MITSUBISHI ELECTRIC

SERVO SYSTEM CONTROLLER TECHNICAL BULLETIN [1 / 57]

[Issue No.] SSC-A-0001
[Title] Migration Guide of Motion Controller [Q17nDSCPU → RnMTCPU]
[Date of Issue] November 2018
[Relevant Models] R32MTCPU, R16MTCPU, Q173DSCPU, Q172DSCPU

Thank you for your continued support of Mitsubishi Electric Servo System Controllers.
This bulletin provides points and cautions when migrating the existing system using Q173DSCPU/Q172DSCPU (hereinafter called Q17nDSCPU) to a new system using R32MTCPU/R16MTCPU (hereinafter called RnMTCPU).
This document, however, does not provide detailed information in changing mechanical system program to advanced synchronous control. Refer to the following migration guide.
• Motion Controller Replacement of Virtual mode with Advanced synchronous control (L(NA)03123)

Point

When no equivalent models exist in MELSEC iQ-R series for the MELSEC-Q series models currently used, use RQ extension base units.
When replacing the controllers of the existing machine, the terminals and connectors may be changed. Refer to the “MELSEC iQ-R Module Configuration Manual” (SH-081262ENG) and user’s manual of the module.

The product lines in this document are based on the ones as of May 2017. As for the Motion controller operating system software (OS), the specifications are based on OS ver.12. As for the engineering tool, MELSOFT MT Works2 with Ver.1.146C is used.
The contents are subject to change without notice due to new product addition and specification change. Please refer to the latest information at the time of considering the 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

[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
MELSEC iQ-R Motion Controller Programming Manual (Machine Control)	IB-0300309

(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 iQ-R Programming Manual (Program Design)	SH-081265ENG
MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks)	SH-081266ENG
MELSEC iQ-R Inter-Module Synchronization Function Reference Manual	SH-081401ENG

(3) Servo amplifier

Manual title	Manual No.
MR-J4-_B_(-RJ) Servo Amplifier Instruction Manual	SH-030106
MR-J4W2-_B/MR-J4W3-_B/MR-J4W2-0303B6 Servo Amplifier Instruction Manual	SH-030105
MR-J3-_B Instruction Manual	SH-030051
Linear Servo MR-J3-_B-RJ004U_ Instruction Manual	SH-030054
Fully Closed Loop Control MR-J3-_B-RJ006 Instruction Manual	SH-030056
MR-J3W-0303BN6/MR-J3W-_B Instruction Manual	SH-030073
Direct Drive Servo MR-J3-_B-RJ080W Instruction Manual	SH-030079
Drive Safety integrated MR-J3-_B Safety Instruction Manual	SH-030084

1. MAIN TARGET MODELS FOR MIGRATION

The main target models and operating system software for replacement are as follows.
If you are using special operating system software or application-specific operating system software, contact your local sales office.

1.1 System Component

1.1.1 Table of system component

When replacing the existing controller with RnMTCPU, be sure to use MELSEC iQ-R series compatible system components.

Product name		Model	
		Q17nDSCPU	RnMTCPU
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
CPU module No.1	PLC CPU module	QnUD(E)(H)(V)CPU	RnCPU
	C Controller module	Q06CCPU-V, Q12DCCPU-V	R12CCPU-V
		Q24/26DHCCPU-□	-
Motion CPU module		Q173DSCPU	R32MTCPU
		Q172DSCPU	R16MTCPU
Input module	AC	QX10(-TS)	RX10
		QX28	RX28
	DC (Positive common)	QX40(-S1)(-TS) QX41(-S1) QX42(-S1)	RX40C7 RX41C4 RX41C6HS RX42C4 (Positive common/negative common shared)
	DC (Negative common)	QX80(-TS) QX81(-S2) QX82(-S1)	
	DC (Positive common/ negative common shared)	QX70 QX71 QX72	
	DC high-speed (Positive common)	QX40H QX70H	RX40PC6H RX61C6HS (Positive common/negative common shared)
	DC high-speed (Positive common)	QX80H QX90H	RX40NC6H RX61C6HS
	DC/AC	QX50	-

[Continued]

Product name		Model	
		Q17nDSCPU	RnMTCPU
Output module	Relay	QY10(-TS) QY18A	RY10R2 RY18R2A
	Triac	QY22	RY2056
	Transistor (Sink type)	QY40P(-TS), QY50 QY41P QY42P QY71	RY40NT5P RY41NT2P RY42NT2P RY41NT2H
		QY70	-
	Transistor (Source type)	QY80(-TS) QY81P QY82P	RY40PT5P RY41PT1P RY42PT1P
	Transistor high-speed (Sink type)	QY41H	RY41NT2H
	Transistor (Independent)	QY68A	-
Input/output composite module	DC Input/transistor output	QH42P	RH42C4NT2P
		QX48Y57 QX41Y41P	-
Analog input module	Voltage/current input	Q64AD Q64ADH	R60AD4 R60ADH4
	Voltage input	Q68ADV	R60ADV8
	Current input	Q68ADI	R60ADI8
Analog input module (channel isolated)	Voltage/current input	Q64AD-GH	-
		Q68AD-G	R60AD8-G
	Current input	Q62AD-DGH Q66AD-DG	-
Analog output module	Voltage/current output	Q62DA(N) Q64DA(N) Q64DAH	R60DA4 R60DAH4
	Voltage output	Q68DAV(N)	R60DAV8
	Current output	Q68DAI(N)	R60DAI8
Analog output module (channel isolated)	Voltage/current output	Q62DA-FG	-
		Q66DA-G	R60DA8-G
Analogue input/output module		Q64AD2DA	-

[Continued]

Product name		Model		
		Q17nDSCPU		RnMTCPU
Servo external signals interface module		Q172DLX (32 points)		[Input module] RX10 (16 points) RX40C7 (16 points) RX41C4 (32 points) RX42C4 (64 points, High-accuracy setting not possible)
Synchronous encoder interface module		Q172DEX		[Servo amplifier for scale measurement function] MR-J4-_B-RJ (Note-1)
Manual pulse generator interface module		Q173DPX		[High-speed counter module] RD62D2 (Differential input, 2CH) RD62P2 (DC input, 2CH) RD62P2E (DC input, source type, 2CH) (Note-2)
Interrupt module		QI60		[Input module] RX10 (16 points) RX40C7 (16 points) RX41C4 (32 points) RX42C4 (64 points, High-accuracy setting not possible)
Safety signal module		Q173DSXY		-
Cable for forced stop input		Q170DEMICBL□M	Use either of them	Unnecessary
Connector for forced stop input		Q170DEMICON		
Serial absolute synchronous encoder		Q170ENC Q171ENC-W8		Q171ENC-W8
Serial absolute synchronous encoder cable		Q170ENCCBL□M		Q170ENCCBL□M-A
Battery unit		Motion CPU built-in battery		Unnecessary
Battery	For CPU module	Q6BAT		Unnecessary
	For synchronous encoder	A6BAT		Unnecessary (battery of a servo amplifier can be used)
Manual pulse generator		MR-HDP01		MR-HDP01 (Note-2)
Optical hub unit		MR-MV200		← (Same as Q17nDSCPU)
SSCNETIII cable		MR-J3BUS□M MR-J3BUS□M-A MR-J3BUS□M-B		← (Same as Q17nDSCPU)

(Note-1): A synchronous encoder is connected via the servo amplifier.
(Note-2): The existing MR-HDP01 can be used continuously with RnMTCPU.
In addition, Mitsubishi Electric has confirmed the operation of the following manual pulse generator.
Contact the manufacture for details.

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

1.1.2 Points and cautions for system components replacement

The following shows the points and cautions for system components replacement.

- The RnMTCPU only controls MELSEC iQ-R series modules. It cannot control MELSEC-Q series modules.
- The RnMTCPU does not have an EMI terminal. When the existing model executes forced stop by using the EMI terminal, input the forced stop signal to an input module, and assign the device of the input module for forced stop in the forced stop input settings ([Motion CPU Common Parameter] → [Basic Setting]).
- **RnMTCPU is battery-less.**

1.2 Operating System software

Use the operating system software (OS) for RnMTCPU. For Q17nDSCPU, the OS is available by application (Conveyor assembly use (SV13) and Automatic machinery use (SV22)). For RnMTCPU, the OS is integrated (SW10DNC-RMTFW only).

Q17nDSCPU				RnMTCPU	
OS type	CPU model	OS model		CPU model	OS type
SV13	Q173DSCPU	SW8DNC-SV13QJ		R32MTCPU R16MTCPU	SW10DNC-RMTFW (installed before shipment)
	Q172DSCPU	SW8DNC-SV13QL			
SV22	Q173DSCPU	SW8DNC-SV22QJ			
	Q172DSCPU	SW8DNC-SV22QL			

1.3 Engineering Environment (required)

Product name	Model	Version
MELSOFT MT Works2	SW1DND-MTW2-E (Note-1), (Note-2)	Ver.1.100E or later

(Note-1): The model name has been changed because it has become available as DVD.

(Note-2): Prepare the MELSOFT GX Works3 to create sequence programs and set R series CPU common parameters.

2. Differences Between Q17nDSCPU and RnMTCPU

2.1 Table of the differences and migration points

Items		Q17nDSCPU	RnMTCPU	Points for migration
Operation cycle (Default value)	SV13	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	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	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.
	SV22	0.444 ms / 1 to 6 axes 0.888 ms / 7 to 16 axes 1.777 ms / 17 to 32 axes		
Control method		Positioning 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, Tightening & press-fit control, Advanced synchronous control	Positioning 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	• 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 4.2.4.)
Motion dedicated PLC instruction	M(P).□	-	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	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)” for details.
	D(P).□	D(P).DDRD , (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	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.	• If the D(P).CHGT or D(P).CHGT2 is used, review the program. (Refer to section 4.2.3.) • Arguments of D(P).DDRD and D(P).DDWR have been changed. However, program revision is not required because the change is automatically reflected by MELSOFT GX Works3 at conversion.
Servo external signal		Q172DLX signal, Amplifier input, DI signal, Bit device	Bit device (When “Inter-module synchronization” is valid, “High accuracy” setting of actual input signal is possible), Amplifier input	
Cancel signal of servo program		Available	Not available	Delete the cancel command in the servo program, and assign the same signal to the external signal (STOP signal) or use “Rq.1140 Stop command”.

[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".
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 points (I/O module + Intelligent function module)	• When the existing program uses PX/PY, revise the program so that the PX/PY devices are replaced with the X/Y devices assigned in the system setting. • Refer to section 4.2.9 for details of the I/O assignment.
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. Review 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)	
	Coasting timers (FT)	1 point (888 μs)	SD718, SD719 (888 μs coasting 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 4.2.13.
		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 5.6.
	Module access device (U□\G)	-	Up to 268435456 points	
Bit specification of word device		Bit specification is possible in Multiple CPU area device only. (U3E□\G□.0 to F)	Bit specification is possible in all word devices.	

[Continued]

Items		Q17nDSCPU	RnMTCPU	Points for migration
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 I45 executing: 32 latch ranges	
	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 2.2.
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 SD0, and then SM0 and 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.
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.	• Change the input from the EMI terminal to 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.

[Continued]

Items		Q17nDSCPU	RnMTCPU	Points for migration
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. Review the setting. (Refer to section 4.3.2.)
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, revise the program. (Refer to section 4.2.11.)
Output mode setting of STOP to RUN		No option (Comparable to “Clear the output (Y)”)	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 MELSOFT 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.)	
LED display		7-segment LED	Dot matrix LED READY, ERROR, CARD READY, CARD ACCESS	More information can be indicated on the LED display, enabling to conduct troubleshooting more easily. (Refer to "MELSEC iQ-R Motion Controller User's Manual".).
Rotary switch		2 (Normal mode, mode operated by ROM, installation mode, all clear, Ethernet IP address display mode)	1 (Normal mode, installation mode, all clear, Ethernet IP address 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. Review 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 MELSOFT MELSOFT MT Works2 Motion CPU memory. • Cleaning built-in memory with Motion CPU rotary switch "C".	
	Latch (2)	It is possible to clear with lath clear (1)(2) of remote latch clear.	• Cleaning built-in memory with Motion CPU rotary switch "C".	

[Continued]

Items			Q17nDSCPU	RnMTCPU	Points for migration
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-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.)		
SSCNETIII/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-J4-_B, MR-J4W-_B, VCII, LJ71MS15		
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 (PHERIPHERAL I/F)			Provided	None	Use the Ethernet port of PLC CPU for the 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)	Revise the program.
Torque limit value			1 [%] unit (Some items are set by 0.1 [%] unit)	0.1 [%] unit	Revise the program. (Refer to section 4.2.2.)
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)	Revise the program.

[Continued]

Items		Q17nDSCPU	RnMTCPU	Points for migration
Operation control program	Motion dedicated instructions	CHGV, CHGVS, CHGT, CHGT2, CHGP	CHGV, CHGVS, CHGT(Note), CHGP (Note): Equivalent to D(P).CHGT2	If CHGT/CHGT2 instruction is used, revise the program. (Refer to section 4.2.3.)
	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, revise the program. (Refer to section 4.2.5.)
	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. Revise the program. (Refer to section 4.2.6.)
	Y/N transition	Provided	Provided	The description method of the program has been changed. Revise the program. (Refer to section 4.2.8.)
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)”.
Operating system software installation method		• MELSOFT MT Works2	• MELSOFT MT Works2 • SD memory card	The installation files have been consolidated into one, making management of the files easier.
Safety Observation function		Safety signal comparison, Safety communication, STO, SS1, SS2, SOS, SLS, SBC, SSM	None	Use the Safety CPU module (R□SFCPU-SET), Safety input/output, and functional safety unit (MR-D30)

[Continued]

Manual title	Q17nDSCPU	RnMTCPU	Points for migration
Other functions (New functions in RnMTCPU)	-	• Label access from external devices • File transmission at boot • ABS direction in degrees • Pressure control • Transient command • Multiple CPU advanced synchronous control setting • Vibration suppression command filter • Multi-axis test operation • Override • Parameter change • Machine control • Sensing module connection • Mixed operation cycle • Virtual servo amplifier • G-code control • File transmission via FTP server	Refer to “MELSEC iQ-R Motion Controller Programming Manual” for details of each function.

2.2 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	H0001 to H3FFF	These are errors such as module self-diagnostic errors that are different for each module.
Detection when communicating between modules	H4000 to H4FFF	CPU module error
	H7000 to H7FFF	Serial communication module error
	HB000 to HBFFF	CC-Link module error
	HC000 to HCFBF	Ethernet module error
	HD000 to HDFFF	CC-Link IE Field Network module error
	HE000 to HEFFF	CC-Link IE controller network module error
	HF000 to HFFFF	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		H0800 to H0FFF	Warnings which do not stop servo programs	• Equivalent to some of the Q17nDSCPU minor errors
Error	Minor	H1000 to H1FFF	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)	H3100 to H3BFF	Motion SFC execution errors The CPU continues to operate (in RUN status).	• Equivalent to Motion SFC errors of Q17nDSCPU.
	Moderate	H2000 to H30FF	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	H3C00 to H3FFF		• 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.

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 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) MELSOFT GX Works3 module diagnostics (error information list)
(d) MELSOFT MT Works2 Motion CPU error batch monitor (Motion error history)
(e) Axis status signals, and axis monitor devices (Error details detected for each axis)
(f) Event history
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)"

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 of MELSEC iQ-R Motion Controller Programming Manual (Common)” for details.

3. PROJECT DIVERSION

3.1 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 servo external signals interface 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 4.2.
	Transition program	△	Refer to section 4.2.
Servo program	Command generation axis program allocation	○	
	Servo program	△	Refer to section 4.2.
Cam data		△	When diverting the data from the virtual mode switching method, cam No. may be changed.

[Continued]

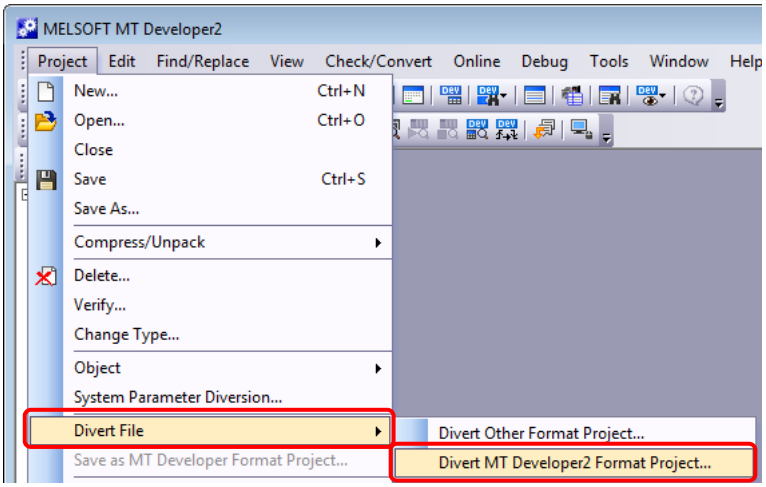
Data name	Divertible/ not divertible	Remarks
Label	○	
Structured data types	○	
Device memory	△	Special relays (SM) and special registers (SD) will not be diverted. Link relays (B), annunciators (F), and Link registers (W) are diverted within the range of the settable device points.
Device comment	○	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

3.2 Project diversion

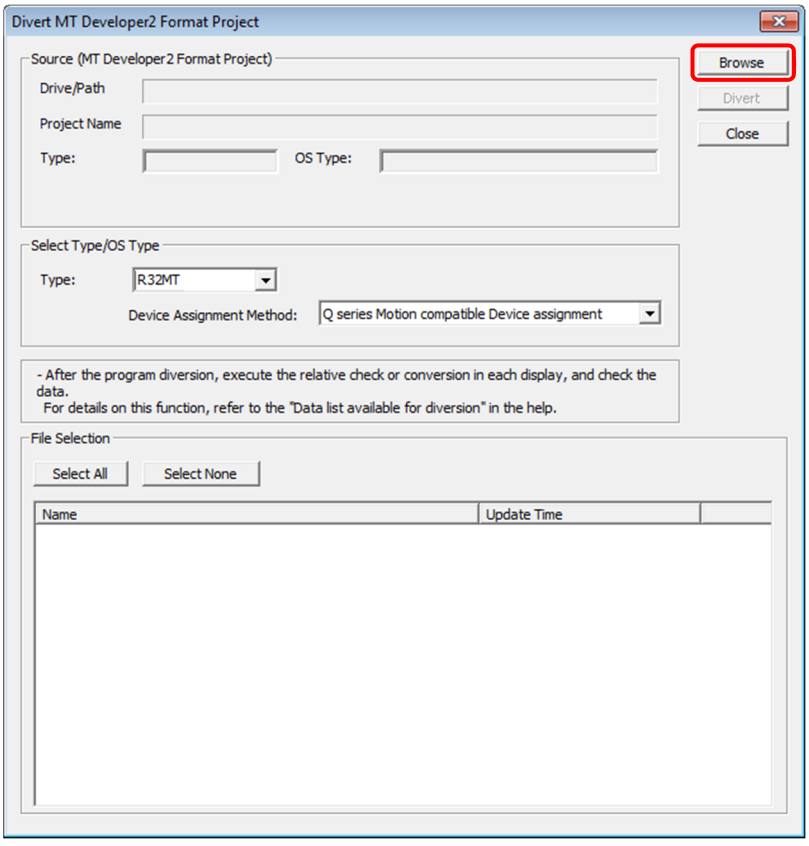
3.2.1 Motion CPU projects diversion by MELSOFT MT Works2

The following shows the procedure for diverting Q17nDSCPU projects by MELSOFT MT Works2.

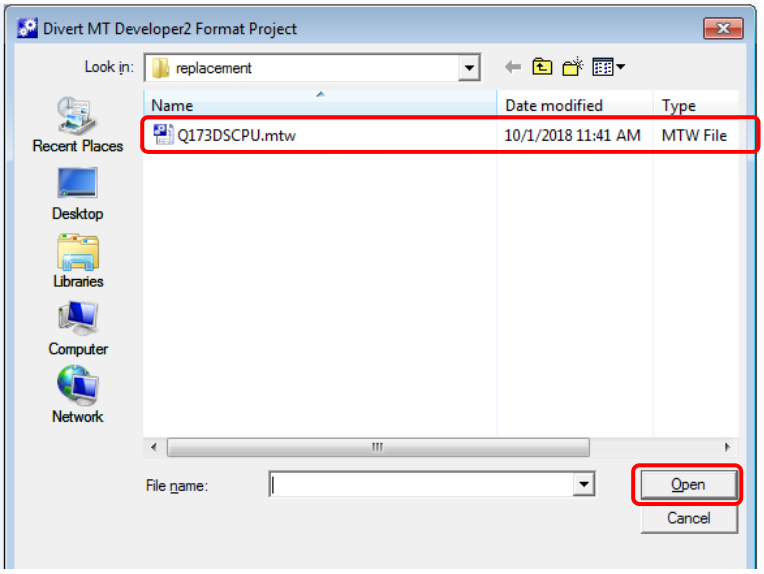
1) Start MELSOFT MT Developer2. Select [Divert file] - [Divert MT Developer2 Format Project…] from "Project" menu.



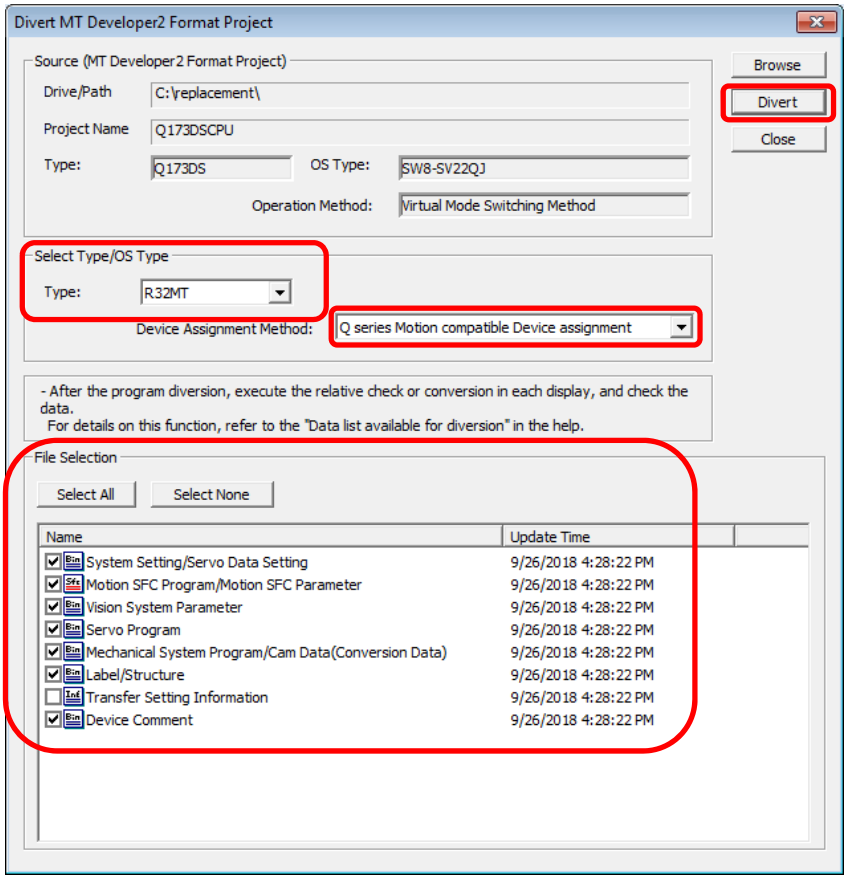
2) Click "Browse" on the "Divert MT Developer2 Format Project" screen.



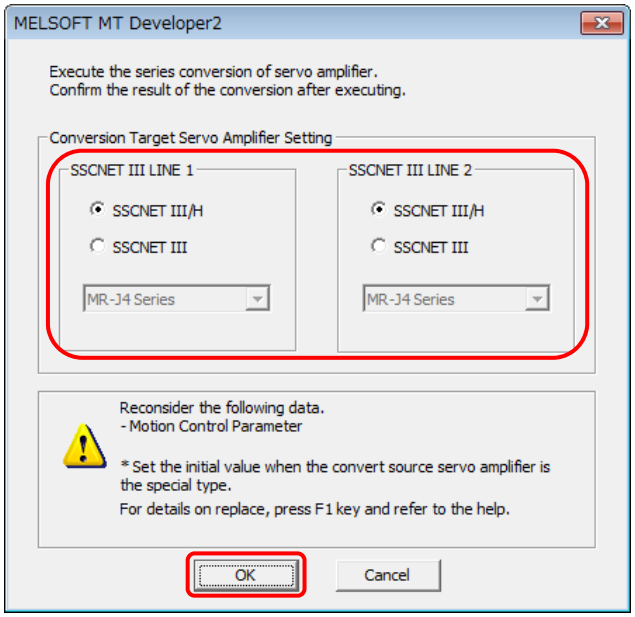
3) Select the project to be diverted on the file selection window. Click [Open] to update the selected project (MT Developer2 Format Project).



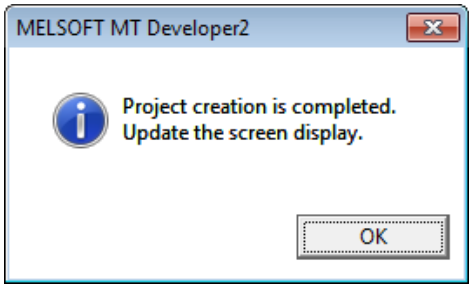
4) Select the replaced model for [Select Type/OS Type] (the setting example below: R32MTCPU). After the “Device Assignment Method” appears and becomes selectable, select “Q series Motion compatible Device assignment”. Then, check the box of the data to be diverted in the “File Selection”. Click “Divert”.



5) Execute the series conversion of the servo amplifier. Select the network to be used (SSCNETIII or SSCNETIII/H) for the replaced servo amplifiers (for Q17nDSCPU), and click [OK].



6) The diversion complete message will appear. After diversion, check each of the diverted data by executing a relative check or by converting the data. For R series common parameters, read them from MELSOFT GX Works3.



(Note): Projects in MELSOFT MT Developer are divertible by selecting “Divert MELSOFT MT Developer2 Format Project…” from the Project menu.
Refer to MELSOFT MT Works2 Help for details of the file diversion.

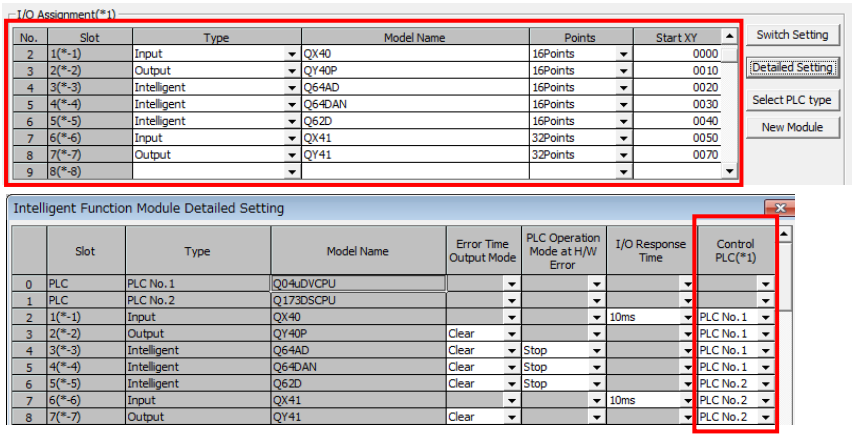
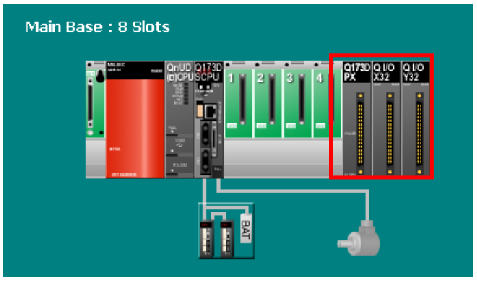
4. MIGRATION POINTS AND PRECAUTIONS

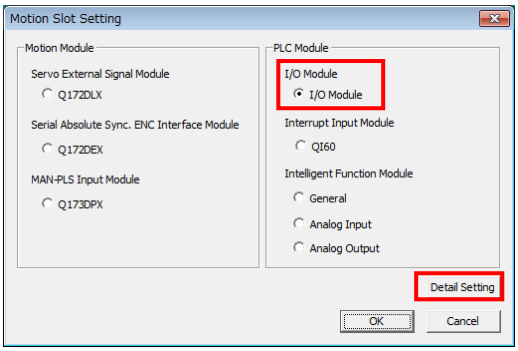
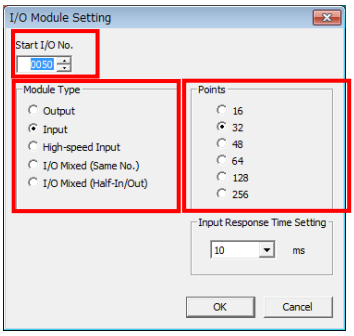
The following shows the points and precautions when migrating the existing system to a new one.

4.1 System Settings

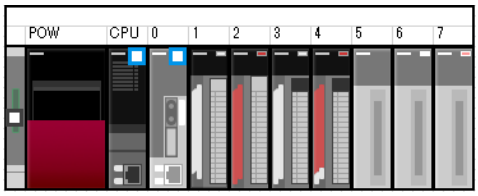
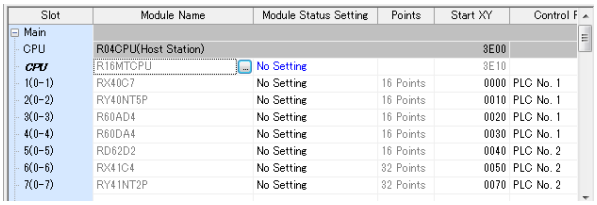
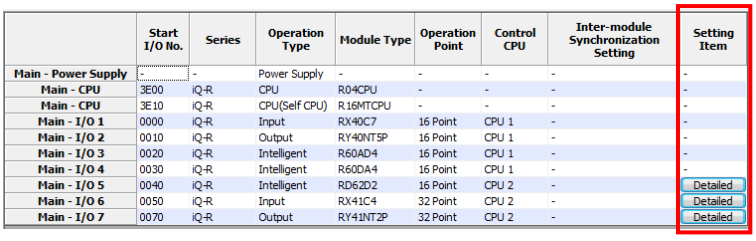
Refer to “MELSEC iQ-R Module Configuration Manual” for details of multiple CPU system settings (system settings including the installation position of CPU module, allocation of CPU No. and I/O No.)
The following shows the differences in system settings between Q17nDSCPU and RnMTCPU.

• System settings (Q17nDSCPU)
The system configuration is set separately in MELSOFT GX Works2 and MELSOFT 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 (MELSOFT GX Works2) (PLC CPU settings)	Set the multiple CPU system and I/O assignment in [Q parameter]. Specify the modules controlled by Motion CPU (CPU No.2) in [Detailed Setting]. 
2)	System configuration settings (MELSOFT MT Developer2) (Motion CPU 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].  Screen example of [System configuration]

	Setting procedure	Description
3)	Settings for each module (MELSOFT 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 MELSOFT GX Works3 is read by MELSOFT MT Developer2, and therefore the setting is not set in MELSOFT MT Developer2. The following shows an example that is equivalent to the example system in the previous “System settings for Q17nDSCPU”.

	Setting procedure	Description
1)	System configuration settings (MELSOFT GX Works3)	Set the system configuration and the common parameters in MELSOFT 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.
2)	System configuration settings (MELSOFT MT Developer2)	Read the parameters set in 1) by MELSOFT MT Works2 [System Parameter Diversion] (These common settings read from MELSOFT GX Works3 cannot be set in MELSOFT MT Developer2.)
3)	Module parameter settings (When using modules)	Set the parameters of the modules controlled by the Motion CPU (R32MTCPU/R16MTCPU). Select [R Series Common Parameters] → [Module Configuration List] → "Detailed” button in "Setting item" in MELSOFT MT Developer2. 

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
6(0-6)	RX41C4	No Setting	32 Points	0050	PLC No. 2
7(0-7)	RY41NT2P	No Setting	32 Points	0070	PLC No. 2

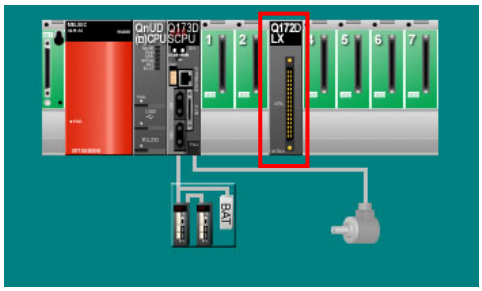
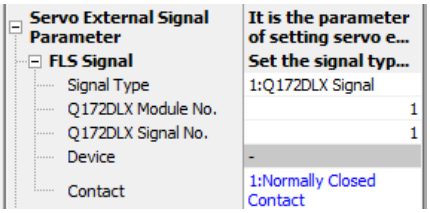
	Start I/O No.	Series	Operation Type	Module Type	Operation Point	Control CPU	Inter-module Synchronization Setting	Setting Item
Main - Power Supply	-	-	Power Supply	-	-	-		-
Main - CPU	3E00	iQ-R	CPU	R04CPU	-	-	-	-
Main - CPU	3E10	iQ-R	CPU(Self CPU)	R16MTCPU	-	-	-	-
Main - I/O 1	0000	iQ-R	Input	RX40C7	16 Point	CPU 1	-	-
Main - I/O 2	0010	iQ-R	Output	RY40NT5P	16 Point	CPU 1	-	-
Main - I/O 3	0020	iQ-R	Intelligent	R60AD4	16 Point	CPU 1	-	-
Main - I/O 4	0030	iQ-R	Intelligent	R60DA4	16 Point	CPU 1	-	-
Main - I/O 5	0040	iQ-R	Intelligent	RD62D2	16 Point	CPU 2	-	Detailed
Main - I/O 6	0050	iQ-R	Input	RX41C4	32 Point	CPU 2	-	Detailed
Main - I/O 7	0070	iQ-R	Output	RY41NT2P	32 Point	CPU 2	-	Detailed

[Points]
- 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 review the program. (An error will occur and writing operation cannot be performed.).

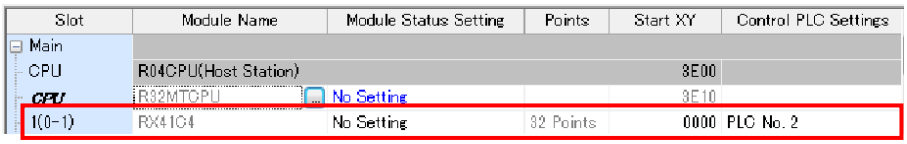
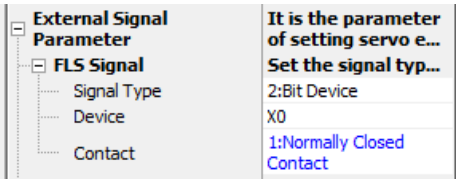
4.1.1 External signal input module

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 MELSOFT MT Developer2.

	Setting item	Setting details
1)	[System configuration] settings of MELSOFT MT Developer2	Set the Q172DLX servo external signal interface module on [System Configuration] screen. 
2)	[Servo Data] settings of MELSOFT 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 MELSOFT GX Works3, and the external signal parameter for each axis is set with MELSOFT MT Developer2.

	Setting item	Setting details
1)	[System parameter] settings of MELSOFT GX Works3	Set RX41C4 input module on the [System parameter] screen. Refer to Section 4.1 for setting method. 
2)	[Axis setting parameter] settings of MELSOFT MT Developer2	Set the external signal parameters (FLS, RLS, STOP, DOG) of the target axes as shown below on the [Axis setting parameter] screen. [Signal type] → 2: Bit device [Device] → X0 (X device No. of the input module set in 1)) 

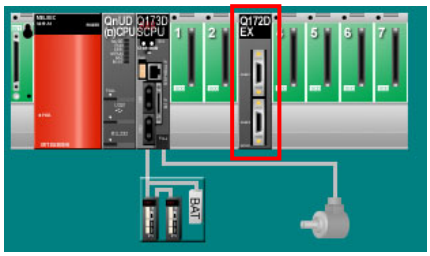
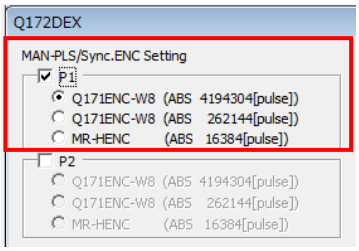
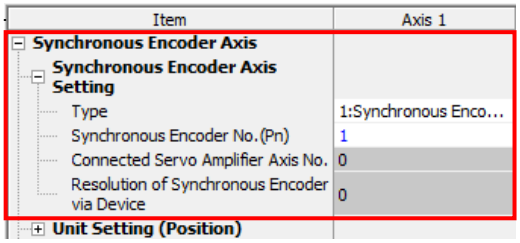
[Points]

- 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 signal to “High-accuracy”.
  Refer to “MELSEC iQ-R Motion Controller Programming Manual (Common)” for how to set the inter-module synchronization function.

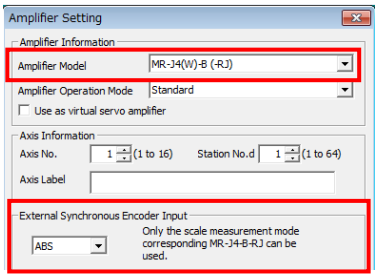
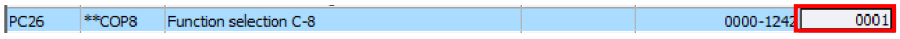
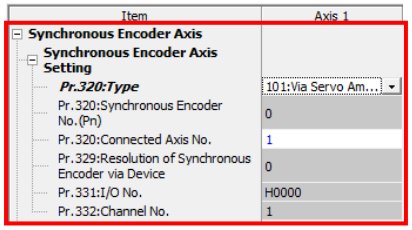
4.1.2 Synchronous encoder interface module

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 MELSOFT MT Developer2.

	Setting item	Setting details
1)	[System configuration] settings of MELSOFT MT Developer2	Set the Q172DEX on [System Configuration] screen  
2)	[Servo Data] settings of MELSOFT 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.) 

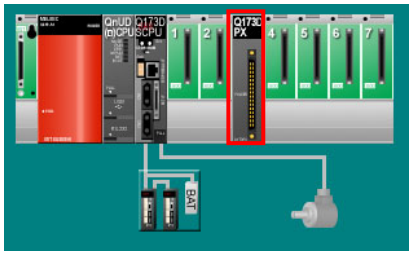
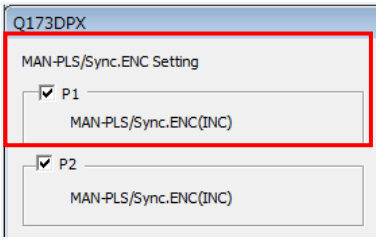
• Synchronous encoder settings (RnMTCPU)
RnMTCPU uses a synchronous encoder via a servo amplifier (MR-J4-_B-RJ).
The following shows a setting example which uses a synchronous encoder via MR-J4-_B-RJ.

Setting procedure		Setting details
1)	[Servo Network] settings of MELSOFT MT Developer2	Set the servo amplifer and the synchronous encoder on [Amplifier Setting].  [Amplifier Model]: MR-J4(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 communication method)”. 
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. 

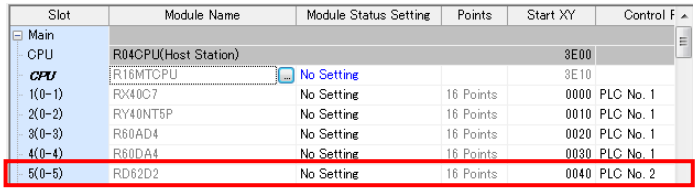
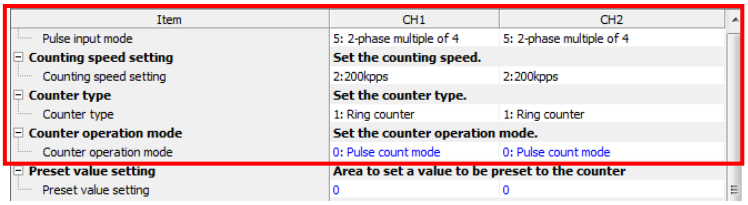
4.1.3 Manual pulse generator interface module

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

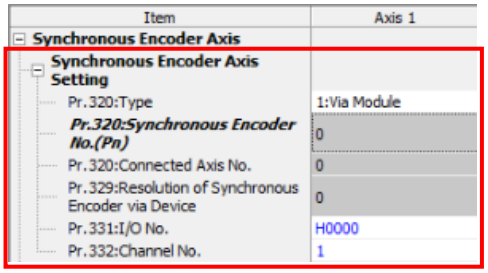
• Manual pulse generator settings (Q17nDSCPU)
Q17nDSCPU uses Q173DPX manual pulse generator interface module.
The module is set with MELSOFT MT Developer2.

	Setting item	Setting details
1)	[System configuration] settings of MELSOFT 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 MELSOFT GX Works3 and the external signals for each axis are set with MELSOFT MT Developer2.

	Setting item	Setting details
1)	[System Parameter] settings of MELSOFT GX Works3	Set the RD62D2 high-speed counter module in system paremter setting. Refer to section 4.1 for setting details. 
2)	[Module Detail Setting] settings of MELSOFT 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 operatoin mode] → Pulse count mode 

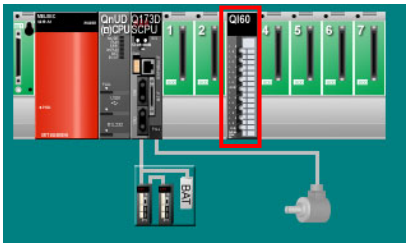
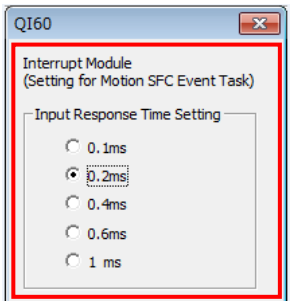
In addition, when the RnMTCPU uses an incremental synchronous encoder, the high-speed counter module can also be used. The high-speed counter module setting is the same as above, however, the axis setting is different. The following shows the setting details. Note that ring counter function, coincidence 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 MELSOFT 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 MELSOFT 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). 

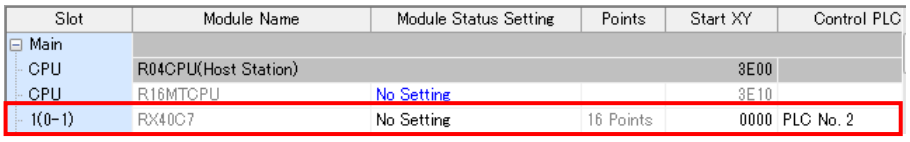
4.1.4 Interrupt module

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

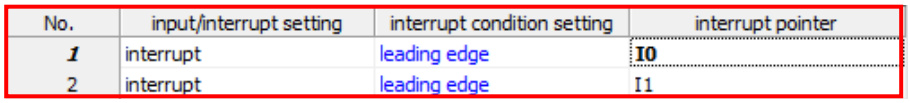
• Interrupt module settings (Q17nDSCPU)
Q17nDSCPU uses QI60 interrupt module.
The interrupt module is set with MELSOFT MT Developer2.

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

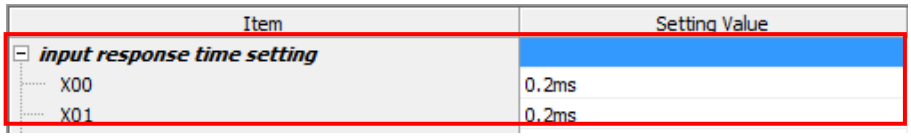
• Interrupt module settings (RnMTCPU)
RnMTCPU uses a common input module with the PLC CPU. The signals to be used for interrupt need to be set as “interrupt” in the “input/interrupt setting”. The following shows an example which uses RX40C7 input module.
The interrupt module is set with MELSOFT GX Works3,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

	Setting item	Setting details
1)	[System Parameter] settings of MELSOFT GX Works3	Set the RX40C7 input module in the system paremter settings. Refer to Section 4.1 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 MELSOFT 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]: 0.2ms 

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 interrput can be used as input devices.
• Change the setting items, [input/interrupt setting], [interrupt pointer], and [Input response time setting] according to your system.

4.2 Programming

4.2.1 Acceleration/deceleration time settings

The setting range of the acceleration/deceleration time is expanded from 1 word to 2 words in RnMTCPU. This change requires program revisions.
Refer to the following conditions for the revisions.

[Items which need a program revision]

Function	Item
Motion control parameter (Parameter block)	Acceleration time
	Deceleration time
	Rapid stop deceleration time
Servo program	Acceleration time
	Deceleration time
	Rapid stop deceleration time
	Fixed position stop, acceleration/deceleration time

[Program change procedure]

No.	Conditions		Program change procedure
1	Direct setting of the acceleration/ deceleration time		• No need to revise the program
2	Indirect setting of the acceleration /deceleration time	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 two 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 two words. Secure two words of devices starting from even number. • If the device is an odd number, an error occurs at program conversion.

4.2.2 Torque limit value settings

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

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 and a program revision is required.
Servo program	Torque limit value (common)	1 [%]		The unit is not automatically converted regardless of direct or indirect designation. A program revision is required.
	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, a program revision is required. Refer to 4.2.3 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 [%]		

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

In the new system, the CHGT/D(P).CHGT instructions have been changed to the instructions equivalent to CHGT2/D(P).CHGT2 of Q17nDSCPU. Therefore, CHGT2/D(P).CHGT2 instructions have been eliminated in RnMTCPU. Following the change, a program revision is required. The following shows the points and revision example.

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

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

[Points]

- The CHGT instruction of Q17nDSCPU sets the same value for both positive and negative directions, however, the CHGT instruction of RnMTCPU changes the value individually for positive and negative directions. Therefore, separate value for each direction is required in RnMTCPU program.
- The torque limit value unit differs between Q17nDSCPU and RnMTCPU (Q17nDSCPU: 1%, RnMTCPU: 0.1%). Be sure to multiply the value tenfold.
- The changes above are not automatically reflected by MELSOFT MT Developer2 or MELSOFT GX Works3 at project diversion. If the program is converted by MELSOFT MT Developer2 without revision, an error will occur at project diversion and write operation cannot be performed. If converted by MELSOFT GX Works3 without revision, the instruction will be changed to SM4095 (coil).

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

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

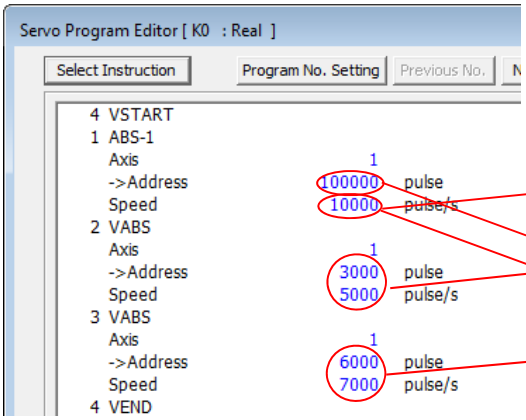
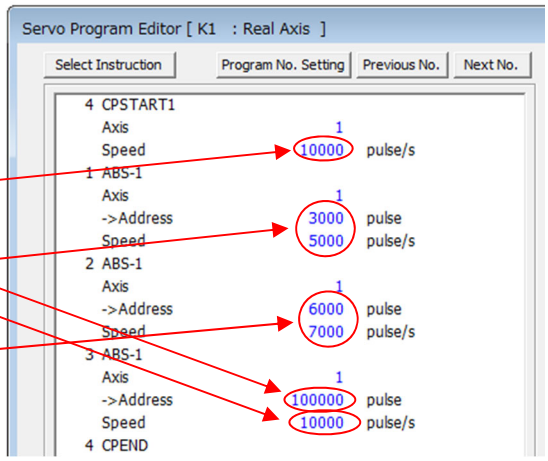
Q17nDSCPU		RnMTCPU
CHGT2 (K1 , K200 , K100)	→	CHGT (K1 , K200 , K100)
D(P).CHGT2 H3E1 “J1” K200 K100		D(P).CHGT H3E1 “J1” K200 K100

[Points]

- The replacement of CHGT2/D(P).CHGT2 instructions is done just by changing the name.
- The name is not automatically changed to the new one by MELSOFT MT Developer2 and MELSOFT GX Works3.
  Be sure to revise the program (If the name is not changed, an error will occur and write operation will not be performed.)

4.2.4 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.
The following shows the replacement points when changing the speed switching control to the continuous trajectory control.

[Speed switching control of Q17nDSCPU]	[Continuous trajectory control of RnMTCPU]
	

[Points]

- The speed switching control program begins with the end point address/movement amount. The speed is described as needed for each speed switching point.
  The continuous trajectory control program describes the address/movement amount and the speed for each point.

4.2.5 Write device data to CPU shared memory/Read device data from CPU shared memory

(1) MULTW/MULTR instructions

MULTW/MULTR instructions need to be 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 TO/FROM instruction, BMOV instruction, or 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

[Points]

- Make sure to review the Motion SFC program since the MULTW/MULTR instructions are not automatically converted for the new system at project diversion by MELSOFT MT Wroks2. (An error will occur at program conversion and write operation cannot be performed.)

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

If the specified I/O number cannot be found (the specified module does not exist) when Q17nDSCPU accesses other modules with MULTR instructions or FROM/TO instructions, a Motion SFC error will be outputted, however, the operation will continue.
With RnMTCPU, whether to stop or continue the program execution can be selected with parameter. ([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), change the parameter to “Continue”.

4.2.6 Synchronous control dedicated functions

An SD memory card is newly selectable for a cam data saving area in RnMTCPU. Due to the new addition, an argument has been added to each of the read/write instructions to specify the cam data read/write area. Following the changes, programs need to be revised.
The following shows the points and 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, and a program revision is required.
The following shows a program revision example.

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, 401H

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

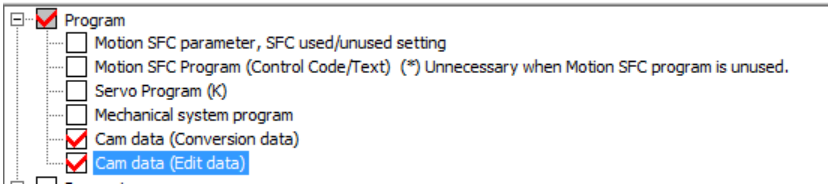
[Points]
- If the write area argument is omitted, the cam data will be written to the same area as when Q17nDSCPU uses CAMWR2.
- Set “401H” for the cam read area when writing the cam data to the same area as when the Q17nDSCPU uses CAMWR.

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

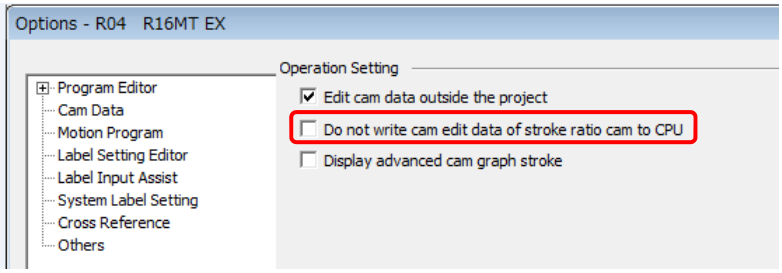
4.2.7 Cam conversion/editing data

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.
In RnMTCPU, these two types of cam data are integrated as “Cam data”. (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 MELSOFT MT Works2 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 ration cam to CPU”.)

[Q17nDSCPU]



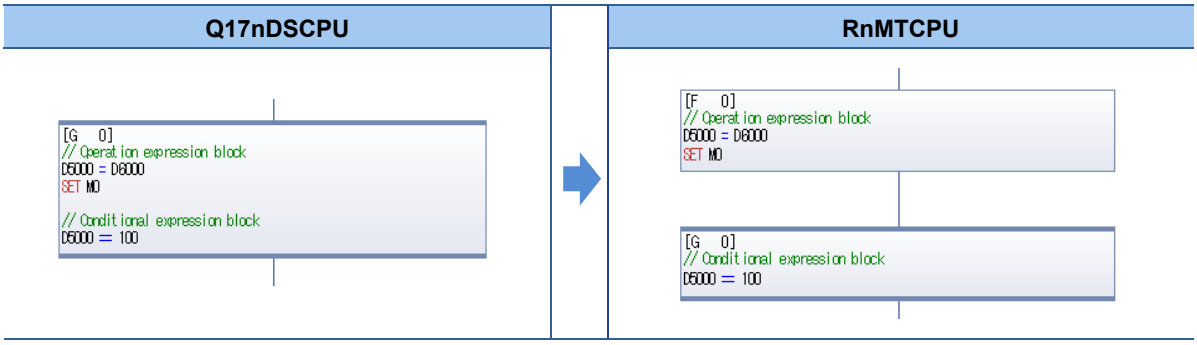
[Optional settings]



4.2.8 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", the program needs to be revised. The following shows a program revision example.

Ex.) “WAIT Y/N” transition



[Points]

- 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 review the Motion SFC program since MELSOFT MT Developer2 does not automatically convert the changes above at project diversion. (An error will occur at project diversion and write operation cannot be executed.)

4.2.9 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. The following shows the cases where a revision is required.

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

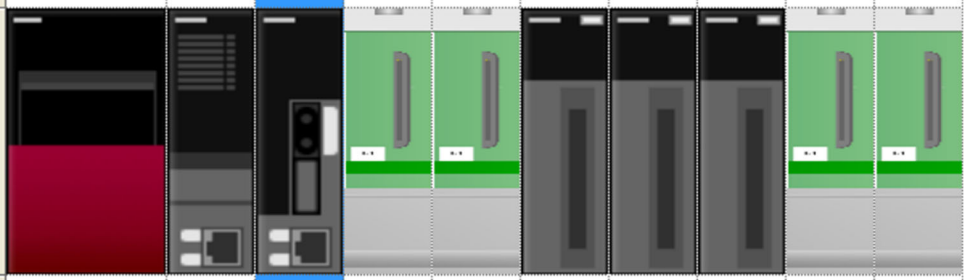
[Q17nDSCPU]

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]

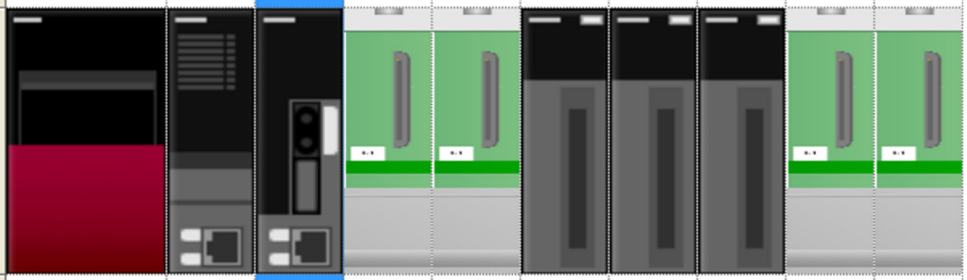
Ex.) When the self CPU controls 1 or more modules, or “I/O reading from outside the group” is valid

	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 Point	16 Point	32 Point	32 Point	32 Point	32 Point	16 Point
Module Name	R61P	R04CPU	R32MTCPU	-	-	RX41C4	RX41C4	RX41C4	-	-
Error Status	-	-	-	-	-	-	-	-	-	-
Module Configuration										
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 Point	16 Point	32 Point	32 Point	32 Point	32 Point	16 Point
Module Name	R61P	R04CPU	R32MTCPU	-	-	RX41C4	RX41C4	RX41C4	-	-
Error Status	-	-	-	-	-	-	-	-	-	-
Module Configuration										
Control CPU	-	-	-	-	-	No. 1	No. 1	No. 1	-	-

CPU No.2 refresh area 

4.2.10 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.

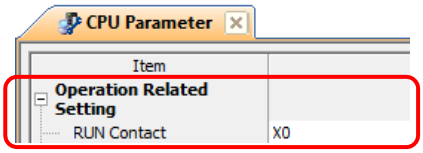
4.2.11 RUN/STOP

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, the program is required to be changed so that a RUN contact for remote operation is used to switch the RUN/STOP status.
The following shows the procedure and point 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 MELSOFT 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 must be in RUN position.

[Points]
- M3072 and D704 have become unusable in RnMTCPU. They cannot be used as a status device
- Note that RUN contact ON is for STOP status and the RUN contact OFF is for RUN status.

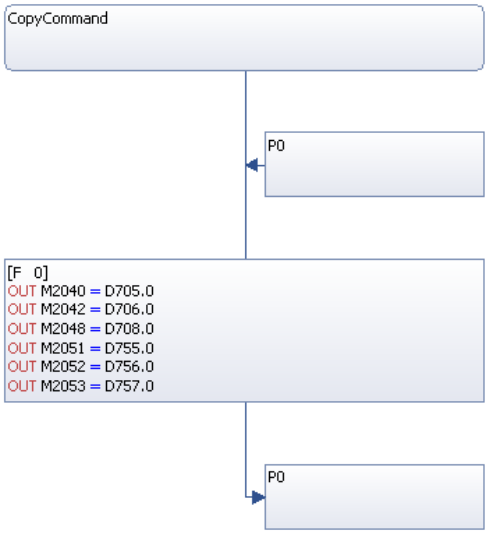
4.2.12 Common bit device SET/RST request register

The positioning dedicated signals of Q17nDSCPU include common devices SET/RST register (D705 to D708, D755 to D757) and bit devices (M3073 to M3079) as command request devices from other CPUs. (The following devices)

Function	Q17nDSCPU		RnMTCPU
	Request device (word)	Request device (bit)	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

These devices do not exist in RnMTCPU. To reflect requests from other CPUs to positioning dedicated signals, create an additional Motion SFC program.

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



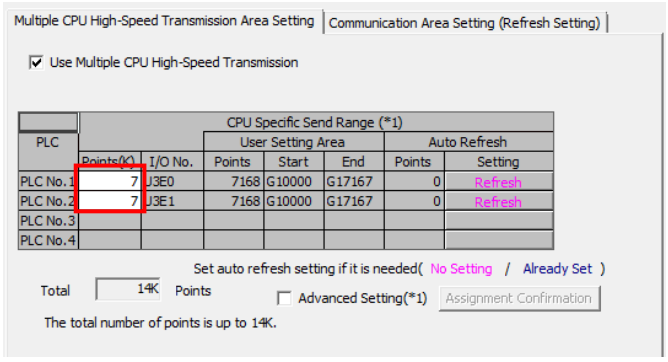
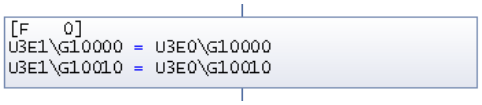
[Points]

- 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.

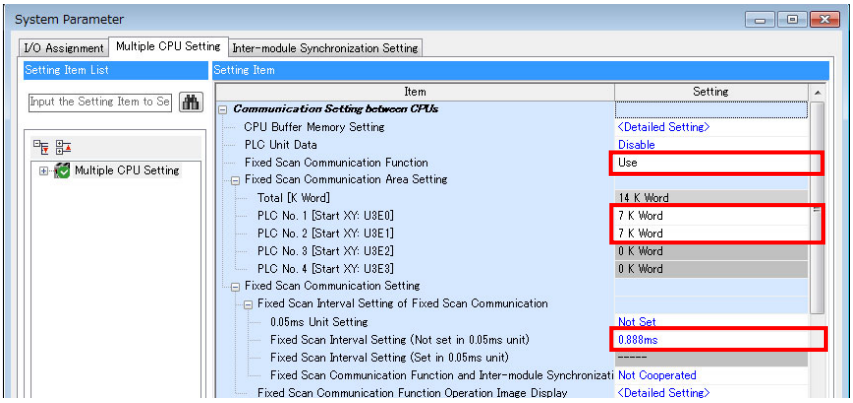
4.2.13 CPU buffer memory (fixed scan communication area)

"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, it is required to change the devices for the transmission area to those for the CPU buffer memory and set the Multiple CPU fixed scan communication.

[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 MELSOFT GX Works3, and read that data by MELSOFT MT Works2. 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]: 0.888ms (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.
	
	`[F  0]`
	`U3E1\HG0 = U3E0\HG0`
	`U3E1\HG10 = U3E0\HG10`

4.3 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.

4.3.1 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>• 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 the existing system, the signal detection direction is set in the each Motion module settings or the system setting of a Motion CPU. In the new migrated system, 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 5000000μs</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>

4.3.2 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.

[Points]
- 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 reviewing the following items:
  - [when using an input module] input response time and compensation time settings
  - [when using amplifier input] input filter and compensation time setting.

5. Device comparison

5.1 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 2.2.
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 devices are assigned for one command signal in Q17nDSCPU, however, these devices have been integrated into one device in RnMTCPU. When using the devices which do not exist in RnMTCPU because of the integration, review the program. Refer to section 4.2.12 for details.
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.

5.2 Data register

• Common devices (command signal)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
D704	M2000	PLC ready flag request	Multiple devices are assigned for one command signal in Q17nDSCPU, however, these devices have been integrated into one device in RnMTCPU. When using the devices which do not exist in RnMTCPU because of the integration, review the program. Refer to section 4.2.12 for details.
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	

5.3 Motion register

• Motion register (monitor device)

Device No.		Name	Remarks
Q17nDSCPU	RnMTCPU		
-	#8016+20n	Servo amplifier vender 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.

5.4 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 2.2.
-	SM1	Latest self-diagnosis error detection (Not including ON of annunciator)	
-	SM4	Warning detection	
-	SM50	Error reset	New device in RnMTCPU
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 2.2.)
SM516	-	Servo program setting error	
SM528	-	No.1 CPU MULTR complete	MULTR instructions have been deleted since RnMTCPU can use CPU buffer memory access device to access the memory. (Refer to section 4.2.5.)
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	

5.5 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 2.2.)
-	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	Errors detected in the Motion CPU are monitored in these devices. (Refer to section 2.2.)
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	Only the advanced synchronous control is supported in RnMTCPU.
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 (14.222ms)	
-	SD585	Motion SFC event task time within operation cycle (7.111ms)	
-	SD586	Motion SFC event task time within operation cycle (3.555ms)	
-	SD587	Motion SFC event task time within operation cycle (1.777ms)	
-	SD588	Motion SFC event task time within operation cycle (0.888ms)	
-	SD589	Motion SFC event task time within operation cycle (0.444ms)	
-	SD590	Motion SFC event task time within operation cycle (0.222ms)	
-	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	888μs free-running timer	
#8736 - #8751	SD740 - SD747	Operating system software version	
-	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	

5.6 CPU buffer memory (CPU shared memory)

• Self CPU operation information area (SD0 to SD199)
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	

6. REVISIONS

Version	Date	Description
-	November 2018	First edition