditional/operation expressions can be described together in "Shift Y/N transition" or "WAIT Y/N transition" (conditional expression must be the last block), however, for RnMTCPU, only conditional expression can be described in the transition program.
If operation expression and conditional expression are described together in "Shift Y/N transition" or "WAIT Y/N transition", modify the programs by referring to the following revision example.

Ex.) "WAIT Y/N" transition



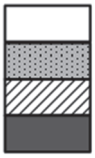
Point

- If operation expression and conditional expression are described together in "Shift Y/N transition" or "WAIT Y/N transition", revise the program so that the operation expression is described in an operation control step, and the conditional expression is in "WAIT Y/N transition".
- Make sure to modify the Motion SFC program since MELSOFT MT Developer2 does not automatically convert the changes above at project diversion. (An error will occur at project conversion and write operation cannot be executed.

5.2.9. Input device (X)

With the Q17nDSCPU, the input devices (X) other than actual I/O devices (PX) assigned in the system settings can be used in the same way as internal relay (M). These X devices can be flexibly set/reset in the program.
In the RnMTCPU system, however, depending on the system configuration, some input devices (X) may not be set/reset.
Modify the program in the following cases.

The system image below shows a general example of refresh area for input devices of a self CPU.

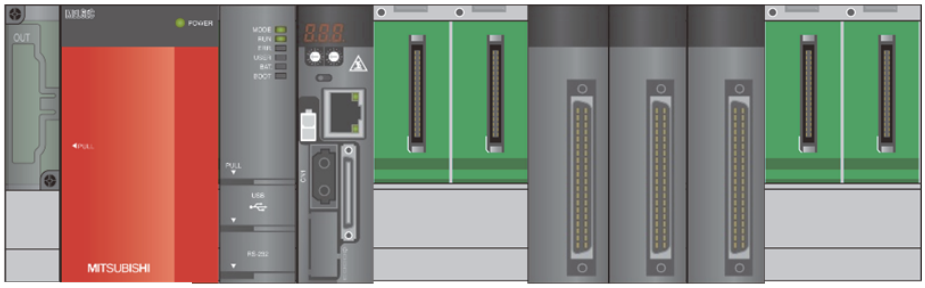


: Fixed to 0 at refresh
: Fixed to 0 at refresh (I/O reading from outside the group enabled)/ Refresh (I/O reading from outside the group disabled)
: Input refresh
: No refresh (SET/RESET possible)

Ex.) The example system configuration:
- Self CPU: CPU No.2
- Input module × 3 (32 points)
  I/O slot No.3 (controlled by CPU No.1), No.4 (controlled by CPU No.2, X40 to X5F),
  No.5 (controlled by CPU No.1)
- With no extension base unit

[Q17nDSCPU]

	Power supply	CPU	I/O0	I/O1	I/O2	I/O3	I/O4	I/O5	I/O6	I/O7
Start I/O No.	-	3E00	3E10	0000	0010	0020	0040	0060	0080	00A0
Points	-	-	-	16 points	16 points	32 points	32 points	32 points	32 points	16 points



CPU No.2 refresh area :

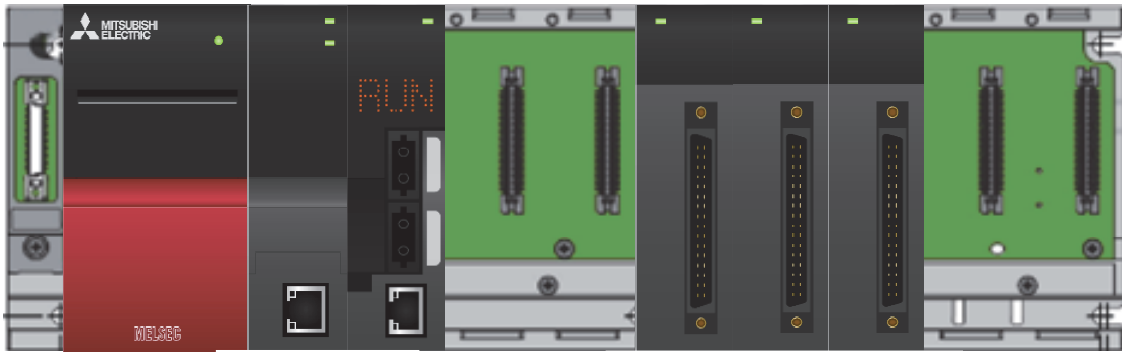


X devices other than X40 to X5F assigned as the actual inputs can be set/reset in the program.

[RnMTCPU]

Ex.) When the self CPU controls 1 or more modules, or "I/O reading from outside the group" is valid

	Power supply	CPU	I/O0	I/O1	I/O2	I/O3	I/O4	I/O5	I/O6	I/O7
Start I/O No.	-	3E00	3E10	0000	0010	0020	0040	0060	0080	00A0
Points	-	-	-	16 points	16 points	32 points	32 points	32 points	32 points	16 points



Control CPU : No.1 No.2 No.1

CPU No.2 refresh area :

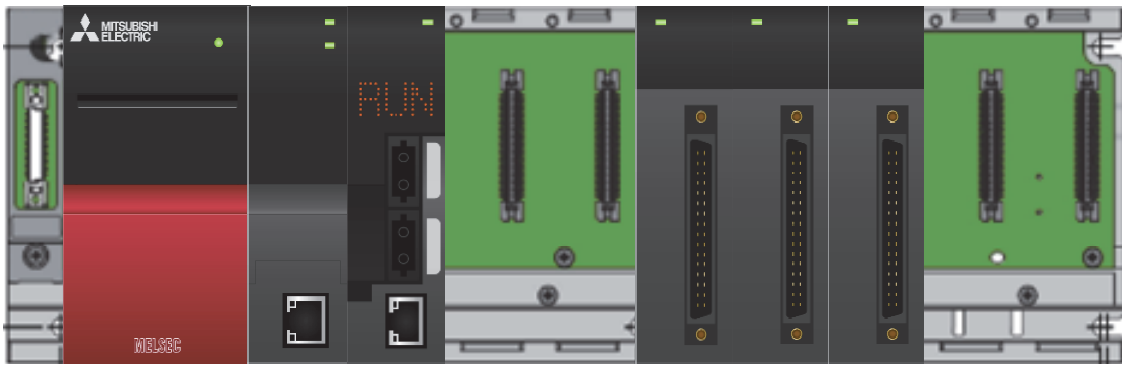


The X devices ranging from I/O No. 0 to actual I/O No. (regardless of control CPU) cannot be set/reset in the program. (In the system above, the X00 to X7F is the device area where the set/reset cannot be performed.) If the current Q17nDSCPU program uses X00 to X3F, and X60 to X7F as an internal relay in the program, replace them with devices of X80 or later.

<Information>

When no module is controlled by the self CPU and "I/O reading from outside the group" is invalid, all the X devices of No.2 CPU can be set/reset in the program.

	Power supply	CPU	I/O0	I/O1	I/O2	I/O3	I/O4	I/O5	I/O6	I/O7
Start I/O No.	-	3E00	3E10	0000	0010	0020	0040	0060	0080	00A0
Points	-	-	-	16 points	16 points	32 points	32 points	32 points	32 points	16 points



Control CPU : No.1 No.1 No.1

CPU No.2 refresh area :



5.2.10. SM/SD device automatic refresh

With Q17nDSCPU, Multiple CPU automatic refresh can be executed for the SM/SD devices, however, with the RnMTCPU, the automatic refresh cannot be executed for the SM/SD devices.
To perform automatic refresh for the SM/SD devices with the RnMTCPU, copy the SM/SD to other user devices (D/M, etc.) for automatic refresh, or exchange the data with DDRD/DDWR instruction of the PLC CPU.

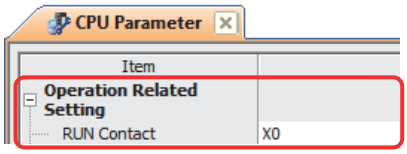
5.2.11. RUN/STOP

The RUN/STOP status of Q17nDSCPU is switched by directly operating M2000 (or M3072, D704) in the program. However, the RUN/STOP status of RnMTCPU cannot be switched by the same method.
Therefore, if RUN/STOP status is changed through direct operation, modify the program so that a RUN contact for remote operation is used to switch the RUN/STOP status.
The following shows the procedure for the program revision.

[Q17nDSCPU]

Procedure	Description
1) Direct operation of M2000 (or M3072, D704) in the program	Changes CPU operation status.

[RnMTCPU]

Procedure	Description
1) Set a RUN contact in the [CPU Parameter] settings of MT Works2	Set a X device for RUN contact (X0 to X2FFF) 
2) Change the X device status	CPU operation status can be changed by changing the status of the X device set in 1). • RUN contact is OFF: CPU module is in RUN status. • RUN contact is ON: CPU module is in STOP status. During this operation, the RUN/STOP switch on product must be in RUN status position.

Point

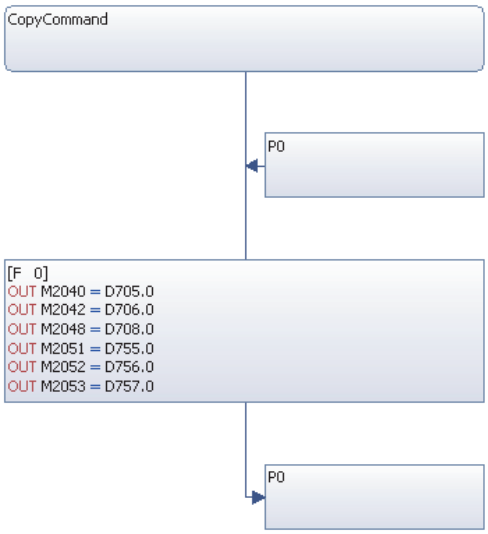
• M3072 and D704 have become unusable in RnMTCPU. They cannot be used as a status device.

5.2.12. Common bit device SET/RST request register

The positioning dedicated signals of Q17nDSCPU include common devices SET/RST request register (D705 to D708, D755 to D757) and bit devices (M3073 to M3079) as command devices from other CPUs. (The following devices)
These devices do not exist in RnMTCPU. To reflect requests from other CPUs to positioning dedicated signals, create an additional Motion SFC program

Function	Q17nDSCPU		RnMTCPU
	Request register	Bit device	Command bit device
Speed switching point specified flag request	D705	M3073	M2040
All-axis servo ON command	D706	M3074	M2042
JOG operation synchronous start command	D708	M3076	M2048
Manual pulse generator 1 enable flag	D755	M3077	M2051
Manual pulse generator 2 enable flag	D756	M3078	M2052
Manual pulse generator 3 enable flag	D757	M3079	M2053

Program example (A program in which D705 to D708, or D755 to D757 are used as the request devices)



Point

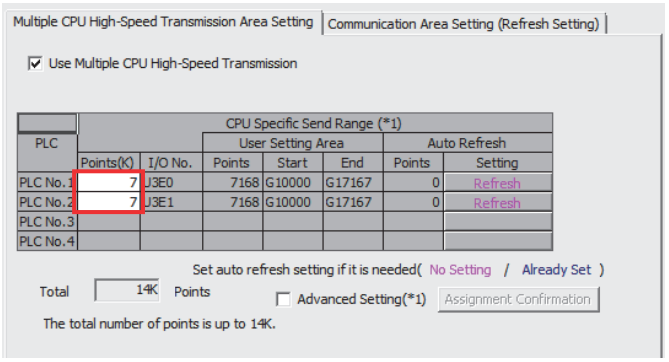
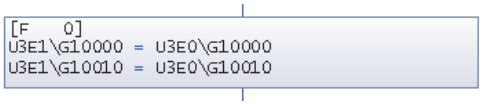
The least significant bits of D705 to D708, D755 to D757 are used for command in Q17nDSCPU. Therefore, in the program above, the least significant bits of the register devices (D) are set to M devices

5.2.13. CPU buffer memory (fixed scan communication area)

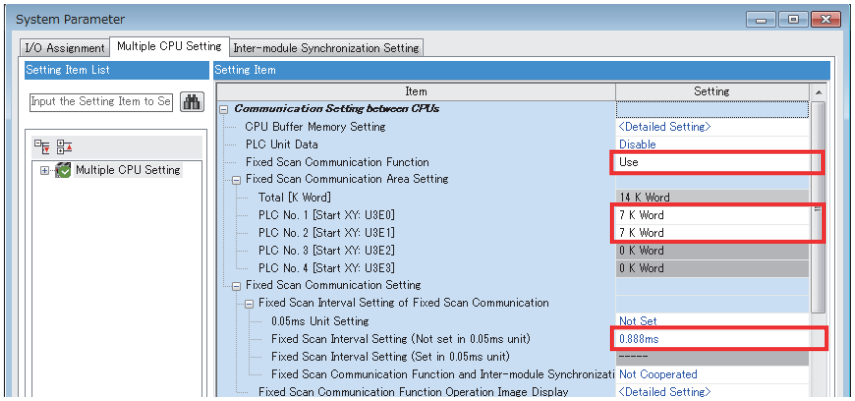
"Multiple CPU high speed transmission area" (from U3E□\G10000) in the existing system has been changed to the "CPU buffer memory (fixed scan communication area)" (from U3E□\HG0) in the MELSEC iQ-R series. Therefore, change the devices for the transmission area to those for the CPU buffer memory and set the CPU fixed scan communication in the system parameters.
When starting the CPU fixed scan communication, execute the EI instruction in a PLC CPU, and with an interrupt enabled, execute the Multiple CPU synchronous interrupt program (I45)(Note-1).

(Note-1): Add a sequence program to execute an I45 interrupt.

[Q17nDSCPU]

Procedure	Description
1) Set "Multiple CPU high speed transmission area".	• Make the same setting for the Motion CPU and the PLC CPU. 
2) Programming	• Create a program by using the devices for multiple CPU high-speed transmission area (from "U3E□\G10000"). 

[RnMTCPU]

Procedure	Description
1) Set "Fixed Scan Communication Setting".	• Set the system parameter in GX Works3, and read that data by MT Developer2. The following shows the details of the system parameter settings. [Fixed Scan Communication Function]: Use [Fixed Scan Communication Area Setting]: The same area as Q17nDSCPU or more [Fixed Scan Interval Setting]: 0.888 ms (to make the data refresh timing equivalent to that of Q17nDSCPU.) Setting example in MELSOFT GX Works3 
2) Programming	• Create a program by using the devices for CPU buffer memory (from "U3E□\HG0") • When Q17nDSCPU projects are diverted, the devices are not automatically replaced for the new system. Replace the devices manually, and if needed, use the replace device batch function, etc. of MT Developer2. 

5.3. High-Speed Input Request Signal, Mark Detection Signal

The following shows the differences in high-speed input request signal and mark detection function between Q17nDSCPU and RnMTCPU.

5.3.1. High-speed input request signal

The following shows the differences in settings and specifications of high-speed input request signal between Q17nDSCPU and RnMTCPU.

<table>
<tr><th>Setting items</th><th colspan="2">Q17nDSCPU</th><th></th><th>Setting items</th><th>RnMTCPU</th></tr>
<tr><td>Signal type</td><td colspan="2">• DI signal
• Q172DLX (DOG/CHANGE)
• Q172DEX (TREN)
• Q173DPX (TREN)</td><td rowspan="6"></td><td>• Bit device
• Amplifier input</td><td>• In RnMTCPU, the signal type has been changed. Select from "Bit device" or "Amplifier input".
• When using signals from input modules, set the signal type to "Bit device", and use the devices assigned to the input module.
• When using input signals from servo amplifiers, set the signal type to "Amplifier input", and set the axis No. of the servo amplifier and the signal type (DI1 to DI3).</td></tr>
<tr><td rowspan="2">Signal detection direction</td><td>DI signal</td><td>CPU input setting of system setting</td><td rowspan="2">• Leading edge
• Trailing edge
• Both direction</td><td rowspan="2">• In Q17nDSCPU, the signal detection direction is set in each module or the system setting of the CPU. In RnMTCPU, the signal detection direction can be set on the "High-speed Input Request Signal" screen.</td></tr>
<tr><td>Motion modules</td><td>Each Motion module setting</td></tr>
<tr><td>Signal accuracy</td><td colspan="2">No setting
(comparable to "general")</td><td>• General purpose
• High-accuracy</td><td>• When selecting "Bit device" for signal type, signals are detected with "high-accuracy". (Inter-module synchronization must be set.).</td></tr>
<tr><td>Compensation time</td><td colspan="2">-5000000 to 5000000 μs</td><td>-5000000 to 5000000 μs</td><td>• The same setting procedure as Q17nDSCPU.</td></tr>
<tr><td>Enable flag</td><td colspan="2">Bit devices (can be omitted)</td><td>Bit devices (can be omitted)</td><td rowspan="2">• The same setting procedure as Q17nDSCPU.</td></tr>
<tr><td>Status</td><td colspan="2">Bit devices (can be omitted)</td><td>Bit devices (can be omitted)</td></tr>
</table>

5.3.2. Mark detection function

With Q17nDSCPU, three options are available for mark detection signal: "Device", "DI signal", or "Q172DLX (DOG)". With RnMTCPU, however, "high-speed input request signal" is only available for mark detection.
In addition, the mark detection data item "Motor actual current value" has been eliminated. If the item is selected, change it to "Real current value". The data to be stored is the same as the "Motor actual current value". Other setting items have not been changed from those of Q17nDSCPU.

Point

- When using the high-speed input request signal (mark detection function), input response time and signal detection accuracy vary depending on the modules and servo amplifiers used.
- Adjust the signal detection timing by checking the following items:
  - [when using an input module] input response time and compensation time settings
  - [when using amplifier input] input filter and compensation time settings

6. DEVICE COMPARISON

6.1. Internal Relays

• Common devices (status)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
M2035	-	Motion error history clear request	The Motion CPU-specific errors have been integrated into the self-diagnostic errors. (Refer to section 3.2.)
M2039	-	Motion error detection	
M2041	-	System setting error	
M2047	-	Motion slot error detection	

• Common devices (command signal)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
M3072	M2000	PLC ready	Multiple command signals have been integrated into one. When using the devices before integration, modify the program. (Refer to section 5.2.12.)
M3073	M2040	Speed switching point	
M3074	M2042	All-axis servo ON command	
M3076	M2048	JOG operation synchronous start command	
M3077	M2051	Manual pulse generator 1 enable	
M3078	M2052	Manual pulse generator 2 enable	
M3079	M2053	Manual pulse generator 3 enable	
M3080	-	Motion error history clear request	"MELSOFT MT Works2 Motion CPU error batch monitor" clears the error history.

6.2. Data Register

• Common devices (command signal)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
D704	M2000	PLC ready flag request	Multiple command signals have been integrated into one. When using the devices before integration, modify the program. (Refer to section 5.2.12.)
D705	M2040	Speed switching point specified flag request	
D706	M2042	All-axis servo ON command request	
D708	M2048	JOG operation synchronous start command request	
D755	M2051	Manual pulse generator 1 enable flag request	
D756	M2052	Manual pulse generator 2 enable flag request	
D757	M2053	Manual pulse generator 3 enable flag request	

6.3. Motion Register

• Motion register (monitor device)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	#8016+20n	Servo amplifier vendor ID	New device in RnMTCPU

(Note): "n" indicates the corresponding axis No. (Axis No.1 to 32: n = 0 to 31).

• Motion register

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
#8640 - #8735	SD10 - SD25	Motion error history device	Motion error history is checked with the MELSOFT MT Works2 Motion CPU error batch monitor.

6.4. Special Relay

• Error information (SM0 to SM199)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	SM0	Latest self-diagnosis error detection	The Motion CPU-specific errors have been monitored with these devices. (Refer to section 3.2.)
-	SM1	Latest self-diagnosis error detection (Not including ON of annunciator)	
-	SM4	Warning detection	
-	SM50	Error reset	New device in RnMTCPU When resetting diagnostic error information in M2039, errors are reset by SM50.
SM51	-	Battery low latch	Not required since the Motion CPU is battery-less.
SM52	-	Battery low	
SM58	-	Battery low warning latch	
SM59	-	Battery low warning	
SM60	-	Fuse blown detection	
-	SM61	I/O module verify error	New device in RnMTCPU
-	SM80	Detailed information 1: Flag in use	
-	SM112	Detailed information 2: Flag in use	

• System information (SM200 to SM399)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	SM203	STOP contact	New device in RnMTCPU
SM211	-	Clock data error	The clock data has become unnecessary since the clock data of RnMTCPU is synchronized with that of CPU No.1.
SM801	SM213	Clock data read request	
SM244	SM230	No.1 CPU error	
SM245	SM231	No.2 CPU error	
SM246	SM232	No.3 CPU error	
SM247	SM233	No.4 CPU error	
SM526	SM360	Over heat warning latch	
SM527	SM361	Over heat warning	

• System clock, system counter (SM400 to SM499)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	SM480	Motion operation cycle over	New device in RnMTCPU
-	SM484	Fixed scan data transmission section over	
-	SM488	Inter-module synchronization signal error detection	

• Motion dedicated information (SM500 to SM799)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
SM510	-	TEST mode request error	Check the error which is displayed on the test mode screen of MELSOFT MT Developer2.
SM513	-	Manual pulse generator axis setting error	Errors detected in the Motion CPU are assigned to self-diagnostic error codes. (Refer to section 3.2.)
SM516	-	Servo program setting error	
SM528	-	No.1 CPU MULTR complete	Not required since MULTR instructions have been deleted. When using MULTR instructions, modify the program by referring to section 5.2.5.
SM529	-	No.2 CPU MULTR complete	
SM530	-	No.3 CPU MULTR complete	
SM531	-	No.4 CPU MULTR complete	
SM561	-	Multiple CPU synchronous control initial complete flag	This device does not exist in RnMTCPU because multiple CPU synchronous control is not supported.
-	SM600	Memory card enabled/disabled flag	New device in RnMTCPU
-	SM601	Memory card protect flag	
-	SM603	Memory card (drive 2) flag	
-	SM605	Memory card remove/attach prohibit flag	
-	SM606	SD memory card forced disable instruction	
-	SM607	SD memory card forced disable state flag	
-	SM634	Number of rewriting operations error to data memory flag	
-	SM752	EI flag	
SM503	SM760	Sampling settings RUN status	
-	SM761	Sampling settings trigger status	
-	SM762	Saving sampling settings	
-	SM765	Sampling settings sampling error	

6.5. Special Register

• Error information (SD0 to SD199)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	SD0	Latest self-diagnostic error code	Errors detected in the Motion CPU are monitored with these devices. (Refer to section 3.2.)
-	SD1	Clock time for latest self-diagnostic error occurrence (Year)	
-	SD2	Clock time for latest self-diagnostic error occurrence (Month)	
-	SD3	Clock time for latest self-diagnostic error occurrence (Day)	
-	SD4	Clock time for latest self-diagnostic error occurrence (Hour)	
-	SD5	Clock time for latest self-diagnostic error occurrence (Minute)	
-	SD6	Clock time for latest self-diagnostic error occurrence (Second)	
-	SD7	Clock time for latest self-diagnostic error occurrence (Day of week)	
-	SD10 - SD25	Self-diagnostic error code	
-	SD61	I/O module verify error module number	
-	SD80	Detailed information 1 information category	
-	SD81 - SD111	Detailed information 1	
-	SD112	Detailed information 2 information category	
-	SD113 - SD143	Detailed information 2	

• System information (SD200 to SD399)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	SD201	LED status	New device in RnMTCPU
SD210 (Upper 8 bit)	SD210	Clock data (Year)	
SD210 (Lower 8 bit)	SD211	Clock data (Month)	
SD211 (Upper 8 bit)	SD212	Clock data (Day)	
SD211 (Lower 8 bit)	SD213	Clock data (Hour)	
SD212 (Upper 8 bit)	SD214	Clock data (Minute)	
SD212 (Lower 8 bit)	SD215	Clock data (Second)	
SD213	SD216	Clock data (Day of week)	
-	SD218	Time zone setting value	
-	SD228	Number of CPU modules	New device in RnMTCPU
SD359	SD229	CPU module number in Multiple CPU system	
-	SD230	CPU No.1 operating status	New device in RnMTCPU
-	SD231	CPU No.2 operating status	
-	SD232	CPU No.3 operating status	
-	SD233	CPU No.4 operating status	
-	SD241	Number of extension base units	
-	SD242	Identification for whether or not Q series module can be mounted	
-	SD243 to SD244	Number of base slots	
-	SD250	Latest I/O for implemented unit	
SD290	SD260 - SD261	X number of points assigned	
SD291	SD262 - SD263	Y number of points assigned	
SD292	SD264 - SD265	M number of points assigned	
SD294	SD266 - SD267	B number of points assigned	
SD295	SD270 - SD271	F number of points assigned	
SD302	SD280 - SD281	D number of points assigned	
SD303	SD282 - SD283	W number of points assigned	
-	SD350 - SD351	Accumulative power-on time	New device in RnMTCPU
-	SD360	Internal temperature of Motion CPU	

• System clock, system counter (SD400 to SD499)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	SD480	Motion CPU operation cycle over count	New device in RnMTCPU
-	SD484	Fixed scan data transmission section over count	

• Motion dedicated information (SD500 to SD799)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
SD500 - SD501	-	Real mode axis information register	The Motion CPU-specific errors have been integrated into the self-diagnostic errors. (Refer to section 3.2.)
SD504 - SD506	-	Real mode/virtual mode switching error information	
SD510 - SD511	-	Test mode request error information	
SD513 - SD515	-	Manual pulse generator axis setting error	
SD516	-	Error program No.	
SD517	-	Error item information	
SD550 - SD551	-	System setting error information	
-	SD554	File transfer status (status)	New device in RnMTCPU
-	SD556 - SD559	Servo parameter change flag	
SD560	-	Operation method	Not required since the operation methods have been integrated.
SD561	-	Multiple CPU synchronous control setting	This device does not exist in RnMTCPU because multiple CPU synchronous control is not supported.
-	SD562 - SD563	Scan time	New device in RnMTCPU
-	SD564 - SD565	Maximum scan time	
-	SD566 - SD567	Motion SFC normal task processing time	
-	SD568 – SD569	Maximum Motion SFC normal task processing time	
-	SD570	Motion SFC event task (14.222 ms) operation time	
-	SD571	Motion SFC event task (7.111 ms) operation time	
-	SD572	Motion SFC event task (3.555 ms) operation time	
-	SD573	Motion SFC event task (1.777 ms) operation time	
-	SD574	Motion SFC event task (0.888 ms) operation time	
-	SD575	Motion SFC event task (0.444 ms) operation time	
-	SD576	Motion SFC event task (0.222 ms) operation time	

[Continued]

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	SD578	Motion SFC event task (External interrupt) operation time	New device in RnMTCPU.
-	SD579	Motion SFC event task (PLC interrupt) operation time	
-	SD580	Motion SFC NMI task processing time	
-	SD581	Fixed-cycle system processing time	
-	SD582	Motion operation task processing time	
-	SD583	CPU refresh (I45 executing) processing time	
-	SD584	Motion SFC event task time within operation cycle (14.222 ms)	
-	SD585	Motion SFC event task time within operation cycle (7.111 ms)	
-	SD586	Motion SFC event task time within operation cycle (3.555 ms)	
-	SD587	Motion SFC event task time within operation cycle (1.777 ms)	
-	SD588	Motion SFC event task time within operation cycle (0.888 ms)	
-	SD589	Motion SFC event task time within operation cycle (0.444 ms)	
-	SD590	Motion SFC event task time within operation cycle (0.222 ms)	
-	SD592	Motion SFC event task (external interrupt) time within operation cycle	
-	SD593	Motion SFC event task (PLC interrupt) time within operation cycle	
-	SD594	Motion SFC NMI task time within operation cycle	
-	SD595	Fixed-cycle system processing time within system operation cycle	
-	SD596	Motion operation task time within operation cycle	
-	SD597	CPU refresh (I45 executing) time within operation cycle	
-	SD600	Memory card mounting status	
-	SD606 - SD607	SD memory card capacity	
-	SD610 - SD611	SD memory card free space	
-	SD622 - SD623	Standard ROM capacity	
-	SD624 - SD625	Standard ROM free space	

[Continued]

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	SD634 - SD635	Index for the number of standard ROM write operations	New device in RnMTCPU
-	SD718 - SD719	888 μs free-running timer	
#8736 - #8751	SD740 - SD747	Operating system software version	
-	SD750	Words used information	
-	SD760	Sampling settings storage target	
-	SD761	Sampling settings results save target	
-	SD762	Sampling settings sampling type	
-	SD764 to SD765	Sampling settings latest file information	
-	SD769	Sampling settings digital oscilloscope error cause	

• Command signal (SD800 to SD1999)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
SD803	SD803	SSCNET control (command)	When using amplifier-less operation function, the setting method has been changed. Refer to "MELSEC iQ-R Motion Controller Programming Manual (Common)".
-	SD820	File transmission request (command)	New device in RnMTCPU.
-	SD860	Sampling settings storage target	

6.6. CPU Buffer Memory (CPU Shared Memory)

• Self CPU operation information area
Replace the Q17nDSCPU devices with the corresponding SD devices of RnMTCPU.

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
U3E□\G0	-	Information availability	
U3E□\G1	SD0	Diagnostic error	
U3E□\G2 - U3E□\G4	SD1 - SD7	Time the diagnostic error occurred	Displays the error occurrence time by Year/Month/Day/Hour/Minutes/Second Q17nDSCPU: 3 words RnMTCPU: 7 words
U3E□\G5	SD80, SD112	Error information identification code	Q17nDSCPU: The latest error data RnMTCPU: The latest two error data
U3E□\G6 - U3E□\G27	SD81 - SD111, SD113 - SD143	Common error information, Individual error information	
U3E□\G29	SD200	Switch status	
U3E□\G31	SD203	CPU operation status	

7. REVISIONS

Version	Date	Description
-	November 2018	First edition
A	November 2025	[Additional correction] When replacing a program with the data in 1 word, corrections to errors, etc.
B	November 2026	Structure revision and error corrections