

Industrial PC

**MELIPC**

MELIPC MI5000 Series  
Programming Manual (VxWorks)

---



# SAFETY PRECAUTIONS

---

(Read these precautions before using this product.)

Before using this product, please read this manual and the relevant manuals carefully and pay full attention to safety to handle the product correctly.

Make sure that the end users read this manual and then keep the manual in a safe place for future reference.

## CONDITIONS OF USE FOR THE PRODUCT

---

(1) Mitsubishi industrial PC ("the PRODUCT") shall be used in conditions;

i) where any problem, fault or failure occurring in the PRODUCT, if any, shall not lead to any major or serious accident; and

ii) where the backup and fail-safe function are systematically or automatically provided outside of the PRODUCT for the case of any problem, fault or failure occurring in the PRODUCT.

(2) The PRODUCT has been designed and manufactured for the purpose of being used in general industries.

MITSUBISHI SHALL HAVE NO RESPONSIBILITY OR LIABILITY (INCLUDING, BUT NOT LIMITED TO ANY AND ALL RESPONSIBILITY OR LIABILITY BASED ON CONTRACT, WARRANTY, TORT, PRODUCT LIABILITY) FOR ANY INJURY OR DEATH TO PERSONS OR LOSS OR DAMAGE TO PROPERTY CAUSED BY the PRODUCT THAT ARE OPERATED OR USED IN APPLICATION NOT INTENDED OR EXCLUDED BY INSTRUCTIONS, PRECAUTIONS, OR WARNING CONTAINED IN MITSUBISHI'S USER, INSTRUCTION AND/OR SAFETY MANUALS, TECHNICAL BULLETINS AND GUIDELINES FOR the PRODUCT.

("Prohibited Application")

Prohibited Applications include, but not limited to, the use of the PRODUCT in;

- Nuclear Power Plants and any other power plants operated by Power companies, and/or any other cases in which the public could be affected if any problem or fault occurs in the PRODUCT.
- Railway companies or Public service purposes, and/or any other cases in which establishment of a special quality assurance system is required by the Purchaser or End User.
- Aircraft or Aerospace, Medical applications, Train equipment, transport equipment such as Elevator and Escalator, Incineration and Fuel devices, Vehicles, Manned transportation, Equipment for Recreation and Amusement, and Safety devices, handling of Nuclear or Hazardous Materials or Chemicals, Mining and Drilling, and/or other applications where there is a significant risk of injury to the public or property.

Notwithstanding the above, restrictions Mitsubishi may in its sole discretion, authorize use of the PRODUCT in one or more of the Prohibited Applications, provided that the usage of the PRODUCT is limited only for the specific applications agreed to by Mitsubishi and provided further that no special quality assurance or fail-safe, redundant or other safety features which exceed the general specifications of the PRODUCTS are required. For details, please contact the Mitsubishi representative in your region.

# CONSIDERATIONS FOR USE

---

## **For the product manufactured by Microsoft Corporation in the United States**

This product comes loaded with Windows 10 IoT Enterprise by Microsoft Corporation in the United States as an operating system. Mitsubishi Electric accepts no responsibility for support or damage related to problems caused by products manufactured by Microsoft Corporation in the United States when using this product.

For the problems or specifications of the Microsoft Corporation product, refer to the corresponding manual or consult Microsoft Corporation.

## **For the Wind River Systems product**

This product is equipped with VxWorks, manufactured by Wind River Systems, Inc., as a real-time operating system. Mitsubishi Electric accepts no responsibility for dealing with or damage from problems caused by products manufactured by Wind River Systems, Inc. when using this product.

For the problems or specifications of the Wind River Systems product, refer to the corresponding manual or consult Wind River Systems, Inc.

Contact information is available on the following website.

Wind River Systems, Inc.: [www.windriver.com](http://www.windriver.com)

# INTRODUCTION

---

Thank you for purchasing the Mitsubishi Electric Industrial PC.

This manual describes the functions required for programming.

Before using the Mitsubishi Electric Industrial PC, please read this manual and the relevant manuals carefully and develop familiarity with the functions and performance to handle the product correctly.

Please make sure that the end users read this manual.

# CONTENTS

SAFETY PRECAUTIONS .....	1
CONDITIONS OF USE FOR THE PRODUCT .....	1
CONSIDERATIONS FOR USE .....	2
INTRODUCTION .....	2
RELEVANT MANUALS .....	6
TERMS .....	7
<b>CHAPTER 1 PROGRAMMING</b> .....	<b>8</b>
<b>1.1 C Controller Module Dedicated Functions</b> .....	<b>8</b>
Program processing .....	8
Argument specifications .....	8
Considerations .....	9
<b>1.2 MELSEC Data Link Functions</b> .....	<b>10</b>
Access ranges and accessible devices .....	10
Program processing .....	16
Argument specifications .....	16
Considerations .....	19
<b>CHAPTER 2 FUNCTION LIST</b> .....	<b>20</b>
<b>2.1 C Controller Module Dedicated Functions</b> .....	<b>20</b>
<b>2.2 C Controller Module Dedicated Functions (For ISR)</b> .....	<b>22</b>
<b>2.3 MELSEC Data Link Functions</b> .....	<b>23</b>
<b>CHAPTER 3 DETAILS OF FUNCTIONS</b> .....	<b>24</b>
<b>3.1 C Controller Module Dedicated Functions</b> .....	<b>24</b>
CCPU_ChangeCCIEFBCCycPrm .....	24
CCPU_ClearError .....	26
CCPU_Control .....	27
CCPU_EndCCIEFBDataAssurance .....	28
CCPU_EndLinkDeviceAssurance .....	29
CCPU_EntryCCIEFBLinkScanEndFunc .....	30
CCPU_EntryLinkScanEndFunc .....	31
CCPU_EntryShutdownFunc .....	32
CCPU_EntryTimerEvent .....	33
CCPU_EntryWDTInt .....	35
CCPU_EntryWinResetFunc .....	36
CCPU_FromBuf .....	37
CCPU_GetCCIEFBDiagnosticInfo .....	38
CCPU_GetCounterMicros .....	40
CCPU_GetCounterMillis .....	41
CCPU_GetCpuStatus .....	42
CCPU_GetDotMatrixLED .....	43
CCPU_GetErrInfo .....	45
CCPU_GetLEDStatus .....	46
CCPU_GetPowerStatus .....	48
CCPU_GetRTC .....	49
CCPU_GetSerialNo .....	50
CCPU_GetSwitchStatus .....	51

CCPU_MELIPCShutdown	52
CCPU_ReadDevice	53
CCPU_ReadLinkDevice	54
CCPU_ReadSharedMemory	55
CCPU_ReceiveInterOSEvent	56
CCPU_RegistEventLog	57
CCPU_Reset	58
CCPU_ResetDevice	59
CCPU_ResetWDT	60
CCPU_RestoreDefaultCCIEFBcycPrm	61
CCPU_SendInterOSEvent	62
CCPU_SetDevice	63
CCPU_SetDotMatrixLED	64
CCPU_SetLEDStatus	66
CCPU_SetRTC	67
CCPU_ShutdownRom	68
CCPU_StartCCIEFBDataAssurance	69
CCPU_StartLinkDeviceAssurance	70
CCPU_StartWDT	71
CCPU_StopWDT	72
CCPU_SysClkRateGet	73
CCPU_SysClkRateSet	74
CCPU_ToBuf	75
CCPU_WaitSwitchEvent	76
CCPU_WaitTimerEvent	77
CCPU_WinReset	78
CCPU_WriteDevice	79
CCPU_WriteLinkDevice	80
CCPU_WriteSharedMemory	81
<b>3.2 C Controller Module Dedicated Functions (for ISR)</b>	<b>82</b>
CCPU_FromBuf_ISR	82
CCPU_GetCounterMicros_ISR	83
CCPU_GetCounterMillis_ISR	84
CCPU_GetDotMatrixLED_ISR	85
CCPU_ReadDevice_ISR	87
CCPU_ReadLinkDevice_ISR	88
CCPU_RegistEventLog_ISR	89
CCPU_ResetDevice_ISR	90
CCPU_SetDevice_ISR	91
CCPU_SetDotMatrixLED_ISR	92
CCPU_SetLEDStatus_ISR	95
CCPU_ToBuf_ISR	96
CCPU_WriteDevice_ISR	97
CCPU_WriteLinkDevice_ISR	98
<b>3.3 MELSEC Data Link Functions</b>	<b>99</b>
mdClose	99
mdControl	100
mdDevRstEx	101
mdDevSetEx	102
mdOpen	103
mdRandREx	104

mdRandWEx .....	107
mdReceiveEx .....	109
mdRemBufReadEx .....	110
mdRemBufWriteEx .....	111
mdSendEx .....	112
mdTypeRead .....	113

**APPENDIX** **116**

---

<b>Appendix 1 Function error codes</b> .....	<b>116</b>
common .....	116
C Controller module dedicated function .....	119
MELSEC data link function .....	122

**INDEX** **124**

---

**FUNCTION INDEX** **126**

---

REVISIONS .....	128
WARRANTY .....	129
TRADEMARKS .....	130

# RELEVANT MANUALS

Manual name [manual number]	Description	Available form
MELIPC MI5000 Series Programming Manual (VxWorks) [SH-081936ENG](this manual)	Explains the programming specifications and dedicated function library.	e-Manual PDF

## Point

e-Manual refers to the Mitsubishi FA electronic book manuals that can be browsed using a dedicated tool.

e-Manual has the following features:

- Required information can be cross-searched in multiple manuals.
- Other manuals can be accessed from the links in the manual.
- Hardware specifications of each part can be found from the product figures.
- Pages that users often browse can be bookmarked.



# TERMS

Unless otherwise specified, this manual uses the following terms.

Terms	Description
C Controller module dedicated function	A dedicated function library used for controlling a MELIPC.
CW Workbench 4	An abbreviation for the engineering tool for Mitsubishi Electric Industrial PC, CW Workbench 4.
Fan module	An abbreviation for a fan module for Mitsubishi Electric Industrial PC, MI5FAN.
Main module	An abbreviation for a main module for Mitsubishi Electric Industrial PC, MI5122.
MELIPC	An abbreviation for Mitsubishi Electric Industrial PC, MI5122-VW It comprises a main module, a power supply module, and a fan module.
MELSEC data link function	A dedicated function library used for accessing another station on a network.
MI Configurator	A product name for SWnDNN-MICONF. ("n" indicates its version)
Power supply module	An abbreviation for a power supply module for Mitsubishi Electric Industrial PC, MI5A1P.
Virtual Ethernet	A virtual network that connects the Windows part and VxWorks part in a MELIPC.
VxWorks	A product name for a real-time operating system manufactured by Wind River Systems, Inc.
VxWorks part	A device in which a real-time operating system VxWorks is embedded. The information acquired from a programmable controller CPU or a CC-Link IE Field Network connected device via CC-Link IE Field Network can be processed in this section.
Windows part	A device in which Windows is embedded. Applications for processing complex calculations, such as data analysis, to process various information can run in this section.

# 1 PROGRAMMING

CW Workbench 4 is used for development of user programs to run on the VxWorks part.

For details of creation of user programs, refer to the following manual.

📖 MELIPC MI5000 Series User's Manual (Startup)

For details on CW Workbench 4, refer to the following manual.

📖 CW Workbench 4 Operating Manual

## 1.1 C Controller Module Dedicated Functions

C Controller dedicated functions are the dedicated function libraries that controls MELIPCs.

The following section shows the argument specifications of the C Controller module dedicated functions and considerations when creating user programs.

### Program processing

The following shows the procedure for processing a user program using a C Controller module dedicated function.

1. Start a task.
2. Read the module status, control the LED status, and access a resource such as a clock by using a C Controller dedicated function.
3. End the task.

### Argument specifications

The following shows the argument specifications of C Controller module dedicated functions.

#### Device type

For the device types to be specified by the C Controller module dedicated functions, either a code or a device name can be specified.

Device names are defined in the include files (CCPUFunc.h).

#### ■ Device type for own station access

The following table shows the device types that can be specified by the argument [Device type (sDevType)].

Device name (device)	Code		Device name specification
	Decimal	Hexadecimal	
Internal relay (M)	4	4H	Dev_CCPU_M
Special relay (SM)	5	5H	Dev_CCPU_SM
Data register (D)	13	DH	Dev_CCPU_D
Special register (SD)	14	EH	Dev_CCPU_SD
Link relay (B)	23	17H	Dev_CCPU_B
Link register (W)	24	18H	Dev_CCPU_W
File register (ZR)	220	DCH	Dev_CCPU_ZR

#### Target functions

📖 Page 53 CCPU\_ReadDevice

📖 Page 59 CCPU\_ResetDevice

📖 Page 63 CCPU\_SetDevice

📖 Page 79 CCPU\_WriteDevice

📖 Page 87 CCPU\_ReadDevice\_ISR

📖 Page 90 CCPU\_ResetDevice\_ISR

📖 Page 91 CCPU\_SetDevice\_ISR

📖 Page 97 CCPU\_WriteDevice\_ISR

## ■ Device types for accessing CC-Link IE Field Networks

The following table shows the device types that can be specified by the argument [Device type (sDevType)].

Device name (device)	Code		Device name specification
	Decimal	Hexadecimal	
Direct link input (RX)	1000	3E8H	Dev_LX
Direct link output (RY)	2000	7D0H	Dev_LY
Direct link register (RWr, RWw) <sup>*1</sup>	24000	5DC0H	Dev_LW
Direct link special relay (SB)	25000	61A8H	Dev_LSB
Direct link special register (SW)	28000	6D60H	Dev_LSW

\*1 To access direct link registers (RWw, RWr), specify the device numbers as follows.

RWw: 0H to 1FFFH

RWr: 2000H to 3FFFH

### Target functions

☞ Page 54 CCPU\_ReadLinkDevice

☞ Page 80 CCPU\_WriteLinkDevice

☞ Page 88 CCPU\_ReadLinkDevice\_ISR

☞ Page 98 CCPU\_WriteLinkDevice\_ISR

## Considerations

The following shows the considerations when using the C Controller module dedicated functions.

### Considerations for the user WDT

If a user WDT (user watchdog timer) cannot be reset due to an out-of-control user program or other problems, a user WDT error will occur.

In this case, take the following corrective actions.

- Make sure that there are no problems with the user program.
- Check whether a task with a high CPU activity ratio is preventing the reset of the user WDT.
- Consider increasing the length of the interval of the user WDT configured by the CCPU\_StartWDT function.

### Considerations on interrupt service routine (ISR)

Fully understand the restrictions of the VxWorks operating system before creating a routine that is executed in an interrupt service routine (ISR) using a C Controller module dedicated function (for ISR). To use another dedicated function in synchronization with an interrupt, implement the notification processing in a user program and perform it in a task.



If an inappropriate value is set for an argument of a C Controller module dedicated function (for ISR), or a function other than a C Controller module dedicated function (for ISR) is executed from an interrupt routine, the VxWorks will be out of control.

# 1.2 MELSEC Data Link Functions

This section shows the access ranges, devices, argument specifications, and considerations of MELSEC communication functions.

## Access ranges and accessible devices

The following shows the access ranges and accessible devices of MELSEC communication functions.

### Communication addressed to own station

The access ranges and accessible devices during communication addressed to own stations are as follows.

#### ■Access range

Only own stations (MELIPC) can be accessed during communication addressed to own station.

#### ■Accessible devices

It indicates an accessible device during communications with the own station.



- The following table shows batch and random access.  
 Batch: batch writing (mdSendEx function), batch reading (mdReceiveEx function)  
 Random: random writing (mdRandWEx function), random reading (mdRandREx function), bit set (mdDevSetEx function), and bit reset (mdDevRstEx function)
- Bit set (mdDevSetEx function) and bit reset (mdDevRstEx function) can access bit devices only.
- Device extension specifications (digit specification, bit specification and index specification) cannot be used.

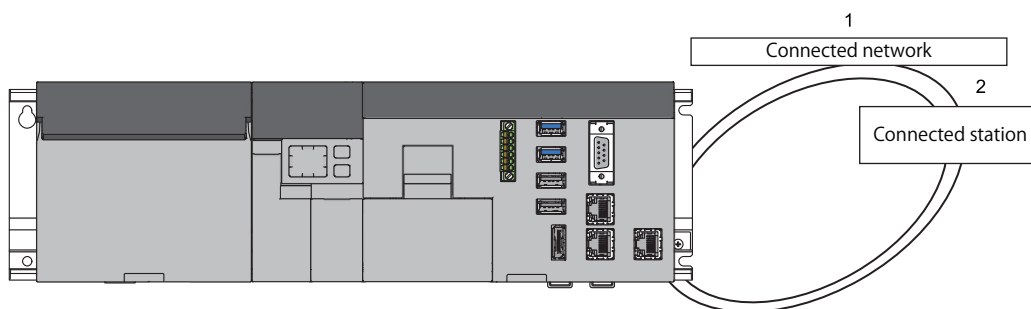
Device name (device)		Device name specification
Internal relay (M)	Batch	DevM
	Random	
Special relay (SM)	Batch	DevSM
	Random	
Data register (D)	Batch	DevD
	Random	
Special register (SD)	Batch	DevSD
	Random	
Link relay (B)	Batch	DevB
	Random	
Link register (W)	Batch	DevW
	Random	
File register (ZR)	Batch	DevZR
	Random	
CPU buffer memory (U3E0\G)	Batch	DevSPB1
	Random	

## CC-Link IE Field Network communication

The following shows the access ranges and accessible devices in CC-Link IE Field Network communications.

### ■access range

The system configuration within an access range for CC-Link IE field network communication is as follows.



- Devices that can be accessed

The following table shows the equipment accessible by the MELIPC.

1. Connected network	2. Connected station
CC-Link IE Field Network	MELIPC
	MELSEC iQ-R series programmable controller
	MELSEC iQ-R series C Controller module
	MELSEC-Q series programmable controller (Q mode)
	MELSEC-Q series C Controller module

### ■Accessible devices

The following section shows the accessible devices during CC-Link IE Field Network communication.



- The following table shows batch and random access.  
 Batch: batch writing (mdSendEx function), batch reading (mdReceiveEx function)  
 Random: random writing (mdRandWEx function), random reading (mdRandREx function), bit set (mdDevSetEx function), and bit reset (mdDevRstEx function)
- Bit set (mdDevSetEx function) and bit reset (mdDevRstEx function) can access bit devices only.
- Device extension specifications (digit specification, bit specification and index specification) cannot be used.

#### During own station access

The following table shows the accessible devices during own station access via CC-Link IE Field Network communication.

Device name (device)		Device name specification
Own station remote input (RX)	Batch	DevLX(0)
	Random	
Own station remote output (RY)	Batch	DevLY(0)
	Random	
Own station direct link special relay (SB)	Batch	DevLSB(0)
	Random	
Own station direct link special register (SW)	Batch	DevLSW(0)
	Random	
Own station remote register (for sending)(RWw) <sup>*1</sup>	Batch	DevLW(0)
	Random	
Own station remote register (for receiving) (RWr) <sup>*1</sup>	Batch	DevLW(0)
	Random	

\*1 To access own-station remote registers, specify the device numbers as follows.

For sending (RWw): 0H to 1FFFH

For receiving (RWr): 2000H to 3FFFH

## During other station access

The following table shows the accessible devices during other station access via CC-Link IE Field Network communication.

No.	Access target
(1)	Q00JCPU, Q00UJCPU, Q00CPU, Q00UCPU, Q01CPU, Q01UCPU, Q02(H)CPU, Q02PHCPU, Q02UCPU, Q03UDCPU, Q03UDECPU, Q03UDVCPU, Q04UDHCPU, Q04UDEHCPU, Q04UDVCPU, Q06HCPU, Q06PHCPU, Q06UDHCPU, Q06UDEHCPU, Q06UDVCPU, Q10UDHCPU, Q10UDEHCPU, Q12HCPU, Q12PHCPU, Q13UDHCPU, Q13UDEHCPU, Q13UDVCPU, Q20UDHCPU, Q20UDEHCPU, Q25HCPU, Q25PHCPU, Q26UDHCPU, Q26UDEHCPU, Q26UDVCPU, Q50UDEHCPU, Q100UDEHCPU
(2)	Q12DCCPU-V (Basic mode), Q12DCCPU-V (Advanced mode), Q24DHCCPU-V, Q24DHCCPU-LS, Q24DHCCPU-VG, Q26DHCCPU-LS
(3)	WinCPU, PC CPU, PC, CC-Link IE field network remote device station <sup>*1</sup>
(4)	L02SCPU, L02CPU, L02CPU-P, L06CPU, L26CPU, L26CPU-BT, L26CPU-PBT
(5)	R00CPU, R01CPU, R02CPU, R04CPU, R04ENCPU, R08CPU, R08ENCPU, R08PCPU <sup>*2</sup> , R08PSFCPU, R08SFCPU, R16CPU, R16ENCPU, R16PCPU <sup>*2</sup> , R16PSFCPU, R16SFCPU, R32CPU, R32ENCPU, R32PCPU <sup>*2</sup> , R32PSFCPU, R32SFCPU, R120CPU, R120ENCPU, R120PCPU <sup>*2</sup> , R120PSFCPU, R120SFCPU
(6)	R12CCPU-V
(7)	MI5122-VW
(8)	LJ72GF15-T2, NZ2GF-ETB, RJ72GF15-T2

\*1 For a CC-Link IE Field Network remote device station, the buffer memory can be accessed only by the mdRemBufWrite function and mdRemBufRead function.

\*2 Only in the process mode.

○: Accessible, ×: Not accessible

Device name (device)	Device name specification	Access target									
		(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)		
Input relay (X)	Batch	DevX	○	○ <sup>*1</sup>	×	○	○	○	×	○	
	Random										
Output relay (Y)	Batch	DevY	○	○ <sup>*1</sup>	×	○	○	○	×	○	
	Random										
Latch relay (L)	Batch	DevL	○	×	×	○	○	×	×	×	
	Random										
Internal relay (M)	Batch	DevM	○	○ <sup>*1</sup>	×	○	○	○	○	×	
	Random										
Special relay (SM)	Batch	DevSM	○	○ <sup>*1</sup>	×	○	○	○	○	○	
	Random										
Annunciator (F)	Batch	DevF	○	×	×	○	○	×	×	×	
	Random										
Timer contact (T)	Batch	DevTT	○	×	×	○	○	×	×	×	
	Random										
Long timer contact (LT)	Batch	DevLTT	×	×	×	×	○	×	×	×	
	Random										
Timer coil (T)	Batch	DevTC	○	×	×	○	○	×	×	×	
	Random										
Long timer coil (LT)	Batch	DevLTC	×	×	×	×	○	×	×	×	
	Random										
Counter contact (C)	Batch	DevCT	○	×	×	○	○	×	×	×	
	Random										
Long counter contact (LC)	Batch	DevLCT	×	×	×	×	○	×	×	×	
	Random										
Counter coil (C)	Batch	DevCC	○	×	×	○	○	×	×	×	
	Random										
Long counter coil (LC)	Batch	DevLCC	×	×	×	×	○	×	×	×	
	Random										
Timer current value (T)	Batch	DevTN	○	×	×	○	○	×	×	×	
	Random										
Long timer current value (LT)	Batch	DevLTN	×	×	×	×	○	×	×	×	
	Random										

Device name (device)		Device name specification	Access target							
			(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Counter current value (C)	Batch	DevCN	○	×	×	○	○	×	×	×
	Random									
Long counter current value (LC)	Batch	DevLCN	×	×	×	×	○	×	×	×
	Random									
Data register (D)	Batch	DevD	○	○ <sup>*1</sup>	×	○	○	○	○	×
	Random									
Special register (SD)	Batch	DevSD	○	○ <sup>*1</sup>	×	○	○	○	○	○
	Random									
Index register (Z)	Batch	DevZ	○	×	×	○	○	×	×	×
	Random									
Long index register (LZ)	Batch	DevLZ	×	×	×	×	○	×	×	×
	Random									
File register (R)	Batch	DevR	○ <sup>*2</sup>	×	×	○	○	×	×	×
	Random									
File register (ZR)	Batch	DevZR	○ <sup>*2</sup>	×	×	○	○	○	○	×
	Random									
Module refresh register (RD)	Batch	DevRD	×	×	×	×	○	×	×	○ <sup>*3</sup>
	Random									
Link relay (B)	Batch	DevB	○	○ <sup>*1</sup>	×	○	○	○	○	×
	Random									
Link register (W)	Batch	DevW	○	○ <sup>*1</sup>	×	○	○	○	○	○
	Random									
Link special relay (SB)	Batch	DevQSB	○	×	×	○	○	×	×	○
	Random									
Retentive timer contact (ST)	Batch	DevSTT	○	×	×	○	○	×	×	×
	Random									
Long retentive timer contact (LST)	Batch	DevLSTT	×	×	×	×	○	×	×	×
	Random									
Retentive timer coil (ST)	Batch	DevSTC	○	×	×	○	○	×	×	×
	Random									
Long retentive timer coil (LST)	Batch	DevLSTC	×	×	×	×	○	×	×	×
	Random									
Link special register (SW)	Batch	DevQSW	○	×	×	○	○	×	×	○
	Random									
Edge relay (V)	Batch	DevQV	○	×	×	○	○	×	×	×
	Random									
Own station random access buffer (—)	Batch	DevMRB	×	×	×	×	×	×	×	×
	Random									
Retentive timer current value (ST)	Batch	DevSTN	○	×	×	○	○	×	×	×
	Random									
Long retentive timer current value (LST)	Batch	DevLSTN	×	×	×	×	○	×	×	×
	Random									
Own station link register (for sending) (—)	Batch	DevWw	×	×	×	×	×	×	×	×
	Random									
Own station link register (for receiving) (—)	Batch	DevWr	×	×	×	×	×	×	×	×
	Random									
Own station buffer memory (—)	Batch	DevSPB	×	×	×	×	×	×	×	×
	Random									
Link direct device link input (Jn\X)	Batch	DevLX(1) to DevLX(255)	○	○	×	×	○	○	×	×
	Random									
Remote input (RX) for CC-Link IE Field Network	Batch	DevLX(1) to DevLX(255)	○	○ <sup>*4</sup>	×	○	○	○	×	×
	Random									

Device name (device)		Device name specification	Access target							
			(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Link direct device link output (Jn\Y)	Batch	DevLY(1) to DevLY(255)	○	○	×	×	○	○	×	×
	Random									
Remote output (RY) for CC-Link IE Field Network	Batch	DevLY(1) to DevLY(255)	○	○ <sup>*4</sup>	×	○	○	○	×	×
	Random									
Link direct device link relay (Jn\B) <sup>*4</sup>	Batch	DevLB(1) to DevLB(255)	○	○	×	×	○	○	×	×
	Random									
Link direct device link register (Jn\W) <sup>*4</sup>	Batch	DevLW(1) to DevLW(255)	○	○	×	×	○	○	×	×
	Random									
Remote register for sending (RWw) for CC-Link IE Field Network <sup>*5</sup>	Batch	DevLW(1) to DevLW(255)	○	○ <sup>*4</sup>	×	○	○	○	×	×
	Random									
Remote register for receiving (RWr) for CC-Link IE Field Network <sup>*5</sup>	Batch	DevLW(1) to DevLW(255)	○	○ <sup>*4</sup>	×	○	○	○	×	×
	Random									
Direct link special relay (other station side) (—)	Batch	DevLSB(1) to DevLSB(255)	○	○ <sup>*4</sup>	×	○	○	○	×	×
	Random									
Direct link special register (other station side) (—)	Batch	DevLSW(1) to DevLSW(255)	○	○ <sup>*4</sup>	×	○	○	○	×	×
	Random									
Intelligent function module device/module access device (Un\G)	Batch	DevSPG(0) to DevSPG(255)	○	○ <sup>*4</sup>	×	○	○	○	×	○ <sup>*3</sup>
	Random									
CPU shared memory, CPU buffer memory CPU No. 1 area (U3E0/G)	Batch	DevSPB1	○	○	×	×	○	○	○	×
	Random		×	×			×	×	×	
CPU shared memory, CPU buffer memory CPU No. 2 area (U3E1/G)	Batch	DevSPB2	○	○	×	×	○	○	×	×
	Random		×	×			×	×		
CPU shared memory, CPU buffer memory CPU No. 3 area (U3E2/G)	Batch	DevSPB3	○	○	×	×	○	○	×	×
	Random		×	×			×	×		
CPU shared memory, CPU buffer memory CPU No. 4 area (U3E3/G)	Batch	DevSPB4	○	○	×	×	○	○	×	×
	Random		×	×			×	×		
Fixed cycle communication area CPU No. 1 area (U3E0/HG)	Batch	DevHSPB1	×	×	×	×	○	○	×	×
	Random						×	×		
Fixed cycle communication area CPU No. 2 area (U3E1/HG)	Batch	DevHSPB2	×	×	×	×	○	○	×	×
	Random						×	×		
Fixed cycle communication area CPU No. 3 area (U3E2/HG)	Batch	DevHSPB3	×	×	×	×	○	○	×	×
	Random						×	×		
Fixed cycle communication area CPU No. 4 area (U3E3/HG)	Batch	DevHSPB4	×	×	×	×	○	○	×	×
	Random						×	×		
Other station buffer memory (—)	Batch	DevRBM	×	×	×	×	×	×	×	×
	Random									
Other station random access buffer (—)	Batch	DevRAB	×	×	×	×	×	×	×	×
	Random									
Remote input for CC-Link (RX)	Batch	DevRX	×	×	×	×	×	×	×	×
	Random									
Remote input for CC-Link (RY)	Batch	DevRY	×	×	×	×	×	×	×	×
	Random									
Other station link register (—)	Batch	DevRW	×	×	×	×	×	×	×	×
	Random									
Link special relay for CC-Link (SB)	Batch	DevSB	×	×	×	×	×	×	×	×
	Random									
Link special register for CC-Link (SW)	Batch	DevSW	×	×	×	×	×	×	×	×
	Random									



- \*1 Only the following devices can be accessed.
  - Q12DCCPU-V with a serial number whose first five digits are '12042' or higher
  - Q24DHCCPU-V, Q24DHCCPU-LS, Q24DHCCPU-VG, Q26DHCCPU-LS
- \*2 Q00JCPU is not accessible.
- \*3 There are no module refresh registers or module access devices for LJ72GF15-2 and BZ2GF-ETB.
- \*4 Only the following devices can be accessed.
  - Q12DCCPU-V (Extended mode)
  - Q24DHCCPU-V, Q24DHCCPU-LS, Q24DHCCPU-VG, Q26DHCCPU-LS
- \*5 To access CC-Link IE field network remote registers, specify the device numbers as follows.
  - For sending (RWw): 0H to 1FFFH
  - For receiving (RWr): 2000H to 3FFFH

# Program processing

The following shows the procedure for processing a use program using a MELSEC data link function.

1. Start a task.
2. Open a communication line. (mdOpen function)
3. Perform a dummy access to the target. (Device read or model name read)
4. Access the target using a MELSEC data link function.
5. Go on to step 6 if not to access the target.  
To access the target, go back to step 4.
6. Close the communication line. (mdClose function)
7. End the task.

## Argument specifications

This section shows the values that can be used when the channels, network numbers, station numbers, and device types are specified as arguments in the MELSEC data link functions.

### Channel

A channel implies a network and communication route to be used when communicating with a MELIPC.

Set a channel number for each connection type to be used in a user program (MELSEC data link function).

Channel number	Channel name	Access target
12	Own station address	The MELIPC internal devices, internal system devices, and buffer memory are accessed.
181	CC-Link IE Field Network	CPU modules and the MELIPC connected to CC-Link IE Field Network link devices and networks of the MELIPC are accessed.

### Network number and station number

This section shows the network numbers and station numbers to be specified by the MELSEC data link functions.

#### ■ Device access

This table shows the network numbers and station numbers to be specified during access to devices.

Communication route	Specification method	Network No.	Station number
Own station address	Own station	0(0H)	255(FFH) <sup>*1</sup>
CC-Link IE Field Network	Own station	0(0H)	255(FFH)
	Other station	1 to 239 (1H to EFH)	0 to 120 (0H to 78H)
	Logical station number <sup>*2</sup>	0(0H)	65 to 239 (41H to EFH)

\*1 Communication to the MELSEC (own station) by using the MELSEC data link functions is possible; however, it may take longer to execute the functions compared to the C Controller module dedicated functions. Use the C Controller module dedicated functions to create a user program in which performance should be ensured (such as control program).

\*2 Logical station numbers are logical numbers that are specified for 'Station number' in a user program (MELSEC data link functions). Logical station numbers are used to access other-station CPUs (other CPUs of a multiple CPU system). When a CPU module managing other stations of the CC-Link IE Field Network is to be accessed directly, it is not necessary to configure the logical number. Use the station number of the CC-Link IE Field Network directly.

#### Target functions

☞ Page 101 mdDevRstEx

☞ Page 102 mdDevSetEx

☞ Page 104 mdRandREx

☞ Page 107 mdRandWEx

☞ Page 109 mdReceiveEx

☞ Page 110 mdRemBufReadEx

☞ Page 111 mdRemBufWriteEx

☞ Page 112 mdSendEx

## Remote operation and model name reading

The following table shows the station numbers to be specified during remote operation and model name reading.

Communication route	Specification method	Station number
Own station address	Own station	255(FFH)
CC-Link IE Field Network	Own station	255(FFH)
	Other station	Refer to the following chart

### Specification of station numbers of other stations

Specify a station number using the following format.

Upper/lower	Setting item		Description
High byte	Network No.	1 to 239 (1H to EFH)	Set this to specify other stations in the own network or each station on other networks. (Use to perform request-to-send to the CC-Link IE field network. )
Low byte	Station number, group number, or all stations	0 to 120 (1H to 78H)	Set the station number of other stations. For CC-Link IE Field Network, the setting range is from 0 to 120.

#### • Logical station number setting method

Set '0' in the upper byte (network number) of the station number above, and specify a logical station number in the lower byte (station number).

The setting range of the logical station number is 65 to 239 (41H to EFH). Set the logical station number with the user program (MELSEC data link functions).

#### Target functions

☞ Page 100 mdControl

☞ Page 113 mdTypeRead

## Device type

For the device types to be specified by the MELSEC data link functions, either a code or a device name can be specified. Device names are defined in the include files (MDFunc.h).

Device name (device)	Code		Device name specification
	Decimal	Hexadecimal	
Input relay (X)	1	1H	DevX
Output relay (Y)	2	2H	DevY
Latch relay (L)	3	3H	DevL
Internal relay (M)	4	4H	DevM
Special relay (SM)	5	5H	DevSM
CPU buffer memory (CPU No. 1 area)(U3E0/G)	501	1F5H	DevSPB1
CPU buffer memory (CPU No. 2 area)(U3E1/G)	502	1F6H	DevSPB2
CPU buffer memory (CPU No. 3 area)(U3E2/G)	503	1F7H	DevSPB3
CPU buffer memory (CPU No. 4 area)(U3E3/G)	504	1F8H	DevSPB4
Fixed cycle communication area (CPU No. 1 area) (U3E0/HG)	511	1FFH	DevHSPB1
Fixed cycle communication area (CPU No. 2 area) (U3E1/HG)	512	200H	DevHSPB2
Fixed cycle communication area (CPU No. 3 area) (U3E2/HG)	513	201H	DevHSPB3
Fixed cycle communication area (CPU No. 4 area) (U3E3/HG)	514	202H	DevHSPB4
Annunciator (F)	6	6H	DevF
Timer contact (T)	7	7H	DevTT
Long timer contact (LT)	41	29H	DevLTT
Timer coil (T)	8	8H	DevTC
Long timer coil (LT)	42	2AH	DevLTC
Counter contact (C)	9	9H	DevCT
Long counter contact (LC)	44	2CH	DevLCT
Counter coil (C)	10	AH	DevCC
Long counter coil (LC)	45	2DH	DevLCC
Timer current value (T)	11	BH	DevTN
Long timer current value (LT)	43	2BH	DevLTN
Counter current value (C)	12	CH	DevCN

Device name (device)		Code		Device name specification
		Decimal	Hexadecimal	
Long counter current value (LC)		46	2EH	DevLCN
Data register (D)		13	DH	DevD
Special register (SD)		14	EH	DevSD
Index register (Z) <sup>*1</sup>		20	14H	DevZ
Long index register (LZ) <sup>*1</sup>		38	26H	DevLZ
File register (R) <sup>*1</sup>		22	16H	DevR
File register (ZR) <sup>*1</sup>		220	DCH	DevZR
Link relay (B)		23	17H	DevB
Link register (W)		24	18H	DevW
Link special relay (SB) <sup>*1</sup>		25	19H	DevQSB
Retentive timer contact (ST)		26	1AH	DevSTT
Long retentive timer contact (LST)		47	2FH	DevLSTT
Retentive timer coil (ST)		27	1BH	DevSTC
Long retentive timer coil (LST)		48	30H	DevLSTC
Link special register (SW) <sup>*1</sup>		28	1CH	DevQSW
Edge relay (V)		30	1EH	DevQV
Retentive timer current value (ST)		35	23H	DevSTN
Long retentive timer current value (LST)		49	31H	DevLSTN
Module refresh register (RD)		39	27H	DevRD
Link direct device (Argument value of device name (1 to 255) <sup>*1</sup> : Network number)	Link input (Jn\X)	1001 to 1255	3E9H to 4E7H	DevLX(1) to DevLX(255)
	Link output (Jn\Y)	2001 to 2255	7D1H to 8CFH	DevLY(1) to DevLY(255)
	Link relay (Jn\B)	23001 to 23255	59D9H to 5AD7H	DevLB(1) to DevLB(255)
	Link register (Jn\W)	24001 to 24255	5DC1H to 5EBFH	DevLW(1) to DevLW(255)
	Link special relay (Jn\SB)	25001 to 25255	61A9H to 62A7H	DevLSB(1) to DevLSB(255)
	Link special register (Jn\SW)	28001 to 28255	6D61H to 6E5FH	DevLSW(1) to DevLSW(255)
Own station direct link	Special relay (SB)	25000	61A8H	DevLSB(0)
	Special register (SW)	28000	6D60H	DevLSW(0)
<ul style="list-style-type: none"> <li>• Intelligent function module device (Un\G)</li> <li>• Module access device (Un\G)</li> </ul> (Argument value of device name (0 to 255 <sup>*1</sup> ): Start I/O No. + 16)		29000 to 29255	7148H to 7247H	DevSPG(0) to DevSPG(255)
Own station remote input (RX)		1000	3E8H	DevLX(0)
Own station remote output (RY)		2000	7D0H	DevLY(0)
<ul style="list-style-type: none"> <li>• Own station remote register (for sending)(RWw)<sup>*2</sup></li> <li>• Own station remote register (for receiving)(RWr)<sup>*2</sup></li> </ul>		24000	5DC0H	DevLW(0)

\*1 Even if a non-existent device is specified in the mdRandREx function, the function may end normally. (All of the bits turn ON in read data. For word devices, the read data is '-1'.)

\*2 To access own-station remote registers, specify the device numbers as follows.

RWw: 0H to 1FFFH

RWr: 2000H to 3FFFH

# Considerations

The following shows the considerations when using the MELSEC data link functions.

## Considerations for programming

### ■Open/close processing of a communication line (mdOpen/mdClose function)

Perform the open/close processing of communication line (the mdOpen/mdClose function) only once at the start of task (task activation) and at the end of task (task completion) respectively in each user program. Opening/closing the line every communication decreases the communication performance.

### ■Execution after using the mdOpen function

At the first execution of the function after using the mdOpen function, it takes longer to execute the function since the CPU module or MELIPC information needs to be acquired. The succeeding processing time can be shortened by performing dummy access the first time.

### ■Count of access to other stations from the same task

Accessing 33 or more other stations simultaneously on the same task of a MELIPC using a user program may decrease the communication performance. To access other stations simultaneously on the same task, limit it to 32 or less stations.

### ■taskDelete execution

Do not execute the taskDelete in a task using the MELSEC data link function. Also, do not delete a task using the MELSEC data link function with the taskDelete. Otherwise, the MELSEC data link function may not operate properly.

## Considerations during device access via CC-Link IE Field Networks

An interlock may be required to be provided depending on the link status of the own station and other stations.

### ■Access to devices on the own station

Create a user program that provides an interlock to enable data reading/writing only when the bit of data link error status of own station (Jn\SB0049) is OFF (normal) and cyclic transmission status (a bit corresponding to the communication target station which is stored to any of Jn\SW00B0 to Jn\SW00B7) is '0' (data link normal station). However, read or write processing for the own station will successfully complete even if the conditions are not satisfied.

### ■Other station transient access (other station remote operation and device access)

Create a user program that provides an interlock to access only when the bit of the baton pass status of the own station (Jn\SB0047) is OFF (normal) and a baton pass status of the station to be accessed (a bit corresponding to the communication target station which is stored to any of Jn\SW00A0 to Jn\SW00A7) is '0' (baton pass normal station).

## Considerations on interrupt service routine (ISR)

The MELSEC data link function cannot be run within an interrupt service routine (ISR). When you want to run the MELSEC data link function in synchronization with an interrupt, implement a notification process in the user program and perform it in a task.

# 2 FUNCTION LIST

This chapter shows the functions that can be used for a MELIPC.

## 2.1 C Controller Module Dedicated Functions

The C Controller module dedicated functions are as listed below.

Function name	Function	Reference
CCPU_ChangeCCIEFBCycPrm	Changes the operation parameter of the cyclic transmission of CC-Link IE Field Network Basic.	Page 24 CCPU_ChangeCCIEFBCycPrm
CCPU_ClearError	Clears continuation errors of a MELIPC.	Page 26 CCPU_ClearError
CCPU_Control	Performs a remote operation (remote RUN/STOP) of a MELIPC.	Page 27 CCPU_Control
CCPU_EndCCIEFBDataAssurance	Ends data assurance for access to a link device of CC-Link IE Field Network Basic.	Page 28 CCPU_EndCCIEFBDataAssurance
CCPU_EndLinkDeviceAssurance	Ends data assurance for access to a link device of CC-Link IE Field Network.	Page 29 CCPU_EndLinkDeviceAssurance
CCPU_EntryCCIEFBLinkScanEndFunc	Registers a routine to be called when a link scan of CC-Link IE Field Network Basic is completed.	Page 30 CCPU_EntryCCIEFBLinkScanEndFunc
CCPU_EntryLinkScanEndFunc	Registers a routine to be called when a link scan of CC-Link IE Field Network is completed.	Page 31 CCPU_EntryLinkScanEndFunc
CCPU_EntryShutdownFunc	Registers a routine to be called before the shutdown function of a MELIPC is executed.	Page 32 CCPU_EntryShutdownFunc
CCPU_EntryTimerEvent	Registers a timer event.	Page 33 CCPU_EntryTimerEvent
CCPU_EntryWDTInt	Registers a routine to be called when a user WDT error interrupt of the VxWorks part occurs.	Page 35 CCPU_EntryWDTInt
CCPU_EntryWinResetFunc	Registers a routine to be called before the Windows forced-restart function of the individual reset function is executed.	Page 36 CCPU_EntryWinResetFunc
CCPU_FromBuf	Reads data from the buffer memory of a MELIPC.	Page 37 CCPU_FromBuf
CCPU_GetCCIEFBDiagnosticInfo	Acquires the diagnostic information of CC-Link IE Field Network Basic.	Page 38 CCPU_GetCCIEFBDiagnosticInfo
CCPU_GetCounterMicros	Acquires a 1 $\mu$ s counter value of a MELIPC.	Page 40 CCPU_GetCounterMicros
CCPU_GetCounterMillis	Acquires a 1 ms counter value of a MELIPC.	Page 41 CCPU_GetCounterMillis
CCPU_GetCpuStatus	Acquires the operating status of a MELIPC.	Page 42 CCPU_GetCpuStatus
CCPU_GetDotMatrixLED	Acquires the values displayed on the dot matrix LED of a MELIPC.	Page 43 CCPU_GetDotMatrixLED
CCPU_GetErrInfo	Acquires error information of a MELIPC.	Page 45 CCPU_GetErrInfo
CCPU_GetLEDStatus	Acquires the LED status of a MELIPC.	Page 46 CCPU_GetLEDStatus
CCPU_GetPowerStatus	Acquires the power supply status of a MELIPC.	Page 48 CCPU_GetPowerStatus
CCPU_GetRTC	Acquires the clock data (local time) of a MELIPC.	Page 49 CCPU_GetRTC
CCPU_GetSerialNo	Acquires the serial number of a MELIPC.	Page 50 CCPU_GetSerialNo
CCPU_GetSwitchStatus	Acquires the switch status of a MELIPC.	Page 51 CCPU_GetSwitchStatus
CCPU_MELIPCSShutdown	Performs a shutdown operation so that the power of a MELIPC can turn OFF.	Page 52 CCPU_MELIPCSShutdown
CCPU_ReadDevice	Reads data from the internal user devices and internal system devices of a MELIPC.	Page 53 CCPU_ReadDevice
CCPU_ReadLinkDevice	Reads data from link devices of CC-Link IE Field Network.	Page 54 CCPU_ReadLinkDevice
CCPU_ReadSharedMemory	Reads data from the shared memory of a MELIPC.	Page 55 CCPU_ReadSharedMemory
CCPU_ReceiveInterOSEvent	Waits for notification of inter OS events.	Page 56 CCPU_ReceiveInterOSEvent
CCPU_RegistEventLog	Registers event logs in the event history of a MELIPC.	Page 57 CCPU_RegistEventLog
CCPU_Reset	Resets a MELIPC.	Page 58 CCPU_Reset
CCPU_ResetDevice	Resets internal user devices and internal system devices (bit devices) of a MELIPC.	Page 59 CCPU_ResetDevice
CCPU_ResetWDT	Resets a user WDT of the VxWorks part.	Page 60 CCPU_ResetWDT
CCPU_RestoreDefaultCCIEFBCycPrm	Restores the operation parameter of cyclic transmission of CC-Link IE Field Network Basic to the default value (which is set to the parameter).	Page 61 CCPU_RestoreDefaultCCIEFBCycPrm
CCPU_SendInterOSEvent	Notifies inter OS events.	Page 62 CCPU_SendInterOSEvent
CCPU_SetDevice	Sets internal user devices and internal system devices (bit devices) of a MELIPC.	Page 63 CCPU_SetDevice

Function name	Function	Reference
CCPU_SetDotMatrixLED	Sets values to be displayed on the dot matrix LED of a MELIPC.	Page 64 CCPU_SetDotMatrixLED
CCPU_SetLEDStatus	Sets the INFO LED status of a MELIPC.	Page 66 CCPU_SetLEDStatus
CCPU_SetRTC	Sets the clock data (local time) of a MELIPC.	Page 67 CCPU_SetRTC
CCPU_ShutdownRom	Shuts down the user drive of the VxWorks part.	Page 68 CCPU_ShutdownRom
CCPU_StartCCIEFBDataAssurance	Starts data assurance for access to a link device of CC-Link IE Field Network Basic.	Page 69 CCPU_StartCCIEFBDataAssurance
CCPU_StartLinkDeviceAssurance	Starts data assurance for access to a link device of CC-Link IE Field Network.	Page 70 CCPU_StartLinkDeviceAssurance
CCPU_StartWDT	Sets the user WDT of the VxWorks part and starts the user WDT.	Page 71 CCPU_StartWDT
CCPU_StopWDT	Stops the user WDT of the VxWorks part.	Page 72 CCPU_StopWDT
CCPU_SysCikRateGet	Reads the system clock rate specified with the CCPU_SysCikRateSet function.	Page 73 CCPU_SysCikRateGet
CCPU_SysCikRateSet	Saves a specified system clock rate to MRAM.	Page 74 CCPU_SysCikRateSet
CCPU_ToBuf	Writes data to the buffer memory of a MELIPC.	Page 75 CCPU_ToBuf
CCPU_WaitSwitchEvent	Waits for a switch interruption event to occur.	Page 76 CCPU_WaitSwitchEvent
CCPU_WaitTimerEvent	Waits for a timer event to occur.	Page 77 CCPU_WaitTimerEvent
CCPU_WinReset	Forces a restart of the Windows part of a MELIPC.	Page 78 CCPU_WinReset
CCPU_WriteDevice	Writes data to internal user devices and internal system devices of a MELIPC.	Page 79 CCPU_WriteDevice
CCPU_WriteLinkDevice	Writes data to link devices of CC-Link IE Field Network.	Page 80 CCPU_WriteLinkDevice
CCPU_WriteSharedMemory	Writes data to the shared memory of a MELIPC.	Page 81 CCPU_WriteSharedMemory

## 2.2 C Controller Module Dedicated Functions (For ISR)

The C Controller module dedicated functions (for ISR) are as listed below.

Function name	Function	Reference
CCPU_FromBuf_ISR	Reads data from the buffer memory of a MELIPC.	Page 82 CCPU_FromBuf_ISR
CCPU_GetCounterMicros_ISR	Acquires a 1 $\mu$ s counter value of a MELIPC.	Page 83 CCPU_GetCounterMicros_ISR
CCPU_GetCounterMillis_ISR	Acquires a 1 ms counter value of a MELIPC.	Page 84 CCPU_GetCounterMillis_ISR
CCPU_GetDotMatrixLED_ISR	Acquires the values displayed on the dot matrix LED of a MELIPC.	Page 85 CCPU_GetDotMatrixLED_ISR
CCPU_ReadDevice_ISR	Reads data from the internal user devices and internal system devices of a MELIPC.	Page 87 CCPU_ReadDevice_ISR
CCPU_ReadLinkDevice_ISR	Reads data from link devices of CC-Link IE Field Network.	Page 88 CCPU_ReadLinkDevice_ISR
CCPU_RegistEventLog_ISR	Registers event logs in the event history of a MELIPC.	Page 89 CCPU_RegistEventLog_ISR
CCPU_ResetDevice_ISR	Resets internal user devices and internal system devices (bit devices) of a MELIPC.	Page 90 CCPU_ResetDevice_ISR
CCPU_SetDevice_ISR	Sets internal user devices and internal system devices (bit devices) of a MELIPC.	Page 91 CCPU_SetDevice_ISR
CCPU_SetDotMatrixLED_ISR	Sets values to be displayed on the dot matrix LED of a MELIPC.	Page 92 CCPU_SetDotMatrixLED_ISR
CCPU_SetLEDStatus_ISR	Sets the INFO LED status of a MELIPC.	Page 95 CCPU_SetLEDStatus_ISR
CCPU_ToBuf_ISR	Writes data to the buffer memory of a MELIPC.	Page 96 CCPU_ToBuf_ISR
CCPU_WriteDevice_ISR	Writes data to internal user devices and internal system devices of a MELIPC.	Page 97 CCPU_WriteDevice_ISR
CCPU_WriteLinkDevice_ISR	Writes data to link devices of CC-Link IE Field Network.	Page 98 CCPU_WriteLinkDevice_ISR



## 2.3 MELSEC Data Link Functions

The MELSEC data link functions are as listed below.

Function name	Function	Reference
mdClose	Closes a communication line (channel).	Page 99 mdClose
mdControl	Performs remote operations (RUN/STOP/PAUSE) for a programmable controller CPU.	Page 100 mdControl
mdDevRstEx	Resets bit devices.	Page 101 mdDevRstEx
mdDevSetEx	Sets bit devices.	Page 102 mdDevSetEx
mdOpen	Opens a communication line (channel).	Page 103 mdOpen
mdRandREx	Reads data by specifying a device type and a range to be read.	Page 104 mdRandREx
mdRandWEx	Writes data by specifying a device type and a range to be written.	Page 107 mdRandWEx
mdReceiveEx	Reads data from devices in a batch.	Page 109 mdReceiveEx
mdRemBufReadEx	Reads data from the buffer memory of a remote device station on CC-Link IE Field Network.	Page 110 mdRemBufReadEx
mdRemBufWriteEx	Writes data to the buffer memory of a remote device station on CC-Link IE Field Network.	Page 111 mdRemBufWriteEx
mdSendEx	Writes data to devices in a batch.	Page 112 mdSendEx
mdTypeRead	Reads the model code of a programmable controller CPU.	Page 113 mdTypeRead

# 3 DETAILS OF FUNCTIONS

This chapter shows the details on the functions used for MELIPC MI5000 series.

## 3.1 C Controller Module Dedicated Functions

This section explains the details of the C Controller module dedicated function.

### CCPU\_ChangeCCIEFBCycPrm

Changes the operation parameter of the cyclic transmission of CC-Link IE Field Network Basic.

#### Format

short CCPU\_ChangeCCIEFBCycPrm(unsigned short usGroupNo, unsigned short usLinkScanTime, unsigned short usTimeout, unsigned short usTimeoutRetryCnt)

#### Argument

Argument	Name	Description	IN/OUT
usGroupNo	Group No.	Specifies a target group number to change the operation parameter of link scan. (0 to 4) (When '0' is specified, the same value is set to all groups.)	IN
usLinkScanTime	Link scan time	Specifies the link scan time of a cyclic transmission in milliseconds. (0 to 10000) (When "0" was specified, the link scan time setting becomes invalid. The next link scan is started soon after a link scan is completed. )	IN
usTimeout	Slave station timeout time	Specify a timeout time (ms) for detecting the disconnection of slave stations. (10 to 65535 ms)	IN
usTimeoutRetryCnt	Number of detections of disconnected slave stations	Specify the number of timeouts for detecting the disconnection of slave stations. (3 to 10)	IN


#### Description

- This function changes the operation parameter of cyclic transmission for the specified group.
- If a value out of the range is specified for the link scan time (usLinkScanTime), the slave station timeout time (usTimeout), or the number of detections of disconnected slave stations (usTimeoutRetryCnt), an error which indicates that the value is out of the range is returned.
- Before running this function, set the CC-Link IE Field Network Basic parameters with MI Configurator. If parameters are not configured, an error is returned.
- Operation with the specified settings begins from the next link scan after the execution of this function.
- This function exits normally after the reflection of operation parameters is completed.
- If the MELIPC is reset, the operation parameters return to the prescribed values (parameter setting values).
- The timeout time (usTimeout) of the specified slave station operates in tick units.

## Return value

Return value	Description
0(0000H)	Normal
-328(FEB8H)	Group number error
-329(FEB7H)	Link time specification error
-330(FEB6H)	Number of detections of disconnected slave stations error
-331(FEB5H)	CC-Link IE Field Network Basic parameter unset
-332(FEB4H)	Slave station timeout time specification range error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 61 CCPU\_RestoreDefaultCCIEFBCycPrm

# CCPU\_ClearError

Clears continuation errors of a MELIPC.

## Format

short CCPU\_ClearError(long\* pErrorInfo)

## Argument

Argument	Name	Description	IN/OUT
pErrorInfo	Error information	Not used (Value is not related to operation. )	IN

## Description

- Clears continuation errors.
- When no continuation error has occurred, this function ends normally.
- When a stop error has occurred, the error cannot be cleared. (This function ends normally. )

## Return value

Return value	Description
0(0000H)	Normal
-28158(9202H)	WDT error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 45 CCPU\_GetErrInfo

# CCPU\_Control

Performs a remote operation (remote RUN/STOP) of a MELIPC.

## Format

short CCPU\_Control(short sCpuNo, short sCode)

## Argument

Argument	Name	Description	IN/OUT
sCpuNo	CPU number	Specify '0'.	IN
sCode	Operation specification code	Specifies the remote operation to be performed. • 0: Remote RUN • 1: Remote STOP	IN


## Description

- Sets the MELIPC to a RUN/STOP status.
- The operation status of the MELIPC prioritizes STOP. To set the RUN state again after a remote STOP, use remote RUN.
- If executed at the VxWorks part during a system or user WDT error, an error occurs.

## Return value

Return value	Description
0(0000H)	Normal
-232(FF18H)	CPU number specification error
-236(FF14H)	Operation specification code error
-28158(9202H)	WDT error

For details on error codes, refer to the following section.

 Page 116 Function error codes

# CCPU\_EndCCIEFBDataAssurance

Ends data assurance for access to a link device of CC-Link IE Field Network Basic.

## Format

short CCPU\_EndCCIEFBDataAssurance(unsigned short usGroupNo)

## Argument

Argument	Name	Description	IN/OUT
usGroupNo	Group No.	Specifies a group number for data assurance. (1 to 4)	IN

## Description

- Ends data assurance for access to a link device of CC-Link IE Field Network Basic for the specified group.
- Execute the CCPU\_StartCCIEFBDataAssurance function to start data assurance.
- If this function is executed, cyclic transmission of the CC-Link IE Field Network Basic function is restarted.
- Data assurance can be ended by this function only by the task that started assurance by the CCPU\_StartCCIEFBDataAssurance function. If this function is executed by a task different from the task that started data assurance during data assurance, a data assurance running error is issued by the other task.
- If this function was executed while data assurance has not been started by the CCPU\_StartCCIEFBDataAssurance function, it ends normally.

## Return value

Return value	Description
0(0000H)	Normal
-328(FEB8H)	Group number error
-331(FEB5H)	CC-Link IE Field Network Basic parameter unset
-377(FE87H)	Data assurance running by other task error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 69 CCPU\_StartCCIEFBDataAssurance

# CCPU\_EndLinkDeviceAssurance

Ends data assurance for access to a link device of CC-Link IE Field Network.

## Format

short CCPU\_EndLinkDeviceAssurance(unsigned short usIoNo)

## Argument

Argument	Name	Description	IN/OUT
usIoNo	Module position	Specify 0x03E0.	IN

## Description

- Ends data assurance for access to a link device of CC-Link IE Field Network.
- Execute the CCPU\_StartLinkDeviceAssurance function before starting data assurance.
- If this function is executed, the following operations result.
  - The statuses of RX and RWr of own station are updated.
  - The statuses of RY and RWw of own station are transmitted.
- If executed while data assurance is not being performed by the CCPU\_StartLinkDeviceAssurance function, a data assurance ended error occurs.

## Return value

Return value	Description
0(0000H)	Normal
-361(FE97H)	Data assurance ended error
-362(FE96H)	CC-Link IE Field Network hardware error
-28632(9028H)	Module position specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 70 CCPU\_StartLinkDeviceAssurance

# CCPU\_EntryCCIEFBLinkScanEndFunc

Registers a routine to be called when a link scan of CC-Link IE Field Network Basic is completed.

## Format

short CCPU\_EntryCCIEFBLinkScanEndFunc(CCPU\_SCANENDFUNCPTR pEndFuncPtr)

## Argument

Argument	Name	Description	IN/OUT
pEndFuncPtr	Registered routine	Specify a routine to be registered. (The routine is deregistered by specifying NULL.)	IN

The data type of the registered routine (pEndFuncPtr) is defined in the header file "CCPUFunc.h" as follows.

### ■Format

void(\*CCPU\_SCANENDFUNCPTR) (unsigned short usGroupNo)

### ■Argument

Argument	Name	Description	IN/OUT
usGroupNo	Group No.	The group number whose link scan is completed is specified.	OUT

## Description

- Registers a routine to be called when a link scan of CC-Link IE Field Network Basic is completed.
- When the link scan of any group is completed, the registered routine is executed.
- Only one routine is executed at once. After the link scan of another group is completed during the execution of a routine, the module waits for the execution of the next routine until the current routine is completed.
- When this function is executed several times, the last registered routine will be in effect.
- The registered routine operates on the task that operates on CC-Link IE Field Network Basic. Do not carry out processes that take time as they may affect the link scan time. Run processes that take time using another user task. The effect on the link scan time can be checked by the maximum link scan time of the buffer memory.

The settings of the tasks to be run by the registered routine are as follows.

Priority: 48

Stack size: 4096 bytes are available for the registered routine

Task option: VX\_FP\_TASK

## Precautions

Do not register routines to run the following functions.

- CCPU\_ChangeCCIEFBCycPrm
- CCPU\_RestoreDefaultCCIEFBCycPrm

## Return value

Return value	Description
0(0000H)	Normal
-331(FEB5H)	CC-Link IE Field Network Basic parameter unset

For details on error codes, refer to the following section.

☞ Page 116 Function error codes



# CCPU\_EntryLinkScanEndFunc

Registers a routine to be called when a link scan of CC-Link IE Field Network is completed.

## Format

short CCPU\_EntryLinkScanEndFunc(unsigned short usloNo, CCPU\_FUNCPTR pFuncPtr)

## Argument

Argument	Name	Description	IN/OUT
usloNo	Module position	Specify 0x03E0.	IN
pFuncPtr <sup>*1</sup>	Registered routine	Specify a routine to be registered. (The routine is deregistered by specifying NULL.)	IN

\*1 The data type is defined as a void type in the include file "CCPUFunc.h".

## Description

- Registers a routine to be called when a link scan of CC-Link IE Field Network is completed.
- When this function is executed several times, the last registered routine will be in effect.
- The registered routine is operated by the interrupt service.
- Do not carry out processes in which blocks are generated or processes in which the CPU is occupied for a long time during routines to be registered. Run processes in which the CPU is occupied for a long time using another user task.

## Precautions

- The registered routine is not executed while the operating system is in the interrupt-disabled state.
- For processing a routine to be registered in the registered routine (pFuncPtr), note the following:  
A routine to be registered must not have an argument. (Do not pass an argument from an interrupt.)  
When registering a routine, observe the considerations on the interrupt service routine (ISR).  
Register minimal processing of a routine so that the processing time is as short as possible.  
Only the C Controller module dedicated function (for ISR) can be used for a routine to be registered. Do not use any other function. (An error of a function to be registered is not checked.)

## Warning

If a routine that does not observe the considerations on interrupt service routine (ISR) is registered, the operating system will be out of control.

Make sure to use the routine after carefully verifying the operation and performance.

## Return value

Return value	Description
0(0000H)	Normal
-28632(9028H)	Module position specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

# CCPU\_EntryShutdownFunc

Registers a routine to be called before the shutdown function of a MELIPC is executed.

## Format

short CCPU\_EntryShutdownFunc(CCPU\_FUNCPTR pFuncPtr)

## Argument

Argument	Name	Description	IN/OUT
pFuncPtr*1	Registered routine	Specify a routine to be registered. (The routine is deregistered by specifying NULL.)	IN

\*1 The data type is defined as a void type in the include file "CCPUFunc.h".

## Description

- Registers a routine to be called before the shutdown function of a MELIPC is executed.
- When this function is executed several times, the last registered routine will be in effect.
- Before the registered routine is executed, the WIN RDY LED starts flashing, and the MELIPC shutdown request event is reported.
- After the registered routine ends, the MELIPC shutdown start request event is reported, and the shutdown process of the Windows part starts.
- The registered routine operates on the task that operates the shutdown process of the MELIPC. The settings of the tasks to be run by the registered routine are as follows.

Priority: 50

Stack size: 4096 bytes are available for the registered routine

Task option: VX\_FP\_TASK

## Precautions

Do not register routines to run the following functions.

- CCPU\_MELIPCShutdown
- CCPU\_WinReset

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

☞ Page 36 CCPU\_EntryWinResetFunc

# CCPU\_EntryTimerEvent

Registers a timer event.

## Format

short CCPU\_EntryTimerEvent(long\* pEvent)

## Argument

Argument	Name	Description	IN/OUT
pEvent	Registered event	Specify a timer event to be registered.	IN

Registration events (pEvent) are specified as follows.

pEvent	Description	
pEvent[0]	Number of timer event settings (1 to 16)	
pEvent[1]	First timer event number (1 to 16)	First event setting
pEvent[2]	Cycle of the first timer event (0: Clear, 1 to 60,000: Cycle [ms])	
pEvent[3]	Synchronization type of the first timer event (0: Batch synchronization, 1: Individual synchronization)	
pEvent[4]	Second timer event number (1 to 16)	Second event setting
pEvent[5]	Cycle of the second timer event (0: Clear, 1 to 60,000: Cycle [ms])	
pEvent[6]	Synchronization type of the second timer event (0: Batch synchronization, 1: Individual synchronization)	
pEvent[7]	Third timer event number (1 to 16)	Third event setting
pEvent[8]	Cycle of the third timer event (0: Clear, 1 to 60,000: Cycle [ms])	
pEvent[9]	Synchronization type of the third timer event (0: Batch synchronization, 1: Individual synchronization)	
:	:	:

When setting the timer event cycle, only the following specification method is applicable.

- For 1 to 9: Specify multiples of 1 (1 ms units)
- For 10 to 1000: Specify multiples of 5 (5 ms units)
- For 1000 to 60,000: Specify multiples of 1000 (1\ s units)


## Description

- This function registers a timer event by specifying a cycle and synchronization type.
- When '0' is specified for a cycle for the registered event (pEvent), the registered timer event (occurrence) is cleared. Deregistration will clear the events that have occurred before that.
- Up to 16 timer events can be set. The cycle (1 ms to 60 sec) and synchronization type (batch synchronization or individual synchronization) can be specified. For details on the synchronization type, refer to the description of the CCPU\_WaitTimerEvent function.
- Specify the timer event number within the registration event (pEvent) without duplication. Otherwise, an error will be returned.
- To change the cycle of a timer event number for which cycle settings are already set, clear registration (specify '0' to the cycle), and then register the cycle (specify the cycle) again. Otherwise, an error will be returned.
- The registered timer event by this function can be placed into a wait state with the CCPU\_WaitTimerEvent function.
- All the timer events are cleared at the initial status.

## Return value

Return value	Description
0(0000H)	Normal
-225(FF1FH)	Event number specification error
-235(FF15H)	Number of event settings specification error
-241(FF0FH)	Cycle specification error
-242(FF0EH)	Synchronization type specification error
-246(FF0AH)	Timer event registration error
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 77 CCPU\_WaitTimerEvent

# CCPU\_EntryWDTInt

Registers a routine to be called when a user WDT error interrupt of the VxWorks part occurs.

## Format

short CCPU\_EntryWDTInt(short sType, CCPU\_FUNCPTR pFuncPtr)

## Argument

Argument	Name	Description	IN/OUT
sType	WDT type	Specify the WDT type. (When 'Reserved' is specified, an error is returned.) • 0: User WDT • Others: Reserved	IN
pFuncPtr <sup>*1</sup>	Registered routine	Specify a routine to be registered. (The routine is deregistered by specifying NULL.)	IN

\*1 The data type is defined as a void type in the include file "CCPUFunc.h".

## Description

- Registers a routine to be called when a user WDT error interrupt of the VxWorks part occurs.
- When this function is executed several times, the last registered routine will be in effect.
- The routine registered with this function is executed as an interrupt service routine (ISR) when a user WDT error occurs. (If the CCPU\_ResetWDT function is not executed within the time interval specified by this function, the WDT error will occur.)

## Precautions

- The registered routine is not executed while the operating system is in the interrupt-disabled state.
- For processing a routine to be registered in the registered routine (pFuncPtr), note the following:  
A routine to be registered must not have an argument. (Do not pass an argument from an interrupt.)  
When registering a routine, observe the considerations on the interrupt service routine (ISR).  
Register minimal processing of a routine so that the processing time is as short as possible.  
Only the C Controller module dedicated function (for ISR) can be used for a routine to be registered. Do not use any other function. (An error of a function to be registered is not checked.)

## Warning

If a routine that does not observe the considerations on interrupt service routine (ISR) is registered, the operating system will be out of control.

Make sure to use the routine after carefully verifying the operation and performance.

## Return value

Return value	Description
0(0000H)	Normal
-220(FF24H)	WDT type error
-28158(9202H)	WDT error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 60 CCPU\_ResetWDT

☞ Page 71 CCPU\_StartWDT

☞ Page 72 CCPU\_StopWDT

# CCPU\_EntryWinResetFunc

Registers a routine to be called before the Windows forced-restart function of the individual reset function is executed.

## Format

short CCPU\_EntryWinResetFunc(CCPU\_FUNCPTR pFuncPtr)

## Argument

Argument	Name	Description	IN/OUT
pFuncPtr*1	Registered routine	Specify a routine to be registered. (The routine is deregistered by specifying NULL.)	IN

\*1 The data type is defined as a void type in the include file "CCPUFunc.h".

## Description

- Registers a routine to be called before the Windows forced-restart function of the individual reset function is executed.
- When this function is executed several times, the last registered routine will be in effect.
- Before the registered routine is executed, the WIN RDY LED starts flashing, and the Windows part forced restart request event is reported.
- After the registered routine ends, the Windows part forced restart start event is reported, and the Windows part forced restart is started.
- The registered routine operates on the task that operates the shutdown process of the MELIPC. The settings of the tasks to be run by the registered routine are as follows.

Priority: 50

Stack size: 4096 bytes are available for the registered routine

Task option: VX\_FP\_TASK

## Precautions

Do not register routines to run the following functions.

- CCPU\_MELIPCShutdown
- CCPU\_WinReset

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

☞ Page 32 CCPU\_EntryShutdownFunc

# CCPU\_FromBuf

Reads data from the buffer memory of a MELIPC.

## Format

short CCPU\_FromBuf(unsigned short usloNo, unsigned long ulOffset, unsigned long ulSize, unsigned short\* pusDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
usloNo	Module position	Specify 0x03E0.	IN
ulOffset	Offset	Specify the offset in word units.	IN
ulSize	Data size	Specify the read data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of read data.	OUT
ulBufSize	Data storage destination size	Specify the data storage destination size in word units.	IN

## Description

- Reads data of the data size (ulSize) from an address moved by the amount of the offset (ulOffset) from the start of the buffer memory, and then stores it in the data storage destination (pusDataBuf).
- When "0" was specified for the data size (ulSize), a memory size specification error occurs.


## Precautions

Note that the size of data storage destination (ulBufSize) should be equal to or bigger than the data size (ulSize).


## Return value

Return value	Description
0(0000H)	Normal
-208(FF30H)	Offset error
-209(FF2FH)	Memory size specification error
-210(FF2EH)	Read area size error
-264(FEF8H)	Pointer error
-28158(9202H)	WDT error
-28632(9028H)	Module position specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 75 CCPU\_ToBuf

# CCPU\_GetCCIEFBDiagnosticInfo

Acquires the diagnostic information of CC-Link IE Field Network Basic.

## Format

short CCPU\_GetCCIEFBDiagnosticInfo(unsigned short usSlave, short\* psStatusBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
usSlave	Slave station number	Specifies a slave station from which to acquire diagnostic information.	IN
psStatusBuf	Diagnostic information storage destination	Specify a storage destination for diagnostic information.	OUT
ulBufSize	Diagnostic information storage destination size	Specify the size of the area which is reserved in the diagnostic information storage destination in word units.	IN

## Description

- This function acquires CC-Link IE Field Network Basic information of the slave station specified to the slave station number (usSlave), and stores it to the diagnostic information storage destination (psStatusBuf).
- It also acquires the information for the size specified to the diagnostic information storage destination size (ulBufSize).
- The information to be stored to the diagnostic information storage destination (psStatusBuf) is as follows.


psStatusBuf	Description	
psStatusBuf[0]	Diagnostic information status flag	Diagnostic information status (valid or invalid) of the specified slave station b0 to b7: Status of diagnostic information 1 b8 to b15: Status of diagnostic information 2 • 0: Invalid • 1: Valid
psStatusBuf[1]	Diagnostic information 1	Number of occupied stations
psStatusBuf[2]		Group No.
psStatusBuf[3]		IP address (lower)
psStatusBuf[4]		IP address (upper)
psStatusBuf[5] to psStatusBuf[10]		Reserved
psStatusBuf[11]	Accumulated number of timeouts	Accumulated number of timeouts of the specified slave station
psStatusBuf[12]	Accumulated number of disconnection detections	Accumulated number of disconnection detections of the specified slave station
psStatusBuf[13] to psStatusBuf[15]	Reserved	
psStatusBuf[16]	Diagnostic information 2	Manufacturer code
psStatusBuf[17]		Reserved
psStatusBuf[18]		Model code (lower)
psStatusBuf[19]		Model code (upper)
psStatusBuf[20]		Device version
psStatusBuf[21]		Reserved
psStatusBuf[22]		Module information
psStatusBuf[23]		Error code
psStatusBuf[24]		Detailed module information (lower)
psStatusBuf[25]		Detailed module information (upper)
psStatusBuf[26] to psStatusBuf[31]		Reserved



## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error
-331(FEB5H)	CC-Link IE Field Network Basic parameter unset
-336(FEB0H)	Slave station number specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

# CCPU\_GetCounterMicros

Acquires a 1  $\mu$ s counter value of a MELIPC.

## Format

short CCPU\_GetCounterMicros(unsigned long\* pulMicros)

## Argument

Argument	Name	Description	IN/OUT
pulMicros	1 $\mu$ s counter value storage destination	Specify the storage destination of the 1 $\mu$ s counter value.	OUT

## Description

- This function acquires a 1  $\mu$ s counter value of a MELIPC, and stores it in the 1  $\mu$ s counter value storage destination (pulMicros).
- The 1  $\mu$ s counter value increases by 1 every 1  $\mu$ s after the power is turned ON.
- The count cycles between 0 and 4294967295.




## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

-  Page 41 CCPU\_GetCounterMillis
-  Page 83 CCPU\_GetCounterMicros\_ISR
-  Page 84 CCPU\_GetCounterMillis\_ISR

# CCPU\_GetCounterMillis

Acquires a 1 ms counter value of a MELIPC.

## Format

short CCPU\_GetCounterMillis(unsigned long\* pulMillis)

## Argument

Argument	Name	Description	IN/OUT
pulMillis	1 ms counter value storage destination	Specify the storage destination of the 1 ms counter value.	OUT


## Description

- This function acquires a 1 ms counter value of a MELIPC, and stores it in the 1 ms counter value storage destination (pulMillis).
- The 1 ms counter value increases by 1 every 1 ms after the power is turned ON.
- The count cycles between 0 and 4294967295.




## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

-  Page 40 CCPU\_GetCounterMicros
-  Page 83 CCPU\_GetCounterMicros\_ISR
-  Page 84 CCPU\_GetCounterMillis\_ISR

# CCPU\_GetCpuStatus

Acquires the operating status of a MELIPC.

## Format

short CCPU\_GetCpuStatus(long\* plStatusBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
plStatusBuf	Operating status storage destination	Specify the storage destination of the operating status.	OUT
ulBufSize	Operating status storage destination size	Specify the size of area reserved in the operating status storage destination in double word units. (When '0' is specified, this function ends normally without processing.)	IN

## Description

- This function acquires the operating status of a MELIPC, and stores it to the operating status storage destination (plStatusBuf).
- It also acquires the information for the size specified to the operating status storage destination size (ulBufSize).
- The information to be stored in the operating status storage destination (plStatusBuf) is as follows.  
(If information to be stored is not supported, '0' is set as its status.)

plStatusBuf	Description		
	Storage position		Status
plStatusBuf[0]	bit31-8	Reserved	—
	bit7-4	STOP cause	0: STOP/MAIN RUN switch 1: Execution of the CCPU_Control function from a user program 2: Error 3: Remote operation Others: Reserved
	bit3-0	MELIPC operating status	0: RUN state 1: Reserved 2: STOP state 3: Reserved Others: Reserved
plStatusBuf[1]	bit31-16	Reserved	—
	bit15-9	Reserved	—
	bit8	Cfast card status	0: Inserted 1: Not inserted
	bit7	Windows part shutdown status	0: Operating 1: Stopping, preparing, or shutting down
	bit6-1	Reserved	—
	bit0	User drive shutdown status	0: Shutdown not performed 1: Shutdown completed
plStatusBuf[2]	bit31-0	Reserved	—

## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 45 CCPU\_GetErrInfo

# CCPU\_GetDotMatrixLED

Acquires the values displayed on the dot matrix LED of a MELIPC.

## Format

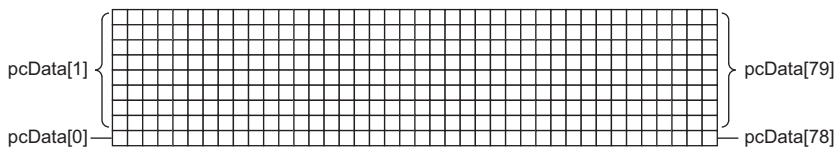
short CCPU\_GetDotMatrixLED(char\* pcData, unsigned long ulDataSize)

## Argument

Argument	Name	Description	IN/OUT
pcData	LED data	Specify the storage destination of LED data.	OUT
ulDataSize	LED data storage destination size	Specify the LED data storage destination size in byte units. (When '0' is specified, this function ends normally without processing.)	IN

## Description

- Acquires the value displayed on the dot matrix LED and stores it in the LED data (pcData).
- It also acquires the information for the size specified to the LED data storage destination size (ulDataSize).
- The value displayed on the dot matrix LED is stored in the LED data (pcData) as shown below.



pcData[0] to pcData[79]: Data on the dot matrix LED (9x40)

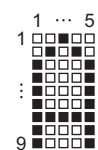
The values displayed are acquired in the following formats.

pcData	Description
pcData [even number]	The lower seven bits are all '0'. Acquires the bit pattern corresponding to line 9 of the dot matrix LED by the upper 1 bit.
pcData [odd number]	Acquires the bit pattern corresponding to lines 1 through 8 of the dot matrix LED. (The most significant bit corresponds to line 1)

When the corresponding bit of the bit pattern section is '1', the LED turns ON.

### Ex.

When the following bit pattern is displayed on the dot matrix LED



1st row: 0011 1111 1000 0000b=3F80H→pcData[0]=0x80, pcData[1]=0x3F  
 2nd row: 0100 0010 0000 0000b=4200H→pcData[2]=0x00, pcData[3]=0x42  
 3rd row: 1000 0010 0000 0000b=8200H→pcData[4]=0x00, pcData[5]=0x82  
 4th row: 0100 0010 0000 0000b=4200H→pcData[6]=0x00, pcData[7]=0x42  
 5th row: 0011 1111 1000 0000b=3F80H→pcData[8]=0x80, pcData[9]=0x3F  
 6th column to 40th column:0000 0000 0000 0000b=0000H→pcData[10] to pcData[79]=0x0000

## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 64 CCPU\_SetDotMatrixLED

# CCPU\_GetErrInfo

Acquires error information of a MELIPC.

## Format

short CCPU\_GetErrInfo(unsigned short\* pusErrorInfo, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
pusErrorInfo	Error information storage destination	Specify the error information storage destination.	OUT
ulBufSize	Error information storage destination size	Specify the error information storage destination size in word units. (When '0' is specified, this function ends normally without processing.)	IN

## Description

- This function acquires the error information of a MELIPC, and stores it in the error information storage destination (pusErrorInfo).
- It also acquires the information for the size specified to the error information storage destination size (ulBufSize).
- The information to be stored in the error information storage destination (pusErrorInfo) is as follows.

pusErrorInfo	Description
pusErrorInfo[0]	Self-diagnostics error code 1
pusErrorInfo[1]	Self-diagnostics error code 2
pusErrorInfo[2]	Self-diagnostics error code 3
pusErrorInfo[3]	Self-diagnostics error code 4
pusErrorInfo[4]	Self-diagnostics error code 5
pusErrorInfo[5]	Self-diagnostics error code 6
pusErrorInfo[6]	Self-diagnostics error code 7
pusErrorInfo[7]	Self-diagnostics error code 8
pusErrorInfo[8]	Self-diagnostics error code 9
pusErrorInfo[9]	Self-diagnostics error code 10
pusErrorInfo[10]	Self-diagnostics error code 11
pusErrorInfo[11]	Self-diagnostics error code 12
pusErrorInfo[12]	Self-diagnostics error code 13
pusErrorInfo[13]	Self-diagnostics error code 14
pusErrorInfo[14]	Self-diagnostics error code 15
pusErrorInfo[15]	Self-diagnostics error code 16

### Point

Up to 16 error codes for errors occurred in the self-diagnostics are stored in order from pusErrorInfo[0]. The error code which has already been stored is not stored.

## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 26 CCPU\_ClearError

# CCPU\_GetLEDStatus

Acquires the LED status of a MELIPC.

## Format

short CCPU\_GetLEDStatus(long lLed, unsigned short\* pusLedInfo, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
lLed	Target LED	Specify a target LED. (When 'Reserved' is specified, this function ends normally without processing.) <b>■Main LED</b> <ul style="list-style-type: none"> <li>• 0: VX RDY LED</li> <li>• 1: WIN RDY LED</li> <li>• 2: MAIN ERR LED</li> <li>• 3: MAIN RUN LED</li> <li>• 4: FAN LED</li> <li>• 5: INFO LED</li> <li>• 6: BATTERY LED</li> </ul> <b>■CC-Link IE Field Network LED</b> <ul style="list-style-type: none"> <li>• 7: RUN LED</li> <li>• 8: ERR LED</li> <li>• 9: MST LED</li> <li>• 10: D LINK LED</li> <li>• 11: SD/RD LED</li> <li>• 12: L ERR LED</li> </ul> <b>■common</b> <ul style="list-style-type: none"> <li>• -1: All of the LEDs above</li> <li>• Others: Reserved</li> </ul>	IN
pusLedInfo	LED status storage destination	Specify the storage destination of the LED status.	OUT
ulBufSize	LED status storage destination size	Specify the LED status storage destination size in word units. (When '0' is specified, this function ends normally without processing.)	IN

## Description

- Acquires the LED status of the MELIPC specified to the target LED (lLed) and stores it in the LED status storage destination (pusLedInfo).
- It also acquires the information for the size specified to the LED status storage destination size (ulBufSize).
- For an unsupported display LED, '0' is set for the LED status.
- The LED status to be stored in the LED status storage destination (pusLedInfo) is as follows.

pusLedInfo	Description
0	OFF
1	ON (Red)
2	Flashing at low speed (Red)
3	Flashing (Red)
4	ON (Green)
5	Flashing at low speed (Green)
6	Flashing (Green)
7	ON (yellow)
9	Flashing (Yellow)




- When '-1' is specified for the target LED (lLed), the LED status is stored in the LED status storage destination (pusLedInfo) as follows in a range that does not exceed the LED status storage destination size (ulBufSize).  
(When a value from 0 through 12 was specified, the LED status specified for pusLedInfo[0] is stored. )

pusLedInfo	Description
pusLedInfo[0]	VX RDY LED status
pusLedInfo[1]	WIN RDY LED status
pusLedInfo[2]	MAIN ERR LED status
pusLedInfo[3]	MAIN ERR LED status
pusLedInfo[4]	FAN LED status
pusLedInfo[5]	INFO LED status
pusLedInfo[6]	BATTERY LED status
pusLedInfo[7]	RUN LED status
pusLedInfo[8]	ERR LED status
pusLedInfo[9]	MST LED status
pusLedInfo[10]	D LINK LED status
pusLedInfo[11]	SD/RD LED status
pusLedInfo[12]	L ERR LED status

## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 45 CCPU\_GetErrInfo

# CCPU\_GetPowerStatus

Acquires the power supply status of a MELIPC.

## Format

short CCPU\_GetPowerStatus(long\* plStatusBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
plStatusBuf	Power status storage destination	Specify the storage destination of the power status.	OUT
ulBufSize	Power status storage destination size	Specify the power status storage destination size in double word units. (When '0' is specified, this function ends normally without processing.)	IN

## Description

- This function acquires the power status of a MELIPC, and stores it in the power status storage destination (plStatusBuf).
- It also acquires the information for the size specified to the power status storage destination size (ulBufSize).
- The information to be stored in the power status storage destination (plStatusBuf) is as follows.

plStatusBuf	Description		
	Storage position		Status
plStatusBuf[0]	bit31-16	Reserved	—
	bit15-0	Number of detected momentary power failures	—

## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 45 CCPU\_GetErrInfo

# CCPU\_GetRTC

Acquires the clock data (local time) of a MELIPC.

## Format

short CCPU\_GetRTC(short\* psGetData, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
psGetData	Clock data storage destination	Specify the storage destination of the clock data (local time).	OUT
ulBufSize	Clock data storage destination size	Specify the clock data (local time) storage destination size in word units. (When '0' is specified, this function ends normally without processing.)	IN

## Description

- This function acquires the clock data (local time) of a MELIPC, and stores it in the clock data storage destination (psGetData).
- It also acquires the information for the size specified to the clock data storage destination size (ulBufSize).
- The information to be stored in the clock data storage destination (psGetData) is as follows.  
(Available range: January 1, 1980 to December 31, 2079)

psGetData	Description
psGetData[0]	Year data (1980 to 2079)
psGetData[1]	Month data (1 to 12)
psGetData[2]	Day data (1 to 31)
psGetData[3]	Hour data (0 to 23)
psGetData[4]	Minute data (0 to 59)
psGetData[5]	Second data (0 to 59)
psGetData[6]	Day data (0 to 6) (0: Sunday, 1: Monday, 2: Tuesday, 3: Wednesday, 4: Thursday, 5: Friday, 6: Saturday)
psGetData[7]	Time zone (Unit: minute)
psGetData[8]	Daylight saving time status flag (0 to 1) (0: Not during daylight saving time, 1: During daylight saving time)

## Return value

Return value	Description
0(0000H)	Normal
-240(FF10H)	Clock data incorrect error
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 67 CCPU\_SetRTC

# CCPU\_GetSerialNo

Acquires the serial number of a MELIPC.

## Format

short CCPU\_GetSerialNo(char\* pcGetData, unsigned long ulDataSize)

## Argument

Argument	Name	Description	IN/OUT
pcGetData	Serial number storage destination	Specify the serial number storage destination.	OUT
ulDataSize	Serial number storage destination size	Specify the serial number storage destination in byte units. (When '0' is specified, this function ends normally without processing.)	IN

## Description

- This function acquires the serial number (16-digits) of a MELIPC, and stores it in the serial number storage destination (pcGetData).
- It also acquires the information for the size specified to the serial number storage destination size (ulDataSize).

## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

# CCPU\_GetSwitchStatus

Acquires the switch status of a MELIPC.

## Format

short CCPU\_GetSwitchStatus(long\* plStatusBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
plStatusBuf	Switch status storage destination	Specify the switch status storage destination.	OUT
ulBufSize	Switch status storage destination size	Specify the switch status storage destination size in double word units. (When '0' is specified, this function ends normally without processing.)	IN

## Description

- This function acquires the switch status of a MELIPC, and stores it in the switch status storage destination (plStatusBuf).
- It also acquires the information for the size specified to the switch status storage destination size (ulBufSize).
- The information to be stored in the switch status storage destination (plStatusBuf) is as follows.

plStatusBuf	Description		
	Storage position		Status
plStatusBuf[0]	bit31-6	Reserved	—
	bit5-3	STOP/MAIN RUN switch status	000: STOP state 100: MAIN RUN status Others: Reserved
	bit2-0	MELIPC RESET/WIN RESET switch status	000: MELIPC RESET status 010: NEUTRAL state 100: WIN RESET status Others: Reserved

## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## CCPU\_MELIPCShutdown

Performs a shutdown operation so that the power of a MELIPC can turn OFF.

### Format

short CCPU\_MELIPCShutdown(void)

### Argument

None

### Description

- Runs processing equivalent to when the SHUTDOWN switch of the MELIPC was operated and sets it to a state in which the power can be turned OFF.
- If application software is running in the Windows part, the shutdown process may not be completed.

### Return value

Return value	Description
0(0000H)	Normal


# CCPU\_ReadDevice

Reads data from the internal user devices and internal system devices of a MELIPC.

## Format

short CCPU\_ReadDevice(short sDevType, unsigned long ulDevNo, unsigned long ulSize, unsigned short\* pusDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number. (Only multiples of 16 can be specified for bit devices.)	IN
ulSize	Data size	Specify the read data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of read data.	OUT
ulBufSize	Data storage destination size	Specify the data storage destination size in word units.	IN

## Description

- This function reads data in a device after one specified to the device type (sDevType) and the start device number (ulDevNo) for the size specified to the data size (ulSize), and stores it in the data storage destination (pusDataBuf).
- When "0" was specified for the data size (ulSize), a specification size error occurs.

## Precautions

Note that the size of data storage destination (ulBufSize) should be equal to or bigger than the data size (ulSize).

## Return value

Return value	Description
0(0000H)	Normal
-210(FF2EH)	Read area size error
-253(FF03H)	Device number specification error
-254(FF02H)	Device type specification error
-255(FF01H)	Size specification error
-264(FEF8H)	Pointer error
-28158(9202H)	WDT error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 79 CCPU\_WriteDevice


# CCPU\_ReadLinkDevice

Reads data from link devices of CC-Link IE Field Network.

## Format

short CCPU\_ReadLinkDevice(unsigned short usloNo, short sDevType, unsigned long ulDevNo, unsigned long ulSize, unsigned short\* pusDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
usloNo	Module position	Specify 0x03E0.	IN
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number. (Only multiples of 16 can be specified for bit devices.)	IN
ulSize	Data size	Specify the read data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of read data.	OUT
ulBufSize	Data storage destination size	Specify the data storage destination size in word units.	IN

## Description

- Reads data in the link device of CC-Link IE Field Network after the one specified to the device type (sDevType) and the start device number (ulDevNo) for the size specified to the data size (ulSize), and stores it to the data storage destination (pusDataBuf).
- When "0" was specified for the data size (ulSize), a specification size error occurs.

## Precautions

Note that the size of data storage destination (ulBufSize) should be equal to or bigger than the data size (ulSize).

## Return value

Return value	Description
0(0000H)	Normal
-210(FF2EH)	Read area size error
-253(FF03H)	Device number specification error
-254(FF02H)	Device type specification error
-255(FF01H)	Size specification error
-264(FEF8H)	Pointer error
-28158(9202H)	WDT error
-28632(9028H)	Module position specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 80 CCPU\_WriteLinkDevice



# CCPU\_ReadSharedMemory

Reads data from the shared memory of a MELIPC.

## Format

short CCPU\_ReadSharedMemory(unsigned long ulOffset, unsigned long ulSize, unsigned char\* pucDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
ulOffset	Offset	Specify the offset in byte units.	IN
ulSize	Data size	Specify the read data size in byte units.	IN
pucDataBuf	Data storage destination	Specify the storage destination of read data.	OUT
ulBufSize	Data storage destination size	Specify the data storage destination size in byte units.	IN

## Description

- Reads data of the data size (ulSize) from an address moved by the amount of the offset (ulOffset) from the start of the shared memory, and then stores it in the data storage destination (pusDataBuf).
- The size of the shared memory area is 256 MB (268,435,456 bytes (10000000H)). Specify the offset (ulOffset) and data size (ulSize) so that data in the shared memory area is read.
- When "0" was specified for the data size (ulSize), a memory size specification error occurs.

## Precautions

Note that the size of data storage destination (ulBufSize) should be equal to or bigger than the data size (ulSize).

## Return value

Return value	Description
0(0000H)	Normal
-208(FF30H)	Offset error
-209(FF2FH)	Memory size specification error
-210(FF2EH)	Read area size error
-264(FEF8H)	Pointer error
-375(FE89H)	Shared memory access error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 81 CCPU\_WriteSharedMemory

# CCPU\_ReceiveInterOSEvent

Waits for notification of inter OS events.

## Format

short CCPU\_ReceiveInterOSEvent(short sOsEventNo, unsigned long ulTimeout)

## Argument

Argument	Name	Description	IN/OUT
sOsEventNo	Inter OS event number	Specifies '1'. If a value other than "1" is specified, an error is returned.	IN
ulTimeout	Timeout	Specify the timeout value in milliseconds. (0H to FFFFFFFFH) (When 'FFFFFFFH' is specified, the function waits for an event infinitely.)	IN

## Description

- Waits for notification of the inter OS event specified by the inter OS event number (sOsEventNo).
- When an inter OS event was already reported when this function was executed, the user application restarts soon after the inter OS event wait status.
- The specified timeout value is rounded (rounding up after the decimal point) to tick units. Specify a timeout value of 1 tick or higher.
- When the same inter OS event was reported multiple times when this function was executed, the user application processes them as a single inter OS event notification.
- When an inter OS event was reported when multiple tasks were waiting for the same inter OS event notification, only the task that executed this function first receives the notification and restarts execution.

## Return value

Return value	Description
0(0000H)	Normal
-231(FF19H)	Event timeout error
-234(FF16H)	Event wait error
-366(FE92H)	Event number specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

# CCPU\_RegistEventLog

Registers event logs in the event history of a MELIPC.

## Format

short CCPU\_RegistEventLog(long IEventCode, char\* pcEventMsg)

## Argument

Argument	Name	Description	IN/OUT
IEventCode	Detailed code	Specify a detailed event code to be registered in the event history.	IN
pcEventMsg	Detailed information	Specify detailed information character string data of an event to be registered in the event history. (The detailed information character string data of an event can be specified up to 200 bytes by Shift JIS. When 'NULL' is specified, the detailed information is not registered. )	IN

## Description

- Registers event logs in the event history of a MELIPC.
- The registration information is as follows.

Item	Description
Occurrence Date	Event registered date and time
Event Type	Operation (fixed)
Status	Information (fixed)
Event Code	25000 (fixed)
Overview	Registration from the user program (fixed)
Source function	Basic function (fixed)
Detailed event code information	Detailed code (hexadecimal) specified to the detailed code (IEventCode)
Detailed event log information	Detailed information specified to the detailed information (pcEventMsg)
Cause	An event history was registered with the C Controller module dedicated function. (Fixed)

- The event history can be stored for the size of the event history file specified with MI Configurator.  
Note that data is deleted starting from older data if the specified file size is exceeded.
- An error occurs if the character string data specified for the detailed information (pcEventMsg) is 201 bytes or bigger.

## Return value

Return value	Description
0(0000H)	Normal
-237(FF13H)	Source/event information character string specification error
-238(FF12H)	Event log registration error

For details on error codes, refer to the following section.

 Page 116 Function error codes

# CCPU\_Reset

Resets a MELIPC.

## Format

short CCPU\_Reset(void)

## Argument

None

## Precautions

When this function is executed, no value is returned because a MELIPC is restarted from an operating system. (All programs are forcibly terminated.)

## Description

- Resets a MELIPC.
- Use this function only to reset and restart, such as when an error has occurred.
- Do not execute this function while a file is being written to the user drive. Data corruption or a file system error may occur.
- If a file is being written to the user drive, always close the file before executing this function. Alternatively, run the CCPU\_ShutdownRom function and shut down the user drive.
- Resets the entire MELIPC, including the Windows part. If this function is executed while file writing is occurring on the Windows part, data corruption or a file system error may occur on the Windows part.
- This function can be executed only when all the following conditions are satisfied. If a condition is not satisfied, the error code in parentheses ( ) is returned.

"Enable" is set to "Remote Reset" (Unset: 16523)

The operation status of the MELIPC is STOP (RUN status: 16400)

## Return value

Return value	Description
0(0000H)	Normal
16400(4010H)	RUN state not possible
16523(408BH)	Remote request cannot be executed

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 68 CCPU\_ShutdownRom


# CCPU\_ResetDevice

Resets internal user devices and internal system devices (bit devices) of a MELIPC.

## Format

short CCPU\_ResetDevice(short sDevType, unsigned long ulDevNo)

## Argument

Argument	Name	Description	IN/OUT
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Device number	Specify a device number.	IN


## Description

Resets (turns OFF) a bit device of a MELIPC specified to the device type (sDevType) and the device number (ulDevNo).

## Return value

Return value	Description
0(0000H)	Normal
-253(FF03H)	Device number specification error
-254(FF02H)	Device type specification error
-28158(9202H)	WDT error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 63 CCPU\_SetDevice

# CCPU\_ResetWDT

Resets a user WDT of the VxWorks part.

## Format

short CCPU\_ResetWDT(short sType)

## Argument

Argument	Name	Description	IN/OUT
sType	WDT type	Specify the WDT type. (When 'Reserved' is specified, an error is returned.) <ul style="list-style-type: none"><li>• 0: User WDT</li><li>• Others: Reserved</li></ul>	IN

## Description

- Resets a user WDT of the VxWorks part.
- When this function is executed without the user WDT running, an error will be returned.

## Return value


Return value	Description
0(0000H)	Normal
-220(FF24H)	WDT type error
-28158(9202H)	WDT error
-28629(902BH)	WDT not started error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 35 CCPU\_EntryWDTInt

 Page 71 CCPU\_StartWDT

 Page 72 CCPU\_StopWDT

# CCPU\_RestoreDefaultCCIEFBCycPrm

Restores the operation parameter of cyclic transmission of CC-Link IE Field Network Basic to the default value (which is set to the parameter).

## Format

short CCPU\_RestoreDefaultCCIEFBCycPrm(void)

## Argument

None


## Description

- Restores the all group operation parameters of cyclic transmission to their default value (parameters setting value).
- Operation with the specified settings begins from the next link scan after the execution of this function.
- This function exits normally after the reflection of operation parameters is completed.


## Return value

Return value	Description
0(0000H)	Normal
-331(FEB5H)	CC-Link IE Field Network Basic parameter unset

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 24 CCPU\_ChangeCCIEFBCycPrm

# CCPU\_SendInterOSEvent

Notifies inter OS events.

## Format

short CCPU\_SendInterOSEvent(short sOsEventNo)

## Argument

Argument	Name	Description	IN/OUT
sOsEventNo	Inter OS event number	Specifies '1'. If a value other than "1" is specified, an error is returned.	IN

## Description

Sends notice of the inter OS event specified by the inter OS event number (sOsEventNo) to the Windows part from VxWorks part.

## Return value

Return value	Description
0(0000H)	Normal
-366(FE92H)	Event number invalid
-376(FE88H)	Inter OS event notification error

For details on error codes, refer to the following section.

 Page 116 Function error codes




# CCPU\_SetDevice

Sets internal user devices and internal system devices (bit devices) of a MELIPC.

## Format

short CCPU\_SetDevice(short sDevType, unsigned long ulDevNo)

## Argument

Argument	Name	Description	IN/OUT
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Device number	Specify a device number.	IN


## Description

Sets (turns ON) the bit device of the MELIPC specified to the device type (sDevType) and the device number (ulDevNo).

## Return value

Return value	Description
0(0000H)	Normal
-253(FF03H)	Device number specification error
-254(FF02H)	Device type specification error
-28158(9202H)	WDT error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 59 CCPU\_ResetDevice

# CCPU\_SetDotMatrixLED

Sets values to be displayed on the dot matrix LED of a MELIPC.

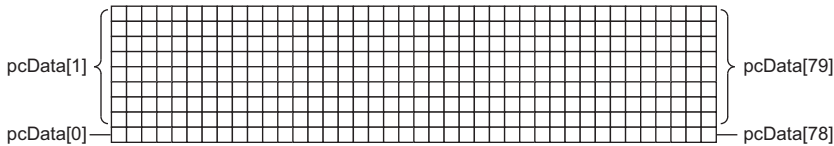
## Format

short CCPU\_SetDotMatrixLED(unsigned short usMode, char\* pcData)

## Argument

Argument	Name	Description	IN/OUT
usMode	Output mode	Specify the output mode to the dot matrix LED. (When 'Reserved' is specified, this function ends normally without processing.) • 0: Dot mode • 1: ASCII mode • Others: Reserved	IN
pcData	LED data	Specifies the data displayed on the dot matrix LED.	IN

- Specifies the LED data (pcData) as follows when the output mode (usMode) is '0' (dot mode).



pcData[0] to pcData[79]: Data on the dot matrix LED (9x40)

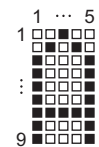
The following table shows the specified data in the following formats.

pcData	Description
pcData [even number]	The lower seven bits are all specified to '0'. The upper 1 bit specifies the bit pattern corresponding to line 9 of the dot matrix LED.
pcData [odd number]	Specifies the bit pattern corresponding to lines 1 through 8 of the dot matrix LED. (The most significant bit corresponds to line 1)

When the corresponding bit of the bit pattern section is '1', the LED turns ON.

**Ex.**

When the following bit pattern is displayed on the dot matrix LED



1st row: 0011 1111 1000 0000b=3F80H→pcData[0]=0x80, pcData[1]=0x3F

2nd row: 0100 0010 0000 0000b=4200H→pcData[2]=0x00, pcData[3]=0x42

3rd row: 1000 0010 0000 0000b=8200H→pcData[4]=0x00, pcData[5]=0x82

4th row: 0100 0010 0000 0000b=4200H→pcData[6]=0x00, pcData[7]=0x42

5th row: 0011 1111 1000 0000b=3F80H→pcData[8]=0x80, pcData[9]=0x3F

6th column to 40th column:0000 0000 0000 0000b=0000H→pcData[10] to pcData[79]=0x0000

- Specifies the LED data (pcData) as follows when the output mode (usMode) is '1' (ASCII mode).

The specified character strings are displayed in pcData[0] to pcData[5].

Available characters (ASCII code) are shown below. Note that NULL indicates null text and SP indicates a space.

	0	1	2	3	4	5	6	7
0	null		SP	0	@	P	'	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			\$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	\	l	
D			-	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

When a character other than above is specified, an error will be returned.

When the character strings are null-terminated, data after the NULL character are not displayed (blank).

### Point

When text is displayed in ASCII mode, the display width may change even with the same number of characters depending on the content to be displayed.

## Description


Displays the data specified for the LED data (pcData) on the dot matrix LED according to the mode specified for the output mode (usMode).

## Precautions

To display data on the dot matrix LED, it is necessary to set the dot matrix LED display status to normal mode and select "User" for the Display mode.

If this function is executed in a state other than the above-mentioned state, an error occurs.

For details on the display status of the dot matrix LED, refer to the following manual.

 MELIPC MI5000 Series User's Manual (Startup)


## Return value

Return value	Description
0(0000H)	Normal
-224(FF20H)	LED setting value error
-264(FEF8H)	Pointer error
-289(FEDFH)	Dot matrix LED Display mode error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 43 CCPU\_GetDotMatrixLED

# CCPU\_SetLEDStatus

Sets the INFO LED status of a MELIPC.

## Format

short CCPU\_SetLEDStatus(long lLed, unsigned short usLedInfo)

## Argument

Argument	Name	Description	IN/OUT
lLed	Target LED	Specify a target LED. (When 'Reserved' is specified, this function ends normally without processing.) • 0: INFO LED • Others: Reserved	IN
usLedInfo	LED status information	Specify the LED status information.	IN

The LED state information (usLedInfo) is specified as follows.

usLedInfo	Description
0	OFF
1	ON (Red)
2	Flashing at low speed (Red)
3	Flashing (Red)
4	ON (Green)
5	Flashing at low speed (Green)
6	Flashing (Green)

## Description

Controls the LED INFO of the MELIPC to the status specified for the LED status information (usLedInfo).

## Return value

Return value	Description
0(0000H)	Normal
-224(FF20H)	LED setting value error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 46 CCPU\_GetLEDStatus

# CCPU\_SetRTC

Sets the clock data (local time) of a MELIPC.

## Format

short CCPU\_SetRTC(short\* psSetData)

## Argument

Argument	Name	Description	IN/OUT
psSetData	Clock data	Specify the clock data (local time) to be set.	IN

- Specify the clock data (local time) to the clock data (psSetData) as follows.  
(Available range: January 1, 1980 to December 31, 2079)

psSetData	Description
psSetData[0]	Year data (1980 to 2079)
psSetData[1]	Month data (1 to 12)
psSetData[2]	Day data (1 to 31)
psSetData[3]	Hour data (0 to 23)
psSetData[4]	Minute data (0 to 59)
psSetData[5]	Second data (0 to 59)

## Description

- This function sets the time (local time) specified to the clock data (psSetData) to the MELIPC.
- If a clock data (psSetData) value is out of the range, an error is returned.
- Once the clock data (local time) is set, the history set to the event history is registered.
- When the adjust clock for daylight saving time settings is enabled, if a time (local time) that is less than one hour from the starting time of the daylight saving time is specified to the clock data (psSetData), an error will be returned.

## Precautions

The clock data (local time) set with this function is not applied to the clock of the operating system (VxWorks).

## Return value

Return value	Description
0(0000H)	Normal
-240(FF10H)	Clock data incorrect error
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 49 CCPU\_GetRTC

# CCPU\_ShutdownRom

Shuts down the user drive of the VxWorks part.

## Format

short CCPU\_ShutdownRom(void)

## Argument

None

## Description

- Shuts down the user drive of the MELIPC and sets the VxWorks part to a state in which the power supply can be turned OFF. (The shutdown status can be checked by the CCPU\_GetCpuStatus function. )
- This function is used to shut down the user drive before turning OFF the power supply of the MELIPC. After shutdown, it is not possible to perform file operations for the user drive (create new files, delete files, or overwrite files). However, the user drive can be browsed.
- Before executing this function, be sure to stop the access to the user drive and close all files; otherwise, data corruption or file system errors may occur.
- Always turn OFF the power or reset the MELIPC after checking that shutdown is complete. If operation is continued, an error occurs when accessing files in the user drive.
- When the user drive has already been shut down, this function ends normally without processing.
- This function shuts down only the user drive of the VxWorks part. The Windows part is not affected. If necessary, set the Windows part to a state in which the power supply can be turned OFF.


## Precautions

When the SHUTDOWN switch is operated or the CPU temperature is very high, this function is automatically executed by the MELIPC built-in firmware after shutdown of the Windows part is complete.

## Return value

Return value	Description
0(0000H)	Normal
-292(FEDCH)	User drive shutdown error

For details on error codes, refer to the following section.

 Page 116 Function error codes

# CCPU\_StartCCIEFBDataAssurance

Starts data assurance for access to a link device of CC-Link IE Field Network Basic.


## Format

short CCPU\_StartCCIEFBDataAssurance(unsigned short usGroupNo, unsigned long ulTimeout)

## Argument

Argument	Name	Description	IN/OUT
usGroupNo	Group No.	Specifies a group number for data assurance. (1 to 4)	IN
ulTimeout	Timeout value	Specifies the timeout time in milliseconds. (0 to 0xFFFFFFFF) (When '0xFFFFFFFF' is specified, the function waits for an event infinitely.)	IN

## Description

- Starts data assurance for access to a link device of CC-Link IE Field Network Basic for the specified group.
- Before executing this function, set the CC-Link IE Field Network Basic parameters with MI Configurator. If parameters are not configured, an error occurs.
- During data assurance, the cyclic transmission of the CC-Link IE Field Network Basic function stops.
- Execute the CCPU\_EndCCIEFBDataAssurance function to end data assurance.
- When this function was executed while the specified group of the CC-Link IE Field Network Basic was link refreshing, it waits until link refreshing is completed. (Waits until the timeout specified value. )
- It is not possible to start two instances of data assurance for one group. If this function is executed during data assurance, operation is as follows.  
When executed by the task that started data assurance: a data assurance running error occurs.  
When executed by a task that did not start data assurance: Waits until data assurance ends. (Waits until the timeout specified value.)
- Only the task that started assurance using this function can end data assurance by the CCPU\_EndCCIEFBDataAssurance function.
- The specified timeout value (ulTimeout) is rounded (rounding up after the decimal point) to tick units.
- For the processing procedure to access a link device that underwent data assurance, refer to the following manual.  
( MELIPC MI5000 Series User's Manual (Application))

## Return value

Return value	Description
0(0000H)	Normal
-328(FEB8H)	Outside of group number range
-331(FEB5H)	CC-Link IE Field Network Basic parameter unset
-335(FEB1H)	Timeout error
-371(FE8DH)	During data assurance

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 28 CCPU\_EndCCIEFBDataAssurance

# CCPU\_StartLinkDeviceAssurance

Starts data assurance for access to a link device of CC-Link IE Field Network.


## Format

short CCPU\_StartLinkDeviceAssurance(unsigned short usloNo, unsigned long ulTimeout)

## Argument

Argument	Name	Description	IN/OUT
usloNo	Module position	Specify 0x03E0.	IN
ulTimeout	Timeout value	Specifies the timeout time in milliseconds. (0 to 0xFFFFFFFF) (When '0xFFFFFFFF' is specified, the function waits for an event infinitely.)	IN

## Description

- Starts data assurance for access to a link device of the CC-Link IE Field Network function.
- Operation is as follows during data assurance.  
The statuses of RX and RWr of own station are as at the start of data assurance and are not updated.  
The values at the start of data assurance are transmitted and retained even when the statuses of RY and RWw of own station are updated.
- Execute the CCPU\_EndLinkDeviceAssurance function to end data assurance.
- The time required until completion depends on the processing state of the CC-Link IE Field Network. (Waits until the timeout specified value. )
- To perform data assurance of a link device, it is necessary to configure "Station unit block guarantee" in the parameters of the CC-Link IE Field Network. If "Do not perform station unit block guarantee" is configured, a configuration error occurs.
- If the link has been cut, a data link error occurs.
- It is not possible to start two instances of data assurance. If this function is executed during data assurance, a data assurance running error occurs.
- The specified timeout value (ulTimeout) is rounded (rounding up after the decimal point) to tick units.
- For the processing procedure to access a link device that underwent data assurance, refer to the following manual.  
( MELIPC MI5000 Series User's Manual (Application))


## Return value

Return value	Description
0(0000H)	Normal
-362(FE96H)	CC-Link IE Field Network hardware error
-367(FE91H)	Timeout error
-368(FE90H)	Configuration error ("Do not perform station unit block guarantee" setting)
-369(FE8FH)	Data link error (not linked)
-371(FE8DH)	During data assurance
-28632(9028H)	Module position specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 29 CCPU\_EndLinkDeviceAssurance



# CCPU\_StartWDT

Sets the user WDT of the VxWorks part and starts the user WDT.

## Format

short CCPU\_StartWDT(short sType, short sInterval)

## Argument

Argument	Name	Description	IN/OUT
sType	WDT type	Specify the WDT type. (When 'Reserved' is specified, an error is returned.) • 0: User WDT • Others: Reserved	IN
sInterval	WDT interval	Specify the interval of WDT in 10 ms units. (Available range is from 10 to 1,000 (100 to 10,000 ms).)	IN

## Description

- The user WDT is the timer for detecting a hardware failure or program error.
- Sets an interval of the WDT of the VxWorks part to the WDT interval (sInterval) ×10 ms and starts the user WDT.
- If the user WDT is not reset periodically (by execution of the CCPU\_ResetWDT function) within the time set for the WDT interval (sInterval), a user WDT error will occur. When a user WDT error occurs, the MELIPC enters a stop error state. (The MAIN ERR LED flashes.)
- When this function is executed while the user WDT is running, an error is returned.

## Return value

Return value	Description
0(0000H)	Normal
-211(FF2DH)	Time setting error
-220(FF24H)	WDT type error
-28158(9202H)	WDT error
-28630(902AH)	WDT startup error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 35 CCPU\_EntryWDTInt

 Page 60 CCPU\_ResetWDT

 Page 72 CCPU\_StopWDT

# CCPU\_StopWDT

Stops the user WDT of the VxWorks part.

## Format

short CCPU\_StopWDT(short sType)

## Argument

Argument	Name	Description	IN/OUT
sType	WDT type	Specify the WDT type. (When 'Reserved' is specified, an error is returned.) <ul style="list-style-type: none"><li>• 0: User WDT</li><li>• Others: Reserved</li></ul>	IN

## Description

- Stops the user WDT of the VxWorks part.
- When this function is executed without the user WDT running, it ends normally.

## Return value

Return value	Description
0(0000H)	Normal
-220(FF24H)	WDT type error


For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 35 CCPU\_EntryWDTInt

 Page 60 CCPU\_ResetWDT

 Page 71 CCPU\_StartWDT

# CCPU\_SysClkRateGet

Reads the system clock rate specified with the CCPU\_SysClkRateSet function from MRAM.

## Format

short CCPU\_SysClkRateGet(short\* psTicks)

## Argument

Argument	Name	Description	IN/OUT
psTicks	Clock rate	Specify the system clock rate in the unit of clock frequency (Hz) per one second. • 0: Default value (60 Hz) • 60 to 1000: Specified clock rate value	OUT

## Description

Reads the system clock rate specified with the CCPU\_SysClkRateSet function from MRAM.


## Precautions

The read value may not correspond to the system clock rate in operation.  
To check the system clock rate in operation, use the sysClkRateGet function of VxWorks.

## Return value

Return value	Description
0(0000H)	Normal
-264(FEF8H)	Pointer error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 74 CCPU\_SysClkRateSet

# CCPU\_SysClkRateSet

Saves a specified system clock rate to MRAM.

## Format

short CCPU\_SysClkRateSet(short sTicks, short\* psRestart)

## Argument

Argument	Name	Description	IN/OUT
sTicks	Clock rate	Specify the system clock rate in the unit of clock frequency (Hz) per one second. • 0: Default value (60 Hz) • 60 to 1000: Specified clock rate value	IN
psRestart	Restart necessity flag	Stores the necessity to restart a MELIPC after the execution of this function. (When 'NULL' is specified, the restart necessity flag is not stored.) • 0: Restart is not required. (C Controller module has already been running at the specified clock rate.) • 1: Restart is required. (C Controller module operates at the specified clock rate after restarting it.)	OUT

## Description

- Saves a system clock rate specified with the clock rate (sTicks) to MRAM.
- The specified system clock rate will be enabled after restarting a MELIPC.
- When the output to the restart necessity flag (psRestart) is '0' (restart is not required), continue the application processing.
- When the output to the restart necessity flag (psRestart) is '1' (restart is required), stop the application processing, and restart the MELIPC (turn the power supply OFF and ON).
- For details on the system clock rate, refer to the VxWorks Manual.

## Precautions

- Execute this function only once after the MELIPC is started.  
If this function is executed again when specifying the same clock rate as the first time, the restart necessity flag (psRestart) will be '0' (restart is not required) regardless of the system clock rate value in operation.
- Use this function when changing the system clock rate.  
If the sysClkRateSet function of VxWorks is used, the operation of VxWorks will be unstable.

## Return value

Return value	Description
0(0000H)	Normal
-320(FEC0H)	Clock rate specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 73 CCPU\_SysClkRateGet

# CCPU\_ToBuf

Writes data to the buffer memory of a MELIPC.

## Format

short CCPU\_ToBuf(unsigned short usloNo, unsigned long ulOffset, unsigned long ulSize, unsigned short\* pusDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
usloNo	Module position	Specify 0x03E0.	IN
ulOffset	Offset	Specify the offset in word units.	IN
ulSize	Data size	Specify the write data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of write data.	IN
ulBufSize	Data storage destination size	Specify '0'.	IN

## Description

- Writes data of the data storage destination (pusDataBuf) of the data size (ulSize) to the address moved by the amount of the offset (ulOffset) from the start of the buffer memory.
- When "0" was specified for the data size (ulSize), a memory size specification error occurs.
- If this function is executed when the operating status of the MELIPC is not RUN, a STOP error occurs.

## Return value

Return value	Description
0(0000H)	Normal
-208(FF30H)	Offset error
-209(FF2FH)	Memory size specification error
-264(FEF8H)	Pointer error
-28158(9202H)	WDT error
-28632(9028H)	Module position specification error
-28640(9020H)	STOP error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 37 CCPU\_FromBuf

# CCPU\_WaitSwitchEvent

Waits for a switch interruption event to occur.

## Format

short CCPU\_WaitSwitchEvent(short sSwitch, unsigned long ulTimeout)

## Argument

Argument	Name	Description	IN/OUT
sSwitch	Switch interrupt event type	Specify a switch interrupt event type. • 0: MAIN RUN switch interrupt event • 1: STOP switch interrupt event If a value other than the above-mentioned is specified, an error is returned.	IN
ulTimeout	Timeout	Specify the timeout value in milliseconds. (0H to FFFFFFFFH) (When 'FFFFFFFH' is specified, the function waits for an event infinitely.)	IN


## Description

- This function waits for a switch interrupt event specified to the switch interrupt event type (sSwitch).
- If an interrupt event has already been reported when this function is executed, this function immediately ends normally.
- If the same switch interrupt event has been reported several times at a time when this function is executed, the user program executes processing as a single switch interrupt event.
- The specified timeout value is rounded to tick unit. Specify a timeout value of one tick or more.

## Return value

Return value	Description
0(0000H)	Normal
-231(FF19H)	Event timeout error
-234(FF16H)	Event wait error
-257(FEFFFH)	Interrupt type specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

# CCPU\_WaitTimerEvent

Waits for a timer event to occur.

## Format

short CCPU\_WaitTimerEvent(long IEventNo)

## Argument

Argument	Name	Description	IN/OUT
IEventNo	Timer event number	Specify a timer event number that waits for a timer event to occur. (1 to 16)	IN

## Description

- This function waits for a timer event specified to the timer event number (IEventNo) to occur.
- The occurrence cycle of the timer event number (1 to 16) can be set, changed, or cleared by the CCPU\_EntryTimerEvent function.
- When reset operation is performed, any event that has occurred prior to reset is discarded.
- Using this function enables a cycle timer task. However, even though an event occurs, the waiting task may not be operated immediately due to the system status (such as the occurrence of an interrupt).
- If waiting for an event with this function to a cleared timer event, the wait status will not be cleared until an event occurs after the registration of the event (and the specified cycle has elapsed) with CCPU\_EntryTimerEvent function.

## Precautions

Note that event waiting (function return) operation depends on the synchronization type value of the timer event number specified by the CCPU\_EntryTimerEvent function.

- When the synchronization type is '0' (batch synchronization), the waiting status of all tasks waiting for events is cleared. When there are no tasks waiting for an event to occur, the waiting status is not cleared even when this function is executed afterwards.
- When the synchronization type is '1' (individual synchronization), the waiting status of one task waiting for an event is cleared. When multiple tasks are waiting for the same event, the waiting status is cleared based on the priority order of tasks (or the waiting execution order when the priority is the same).  
When there are no tasks waiting for an event to occur, the waiting status is cleared immediately when this function is executed afterwards.

## Return value

Return value	Description
0(0000H)	Normal
-225(FF1FH)	Event number specification error
-234(FF16H)	Event wait error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 33 CCPU\_EntryTimerEvent

# CCPU\_WinReset

Forces a restart of the Windows part of a MELIPC.

## Format

short CCPU\_WinReset(void)

## Argument

None

## Description

- Runs processing equivalent to when the WIN RESET switch of the MELIPC was operated and forces a restart of the Windows part of the MELIPC.
- If this function is executed while file writing is occurring on the Windows part, data corruption or a file system error may occur.
- To execute this function, it is necessary to allow a forced restart of the Windows part in the parameters. Otherwise, an error occurs.  
[Basic Parameters] ⇔ "Operation Related Settings"⇔ "Windows part forced restart"
- If the MELIPC shutdown function is run, and the MELIPC enters a state in which the power supply can be turned OFF, a forced restart of the Windows part is not performed, and this function ends normally.

## Return value

Return value	Description
0(0000H)	Normal
-374(FE8AH)	Windows part forced restart not permitted

For details on error codes, refer to the following section.

 Page 116 Function error codes




# CCPU\_WriteDevice

Writes data to internal user devices and internal system devices of a MELIPC.

## Format

short CCPU\_WriteDevice(short sDevType, unsigned long ulDevNo, unsigned long ulSize, unsigned short\* pusDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number. (Only multiples of 16 can be specified for bit devices.)	IN
ulSize	Data size	Specify the write data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of write data.	IN
ulBufSize	Data storage destination size	Specify '0'.	IN

## Description

- This function writes data in the data storage destination (pusDataBuf) for the size specified to the data size (ulSize) to a device after one specified to the device type (sDevType) and the start device number (ulDevNo).
- When "0" was specified for the data size (ulSize), a specification size error occurs.

## Return value

Return value	Description
0(0000H)	Normal
-253(FF03H)	Device number specification error
-254(FF02H)	Device type specification error
-255(FF01H)	Size specification error
-264(FEF8H)	Pointer error
-28158(9202H)	WDT error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 53 CCPU\_ReadDevice


# CCPU\_WriteLinkDevice

Writes data to link devices of CC-Link IE Field Network.

## Format

short CCPU\_WriteLinkDevice(unsigned short usloNo, short sDevType, unsigned long ulDevNo, unsigned long ulSize, unsigned short\* pusDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
usloNo	Module position	Specify 0x03E0.	IN
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number. (Only multiples of 16 can be specified for bit devices.)	IN
ulSize	Data size	Specify the write data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of write data.	IN
ulBufSize	Data storage destination size	Specify '0'.	IN

## Description

- Writes data in the data storage destination (pusDataBuf) for the size specified to the data size (ulSize) to the link device of CC-Link IE Field Network after the one specified by the device type (sDevType) and the start device number (ulDevNo).
- When "0" was specified for the data size (ulSize), a specification size error occurs.

## Return value

Return value	Description
0(0000H)	Normal
-253(FF03H)	Device number specification error
-254(FF02H)	Device type specification error
-255(FF01H)	Size specification error
-264(FEF8H)	Pointer error
-28158(9202H)	WDT error
-28632(9028H)	Module position specification error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 54 CCPU\_ReadLinkDevice

# CCPU\_WriteSharedMemory

Writes data to the shared memory of a MELIPC.

## Format

short CCPU\_WriteSharedMemory(unsigned long ulOffset, unsigned long ulSize, unsigned char\* pucDataBuf)

## Argument

Argument	Name	Description	IN/OUT
ulOffset	Offset	Specify the offset in word units.	IN
ulSize	Data size	Specify the write data size in byte units.	IN
pucDataBuf	Data storage destination	Specify the storage destination of write data.	IN


## Description

- Reads data of the data size (ulSize) from an address moved by the amount of the offset (ulOffset) from the start of the shared memory, and then stores it in the data storage destination (pusDataBuf).
- When "0" was specified for the data size (ulSize), a memory size specification error occurs.
- The size of the shared memory area is 256 MB (268,435,456 bytes (10000000H)). Specify the offset (ulOffset) and data size (ulSize) so that data in the shared memory area is written.

## Return value

Return value	Description
0(0000H)	Normal
-208(FF30H)	Offset error
-209(FF2FH)	Memory size specification error
-227(FF1DH)	Control code send error
-264(FEF8H)	Pointer error
-375(FE89H)	Shared memory access error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 55 CCPU\_ReadSharedMemory

## 3.2 C Controller Module Dedicated Functions (for ISR)

The following table shows details on the C Controller module dedicated functions (for ISR).

### CCPU\_FromBuf\_ISR

Reads data from the buffer memory of a MELIPC.

#### Format

short CCPU\_FromBuf\_ISR(unsigned short usIoNo, unsigned long ulOffset, unsigned long ulSize, unsigned short\* pusDataBuf)

#### Argument

Argument	Name	Description	IN/OUT
usIoNo	Module position	Specify 0x03E0.	IN
ulOffset	Offset	Specify the offset in word units.	IN
ulSize	Data size	Specify the read data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of read data.	OUT

#### Description

Reads data of the data size (ulSize) from an address moved by the amount of the offset (ulOffset) from the start of the buffer memory, and then stores it in the data storage destination (pusDataBuf).

#### Restriction

Do not execute this function in a routine other than an interrupt routine.

#### Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Specify the address for the read data as a multiple of 2.  
Assure a read data area of the read word count.  
Do not specify nonexistent CPU buffer memory.  
Do not specify nonexistent buffer memory.

#### Return value

Return value	Description
0(0000H)	Normal
-28158(9202H)	WDT error

For details on error codes, refer to the following section.

 Page 116 Function error codes

#### Relevant function

 Page 96 CCPU\_ToBuf\_ISR

# CCPU\_GetCounterMicros\_ISR

Acquires a 1  $\mu$ s counter value of a MELIPC.

## Format

short CCPU\_GetCounterMicros\_ISR(unsigned long\* pulMicros)

## Argument

Argument	Name	Description	IN/OUT
pulMicros	1 $\mu$ s counter value storage destination	Specify the storage destination of the 1 $\mu$ s counter value.	OUT

3

## Description

- This function acquires a 1  $\mu$ s counter value of a MELIPC, and stores it in the 1  $\mu$ s counter value storage destination (pulMicros).
- The 1  $\mu$ s counter value increases by 1 every 1  $\mu$ s after the power is turned ON.
- The count cycles between 0 and 4294967295.

### Restriction

Do not execute this function in a routine other than an interrupt routine.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

- ☞ Page 40 CCPU\_GetCounterMicros
- ☞ Page 41 CCPU\_GetCounterMillis
- ☞ Page 84 CCPU\_GetCounterMillis\_ISR

# CCPU\_GetCounterMillis\_ISR

Acquires a 1 ms counter value of a MELIPC.

## Format

short CCPU\_GetCounterMillis\_ISR(unsigned long\* pulMillis)

## Argument

Argument	Name	Description	IN/OUT
pulMillis	1 ms counter value storage destination	Specify the storage destination of the 1 ms counter value.	OUT

## Description

- This function acquires a 1 ms counter value of a MELIPC, and stores it in the 1 ms counter value storage destination (pulMillis).
- The 1 ms counter value increases by 1 every 1 ms after the power is turned ON.
- The count cycles between 0 and 4294967295.

## Restriction

Do not execute this function in a routine other than an interrupt routine.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

- ☞ Page 40 CCPU\_GetCounterMicros
- ☞ Page 41 CCPU\_GetCounterMillis
- ☞ Page 83 CCPU\_GetCounterMicros\_ISR

# CCPU\_GetDotMatrixLED\_ISR

Acquires the values displayed on the dot matrix LED of a MELIPC.

## Format

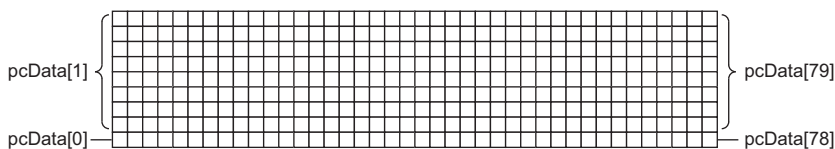
```
short CCPU_GetDotMatrixLED_ISR(char* pcData, unsigned long ulDataSize)
```

## Argument

Argument	Name	Description	IN/OUT
pcData	LED data storage destination	Specify the storage destination of LED data.	OUT
ulDataSize	LED data storage destination size	Specify the LED data storage destination size in byte units.	IN

## Description

- This function acquires the value displayed on the dot matrix LED, and stores it in the LED data storage destination (pcData).
- It also acquires the information for the size specified to the LED data storage destination size (ulBufSize).
- The value displayed on the dot matrix LED is stored in the LED data storage destination (pcData) as shown below.



pcData[0] to pcData[79]: Data on the dot matrix LED (9×40)

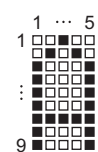
The values displayed are acquired in the following formats.

pcData	Description
pcData [even number]	The lower seven bits are all '0'. Acquires the bit pattern corresponding to line 9 of the dot matrix LED by the upper 1 bit.
pcData [odd number]	Acquires the bit pattern corresponding to lines 1 through 8 of the dot matrix LED. (The most significant bit corresponds to line 1)

When the corresponding bit of the bit pattern section is '1', the LED turns ON.

### Ex.

When the following bit pattern is displayed on the dot matrix LED



- 1st row: 0011 1111 1000 0000b=3F80H→pcData[0]=0x80, pcData[1]=0x3F  
 2nd row: 0100 0010 0000 0000b=4200H→pcData[2]=0x00, pcData[3]=0x42  
 3rd row: 1000 0010 0000 0000b=8200H→pcData[4]=0x00, pcData[5]=0x82  
 4th row: 0100 0010 0000 0000b=4200H→pcData[6]=0x00, pcData[7]=0x42  
 5th row: 0011 1111 1000 0000b=3F80H→pcData[8]=0x80, pcData[9]=0x3F  
 6th column to 40th column: 0000 0000 0000 0000b=0000H→pcData[10] to pcData[79]=0x0000

### Restriction

Do not execute this function in a routine other than an interrupt routine.

## Warning

If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

- ☞ Page 43 CCPU\_GetDotMatrixLED
- ☞ Page 64 CCPU\_SetDotMatrixLED
- ☞ Page 92 CCPU\_SetDotMatrixLED\_ISR




# CCPU\_ReadDevice\_ISR

Reads data from the internal user devices and internal system devices of a MELIPC.

## Format

short CCPU\_ReadDevice\_ISR(short sDevType, unsigned long ulDevNo, unsigned long ulSize, unsigned short\* pusDataBuf)

## Argument

Argument	Name	Description	IN/OUT
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number. (Only multiples of 16 can be specified for bit devices.)	IN
ulSize	Data size	Specify the read data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of read data.	OUT

## Description

This function reads data in a device after one specified to the device type (sDevType) and the start device number (ulDevNo) for the size specified to the data size (ulSize), and stores it in the data storage destination (pusDataBuf).

### Restriction

Do not execute this function in a routine other than an interrupt routine.


## Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Assure a read data area of the read word count.  
Do not specify a device outside of the range.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

 Page 97 CCPU\_WriteDevice\_ISR


# CCPU\_ReadLinkDevice\_ISR

Reads data from link devices of CC-Link IE Field Network.

## Format

short CCPU\_ReadLinkDevice\_ISR(unsigned short usloNo, short sDevType, unsigned long ulDevNo, unsigned long ulSize, unsigned short\* pusDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
usloNo	Module position	Specify 0x03E0.	IN
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number. (Only multiples of 16 can be specified for bit devices.)	IN
ulSize	Data size	Specify the read data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of read data.	OUT
ulBufSize	Data storage destination size	Specify the data storage destination size in word units.	IN

## Description

This function reads data in a device after one specified to the device type (sDevType) and the start device number (ulDevNo) for the size specified to the data size (ulSize), and stores it in the data storage destination (pusDataBuf).

## Precautions

Note that the size of data storage destination (ulBufSize) should be equal to or bigger than the data size (ulSize).

### Restriction

Do not execute this function in a routine other than an interrupt routine.

## Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Assure a read data area of the read word count.  
Do not specify a device outside of the range.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

 Page 98 CCPU\_WriteLinkDevice\_ISR

# CCPU\_RegistEventLog\_ISR

Registers event logs in the event history of a MELIPC.

## Format

short CCPU\_RegistEventLog\_ISR(long IEventCode, char\* pcEventMsg)

## Argument

Argument	Name	Description	IN/OUT
IEventCode	Detailed code	Specify a detailed event code to be registered in the event history.	IN
pcEventMsg	Detailed information	Specify detailed information character string data of an event to be registered in the event history. (The detailed information character string data of an event can be specified up to 200 bytes by Shift JIS. When 'NULL' is specified, the detailed information is not registered. )	IN

## Description

- The registration information is as follows.

Item	Description
Occurrence Date	Event registered date and time
Event Type	Operation (fixed)
Status	Information (fixed)
Event Code	25000 (fixed)
Overview	Registration from the user program (fixed)
Source function	Basic function (fixed)
Detailed event code information	Detailed code (hexadecimal) specified to the detailed code (IEventCode)
Detailed event log information	Detailed information specified to the detailed information (pcEventMsg)
Cause	An event history was registered with the C Controller module dedicated function. (Fixed)

- The event history can be stored for the size of the event history file specified with MI Configurator.  
Note that data is deleted starting from older data if the specified file size is exceeded.

### Restriction

Do not execute this function in a routine other than an interrupt routine.

## Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Do not specify detailed information outside of the range.

## Return value

Return value	Description
0(0000H)	Normal


# CCPU\_ResetDevice\_ISR

Resets internal user devices and internal system devices (bit devices) of a MELIPC.

## Format

short CCPU\_ResetDevice\_ISR(short sDevType, unsigned long ulDevNo)

## Argument

Argument	Name	Description	IN/OUT
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number.	IN

## Description

Resets (turns OFF) the bit device of the MELIPC specified to the device type (sDevType) and the start device number (ulDevNo).

### Restriction

Do not execute this function in a routine other than an interrupt routine.


## Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Do not specify a device outside of the range.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

 Page 91 CCPU\_SetDevice\_ISR


# CCPU\_SetDevice\_ISR

Sets internal user devices and internal system devices (bit devices) of a MELIPC.

## Format

short CCPU\_SetDevice\_ISR(short sDevType, unsigned long ulDevNo)

## Argument

Argument	Name	Description	IN/OUT
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Device number	Specify a device number.	IN

## Description

Sets (turns ON) the bit device of the MELIPC specified to the device type (sDevType) and the device number (ulDevNo).

### Restriction

Do not execute this function in a routine other than an interrupt routine.

## Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Do not specify a device outside of the range.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

 Page 90 CCPU\_ResetDevice\_ISR

# CCPU\_SetDotMatrixLED\_ISR

Sets values to be displayed on the dot matrix LED of a MELIPC.

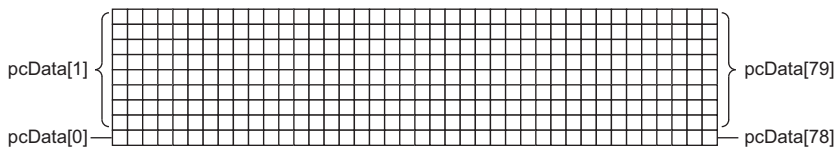
## Format

short CCPU\_SetDotMatrixLED\_ISR(unsigned short usMode, char\* pcData)

## Argument

Argument	Name	Description	IN/OUT
usMode	Output mode	Specify the output mode to the dot matrix LED. (When 'Reserved' is specified, this function ends normally without processing.) • 0: Dot mode • 1: ASCII mode • Others: Reserved	IN
pcData	LED data	Specifies the data displayed on the dot matrix LED.	IN

- Specifies the LED data (pcData) as follows when the output mode (usMode) is '0' (dot mode).



pcData[0] to pcData[79]: Data on the dot matrix LED (9x40)

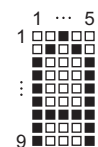
The following table shows the specified data in the following formats.

pcData	Description
pcData [even number]	The lower seven bits are all specified to '0'. The upper 1 bit specifies the bit pattern corresponding to line 9 of the dot matrix LED.
pcData [odd number]	Specifies the bit pattern corresponding to lines 1 through 8 of the dot matrix LED. (The most significant bit corresponds to line 1)

When the corresponding bit of the bit pattern section is '1', the LED turns ON.

### Ex.

When the following bit pattern is displayed on the dot matrix LED



1st row: 0011 1111 1000 0000b=3F80H→pcData[0]=0x80, pcData[1]=0x3F

2nd row: 0100 0010 0000 0000b=4200H→pcData[2]=0x00, pcData[3]=0x42

3rd row: 1000 0010 0000 0000b=8200H→pcData[4]=0x00, pcData[5]=0x82

4th row: 0100 0010 0000 0000b=4200H→pcData[6]=0x00, pcData[7]=0x42

5th row: 0011 1111 1000 0000b=3F80H→pcData[8]=0x80, pcData[9]=0x3F

6th column to 40th column:0000 0000 0000 0000b=0000H→pcData[10] to pcData[79]=0x0000

- Specifies the LED data (pcData) as follows when the output mode (usMode) is '1' (ASCII mode).

The specified character strings are displayed in pcData[0] to pcData[5].

Available characters (ASCII code) are shown below. Note that NULL indicates null text and SP indicates a space.

	0	1	2	3	4	5	6	7
0	null		SP	0	@	P	'	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			\$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	\	l	
D			-	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

When a character other than above is specified, an error will be returned.

When the character strings are null-terminated, data after the NULL character are not displayed (blank).

#### Point

When text is displayed in ASCII mode, the display width may change even with the same number of characters depending on the content to be displayed.

## Description

Displays the data specified for the LED data (pcData) on the dot matrix LED according to the mode specified for the output mode (usMode).

## Precautions

To display data on the dot matrix LED, it is necessary to set the dot matrix LED display status to normal mode and select "User" for the Display mode.

If this function is executed in a state other than the above-mentioned state, an error occurs.

For details on the display status of the dot matrix LED, refer to the following manual.

 MELIPC MI5000 Series User's Manual (Startup)

## Restriction

Do not execute this function in a routine other than an interrupt routine.

## Warning

If any function in which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.

## Return value

Return value	Description
0(0000H)	Normal
-289(FEDFH)	Dot matrix LED Display mode error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 43 CCPU\_GetDotMatrixLED

☞ Page 64 CCPU\_SetDotMatrixLED

☞ Page 85 CCPU\_GetDotMatrixLED\_ISR



# CCPU\_SetLEDStatus\_ISR

Sets the INFO LED status of a MELIPC.

## Format

short CCPU\_SetLEDStatus\_ISR(long lLed, unsigned short usLedInfo)

## Argument

Argument	Name	Description	IN/OUT
lLed	Target LED	Specify a target LED. (When 'Reserved' is specified, this function ends normally without processing.) • 0: INFO LED • Others: Reserved	IN
usLedInfo	LED status information	Specify the LED status information.	IN

The LED state information (usLedInfo) is specified as follows.

usLedInfo	Description
0	OFF
1	ON (Red)
2	Flashing at low speed (Red)
3	Flashing (Red)
4	ON (Green)
5	Flashing at low speed (Green)
6	Flashing (Green)

## Description

Controls the LED INFO of the MELIPC to the status specified for the LED status information (usLedInfo).

### Restriction

Do not execute this function in a routine other than an interrupt routine.

## Warning

If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

Page 66 CCPU\_SetLEDStatus

# CCPU\_ToBuf\_ISR

Writes data to the buffer memory of a MELIPC.

## Format

short CCPU\_ToBuf\_ISR(unsigned short usIoNo, unsigned long ulOffset, unsigned long ulSize, unsigned short\* pusDataBuf)

## Argument

Argument	Name	Description	IN/OUT
usIoNo	Module position	Specify 0x03E0.	IN
ulOffset	Offset	Specify the offset in word units.	IN
ulSize	Data size	Specify the write data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of write data.	IN

## Description

- Writes data of the data storage destination (pusDataBuf) of the data size (ulSize) to the address moved by the amount of the offset (ulOffset) from the start of the buffer memory.
- If this function is executed when the operating status of the CPU is not RUN, an error occurs.

## Restriction

- Do not execute this function in a routine other than an interrupt routine.
- If data is written to the same CPU buffer memory (host CPU) from a routine other than the interrupt routine, the output value may be intermingled, resulting in an invalid value. Manage resources so data is not written to the same CPU buffer memory (host CPU).

## Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Specify the address for the write data as a multiple of 2.  
Do not specify nonexistent CPU buffer memory.  
Do not specify nonexistent buffer memory.

## Return value

Return value	Description
0(0000H)	Normal
-28158(9202H)	WDT error
-28640(9020H)	STOP error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 82 CCPU\_FromBuf\_ISR


# CCPU\_WriteDevice\_ISR

Writes data to internal user devices and internal system devices of a MELIPC.

## Format

short CCPU\_WriteDevice\_ISR(short sDevType, unsigned long ulDevNo, unsigned long ulSize, unsigned short\* pusDataBuf)

## Argument

Argument	Name	Description	IN/OUT
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number. (Only multiples of 16 can be specified for bit devices.)	IN
ulSize	Data size	Specify the write data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of write data.	IN

## Description

This function writes data in the data storage destination (pusDataBuf) for the size specified to the data size (ulSize) to a device after one specified to the device type (sDevType) and the start device number (ulDevNo).

### Restriction

- Do not execute this function in a routine other than an interrupt routine.
- If data is written to the same device from a routine other than an interrupt routine, the output value may be intermingled, resulting in an invalid value. Manage resources so data is not written to the same device.

## Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Specify the address for the write data as a multiple of 2.  
Do not specify a device outside of the range.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

 Page 87 CCPU\_ReadDevice\_ISR


# CCPU\_WriteLinkDevice\_ISR

Writes data to link devices of CC-Link IE Field Network.

## Format

short CCPU\_WriteLinkDevice\_ISR(unsigned short usIoNo, short sDevType, unsigned long ulDevNo, unsigned long ulSize, unsigned short\* pusDataBuf, unsigned long ulBufSize)

## Argument

Argument	Name	Description	IN/OUT
usIoNo	Module position	Specify 0x03E0.	IN
sDevType	Device type	Specify the device type.  Page 8 Device type	IN
ulDevNo	Start device number	Specify the start device number. (Only multiples of 16 can be specified for bit devices.)	IN
ulSize	Data size	Specify the write data size in word units.	IN
pusDataBuf	Data storage destination	Specify the storage destination of write data.	IN
ulBufSize	Data storage destination size	Specify '0'.	IN

## Description

Writes data in the data storage destination (pusDataBuf) for the size specified to the data size (ulSize) to the link device of CC-Link IE Field Network after the one specified by the device type (sDevType) and the start device number (ulDevNo).

### Restriction

- Do not execute this function in a routine other than an interrupt routine.
- If data is written to the same device from a routine other than an interrupt routine, the output value may be intermingled, resulting in an invalid value. Manage resources so data is not written to the same device.

## Warning

- If any function to which an invalid argument is specified is executed, an error such as hardware failure (3C02H) may occur on the MELIPC.
- This function does not check the specified arguments.  
Pay attention to the following points when creating a program.  
Specify the address for the write data as a multiple of 2.  
Do not specify a device outside of the range.

## Return value

Return value	Description
0(0000H)	Normal

## Relevant function

 Page 88 CCPU\_ReadLinkDevice\_ISR

## 3.3 MELSEC Data Link Functions

This section shows the details of MELSEC data link functions.

### mdClose

Closes a communication line (channel).

#### Format

short mdClose(long IPath)

#### Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN


#### Description

- This function closes the channel opened by the mdOpen function.
- When using multiple channels, close the channel one by one.


#### Return value

Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

 Page 116 Function error codes

#### Relevant function

 Page 103 mdOpen


# mdControl

Performs remote operations (RUN/STOP/PAUSE) for a programmable controller CPU.

## Format

short mdControl(long IPath, short sStNo, short sCode)

## Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN
sStNo	Network number and station number	Specify the network number and station number of the target module.  Page 16 Network number and station number	IN
sCode	Instruction code	Specify the contents of the remote operation in numerical value.	IN

- The following remote operations can be specified by instruction code (sCode).

sCode	Description
0	Remote RUN
1	Remote STOP
2	Remote PAUSE

## Description

Changes the status of a CPU module specified to the network number and station number (sStNo) to the status specified to the instruction code (sCode).

### Restriction

This function cannot be executed for the MELIPC, C Controller module, PC CPU module, or WinCPU module.


## Return value


Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 99 mdClose

 Page 103 mdOpen



# mdDevRstEx

Resets bit devices.

## Format

long mdDevRstEx(long IPath, long INetNo, long IStNo, long IDevType, long IDevNo)

## Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN
INetNo	Network number	Specify the network number of target module.	IN
IStNo	Station number	Specify the station number of target module.  Page 16 Network number and station number	IN
IDevType	Device type	Specify the device type of bit device.  Page 17 Device type	IN
IDevNo	Device number	Specify the device number of bit device.	IN

## Description

- This function resets (turns OFF) the bit device of the module specified to the network number (INetNo), the station number (IStNo), the device type (IDevType), and the device number (IDevNo).
- This function is exclusively for bit devices such as a link relay (B) or internal relay (M).


## Return value


Return value	Description
0(0000H)	Normal
Other than 0	Error


For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 99 mdClose

 Page 102 mdDevSetEx

 Page 103 mdOpen



# mdDevSetEx

Sets bit devices.

## Format

long mdDevSetEx(long IPath, long INetNo, long IStNo, long IDevType, long IDevNo)

## Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN
INetNo	Network number	Specify the network number of target module.	IN
IStNo	Station number	Specify the station number of target module.  Page 16 Network number and station number	IN
IDevType	Device type	Specify the device type of bit device.  Page 17 Device type	IN
IDevNo	Device number	Specify the device number of bit device.	IN

## Description

- This function sets (turns ON) the bit device of the module specified to the network number (INetNo), station number (IStNo), device type (IDevType), and device number (IDevNo).
- This function is exclusively for bit devices such as a link relay (B) or internal relay (M).


## Return value


Return value	Description
0(0000H)	Normal
Other than 0	Error


For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 99 mdClose

 Page 101 mdDevRstEx

 Page 103 mdOpen




# mdOpen

Opens a communication line (channel).

## Format

short mdOpen(short sChan, short sMode, long\* plPath)

## Argument

Argument	Name	Description	IN/OUT
sChan	Channel	Specify a communication line (channel).  Page 16 Channel	IN
sMode	Mode	Specify '-1'.	IN
plPath	Path of channel	Specify the storage destination (address) of the path of the channel. (The path of the opened channel is stored.)	OUT


## Description

- When executing another MELSEC data link function, use the path of a channel opened by this function.
- To end a user program, close the opened path of a channel with the mdClose function.
- When using multiple channels, open the channel one by one.


## Return value

Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 99 mdClose

# mdRandREx

Reads data by specifying a device type and a range to be read.

## Format

long mdRandREx(long IPath, long INetNo, long IStNo, long\* pIDev, short\* psBuf, long IBufSize)

## Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN
INetNo	Network number	Specify the network number of target module. ☞ Page 16 Network number and station number	IN
IStNo	Station number	Specify the station number of target module. ☞ Page 16 Network number and station number	IN
pIDev	Randomly selected device	Specify the number of blocks, device type, start device number, and device points of devices to be read.	IN
psBuf	Read data storage destination	Specify the storage destination (address) of read data.	OUT
IBufSize	Read data storage destination size	Specifies the area size reserved in the read data storage destination in word units.	IN

The specification method of the randomly selected device (pIDev) is as follows:

pIDev	Description
pIDev[0]	Number of blocks
pIDev[1]	Device type
pIDev[2]	Start device number
pIDev[3]	Number of read points
pIDev[4]	Device type
pIDev[5]	Start device number
pIDev[6]	Number of read points
⋮	⋮
pIDev[3(n-1)+1]	Device type
pIDev[3(n-1)+2]	Start device number
pIDev[3(n-1)+3]	Number of read points

## Description

- This function reads devices specified to the randomly selected device (pIDev) from a module specified to the network number (INetNo) and the station number (IStNo).
- The read data is stored in the read data storage destination (psBuf) in word units in order of the specification to the randomly selected device (pIDev). A bit device is stored per 16 points, a word device is stored per 1 point, and a double-word device is stored in word units.
- Specify so that the total number of read points specified for each block is 10,240 points or less. Otherwise, a size error occurs.
- Communication time varies significantly depending on the contents specified to the randomly selected device (pIDev). To reduce communication time, use the mdReceiveEx function.
- To access the own station, set the station number to 255. When the actual station number is specified, an error occurs.

## Example

The following tables show the examples of values configured to the random-specified devices (plDev), data read to the read data storage destination (psBuf), and the number of bytes of read data.

Device to be read randomly	Current value
M100 to M115	All bits are OFF.
D10 to D13	10 is stored in D10, 200 is stored in D11, 300 is stored in D12, and 400 is stored in D13.
M0 to 13	All bits are ON.
T10 current value	The current value of T10 is one second (10).
LCN100 to LCN101	0x1 is stored in LCN100 and 0x10000 is stored in LCN101.

- Values specified to the randomly selected device (plDev)

plDev	Specified value	Description	
plDev[0]	5	Number of blocks = 5	—
plDev[1]	DevM	Device type = M	Block 1: M100 to M115
plDev[2]	100	Start device number = 100	
plDev[3]	16	Number of read points = 16	
plDev[4]	DevD	Device type = D	
plDev[5]	10	Start device number = 10	Block 2: D10 to D13
plDev[6]	4	Number of read points = 4	
plDev[7]	DevM	Device type = M	
plDev[8]	0	Start device number = 0	
plDev[9]	14	Number of read points = 14	Block 3: M0 to M13
plDev[10]	DevTN	Device type = T	
plDev[11]	10	Start device number = 10	
plDev[12]	1	Number of read points = 1	
plDev[13]	DevLCN	Device type = LCN	Block 4: T10
plDev[14]	100	Start device number = 100	
plDev[15]	2	Number of read points = 2	
			Block 5: LCN100 to LCN101

- Data to be read to the read data storage destination (psBuf)

psBuf	Read device	Read value	Description
psBuf[0]	M100 to M115	0	All the bit devices from M100 to M115 are OFF.
psBuf[1]	D10	10	D10=10
psBuf[2]	D11	200	D11=200
psBuf[3]	D12	300	D12=300
psBuf[4]	D13	400	D13=400
psBuf[5]	M0 to M13	3FFFH	All the bit devices from M0 to M13 are ON.
psBuf[6]	T10	10	The current value of T10 is 10 (one second).
psBuf[7]	LCN100	0x1	Lower bit of LCN100 = 0x0001
psBuf[8]			Upper bit of LCN100 = 0x0000
psBuf[9]	LCN101	0x10000	Lower bit of LCN101 = 0x0000
psBuf[10]			Upper bit of LCN101 = 0x0001


- Number of bytes of read data

(psBuf[0] to psBuf[10] = 11) × 2 = 22


## Return value


Return value	Description
0(0000H)	Normal
Other than 0	Error


For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 99 mdClose

 Page 103 mdOpen

 Page 107 mdRandWEx

# mdRandWEx

Writes data by specifying a device type and a range to be written.

## Format

long mdRandWEx(long IPath, long INetNo, long IStNo, long\* pIDev, short\* psBuf, long lBufSize)

## Argument

Argument	Name	Description	IN/OUT
IPath	Channel	Specify the path of the channel.	IN
INetNo	Network number	Specify the network number of target module. ☞ Page 16 Network number and station number	IN
IStNo	Station number	Specify the station number of target module. ☞ Page 16 Network number and station number	IN
pIDev	Randomly selected device	Specify the number of blocks, device type, start device number, and device points of devices to be written.	IN
psBuf	Write data storage destination	Specify the storage destination (address) of write data. Reserve a continuous area for the write data storage destination.	IN
lBufSize	Data storage destination size	Unused (Even if a value is configured, operation is not affected.)	IN

The specification method of the randomly selected device (pIDev) is as follows:

pIDev	Description	
pIDev[0]	Number of blocks	
pIDev[1]	Device type	Block 1
pIDev[2]	Start device number	
pIDev[3]	Number of write points	
pIDev[4]	Device type	Block 2
pIDev[5]	Start device number	
pIDev[6]	Number of write points	
⋮	⋮	⋮
pIDev[3(n-1)+1]	Device type	Block n
pIDev[3(n-1)+2]	Start device number	
pIDev[3(n-1)+3]	Number of write points	

## Description

- This function writes data to a device, which is specified to the randomly selected device (pIDev), of a module specified to the network number (INetNo) and the station number (IStNo).
- The data to be written is stored to the write data storage destination (psBuf) in word units. A bit device is stored per 16 points, a word device is stored per 1 point, and a double-word device is stored in word units.
- Specify so that the total number of write points specified for each block is 10,240 points or less. Otherwise, a size error occurs.
- Note that the extension comment information is deleted when the data is written to the block to which an extension comment is assigned (extension file register).
- Also, note that sub 2 or sub 3 program will be deleted when data is written to a block (extension file register) overlapping with the program setting area for sub 2 or sub 3.

## Example

The following tables show the examples of random-specified devices (plDev), data configured for the write data storage destination (psBuf), and the number of bytes of write data.

Device to be written randomly	Description
M100 to M115	All bits are OFF.
D10 to D13	Stores 10 in D10, 200 in D11, 300 in D12, and 400 in D13.
LCN100 to LCN101	Stores 0x1 in LCN100, and 0x10000 in LCN101.

- Values specified to the randomly selected device (plDev)

plDev	Specified value	Description	
plDev[0]	3	Number of blocks = 3	—
plDev[1]	DevM	Device type = M	Block 1: M100 to M115
plDev[2]	100	Start device number = 100	
plDev[3]	16	Number of read points = 16	
plDev[4]	DevD	Device type = D	
plDev[5]	10	Start device number = 10	Block 2: D10 to D13
plDev[6]	4	Number of read points = 4	
plDev[7]	DevLCN	Device type = LCN	
plDev[8]	100	Start device number = 100	Block 3: LCN100 to LCN101
plDev[9]	2	Number of read points = 2	

- Data configured to the write data storage destination (psBuf)

psBuf	Specified value	Description
psBuf[0]	0	All the bit devices from M100 to M115 are OFF.
psBuf[1]	10	D10=10
psBuf[2]	200	D11=200
psBuf[3]	300	D12=300
psBuf[4]	400	D13=400
psBuf[5]	0x0001	Lower bit of LCN100
psBuf[6]	0x0000	Upper bit of LCN100
psBuf[7]	0x0000	Lower bit of LCN101
psBuf[8]	0x0001	Upper bit of LCN101

- Number of bytes of write data

(psBuf[0] to psBuf[8] = 9) × 2 = 18

## Return value

Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 99 mdClose

☞ Page 103 mdOpen

☞ Page 104 mdRandREx

# mdReceiveEx

Reads data from devices in a batch.

## Format

long mdReceiveEx(long IPath, long INetNo, long IStNo, long IDevType, long IDevNo, long\* pISize, short\* psData)

## Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN
INetNo	Network number	Specify the network number of target module. ☞ Page 16 Network number and station number	IN
IStNo	Station number	Specify the station number of target module. ☞ Page 16 Network number and station number	IN
IDevType	Device type	Specifies the device type for the device to be read in batch.	IN
IDevNo	Start device number	Specifies the start device number for the device to be read in batch. (For bit devices, specify the device number in multiples of 8.)	IN
pISize	Read data size	Specify the read size in byte units. (Specifies the value in multiples of 4 when a double-word device (LZ, LTN, LCN, LSTN) is specified and specifies the value in multiples of 2 when a word device or bit device is specified. If other values are specified, a size error occurs.)	IN/OUT
psData	Read data storage destination	Specify the storage destination (address) of read data.	OUT

## Description

- This function reads data for the size specified for the read data size (pISize) from a device specified to the device type (IDevType) and the start device number (IDevNo) of a module specified to the network number (INetNo) and the station number (IStNo).
- When the read size exceeds the device range, the readable device size (the maximum value that can be specified for the read size (pISize) when '0 was specified for the start device number (IDevNo)) is returned to the read data size (pISize).

## Return value

Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 99 mdClose

☞ Page 103 mdOpen

☞ Page 112 mdSendEx



# mdRemBufReadEx

Reads data from the buffer memory of a remote device station on CC-Link IE Field Network.

## Format

long mdRemBufReadEx(long IPath, long INetNo, long IStNo, long IOffset, long\* pISize, short\* psData)

## Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN
INetNo	Network number	Specify the network number of target module.  Page 16 Network number and station number	IN
IStNo	Station number	Specify the station number of target module.  Page 16 Network number and station number	IN
IOffset	Offset	Specify the start address of the buffer memory of the remote device station to read data.	IN
pISize	Read byte size	Specifies the byte size of the read data as an even number. (2 to 480)	IN/OUT
psData	Read data	Stores the data that was read.	OUT

## Description

- Reads data from the buffer memory of a remote device station on CC-Link IE Field Network specified by the network number (INetNo) and station number (IStNo).
- If a value outside the range or an odd number is specified for the read byte size (pISize), '0 is stored as the read byte size (pISize) value.
- Make sure that the read target range does not exceed the buffer memory range of the remote device station. If the buffer memory range is exceeded, a timeout error may occur or unintended data may be read. For details on the range of the buffer memory of remote device stations, refer to the manual of each remote device station.
- Data can be read only from the buffer memory of a remote device station on CC-Link IE Field Network by this function. Data cannot be read from the buffer memory of an intelligent function module controlled by a head module on CC-Link IE Field Network.
- To read data from the buffer memory of an intelligent function module that is controlled by a head module on CC-Link IE Field Network, apply the mdReceiveEx/mdReceive function.

## Return value

Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 111 mdRemBufWriteEx





# mdRemBufWriteEx

Writes data to the buffer memory of a remote device station on CC-Link IE Field Network.

## Format

long mdRemBufWriteEx(long IPath, long INetNo, long IStNo, long IOffset, long\* pISize, short\* psData)

## Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN
INetNo	Network number	Specify the network number of target module.  Page 16 Network number and station number	IN
IStNo	Station number	Specify the station number of target module.  Page 16 Network number and station number	IN
IOffset	Offset	Specifies the start address of the buffer memory of the remote device station to write data.	IN
pISize	Write byte size	Specifies the byte size of the write data as an even number. (2 to 480)	IN/OUT
psData	Write data	Specifies the write data as a single integer array.	IN

## Description

- Writes data to the buffer memory of a remote device station on CC-Link IE Field Network specified by the network number (INetNo) and station number (IStNo).
- If a value outside the range or an odd number is specified for the write byte size (pISize), '0 is stored as the write byte size (pISize) value.
- Make sure that the write target range does not exceed the buffer memory range of the remote device station. When the buffer memory range is exceeded, a timeout error may occur or the remote device station may no longer operate normally. For details on the range of the buffer memory of remote device stations, refer to the manual of each remote device station.
- Data can be written only to the buffer memory of a remote device station on CC-Link IE Field Network by this function. Data cannot be written to the buffer memory of an intelligent function module controlled by a head module on CC-Link IE Field Network.
- To write data to the buffer memory of an intelligent function module that is controlled by a head module on CC-Link IE Field Network, apply the mdSendEx/mdSend function.

## Return value

Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 110 mdRemBufReadEx

# mdSendEx

Writes data to devices in a batch.

## Format

long mdSendEx(long IPath, long INetNo, long IStNo, long IDevType, long IDevNo, long\* plSize, short\* psData)

## Argument

Argument	Name	Description	IN/OUT
IPath	Path of channel	Specify the path of the opened channel.	IN
INetNo	Network number	Specify the network number of target module. ☞ Page 16 Network number and station number	IN
IStNo	Station number	Specify the station number of target module. ☞ Page 16 Network number and station number	IN
IDevType	Device type	Specify the device type for device to be written in batch.	IN
IDevNo	Start device number	Specify the start device number to be written in batch. (For bit devices, specify the device number in multiples of 8.)	IN
plSize	Write data size	Specify the write data size in byte units. (Specifies the value in multiples of 4 when a double-word device (LZ, LTN, LCN, LSTN) is specified and specifies the value in multiples of 2 when a word device or bit device is specified. If other values are specified, a size error occurs. )	IN/OUT
psData	Write data storage destination	Specify the storage destination (address) of write data. Reserve a continuous area for the write data storage destination.	IN

## Description

- Writes data to the module specified by the network number (INetNo) and the station number (IStNo).
- Writes data in the size specified for the write data size (plSize) to the start of the device specified by the device type (IDevType) and the start device number (IDevNo).
- It checks the arguments and verifies whether the address + size determined by the arguments is within the device memory range.
- When the write data size exceeds the device range, the writable device size (the maximum value that can be specified for the write data size (plSize) when '0' was specified for the start device number (IDevNo)) is returned to the write data size (plSize).
- Note that the extension comment information is deleted when the data is written to the block to which an extension comment is assigned (extension file register).

## Return value

Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

☞ Page 116 Function error codes

## Relevant function

☞ Page 99 mdClose

☞ Page 103 mdOpen

☞ Page 109 mdReceiveEx


# mdTypeRead

Reads the model code of a module.

## Format

short mdTypeRead(long lPath, short sStNo, short\* psCode)

## Argument

Argument	Name	Description	IN/OUT
lPath	Path of channel	Specify the path of the opened channel.	IN
sStNo	Station number	Specify the network number and station number of the target module.  Page 16 Network number and station number	IN
psCode	Model code	Specify the storage destination (address) of the model code. (Stores the read model code.)	OUT

## Description

Reads the model code of a module specified by the station number (sStNo).

For modules other than the following, an undefined value is stored to the model name code (psCode).


Model code (hexadecimal)	CPU module
0041H	Q02CPU, Q02HCPU
0042H	Q06HCPU
0043H	Q12HCPU
0044H	Q25HCPU
0049H	Q12PHCPU
004AH	Q25PHCPU
004DH	Q02PHCPU
004EH	Q06PHCPU
0250H	Q00JCPU
0251H	Q00CPU
0252H	Q01CPU
0260H	Q00UJCPU
0261H	Q00UCPU
0262H	Q01UCPU
0263H	Q02UCPU
0266H	Q10UDHCPU
0267H	Q20UDHCPU
0268H	Q03UDCPU
0269H	Q04UDHCPU
026AH	Q06UDHCPU
026BH	Q13UDHCPU
026CH	Q26UDHCPU
02E6H	Q10UDEHCPU
02E7H	Q20UDEHCPU
02E8H	Q03UDECPU
02E9H	Q04UDEHCPU
02EAH	Q06UDEHCPU
02EBH	Q13UDEHCPU
02ECH	Q26UDEHCPU
02EDH	Q50UDEHCPU
02EEH	Q100UDEHCPU
0366H	Q03UDVCPU
0367H	Q04UDVCPU
0368H	Q06UDVCPU

Model code (hexadecimal)	CPU module
036AH	Q13UDVCPU
036CH	Q26UDVCPU
0541H	L02CPU
0543H	L02SCPU
0544H	L06CPU
0545H	L26CPU
0548H	L26CPU-BT
0549H	L02CPU-P
054AH	L26CPU-PBT
0641H	LJ72GF15-T2
0642H	NZ2GF-ETB
2014H	Q172DCPU(-S1)
2015H	Q173DCPU(-S1)
2018H	Q172DSCPU
2019H	Q173DSCPU
2043H	Q12DCCPU-V
2044H	Q24DHCCPU-V
2045H	Q24DHCCPU-LS
2046H	Q24DHCCPU-VG
2047H	Q26DHCCPU-LS
4800H	R04CPU
4801H	R08CPU
4802H	R16CPU
4803H	R32CPU
4804H	R120CPU
4805H	R04ENCPU
4806H	R08ENCPU
4807H	R16ENCPU
4808H	R32ENCPU
4809H	R120ENCPU
4820H	R12CCPU-V
4841H	R08PCPU
4842H	R16PCPU
4843H	R32PCPU
4844H	R120PCPU
4851H	R08PSFCPU
4852H	R16PSFCPU
4853H	R32PSFCPU
4854H	R120PSFCPU
4860H	RJ72GF15-T2
4861H	RJ72GF15-T2(SR)
4862H	RJ72GF15-T2(LR)
4891H	R08SF CPU
4892H	R16SF CPU
4893H	R32SF CPU
4894H	R120SF CPU
48A0H	R00CPU
48A1H	R01CPU
48A2H	R02CPU
4C00H	R16MTCPU
4C01H	R32MTCPU
4C02H	R64MTCPU
4E01H	MI5122-VW


## Return value


Return value	Description
0(0000H)	Normal
Other than 0	Error

For details on error codes, refer to the following section.

 Page 116 Function error codes

## Relevant function

 Page 99 mdClose

 Page 103 mdOpen

# APPENDIX

## Appendix 1 Function error codes

This chapter shows the codes for errors occurred in the dedicated function library and the corrective actions.

### common

The following table shows the common error codes for the dedicated function library.

Error code*1		Description	Corrective action
Decimal	Hexadecimal		
1	0001H	<ul style="list-style-type: none"> <li>■Driver not started</li> </ul> The driver is not started.	<ul style="list-style-type: none"> <li>• Check the channel number.</li> <li>• Correct the error that occurred when the driver is started.</li> <li>• Check the status of the system drive of the MELIPC.</li> <li>• Check if the operating system is running normally.</li> </ul>
2	0002H	<ul style="list-style-type: none"> <li>■Timeout error</li> </ul> <ul style="list-style-type: none"> <li>• A timeout occurred while waiting for the response.</li> <li>• The module specified as the communication target is not supported.</li> </ul>	<ul style="list-style-type: none"> <li>• Review the operating status and mounting condition of the access target station.</li> <li>• Retry on the user program.</li> <li>• Increase the timeout value of MELSEC data link function.</li> <li>• Check if the module specified as the communication target is supported.</li> </ul>
66	0042H	<ul style="list-style-type: none"> <li>■Already opened error</li> </ul> The specified channel has already been opened.	The open processing is required only one time. (If this error occurred, the path of the correct channel will be returned to the argument.)
67	0043H	<ul style="list-style-type: none"> <li>■Already closed error</li> </ul> The specified channel has already been closed.	The close processing is required only one time.
69	0045H	<ul style="list-style-type: none"> <li>■Unsupported function performing error</li> </ul> An unsupported function in the target station was performed.	<ul style="list-style-type: none"> <li>• Check the path of the channel, network number, and station number.</li> <li>• Check if the function performed in the target station is supported.</li> </ul>
70	0046H	<ul style="list-style-type: none"> <li>■Station number error</li> </ul> <ul style="list-style-type: none"> <li>• The specified station number is incorrect.</li> <li>• The request for other stations was issued to the own station, or the network number was not '0' even though the station number was the own station (FFH).</li> </ul>	Correct the network number and station number specified in the user program.
77	004DH	<ul style="list-style-type: none"> <li>■Memory reservation error</li> <li>■Resource shortage error</li> <li>■Task over error</li> </ul> Securing sufficient memory failed. Or, there are too many tasks using the dedicated function library.	<ul style="list-style-type: none"> <li>• The memory may be insufficient. End other running tasks. Alternatively, reduce the excess size.</li> <li>• Reduce the number of tasks using the dedicated function library and retry the operation.</li> <li>• Review the size or number specified to the arguments of the user program.</li> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
102	0066H	<ul style="list-style-type: none"> <li>■Data send error</li> <li>■Restart error</li> </ul> Sending data failed, or an attempt was made to send data during restart.	<ul style="list-style-type: none"> <li>• Retry.</li> <li>• Retry after completion of the restart.</li> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC.</li> </ul>
103	0067H	<ul style="list-style-type: none"> <li>■Reception error</li> </ul> Receiving data failed.	<ul style="list-style-type: none"> <li>• Retry.</li> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
130	0082H	<ul style="list-style-type: none"> <li>■Device number error</li> </ul> <ul style="list-style-type: none"> <li>• The specified device number is out of range.</li> <li>• The specified bit device number is not a multiple of 8.</li> </ul>	Check the device number.
131	0083H	<ul style="list-style-type: none"> <li>■Number of device points error</li> </ul> <ul style="list-style-type: none"> <li>• The specified number of device points is out of range.</li> <li>• The specified number of bit device points is not a multiple of 8.</li> </ul>	Check the specified number of device points.
16384 to 20479	4000H to 4FFFH	<ul style="list-style-type: none"> <li>■Errors detected in the access target CPU module</li> </ul>	Refer to the user's manual of the access target CPU module.

Error code*1		Description	Corrective action
Decimal	Hexadecimal		
16480	4060H	<p>■Online registration error An online debugging function (online change, etc.) or remote operation is being executed by another engineering tool or another CPU module.</p>	<ul style="list-style-type: none"> <li>• Complete the operation of the other engineering tool or other CPU module, and then execute it again.</li> <li>• If operation of the other engineering tool or other CPU module is suspended, execute the operation again using the other engineering tool or other CPU module so that it completes normally, and then execute again.</li> </ul>
-25056	9E20H	<p>■Processing code error A request which cannot be performed in the request destination was issued.</p>	Check the network number and station number of the request destination.
-26334	9922H	<p>■Reset error</p> <ul style="list-style-type: none"> <li>• Another task using the same channel was reset while accessing another station.</li> <li>• A reset operation was performed while monitoring with MI Configurator.</li> </ul>	<ul style="list-style-type: none"> <li>• Retry.</li> <li>• Monitor again.</li> </ul>
-28150	920AH	<p>■Device access error during data link stop The devices (RX, RY, RWw, and RWr) of the own station were accessed when the data link was not performed.</p>	<ul style="list-style-type: none"> <li>• Check the specified device start number and size, or the device range of the parameter for the master station.</li> <li>• Restart the data link.</li> </ul> <p>(Note that data is written/read despite this error, however, the contents of the data will not be guaranteed.)</p>
-28151	9209H	<p>■Abnormal data reception error Abnormal response data received.</p>	Check if an error occurred in the request destination CPU module or link module. (If the status is normal, resend the request.)
-28158	9202H	<p>■WDT error WDT (system/user) error occurred.</p>	Reset the MELIPC, or turn the power OFF and ON.
-28410	9106H	<p>■Target CPU busy error The target CPU module is busy.</p>	<ul style="list-style-type: none"> <li>• Add the processing to wait for the completion of the target operation or to retry the operation in the user program.</li> <li>• Increase the timeout time specified to the argument in the user program.</li> </ul>
-28412	9104H	<p>■Target CPU unsupported error An unsupported request was issued to the target CPU module.</p>	Change the target CPU module specified in the user program.
-28413	9103H	<p>■Target CPU down error The target CPU module is down.</p>	Check the operating status of the target CPU module. In case of an error, troubleshoot while following the user manual of the target CPU module.
-28414	9102H	<p>■Target CPU abnormal start error A request was issued to the CPU module which is not operating normally.</p>	Check the operating status of the target CPU module. In case of an error, troubleshoot while following the user manual of the target CPU module.
-28415	9101H	<p>■Target CPU major error A request was issued to the CPU module in which a major error occurred.</p>	Check the operating status of the target CPU module. In case of an error, troubleshoot while following the user manual of the target CPU module.
-28416	9100H	<p>■Target CPU mounting error A request was issued by specifying the CPU number in the state where no CPU module is mounted.</p>	<ul style="list-style-type: none"> <li>• Check the mounting condition of the target CPU module.</li> <li>• Change the target CPU number specified in the user program.</li> </ul>
-28622	9032H	<p>■Target module busy error</p> <ul style="list-style-type: none"> <li>• The target module is busy.</li> <li>• The own station channel or the target station storage channel is used for other instructions, or multiple identical instructions are being executed.</li> </ul>	Add the processing to wait for the completion of the target operation or to retry the operation in the user program.
-28624	9030H	<p>■Function unsupported error</p> <ul style="list-style-type: none"> <li>• Any processing was performed to the module which does not support the station-based block data assurance function for cyclic data.</li> <li>• Any processing was performed to the module on which the station-based block data assurance function for cyclic data is not set.</li> <li>• An attempt was made to access a module which was not controlled by the host CPU module.</li> </ul>	<ul style="list-style-type: none"> <li>• Check if the target CC-Link module supports the station-based block data assurance function for cyclic data.</li> <li>• Check if the station-based block data assurance function for cyclic data is set for the target module.</li> <li>• Check if the control CPU of the target module is the host CPU module.</li> </ul>
-28626	902EH	<p>■Control data setting value out of range error The specified control data is out of range.</p>	Check the value set to the control data.
-28627	902DH	<p>■Transient unsupported error Transient transmission cannot be performed via the specified communication route and target. (Another station was specified when the station number of the own station is '64' during CC-Link communication.)</p>	<ul style="list-style-type: none"> <li>• Check the communication route and target which support the transient request.</li> <li>• Change the station number of the own station.</li> </ul>

Error code*1		Description	Corrective action
Decimal	Hexadecimal		
-28628	902CH	<b>■Pointer address specification error</b> An incorrect address was specified to the argument pointer.	Check the address of the specified pointer.
-28629	902BH	<b>■WDT not started error</b> An attempt was made to reset a WDT before starting it.	Reset the WDT after starting it.
-28630	902AH	<b>■WDT startup error</b> An attempt was made to start WDT while the other WDT is starting up.	Start the WDT after stopping the WDT which is starting up.
-28631	9029H	<b>■Buffer access range error</b> <ul style="list-style-type: none"> <li>• The specified offset is out of range.</li> <li>• The specified offset and its size are out of range.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the specified offset.</li> <li>• Check the specified buffer size.</li> <li>• Check the offset and its size.</li> </ul>
-28632	9028H	<b>■I/O number error</b> <ul style="list-style-type: none"> <li>• The specified I/O number is out of range.</li> <li>• No accessible module is mounted on the specified I/O number.</li> </ul>	Check the specified I/O number.
-28640	9020H	<b>■STOP error</b> The output or buffer memory writing was requested when the operating status of the CPU module is STOP.	Change the operation status of the CPU module to RUN.
-28653	9013H	<b>■I/O assignment error</b> <ul style="list-style-type: none"> <li>• An attempt was made to read the input value (X) from an output module.</li> <li>• An attempt was made to write the output value (Y) to an input module.</li> <li>• An attempt was made to read the output value (Y) from an input module.</li> </ul>	Check the input signal (X) and output signal (Y).
-28660	900CH	<b>■Access size error</b> The specified size is out of range.	Review the specified offset and size.
-28661	900BH	<b>■Inaccessible error</b> Inaccessible area was specified.	Review the specified offset and size.
-28662	900AH	<b>■CPU number specification error</b> The specified CPU number is out of range or unavailable.	<ul style="list-style-type: none"> <li>• Review the specified CPU number.</li> <li>• Check the operating status of the specified CPU module.</li> </ul>
-28664	9008H	<b>■Data send area occupied</b>	Retry.
-28665	9007H	<b>■No registration data error</b>	Reset the MELIPC, or turn the power OFF and ON.
-28666	9006H	<b>■Data length error</b>	Reset the MELIPC, or turn the power OFF and ON.
-28668	9004H	<b>■Reply data stored error</b>	Resend the request.
-28669	9003H	<b>■Area number error</b> The specified area number, offset address, and mode are out of range.	Review the area number, offset address, and mode.
-28671	9001H	<b>■Module identification error</b>	<ul style="list-style-type: none"> <li>• Review the parameters.</li> <li>• Check the specified module.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
-28672	9000H	<b>■Processing code error</b>	Reset the MELIPC, or turn the power OFF and ON.

\*1 When the function of which the return value is a long-type, the value will be eight digits in hexadecimal.



## C Controller module dedicated function

The following table shows the error codes of the C Controller module dedicated functions.

Error code		Description	Corrective action
Decimal	Hexadecimal		
-201	FF37H	■Module identification error The specified module identification is unavailable.	Check the specified module identification.
-203	FF35H	■I/O signal error The specified I/O signal is out of range.	Check the specified I/O signal.
-204	FF34H	■I/O access size error The specified access size of I/O signal is out of range.	Check the specified access size of I/O signal (I/O number and read/write size in words).
-208	FF30H	■Offset error • The specified offset is out of range. • An AnS series module (buffer memory) was accessed.	Check the specified offset.
-209	FF2FH	■Memory size specification error • The specified offset and its size are out of range. • The address of data storage buffer pointer is 0. • The specified size is 0.	<ul style="list-style-type: none"> <li>• Check the specified memory size.</li> <li>• Check the offset and its size.</li> <li>• Check the specified data storage buffer pointer.</li> </ul>
-210	FF2EH	■Read area size error The read area size is smaller than the read size.	<ul style="list-style-type: none"> <li>• Check the read size.</li> <li>• Check the read area size.</li> </ul>
-211	FF2DH	■Time setting error The specified time is out of range.	Check the specified time.
-217	FF27H	■Driver not started The driver is not started.	Check if the driver is started.
-220	FF24H	■WDT type error The specified WDT type is out of range.	Check the specified WDT type.
-223	FF21H	■Memory reservation error Reserving sufficient memory failed.	Check if sufficient memory is available.
-224	FF20H	■LED setting value error The specified LED setting value is out of range.	Check the specified LED setting value.
-225	FF1FH	■Event number specification error The specified event number is out of range or duplicated.	Check the specified event number.
-227	FF1DH	■Control code send error Sending control code failed.	<ul style="list-style-type: none"> <li>• Retry.</li> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC.</li> </ul>
-231	FF19H	■Event timeout error A timeout occurred while waiting for an event.	<ul style="list-style-type: none"> <li>• Increase the timeout time.</li> <li>• Check if the interrupt event number (interrupt pointer number) is set correctly.</li> </ul>
-232	FF18H	■CPU number specification error The specified CPU number is incorrect.	Check the value of the specified CPU number.
-234	FF16H	■Event wait error An error other than timeout occurred while the function waits for the event.	<ul style="list-style-type: none"> <li>• Check if a program is forcibly being terminated.</li> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
-235	FF15H	■Number of event settings specification error The specified number of event settings is out of range.	Check the number of specified event settings.
-236	FF14H	■Remote operation specification code error The remote operation specification code is out of range.	Check the specified remote operation specification code.
-237	FF13H	■Detailed information character string specification error The length of the specified character string was out of range or characters which cannot be specified was specified.	Correct the length of the specified character string or character string data.
-238	FF12H	■Event log registration error Registering an event log failed.	Reset the MELIPC, or turn the power OFF and ON.
-240	FF10H	■Clock data incorrect error The clock data to be set or the read clock data is incorrect.	<ul style="list-style-type: none"> <li>• Check the clock data to be set.</li> <li>• If this error occurs when reading the clock data, set the data again.</li> </ul>
-241	FF0FH	■Cycle specification error • The specified cycle is out of range. • The cycle was set even when it had already been set.	<ul style="list-style-type: none"> <li>• Check the specified cycle.</li> <li>• Check if the cycle has been already set.</li> </ul>
-242	FF0EH	■Synchronization type specification error The specified synchronization type is out of range.	Check the specified synchronization type.









Error code		Description	Corrective action
Decimal	Hexadecimal		
-245	FF0BH	<p>■Not executable during interrupt service routine A function was executed from interrupt service routine without specifying '1' (ISR) to the call source flag.</p>	Specify '1' to the call source flag (ISR) and execute the function again.
-246	FF0AH	<p>■Timer event registration error Registering a timer event failed.</p>	<ul style="list-style-type: none"> <li>• Retry.</li> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
-253	FF03H	<p>■Device number specification error</p> <ul style="list-style-type: none"> <li>• The specified device number is out of range.</li> <li>• The specified bit device number is not a multiple of 16.</li> </ul>	Correct the start device number of the specified device.
-254	FF02H	<p>■Device type specification error The specified device type is unavailable.</p>	Check the specified device type.
-255	FF01H	<p>■Size specification error</p> <ul style="list-style-type: none"> <li>• The specified number of words is out of range.</li> <li>• The specified size is 0.</li> </ul>	Correct the specified start device number and number of words.
-256	FF00H	<p>■Response completion wait timeout error A timeout occurred while waiting for completion of a response of a processing requested to other CPU modules.</p>	<ul style="list-style-type: none"> <li>• Increase the timeout time specified to the argument.</li> <li>• Review and correct the user program (including other tasks which execute motion CPU interaction functions).</li> <li>• Review the program used for the request destination CPU module and correct it to perform the processing requested from other CPU modules, for example, by adding the WAIT instruction.</li> </ul>
-257	FEFFH	<p>■Interrupt event type specification error The value specified to the interrupt event type is out of range.</p>	Check the specified value.
-258	FEFEH	<p>■Interrupt pointer number specification error The value specified as the interrupt pointer number is out of range.</p>	Check the specified value.
-259	FEFDH	<p>■Interrupt service routine unregistered error The processing was not registered when enabling the processing corresponding to an event (interrupt).</p>	Register the processing for the event (interrupt) and perform the operation again.
-263	FEF9H	<p>■Caller flag error The value specified to the caller flag is out of range.</p>	Review the specified value, and specify a value within the range.
-264	FEF8H	<p>■Pointer error The address of the specified pointer is incorrect.</p>	Check the address of the specified pointer.
-265	FEF7H	<p>■Target system specification error The value specified in the target system is out of range.</p>	Check the specified value.
-266	FEF6H	<p>■WDT start error The user WDT is being started.</p>	Check the user WDT settings.
-267	FEF5H	<p>■Authentication error The username or password is incorrect.</p>	Check the specified username and password.
-268	FEF4H	<p>■Security error The setting content of the security function is incorrect.</p>	Check the settings of the specified security function.
-279	FEE9H	<p>■File specification error</p> <ul style="list-style-type: none"> <li>• The specified file does not exist.</li> <li>• A file with the same name already exists. (The existing file is overwrite-protected.)</li> <li>• A file cannot be created in the specified path, or the specified path does not exist.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the specified file.</li> <li>• Check if the existing file is overwrite-protected.</li> <li>• Check if no files exist with the same name as the file to be created.</li> </ul>
-280	FEE8H	<p>■File access error The specified file is in use.</p>	Check if the specified file is in use.
-288	FEE0H	<p>■Individual identification information read error Reading individual identification information failed.</p>	<ul style="list-style-type: none"> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
-289	FEDFH	<p>■Dot matrix LED Display mode error The dot matrix LED is in standard mode, but a setting other than 'User' is selected for the Display mode.</p>	Using the menu operation, select 'User' as the Display mode, and then close the menu.
-290	FEDEH	<p>■Dot matrix LED display status error The dot matrix LED is not in normal mode.</p>	Using the menu operation, select 'User' as the Display mode, and then close the menu.
-292	FEDCH	<p>■User drive shutdown error The shutdown processing of the user drive failed.</p>	<ul style="list-style-type: none"> <li>• Check if files on the user drive are being accessed.</li> <li>• Check if all files on the user drive have been closed.</li> </ul>
-295	FED9H	<p>■Selected operation is being checked This function was executed while checking the selected operation.</p>	Execute the function after checking the operation.

Error code		Description	Corrective action
Decimal	Hexadecimal		
-296	FED8H	■Setting data size error The setting data size is out of range.	Check the setting data size.
-328	FEB8H	■Group number error The specified group number is out of range.	Check the specified group number.
-329	FEB7H	■Link time specification error The specified link scan time is out of range.	Check the specified link scan time.
-330	FEB6H	■Number of detections of disconnected slave stations specification error The specified number of detections of disconnected slave stations is out of range.	Check the specified number of detections of disconnected slave stations.
-331	FEB5H	■CC-Link IE Field Network Basic parameter unset The function was executed without setting CC-Link IE Field Network Basic parameters.	Set the CC-Link IE Field Network Basic parameters with MI Configurator.
-332	FEB4H	■Slave station timeout time specification range error	Check the timeout time for the disconnection detection of the specified slave station.
-335	FEB1H	■Link scan data assurance wait time timed out The specified data assurance wait time is elapsed.	Increase the timeout time specified to the argument.
-336	FEB0H	■Slave station number specification error The specified slave station number does not exist in the slave station.	Check the specified slave station number.
-361	FE97H	■Data assurance ended error Ending of data assurance was attempted when data assurance was not being performed.	Make sure that data assurance is ended only when the user program is performing data assurance.
-362	FE96H	■CC-Link IE Field Network hardware error Data assurance of the CC-Link IE field network cannot be started due to a hardware error.	<ul style="list-style-type: none"> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC.</li> <li>• Retry.</li> </ul>
-366	FE92H	■Event number specification error The specified event number is out of range.	Check the specified event number.
-367	FE91H	■Data assurance start timeout error A timeout occurred at the start of data assurance.	<ul style="list-style-type: none"> <li>• Increase the timeout time of the argument to be specified by the user program.</li> <li>• Retry.</li> </ul>
-368	FE90H	■Data assurance setting error There was an attempt to start data assurance when "Station-based Block Data Assurance" was "Disable" in the parameters of the CC-Link IE Field Network.	In the parameter of the CC-Link IE Field Network, change the setting in "Station-based Block Data Assurance" to "Enable".
-369	FE8FH	■Data link error Starting of data assurance was attempted when there was no data linking.	Restart the data link.
-371	FE8DH	■Data assurance running error Starting of data assurance was attempted when data assurance was already being performed.	Make sure that data assurance is not started in two instances by the user program.
-372	FE8CH	■Timeout value invalid The specified timeout value is out of range for the OS linking function.	Check the specified timeout value.
-373	FE8BH	■Timeout error A timeout occurred during inter OS shared semaphore acquisition.	Consider the synchronization, or the time required for processing by a target under exclusive control, and then set a sufficiently long timeout value.
-374	FE8AH	■Windows part forced restart not permitted There was an attempt by a C Controller module dedicated function to perform a Windows part forced restart when the "Windows part forced restart" setting in Basic parameters was "Disable".	Configure "Enable" for the "Windows part forced restart" setting in the Basic parameters.
-375	FE89H	■Shared memory access error Access of the shared memory of the OS linking function has failed.	<ul style="list-style-type: none"> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC.</li> <li>• Retry.</li> </ul>
-376	FE88H	■Inter OS event notification error Notification of an inter-OS event of the OS linking function has failed.	<ul style="list-style-type: none"> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC.</li> <li>• Retry.</li> </ul>
-377	FE87H	■Data assurance running by other task error Ending of data assurance was attempted when data assurance was being performed by another task.	Revise the user program so that the same task starts and ends data assurance.

# MELSEC data link function

The following table shows the error codes of MELSEC data link functions.

Error code <sup>*1</sup>		Description	Corrective action
Decimal	Hexadecimal		
-1	FFFFH	<b>■Path error</b> <ul style="list-style-type: none"> <li>The specified path is unavailable.</li> <li>The taskDelete was executed in the task using a MELSEC data link function.</li> <li>The task using a MELSEC data link function was deleted with the taskDelete.</li> </ul>	<ul style="list-style-type: none"> <li>Use a path pointer returned with the mdOpen function.</li> <li>Check if the taskDelete was executed in the task using a MELSEC data link function.</li> <li>Check if the task using a MELSEC data link function was deleted with the taskDelete.</li> </ul>
-2	FFFEH	<b>■Device number error</b> <ul style="list-style-type: none"> <li>The specified device number is out of range.</li> <li>The specified bit device number is not a multiple of 8.</li> <li>The device number and the points for the same block specified for reading/writing device randomly exceeds the device range.</li> </ul>	<ul style="list-style-type: none"> <li>Check the start device number of the specified device.</li> <li>Check the device number plus the number of points.</li> <li>Specify the start device number of bit device in multiples of 8.</li> <li>Check if the specified device is available in the CPU module on the target station.</li> </ul>
-3	FFFDH	<b>■Device type error</b> The specified device type is unavailable.	<ul style="list-style-type: none"> <li>Check the specified device type.</li> <li>Check if the specified device is available in the target station.</li> </ul>
-5	FFFBH	<b>■Size error</b> <ul style="list-style-type: none"> <li>The device number and the size exceeds the device range.</li> <li>The device number and the size exceeds the range for the same block.</li> <li>The access was made with an odd-number bytes.</li> <li>The total points that are specified for each block number of the mdRandREx/mdRandWEx function exceeds 10,240.</li> </ul>	<ul style="list-style-type: none"> <li>Check the specified device size.</li> <li>Check the device number and the size.</li> <li>Specify an even-number byte.</li> <li>Reduce the total points that are specified for each block number of the mdRandREx/mdRandWEx function to 10,240 or less.</li> </ul>
-6	FFFAH	<b>■Number of blocks error</b> The number of blocks specified to the function for reading/writing device randomly is out of range.	Check the number of the specified blocks.
-8	FFF8H	<b>■Channel number error</b> The channel number specified with the mdOpen function is unavailable.	Check the specified channel number.
-11	FFF5H	<b>■Insufficient buffer area error</b> The area size of the read data storage destination is smaller than the read data size.	Check the area size of the read data storage destination and the read data size.
-12	FFF4H	<b>■Block number error</b> The specified block number is unavailable.	<ul style="list-style-type: none"> <li>Check the block number (device type) of the specified device.</li> <li>Check if the specified device and block number are available in the target.</li> </ul>
-13	FFF3H	<b>■Write protect error</b> The block number of the specified extended file register overlaps with the write-protected area of the memory cassette.	<ul style="list-style-type: none"> <li>Check the block number (device type) of the extension file register.</li> <li>Check the write protection switch of the memory cassette.</li> </ul>
-16	FFF0H	<b>■Station number/network number error</b> <ul style="list-style-type: none"> <li>The specified station number or network number is out of range.</li> <li>A device which cannot be accessed by the target station is specified.</li> </ul>	<ul style="list-style-type: none"> <li>Check the specified station number and network number.</li> <li>Check the devices which can be accessed by the target station.</li> </ul>
-17	FFEFH	<b>■All stations/group number specification error</b> A function which does not support specifying all stations and group number was specified.	<ul style="list-style-type: none"> <li>Check if the function allows specifying all stations and group number.</li> <li>When "All stations" or "Group number" is specified to the station number, specify "Without arrival confirmation" to the device type.</li> </ul>
-18	FFEEH	<b>■Remote operation error</b> The specification code specified with the mdControl function is unavailable.	Check the specified specification code.
-31	FFE1H	<b>■Module load error</b> Loading modules required for executing functions failed.	<ul style="list-style-type: none"> <li>The memory may be insufficient. End other running tasks. Alternatively, reduce the excess size.</li> <li>Check the status of the system drive of the MELIPC.</li> </ul>
-32	FFE0H	<b>■Resource timeout error</b> The resource is being used by another task/thread and is not released within 30 seconds.	<ul style="list-style-type: none"> <li>Retry.</li> <li>The memory may be insufficient. End other running tasks.</li> <li>Check if the MELIPC is running normally.</li> <li>Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
-33	FFDFH	<b>■Communication target unsupported error</b> The module specified as the communication target by the network number and station number is not supported.	<ul style="list-style-type: none"> <li>Check if the module specified as the communication target by a network number and station number is supported.</li> <li>Check the settings of the access target set in MI Configurator.</li> </ul>

Error code*1		Description	Corrective action
Decimal	Hexadecimal		
-34	FFDEH	■Registry open error Opening parameter files in the registry failed.	Check if the access target is correctly set with MI Configurator.
-35	FFDDH	■Registry read error Reading parameter files from the registry failed.	<ul style="list-style-type: none"> <li>• Check if the access target is correctly set with MI Configurator.</li> <li>• Check if the setting for the channel number is enabled.</li> <li>• After checking the parameters with MI Configurator again and writing them, reset the MELIPC or turn the power supply OFF to ON.</li> </ul>
-36	FFDCH	■Registry write error Writing parameter files to the registry failed.	<ul style="list-style-type: none"> <li>• Check if the standard ROM has already been shutdown.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
-37	FFDBH	■Communications initialization error Initializing the setting for communication failed.	<ul style="list-style-type: none"> <li>• Retry.</li> <li>• The memory may be insufficient. End other running tasks.</li> <li>• Check the available memory capacity.</li> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
-42	FFD6H	■Close error Communications cannot be closed.	<ul style="list-style-type: none"> <li>• Retry.</li> <li>• Check if the MELIPC is running normally.</li> <li>• Reset the MELIPC, or turn the power OFF and ON.</li> </ul>
-43	FFD5H	■ROM operation error A TC setting value was written to the CPU module during ROM operation.	Change the TC setting value during RAM operation.
-52	FFCCH	■MELSEC data link function service error MELSEC data link function service is disabled.	Enable the MELSEC data link function service using MI Configurator.
-53	FFCBH	■Timeout value error The specified timeout value is out of range.	Check the specified time out value.
-54	FFCAH	■I/O number error The specified I/O number is out of range.	Check the specified I/O number.
-55	FFC9H	■Logical station number error The specified logical station number is out of range.	Check the specified logical station number.
-56	FFC8H	■Target CPU error The specified target CPU is out of range.	Check the specified target CPU.
-71	FFB9H	■Remote device station access error Failed to access the buffer memory of the remote device station.	<ul style="list-style-type: none"> <li>• Check whether the target station is the remote device station of the CC-Link IE Field Network.</li> <li>• Check the network number and station number.</li> <li>• Check whether the offset or the offset plus the read/write byte size is within range of the buffer memory of the target station.</li> </ul>
-80	FFB0H	■Connection destination CPU error The connection destination CPU is not an RCP.	Connect an RCP.
-4097 to -8192	EFFFH to E000H	Refer to the following manuals.  MELSEC iQ-R CC-Link IE Controller Network User's Manual (Application)  MELSEC-Q CC-Link IE Controller Network Reference Manual	
-8193 to -12288	DFFFH to D000H	Refer to the following manuals.  MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)  MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual  MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual	
-16385 to -20480	BFFFH to B000H	Refer to the following manuals.  MELSEC iQ-R CC-Link System Master/Local Module User's Manual (Application)  MELSEC-Q CC-Link System Master/Local Module User's Manual  MELSEC-L CC-Link System Master/Local Module User's Manual	

\*1 When the function of which the return value is a long-type, the value will be eight digits in hexadecimal.

# INDEX

---

## C

---

CC-Link IE Field Network communication . . . . .	11
Channel . . . . .	16
Communications addressed to own station . . . . .	10

## D

---

Device type . . . . .	8,17
-----------------------	------

# MEMO

---

# FUNCTION INDEX

---

## C

---

CCPU_ChangeCCIEFBCycPrm . . . . .	24
CCPU_ClearError . . . . .	26
CCPU_Control . . . . .	27
CCPU_EndCCIEFBDataAssurance . . . . .	28
CCPU_EndLinkDeviceAssurance . . . . .	29
CCPU_EntryCCIEFBLinkScanEndFunc . . . . .	30
CCPU_EntryLinkScanEndFunc . . . . .	31
CCPU_EntryShutdownFunc . . . . .	32
CCPU_EntryTimerEvent . . . . .	33
CCPU_EntryWDTInt . . . . .	35
CCPU_EntryWinResetFunc . . . . .	36
CCPU_FromBuf . . . . .	37
CCPU_FromBuf_ISR . . . . .	82
CCPU_GetCCIEFBDiagnosticInfo . . . . .	38
CCPU_GetCounterMicros . . . . .	40
CCPU_GetCounterMicros_ISR . . . . .	83
CCPU_GetCounterMillis . . . . .	41
CCPU_GetCounterMillis_ISR . . . . .	84
CCPU_GetCpuStatus . . . . .	42
CCPU_GetDotMatrixLED . . . . .	43
CCPU_GetDotMatrixLED_ISR . . . . .	85
CCPU_GetErrInfo . . . . .	45
CCPU_GetLEDStatus . . . . .	46
CCPU_GetPowerStatus . . . . .	48
CCPU_GetRTC . . . . .	49
CCPU_GetSerialNo . . . . .	50
CCPU_GetSwitchStatus . . . . .	51
CCPU_MELIPCSShutdown . . . . .	52
CCPU_ReadDevice . . . . .	53
CCPU_ReadDevice_ISR . . . . .	87
CCPU_ReadLinkDevice . . . . .	54
CCPU_ReadLinkDevice_ISR . . . . .	88
CCPU_ReadSharedMemory . . . . .	55
CCPU_ReceiveInterOSEvent . . . . .	56
CCPU_RegistEventLog . . . . .	57
CCPU_RegistEventLog_ISR . . . . .	89
CCPU_Reset . . . . .	58
CCPU_ResetDevice . . . . .	59
CCPU_ResetDevice_ISR . . . . .	90
CCPU_ResetWDT . . . . .	60
CCPU_RestoreDefaultCCIEFBCycPrm . . . . .	61
CCPU_SendInterOSEvent . . . . .	62
CCPU_SetDevice . . . . .	63
CCPU_SetDevice_ISR . . . . .	91
CCPU_SetDotMatrixLED . . . . .	64
CCPU_SetDotMatrixLED_ISR . . . . .	92
CCPU_SetLEDStatus . . . . .	66
CCPU_SetLEDStatus_ISR . . . . .	95
CCPU_SetRTC . . . . .	67
CCPU_ShutdownRom . . . . .	68
CCPU_StartCCIEFBDataAssurance . . . . .	69
CCPU_StartLinkDeviceAssurance . . . . .	70
CCPU_StartWDT . . . . .	71
CCPU_StopWDT . . . . .	72
CCPU_SysClkRateGet . . . . .	73
CCPU_SysClkRateSet . . . . .	74
CCPU_ToBuf . . . . .	75
CCPU_ToBuf_ISR . . . . .	96
CCPU_WaitSwitchEvent . . . . .	76
CCPU_WaitTimerEvent . . . . .	77

CCPU_WinReset . . . . .	78
CCPU_WriteDevice . . . . .	79
CCPU_WriteDevice_ISR . . . . .	97
CCPU_WriteLinkDevice . . . . .	80
CCPU_WriteLinkDevice_ISR . . . . .	98
CCPU_WriteSharedMemory . . . . .	81

## M

---

mdClose . . . . .	99
mdControl . . . . .	100
mdDevRstEx . . . . .	101
mdDevSetEx . . . . .	102
mdOpen . . . . .	103
mdRandREx . . . . .	104
mdRandWEx . . . . .	107
mdReceiveEx . . . . .	109
mdRemBufReadEx . . . . .	110
mdRemBufWriteEx . . . . .	111
mdSendEx . . . . .	112
mdTypeRead . . . . .	113



# MEMO

---

# REVISIONS

---

\*The manual number is given on the bottom left of the back cover.

Revision date	*Manual number	Description
May 2018	SH(NA)-081936ENG-A	First edition
May 2018	SH(NA)-081936ENG-B	Partial correction

Japanese manual number: SH-081935-C

---

This manual confers no industrial property rights of any other kind, nor does it confer any patent licenses. Mitsubishi Electric Corporation cannot be held responsible for any problems involving industrial property rights which may occur as a result of using the contents noted in this manual.

---

© 2018 MITSUBISHI ELECTRIC CORPORATION

# WARRANTY

---

Please confirm the following product warranty details before using this product.

## **1. Gratis Warranty Term and Gratis Warranty Range**

If any faults or defects (hereinafter "Failure") found to be the responsibility of Mitsubishi occurs during use of the product within the gratis warranty term, the product shall be repaired at no cost via the sales representative or Mitsubishi Service Company.

However, if repairs are required onsite at domestic or overseas location, expenses to send an engineer will be solely at the customer's discretion. Mitsubishi shall not be held responsible for any re-commissioning, maintenance, or testing on-site that involves replacement of the failed module.

[Gratis Warranty Term]

The gratis warranty term of the product shall be for one year after the date of purchase or delivery to a designated place. Note that after manufacture and shipment from Mitsubishi, the maximum distribution period shall be six (6) months, and the longest gratis warranty term after manufacturing shall be eighteen (18) months. The gratis warranty term of repair parts shall not exceed the gratis warranty term before repairs.

[Gratis Warranty Range]

- (1) The range shall be limited to normal use within the usage state, usage methods and usage environment, etc., which follow the conditions and precautions, etc., given in the instruction manual, user's manual and caution labels on the product.
- (2) Even within the gratis warranty term, repairs shall be charged for in the following cases.
  1. Failure occurring from inappropriate storage or handling, carelessness or negligence by the user. Failure caused by the user's hardware or software design.
  2. Failure caused by unapproved modifications, etc., to the product by the user.
  3. When the Mitsubishi product is assembled into a user's device, Failure that could have been avoided if functions or structures, judged as necessary in the legal safety measures the user's device is subject to or as necessary by industry standards, had been provided.
  4. Failure that could have been avoided if consumable parts (battery, backlight, fuse, etc.) designated in the instruction manual had been correctly serviced or replaced.
  5. Failure caused by external irresistible forces such as fires or abnormal voltages, and Failure caused by force majeure such as earthquakes, lightning, wind and water damage.
  6. Failure caused by reasons unpredictable by scientific technology standards at time of shipment from Mitsubishi.
  7. Any other failure found not to be the responsibility of Mitsubishi or that admitted not to be so by the user.

## **2. Onerous repair term after discontinuation of production**

- (1) Mitsubishi shall accept onerous product repairs for seven (7) years after production of the product is discontinued. Discontinuation of production shall be notified with Mitsubishi Technical Bulletins, etc.
- (2) Product supply (including repair parts) is not available after production is discontinued.

## **3. Overseas service**

Overseas, repairs shall be accepted by Mitsubishi's local overseas FA Center. Note that the repair conditions at each FA Center may differ.

## **4. Exclusion of loss in opportunity and secondary loss from warranty liability**

Regardless of the gratis warranty term, Mitsubishi shall not be liable for compensation to:

- (1) Damages caused by any cause found not to be the responsibility of Mitsubishi.
- (2) Loss in opportunity, lost profits incurred to the user by Failures of Mitsubishi products.
- (3) Special damages and secondary damages whether foreseeable or not, compensation for accidents, and compensation for damages to products other than Mitsubishi products.
- (4) Replacement by the user, maintenance of on-site equipment, start-up test run and other tasks.

## **5. Changes in product specifications**

The specifications given in the catalogs, manuals or technical documents are subject to change without prior notice.

# TRADEMARKS

---

Ethernet is a registered trademark of Fuji Xerox Co., Ltd. in Japan.

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

Unicode is either a registered trademark or a trademark of Unicode, Inc. in the United States and other countries.

The company names, system names and product names mentioned in this manual are either registered trademarks or trademarks of their respective companies.

In some cases, trademark symbols such as <sup>™</sup> or <sup>®</sup> are not specified in this manual.



SH(NA)-081936ENG-B(1805)  
MODEL:MELIPC-MI5000-P-VX-E

## **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

When exported from Japan, this manual does not require application to the  
Ministry of Economy, Trade and Industry for service transaction permission.

Specifications subject to change without notice.