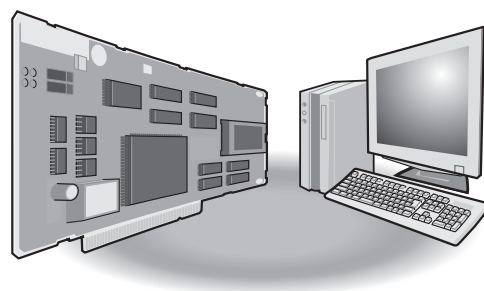


网络接口板

MELSEC

MELSEC数据链接库参考手册

-Q80BD-J61BT11N
-Q81BD-J61BT11
-Q80BD-J71LP21-25
-Q80BD-J71LP21S-25
-Q81BD-J71LP21-25
-Q80BD-J71LP21G
-Q80BD-J71BR11
-Q80BD-J71GP21-SX
-Q80BD-J71GP21S-SX
-Q81BD-J71GP21-SX
-Q81BD-J71GP21S-SX
-Q80BD-J71GF11-T2
-Q81BD-J71GF11-T2



安全注意事项

(使用之前请务必阅读)

在使用本产品之前，应仔细阅读本手册以及本手册中所介绍的关联手册，同时在充分注意安全的前提下正确地操作。请妥善保管本手册以备需要时阅读，并应将本手册交给最终用户。

关于产品的应用

- (1) 在使用三菱电机可编程控制器时，应该符合以下条件：即使在可编程控制器设备出现问题或故障时也不会导致重大事故，并且应在设备外部系统地配备能应付任何问题或故障的备用设备及失效安全功能。
- (2) 三菱电机可编程控制器是以一般工业用途等为对象设计和制造的通用产品。
因此，三菱电机可编程控制器不应用于以下设备・系统等特殊用途。如果用于以下特殊用途，对于三菱电机可编程控制器的质量、性能、安全等所有相关责任（包括但不限于债务未履行责任、瑕疵担保责任、质量保证责任、违法行为责任、制造物责任），三菱电机将不负责。
 - 面向各电力公司的核电站以及其它发电厂等对公众有较大影响的用途。
 - 用于各铁路公司或公用设施目的等有特殊质量保证体系要求的用途。
 - 航空航天、医疗、铁路、焚烧・燃料装置、载人移动设备、载人运输装置、娱乐设备、安全设备等预计对人身财产有较大影响的用途。然而，对于上述应用，如果在限于具体用途，无需特殊质量（超出一般规格的质量等）要求的条件下，经过三菱电机的判断也可以使用三菱电机可编程控制器，详细情况请与当地三菱电机代表机构协商。
- (3) 因拒绝服务攻击（DoS攻击）、非法访问、计算机病毒以及其他网络攻击引发的系统方面的各种问题，三菱电机不承担责任。

前言

在此感谢贵方购买了三菱电机网络接口板。

本手册是用于让用户了解使用MELSEC数据链接库时的必要编程步骤、函数规格有关内容的手册。

在使用之前应熟读本手册及关联手册，在充分了解MELSEC数据链接库的功能・性能的基础上正确地使用本产品。

应将本手册交给最终用户。

对象模块

项目	型号
CC-Link Ver. 2板	Q80BD-J61BT11N、Q81BD-J61BT11
MELSECNET/H板	Q80BD-J71LP21-25、Q81BD-J71LP21-25、Q80BD-J71LP21S-25、Q80BD-J71LP21G、Q80BD-J71LP21GE、Q80BD-J71BR11
CC-Link IE控制网络板	Q80BD-J71GP21-SX、Q80BD-J71GP21S-SX、Q81BD-J71GP21-SX、Q81BD-J71GP21S-SX
CC-Link IE现场网络板	Q80BD-J71GF11-T2、Q81BD-J71GF11-T2

目录

安全注意事项	1
关于产品的应用	1
前言	2
关联手册	6
术语	7
第1章 概要	8
第2章 编程	10
2.1 编程步骤	10
2.2 使用MELSEC数据链接库时的注意事项	11
编程时的注意事项	11
访问本站链接软元件及其它站可编程控制器CPU的软元件时的注意事项	13
2.3 用于使用函数的设置	14
使用Visual Basic 5.0、Visual Basic 6.0的情况下	14
使用Visual C++ 5.0、Visual C++ 6.0的情况下	15
使用Visual Basic .NET的情况下	17
使用Visual C++ .NET 2003、Visual C++ 2005、Visual C++ 2008的情况下	18
使用Visual Studio 2010、Visual Studio 2012的Visual C++的情况下	20
使用Visual Studio 2013、Visual Studio 2015及Visual Studio 2017的Visual C++的情况下	27
第3章 可访问软元件及范围	35
3.1 访问目标	35
3.2 可访问范围	36
本网络内的访问目标	36
经由了网络情况下的访问目标	37
3.3 可访问软元件	38
至本站的链接软元件、缓冲存储器的访问	38
至MELSEC iQ-R系列模块的软元件的访问	39
至MELSEC QnA、Q、L系列模块的软元件的访问	40
至MELSEC A系列模块的软元件的访问	41
至以太网适配器模块、起始模块、CC-Link IE现场网络远程设备站的访问	42
至CC-Link的其它站缓冲存储器的访问	42
通过SEND功能、RECV功能的访问	43
第4章 函数	44
4.1 函数一览	44
4.2 函数通用规格	46
通道No. 指定	46
站号指定	47
扩展函数的网络No. 及站号指定	49
软元件类型指定	51
4.3 函数详细内容	54
mdOpen (通信线路的打开)	55
mdClose (通信线路的关闭)	56
mdSendEx (扩展软元件批量写入)	57
mdSendEx (SEND功能)	59

mdReceiveEx(扩展软元件批量读取)	61
mdReceiveEx(RECV功能)	63
mdRandWEx(扩展软元件随机写入)	65
mdRandREx(扩展软元件随机读取)	68
mdDevSetEx(扩展位软元件设置)	71
mdDevRstEx(扩展位软元件复位)	72
mdRemBufWriteEx(远程设备站的缓冲存储器写入)	73
mdRemBufReadEx(远程设备站的缓冲存储器读取)	74
mdTypeRead(CPU型号读取)	75
mdControl(远程RUN/STOP/PAUSE)	79
mdWaitBdEvent(事件等待)	80
mdBdRst(板复位)	83
mdBdModSet(板模式设置)	84
mdBdModRead(板模式读取)	86
mdBdLedRead(板LED读取)	87
mdBdSwRead(板开关状态读取)	93
mdBdVerRead(板版本读取)	94
mdInit(可编程控制器软元件信息表的初始化)	96
mdSend(软元件批量写入)	97
mdSend(SEND功能)	99
mdReceive(软元件批量读取)	101
mdReceive(RECV功能)	103
mdRandW(软元件随机写入)	105
mdRandR(软元件随机读取)	107
mdDevSet(位软元件设置)	108
mdDevRst(位软元件复位)	109

第5章 样本程序 110

5.1 使用Visual Basic .NET的情况下	111
MTEST及MTEST2	111
5.2 使用Visual Basic 5.0、Visual Basic 6.0的情况下	112
5.3 使用Visual C++的情况下	112

第6章 软元件监视实用程序 113

6.1 操作流程	114
6.2 菜单一览	115
6.3 画面配置及基本操作	116
启动及结束	116
总体画面配置及基本操作	117
帮助	118
6.4 选择监视方法	119
批量监视	119
16点登录监视	119
6.5 设置监视目标	120
6.6 设置监视的软元件	121
6.7 更改当前值	122
更改字软元件/双字软元件值	122
将位软元件置为ON、OFF	124
6.8 切换显示格式	125
6.9 通过数值输入板输入	126

第7章 出错代码	127
附录	132
附1 扩大计算机的最小工作集区域的方法	132
样本程序的处理概要	132
样本程序	133
函数索引	136
修订记录	138
质保	139
商标	140

关联手册

手册名称[手册编号]	内容	提供形态
MELSEC数据链接库参考手册 [SH-082183CHN] (本手册)	介绍MELSEC数据链接库的编程、可访问软元件•范围、函数、样本程序及出错代码有关内容。	#### e-Manual PDF
Q80BD-J61BT11N/Q81BD-J61BT11型CC-Link系统主站/ 本地站接口板用户手册 (SW1DNC-CCBD2-B系列) [SH-080527ENG]	记载了Q80BD-J61BT11N/Q81BD-J61BT11型CC-Link系统主站•本地站接口板的系统配置、规格、功能、处理、配线及故障排除有关内容。	#### PDF
MELSECNET/H接口板用户手册 (SW0DNC-MNETH-B系列) [SH-080128]	记载了MELSECNET/H接口板的系统配置、规格、功能、处理、配线及故障排除有关内容。	#### PDF
CC-Link IE控制网络接口板用户手册 (SW1DNC-MNETG-B系列) [SH-082181CHN]	记载了CC-Link IE控制网络接口板的系统配置、规格、功能、处理、配线及故障排除有关内容。	#### PDF
CC-Link IE现场网络接口板用户手册 (对应SW1DNC-CCIEF-B) [SH-080980ENG]	记载了CC-Link IE现场网络接口板的系统配置、规格、功能、处理、配线及故障排除有关内容。	#### e-Manual PDF

要点

e-Manual是可以使用专用工具进行浏览的三菱电机FA电子书籍手册。

e-Manual具有以下特点。

- 可以从多本手册同时搜索需要的信息 (跨手册搜索)
- 可以通过手册内的链接浏览其他手册
- 可以通过产品插图的各部分浏览想要了解的硬件规格
- 可以将频繁浏览的信息登录到收藏夹

术语

本手册中除了特别标明的情况外，将使用下述术语进行说明。

术语	内容
CC-Link IE控制网络板	是Q80BD-J71GP21-SX、Q80BD-J71GP21S-SX、Q81BD-J71GP21-SX、Q81BD-J71GP21S-SX型CC-Link IE控制网络接口板的总称。
CC-Link IE现场网络板	是Q80BD-J71GF11-T2、Q81BD-J71GF11-T2型CC-Link IE现场网络接口板的总称。
CC-Link Ver. 2板	是Q80BD-J61BT11N、Q81BD-J61BT11型CC-Link系统主站・本地站接口板的总称。
MELSECNET/H板	是Q80BD-J71LP21-25、Q81BD-J71LP21-25、Q80BD-J71LP21S-25、Q80BD-J71LP21G、Q80BD-J71LP21GE、Q80BD-J71BR11型MELSECNET/H接口板的总称。
板	是MELSEC数据链接库支持的计算机用接口板的总称。
实用程序	是MELSEC数据链接库支持的计算机用接口板附带的实用程序的总称。

1 概要

以下介绍MELSEC数据链接库的概要。

关于MELSEC数据链接库

MELSEC数据链接库是指用于进行至板的本站链接软元件的访问，以及使用了板的其它站可编程控制器CPU的软元件存储器的访问的库。

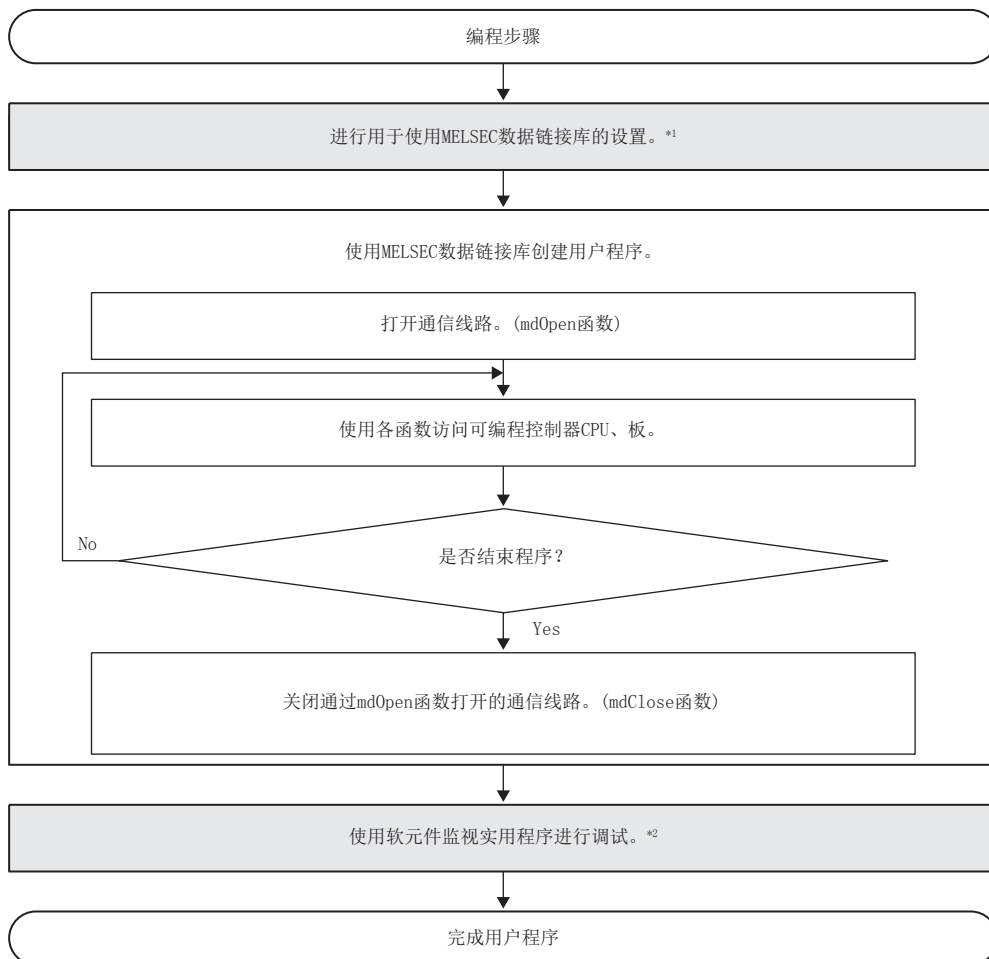
通过使用MELSEC数据链接库，可以无需理会通信函数轻松创建进行访问的程序。

2 编程

以下介绍MELSEC数据链接库的使用方法。

2.1 编程步骤

在安装了软件包的计算机中，使用MELSEC数据链接库，创建用户程序的步骤如下所示。



*1 14页 用于使用函数的设置
*2 113页 软件监视实用程序

2.2 使用MELSEC数据链接库时的注意事项

编程时的注意事项

头文件

若要进行C或C++的编程，则应对Windows头文件（windows.h）进行包含（include）。

多线程通信

不能通过同一过程内的多个线程使用MELSEC数据链接库。
应通过单线程使用MELSEC数据链接库。

通信线路的打开/关闭

通信线路的打开/关闭(mdOpen/mdClose)处理应仅在程序的最初及最后进行1次。
如果在每次通信中重复进行打开/关闭，通信性能将变差。

其它站访问的站数

通过用户程序进行其它站访问的情况下，总访问站数应设置为256站及以下。
将总访问站数设置为257站及以上时通信性能将变差。

初次函数执行时间

MELSEC数据链接库初次访问可编程控制器CPU时，获取可编程控制器的软元件的信息。
因此，初次函数执行时间将变长。

用户程序的强制结束

强制结束正在执行MELSEC数据链接库的用户程序的情况下，有可能发生下述现象。

- 无法结束试图强制结束的应用程序。
- 其它应用程序中MELSEC数据链接库发生出错。
- 对三菱电机的软件包(MX Component、GX Works2等)带来影响。

执行时间

受Windows®的处理以及其它应用程序等的影响，MELSEC数据链接库函数的执行时间及执行间隔有可能暂时延长。
创建程序时应考虑这些因素。

static型的变量

MELSEC数据链接库的各函数的输出参数中请勿指定通过static进行了声明的变量。

服务应用程序

不能通过Windows服务应用程序使用MELSEC数据链接库。
应通过用户应用程序使用MELSEC数据链接库。

板复位

通过用户程序执行mdBdRst函数或mdBdModSet函数的情况下，在函数返回了返回值的时刻完成通过各函数进行的板复位处理。
应创建必须确认函数的返回值的程序。

64位版用户程序

■至QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块的访问

无法通过64位版用户程序访问QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块。

应使用32位版用户程序。

■64位版用户程序的创建方法

创建64位版用户程序的情况下，需要配置64位平台为对象平台的工程。

关于配置工程并设置对象平台的方法，请参阅所使用的Visual Studio®的帮助(“方法：配置工程并设置对象平台”)。

■创建Visual Basic的64位版用户程序的情况下

需要.NET Framework 4.0或与.NET Framework 4.0兼容的.NET Framework。

应使用Visual Studio 2010及以上。

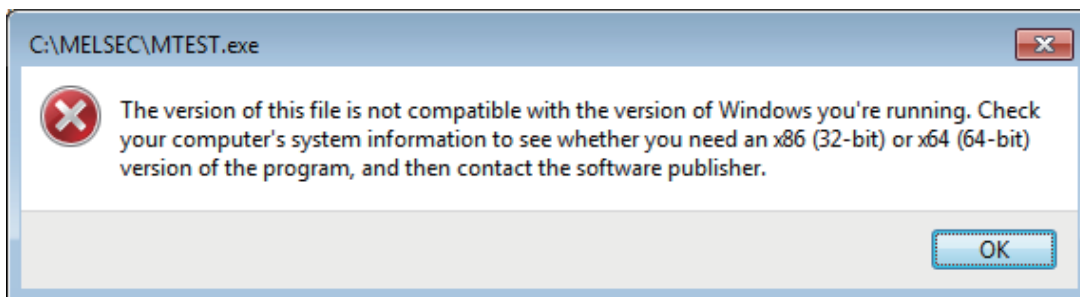
■创建64位版用户程序时的限制事项

在32位版OS上也可创建64位版用户程序。

但是，在32位版OS上将显示下述画面，无法执行。

例

<使用Windows 7(32位版)时>



/SAFESEH(包括安全例外处理程序的图像)选项

进行链接器选项的设置的情况下，请勿使用/SAFESEH(包括安全例外处理程序的图像)选项。

使用的情况下，将无法正确编译。

OS及其它应用程序的影响

由于OS及其它应用程序的更新程序的自动启动、通过其它应用程序进行软元件访问的影响导致OS的系统资源不足时，执行MELSEC数据链接库函数时有可能发生“板驱动器I/F出错102(0066H)”。

应根据需要进行下述处理。

- MELSEC数据链接库函数的重试处理
- OS及其它应用程序的自动更新的停止
- 其它应用程序的停止

循环数据保证时的软元件访问

进行软元件访问时的循环数据保证(32位数据保证及站单位块保证)的情况下，应使用批量写入/批量读取(mdSendEx/mdSend/mdReceiveEx/mdReceive)。

通过随机写入/随机读取(mdRandWEx/mdRandW/mdRandREx/mdRandR)进行软元件访问时，将不进行循环数据保证(32位数据保证及站单位块保证)。

访问本站链接软元件及其它站可编程控制器CPU的软元件时的注意事项

需要根据本站及其它站的链接状态采取互锁。
只有下述条件成立的情况下才启用数据。

MELSECNET/H的情况下

■至本站链接软元件(LX、LY、LB、LW)的访问

只有本站令牌传递状态(SB47)、本站数据链接状态(SB49)的位为OFF(正常),本站模块状态(SB20)的位为OFF(正常)的情况下,才启用至本站链接软元件的数据写入/读取。

但是,即使上述条件不成立的情况下对MELSECNET/H板的写入/读取处理也将正常结束。

■其它站瞬时访问(其它站可编程控制器CPU远程操作及软元件访问)

除勾选了链接软元件访问的软元件之外,访问站的令牌传递状态的位(从本站读取的SW70~73的通信对象站相应的位)为OFF(正常)及循环状态的位(从本站读取的SW74~77的通信对象站相应的位)为OFF(循环传送中)的情况下也可以访问。

CC-Link IE控制网络的情况下

■至本站链接软元件(LX、LY、LB、LW)的访问

只有本站令牌传递状态(SB47)、本站数据链接状态(SB49)的位为OFF(正常),本站模块状态(SB20)的位为OFF(正常)的情况下,才启用至本站链接软元件的数据写入/读取。

但是,即使上述条件不成立的情况下对CC-Link IE控制网络板的写入/读取处理也将正常结束。

■其它站瞬时访问(其它站可编程控制器CPU远程操作及软元件访问)

除勾选了链接软元件访问的软元件之外,访问站的令牌传递状态(从本站读取的SWA0~A7的通信对象站相应的位)及数据链接状态(从本站读取的SWB0~B7的通信对象站相应的位)为OFF(正常)的情况下也可以访问。

CC-Link IE现场网络的情况下

■至本站链接软元件(RX、RY、RW)的访问

只有本站令牌传递状态(SB47)、本站数据链接状态(SB49)的位为OFF(正常)的情况下,才启用至本站链接软元件的数据写入/读取。

但是,即使上述条件不成立的情况下对CC-Link IE现场网络板的写入/读取处理也将正常结束。

■其它站瞬时访问(其它站可编程控制器CPU远程操作及软元件访问)

除勾选了链接软元件访问的软元件之外,访问站的令牌传递状态(从本站读取的SWA0~A7的通信对象站相应的位)及数据链接状态(从本站读取的SWB0~B7的通信对象站相应的位)为OFF(正常)的情况下也可以访问。

2.3 用于使用函数的设置

以下介绍用于使用MELSEC数据链接库的函数的开发工具的设置方法有关内容。
关于各板支持的编程语言，请参阅各板的手册。

要点

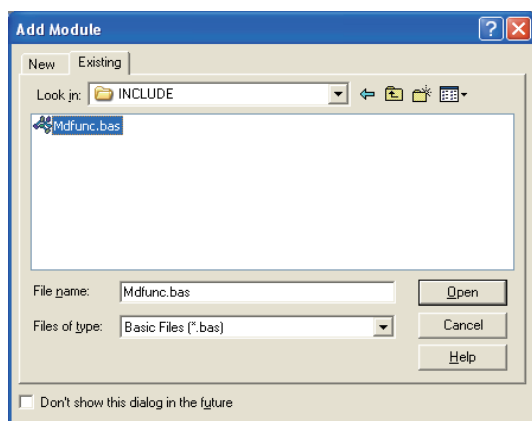
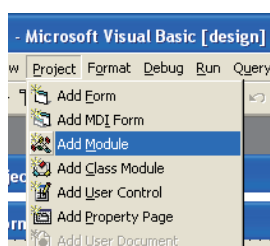
创建64位版用户应用程序的情况下

- 需要配置64位平台为对象平台的工程。
- 关于配置工程并设置对象平台的方法，请参阅所使用的Visual Studio的帮助(“方法：配置工程并设置对象平台”)。

使用Visual Basic，创建64位版用户程序的情况下

- 需要.NET Framework 4.0及与.NET Framework 4.0兼容的.NET Framework。
- 开发环境应使用Visual Studio 2010及以上。

使用Visual Basic 5.0、Visual Basic 6.0的情况下



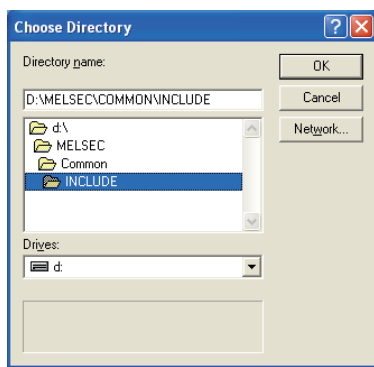
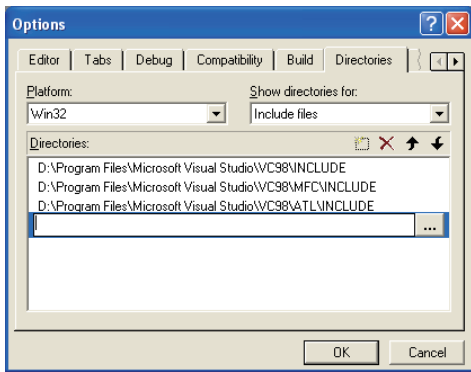
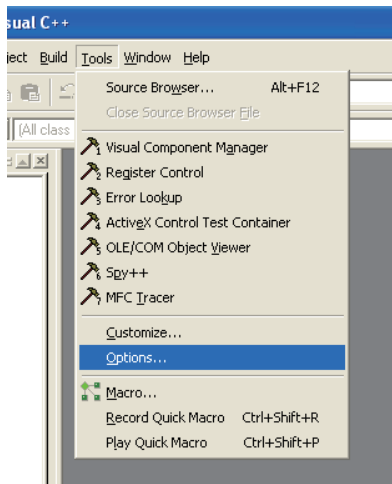
1. 启动Visual Basic®选择[Project(工程)]⇒[Add Module(添加标准模块)]菜单。

2. 选择[Existing(已存在的文件)]标签，选择“Mdfunc.bas”。

- “Mdfunc.bas”存储在安装的文件夹的“COMMON\INCLUDE”中。

使用Visual C++ 5.0、Visual C++ 6.0的情况下

进行包含文件设置的情况下



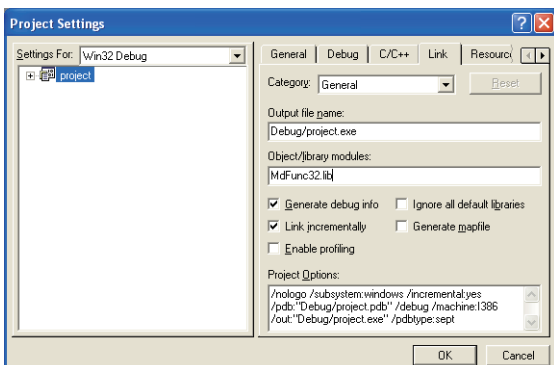
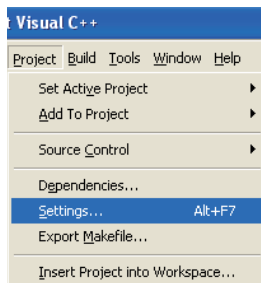
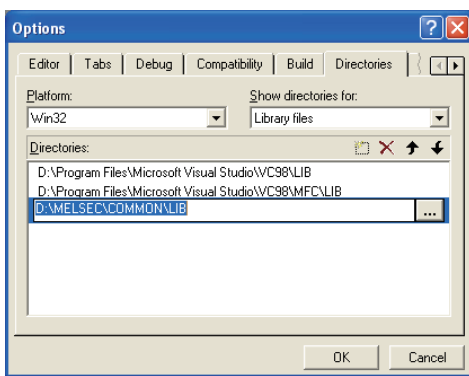
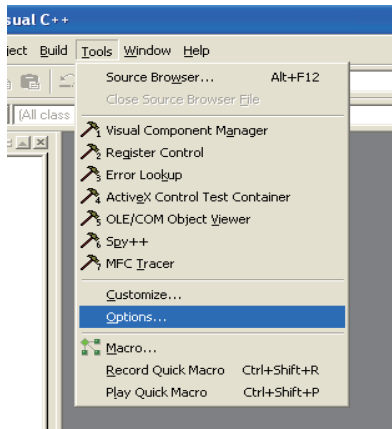
1. 启动Visual C++[®]，选择[Tools(工具)]⇒[Options(选项)]菜单。

2. 选择[Directories(目录)]标签，将“Show directories for(显示的目录)”设置为“Include files(包含文件)”。

3. 双击设置的项目，浏览存储了包含文件的文件夹。
• “Mdfunc.h”存储在安装的文件夹的“COMMON\INCLUDE”中。

4. 在程序的起始处，添加#include<Mdfunc.h>。

进行库文件设置的情况下



1. 启动Visual C++, 选择[Tools(工具)]⇒[Options(选项)]菜单。

2. 选择[Directories(目录)]标签, 将“Show directories for(显示的目录)”设置为“Library files(库文件)”。

3. 双击设置的项目, 浏览存储了库文件的文件夹。

- “MdFunc32.lib”存储在安装的文件夹的“COMMON\LIB”中。

4. 打开创建的工作区, 选择[Project(工程)]⇒[Settings(设置)]菜单。

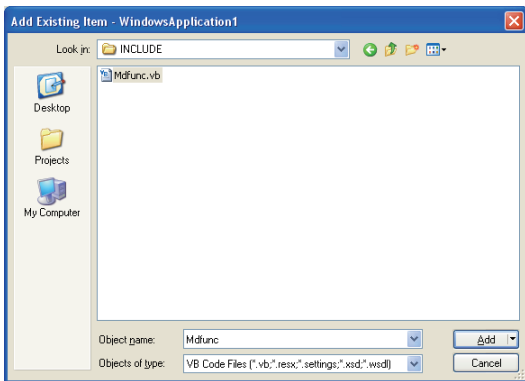
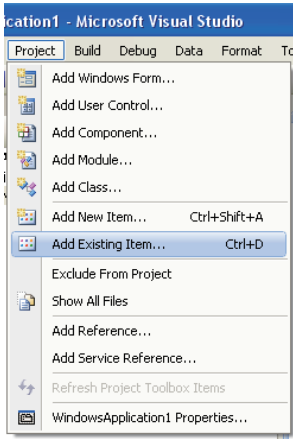
5. 选择[Link(链接)]标签, 在类别中选择“General(常规)”, 在对象/库模块中输入“MdFunc32.lib”。

使用Visual Basic .NET的情况下

使用Visual Studio及Visual Studio .NET 2003的Visual Basic情况下的设置操作如下所示。

要点

本项的说明使用Visual Studio 2008 Visual Basic的画面。
其它的Visual Basic的情况下，画面的显示内容有所不同。



1. 启动Visual Basic，选择[Add Existing Item(添加已存在的项目)]菜单。
 - Visual Studio .NET 2003以外的情况下：[Project(工程)]⇒[Add Existing Item(添加已存在的项目)]
 - Visual Studio .NET 2003的情况下：[File(文件)]⇒[Add Existing Item(添加已存在的项目)]

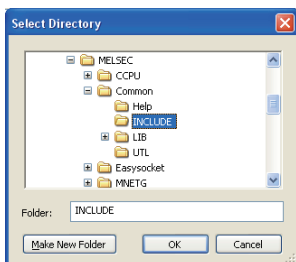
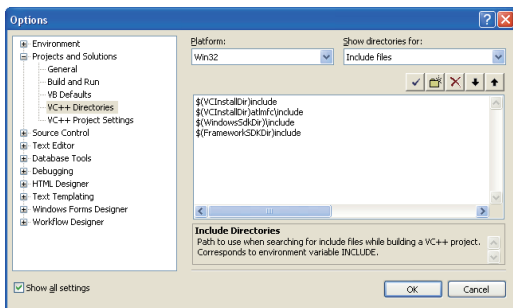
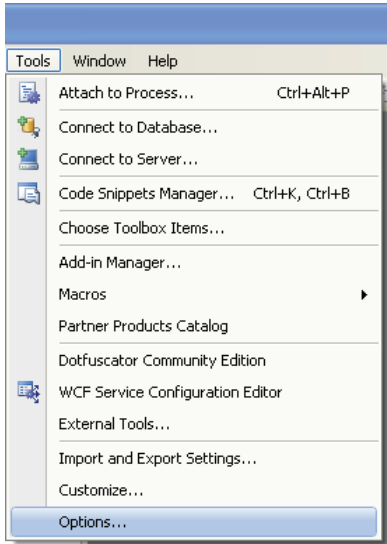
2. 在“Add Existing Item(添加已存在的项目)”画面中，选择“Mdfunc.vb”。
 - “Mdfunc.vb”存储在安装的文件夹的“COMMON\INCLUDE”中。

使用Visual C++ .NET 2003、Visual C++ 2005、Visual C++ 2008的情况下

要点

本项的说明使用Visual Studio 2008 Visual C++的画面。
其它的Visual C++的情况下，画面的显示内容有所不同。

进行包含文件设置的情况下




1. 启动Visual C++，选择[Tools(工具)]⇒[Options(选项)]菜单。

2. 在“Options(选项)”画面的文件夹栏中，选择“VC++ Directories(VC++目录)”。

• Visual C++ .NET 2003以外的情况下：“Projects and Solutions(工程及解决方案)”⇒“VC++ Directories(VC++目录)”

• Visual C++ .NET 2003的情况下：“Projects(工程)”⇒“VC++ Directories(VC++目录)”

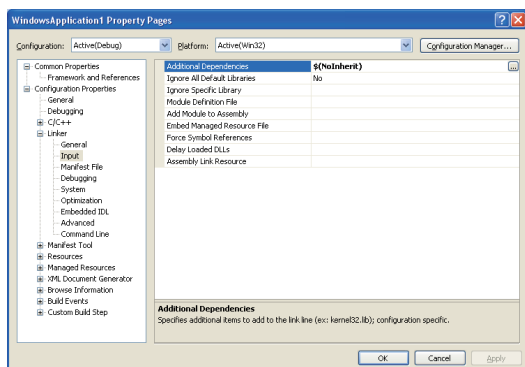
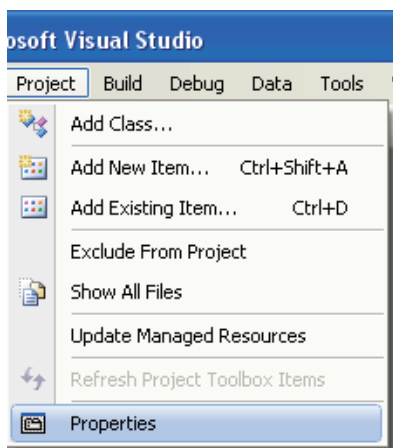
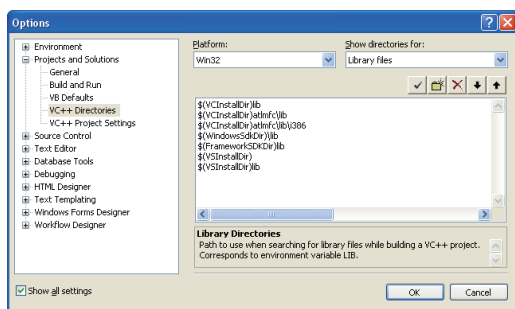
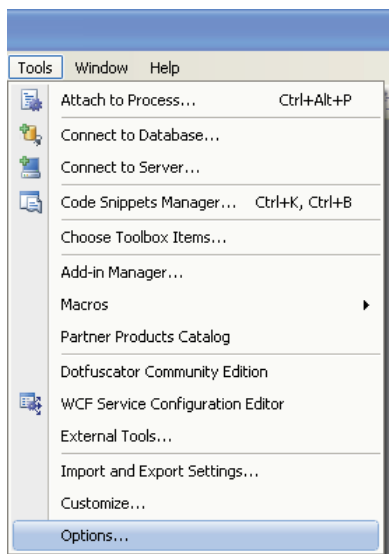
3. 将“Show directories for(显示目录的工程)”设置为“Include files(包含文件)”， 点击按钮。

4. 浏览存储了包含文件的文件夹。

• “Mdfunc.h”存储在安装的文件夹的“COMMON\INCLUDE”中。

5. 在程序的起始处，添加#include<Mdfunc.h>。


进行库文件设置的情况下



1. 启动Visual C++，选择[Tools(工具)]⇒[Options(选项)]菜单。

2. 在“Options(选项)”画面的文件夹栏中，选择“VC++ Directories(VC++目录)”。

- Visual C++ .NET 2003以外的情况下：“Projects and Solutions(工程及解决方案)”⇒“VC++ Directories(VC++目录)”
- Visual C++ .NET 2003的情况下：“Projects(工程)”⇒“VC++ Directories(VC++目录)”

3. 将“Show directories for(显示目录的工程)”设置为“Library files(库文件)”， 点击按钮。

4. 浏览存储了库文件的文件夹。

- 32位版用户应用程序创建用的情况下：“MdFunc32.lib”存储在安装的文件夹的“COMMON\LIB”中。
- 64位版用户应用程序创建用的情况下：“MdFunc32.lib”存储在安装的文件夹的“COMMON\LIB\x64”中。

5. 打开创建的工程，选择[Project(工程)]⇒[Properties(属性)]菜单。

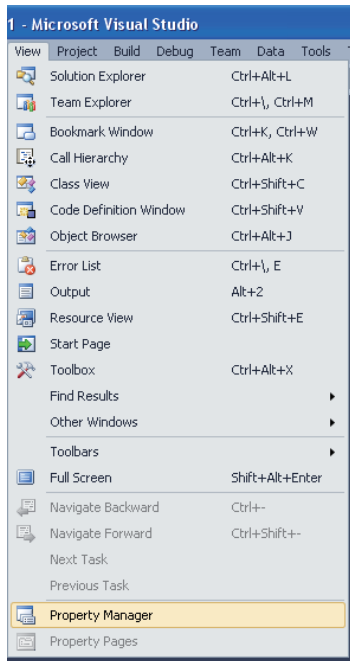
6. 在“Property Pages(属性页面)”画面的文件夹栏中，选择“Configuration Properties(配置属性)”⇒“Linker(链接器)”⇒“Input(输入)”，在“Additional Dependencies(添加的依存文件)”中记载“MdFunc32.lib”。

使用Visual Studio 2010、Visual Studio 2012的Visual C++的情况下

要点

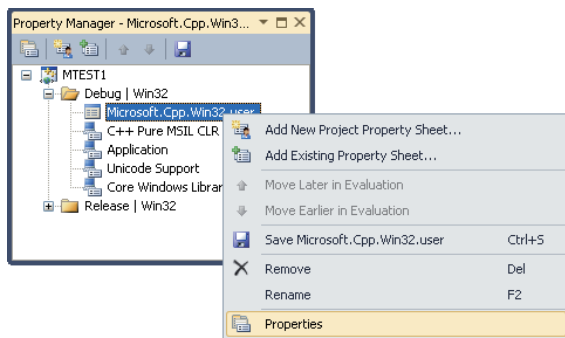
本项的说明使用Visual Studio 2010 Visual C++的画面。
其它的Visual C++的情况下，画面的显示内容有所不同。

进行包含文件设置的情况下



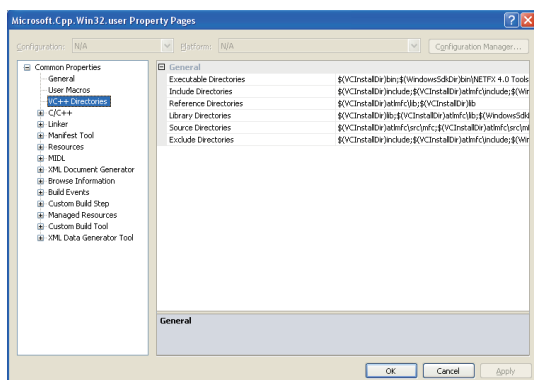
1. 打开创建的工程，选择[View(显示)]⇒[Property Manager (属性管理器)]*1菜单。

*1 根据开发设置及自定义，菜单的配置有可能不同。

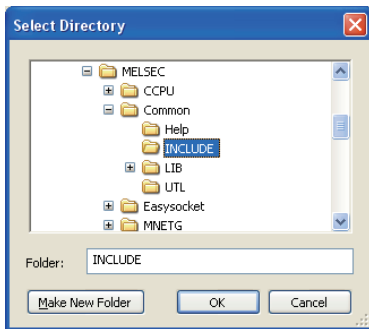
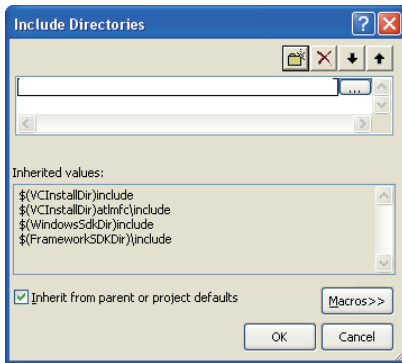
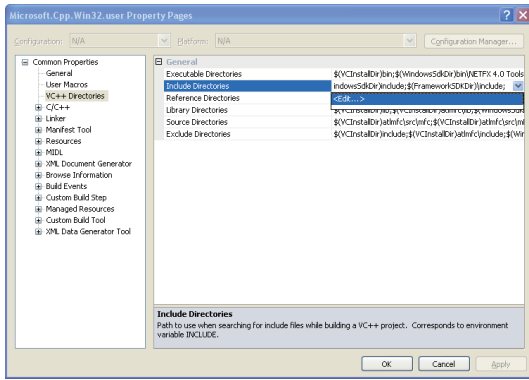



2. 右击用户属性表，选择快捷菜单的[Properties (属性)]。



将显示“Property Pages (属性页面)”画面。



3. 在“Property Pages (属性页面)”画面中，选择“Common Properties (通用属性)”⇒“VC++ Directories (VC++目录)”。



4. 选择“Include Directories(包含目录)”。 点击按钮，选择“<Edit... (编辑...)>”。

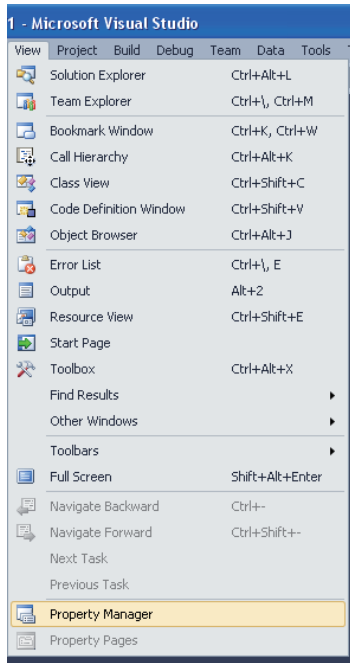
5. 在“Include Directories(包含目录)”画面中， 点击按钮，接着  点击。

6. 在“Select Directory(目录的选择)”画面中，浏览存储了包含文件的文件夹。

- “Mdfunc.h” 存储在安装的文件夹的“COMMON\INCLUDE”中。

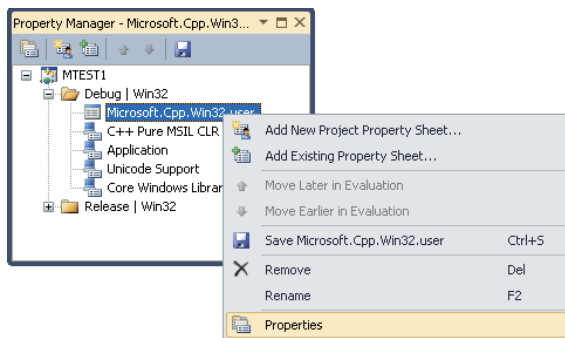
7. 在程序的起始处，添加#include<Mdfunc.h>。

进行库文件设置的情况下



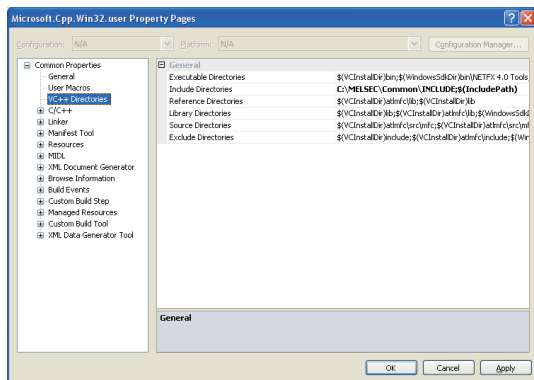
1. 打开创建的工程，选择[View(显示)]⇒[Property Manager(属性管理器)]*1菜单。

*1 根据开发设置及自定义，菜单的配置有可能不同。

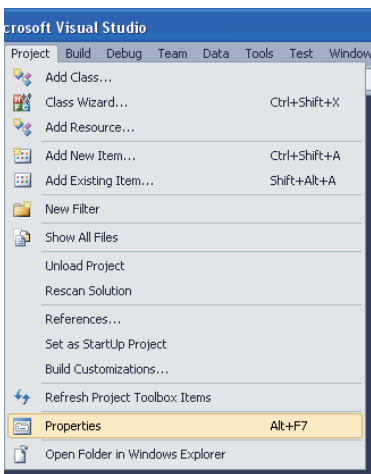
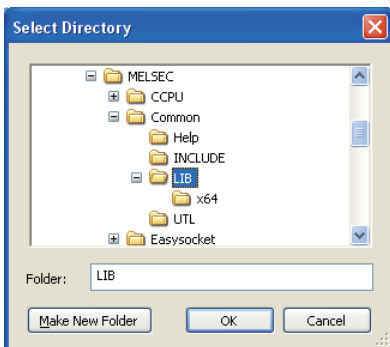
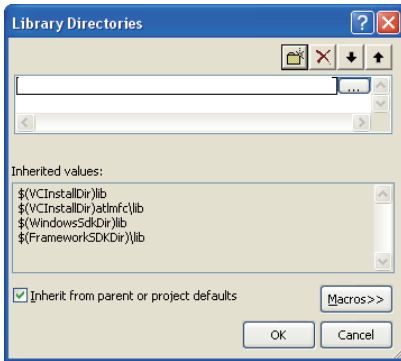
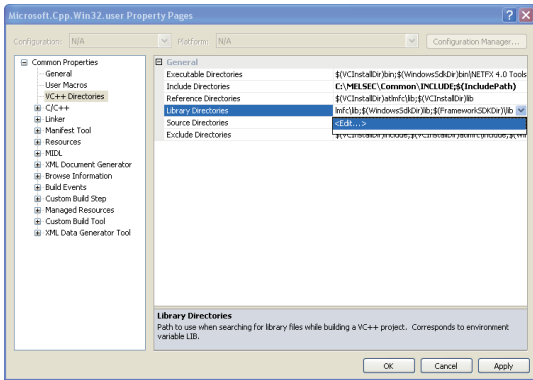



2. 右击用户属性表，选择快捷菜单的[Properties(属性)]。



将显示“Property Pages(属性页面)”画面。



3. 在“Property Pages(属性页面)”画面中，选择“Common Properties(通用属性)”⇒“VC++ Directories(VC++目录)”。



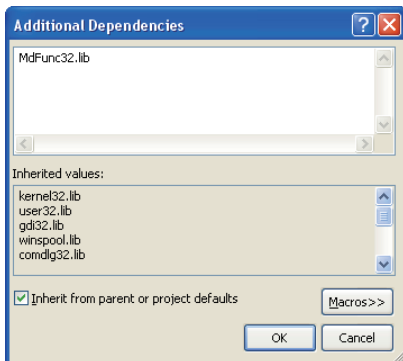
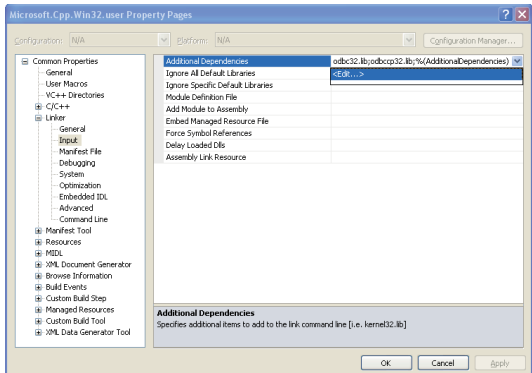
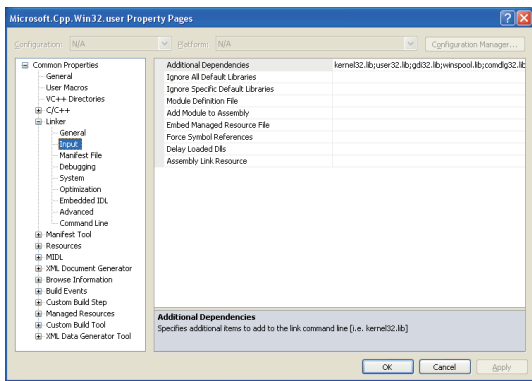
4. 选择“Library Directories(库目录)”。 点击按钮，选择“<Edit... (编辑...)>”。

5. 在“Library Directories(库目录)”画面中， 点击按钮，接着  点击。

6. 在“Select Directory(目录的选择)”画面中，浏览存储了库文件的文件夹。

- 32位版用户应用程序创建用的情况下：“MdFunc32.lib”存储在安装的文件夹的“COMMON\LIB”中。
- 64位版用户应用程序创建用的情况下：“MdFunc32.lib”存储在安装的文件夹的“COMMON\LIB\x64”中。

7. 打开创建的工程，选择[Project(工程)]⇒[Properties(属性)]菜单。



8. 在“Property Pages (属性页面)”画面的文件夹栏中，选择“Configuration Properties (配置属性)”⇒“Linker (链接器)”⇒“Input (输入)”。

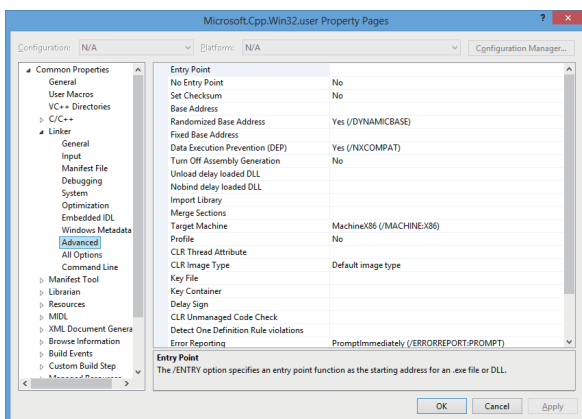
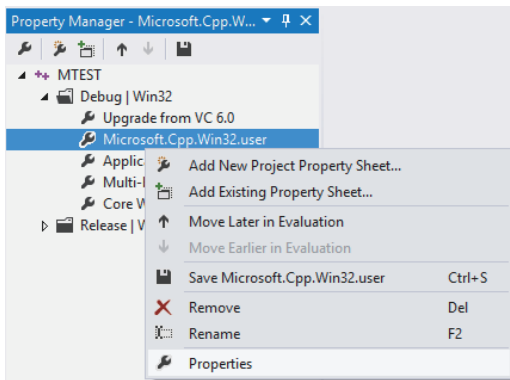
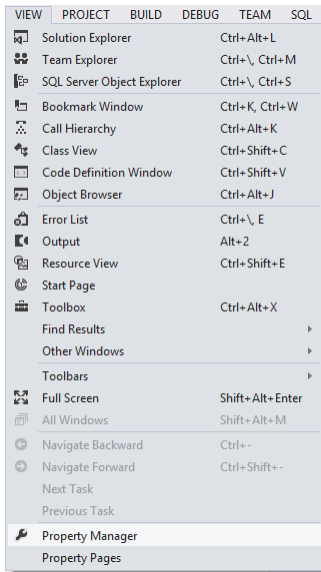
9. 选择“Additional Dependencies (添加的依存文件)”。
 ▼ 点击按钮，选择“<Edit... (编辑...)>”。

10. 在“Additional Dependencies (添加的依存文件)”画面中，记载“MdFunc32.lib”。

/SAFESEH(包括安全例外处理程序的图像)选项的删除步骤

在链接器选项中，正在使用/SAFESEH(包括安全例外处理程序的图像)选项的情况下，将无法编译。

在Visual Studio 2012 Visual C++中，在初始设置中已设置了选项，因此应通过下述操作删除/SAFESEH选项。



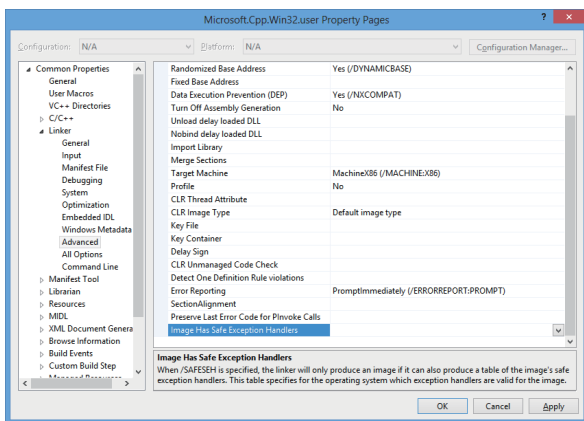
1. 打开创建的工程，选择[View(显示)]⇒[Property Manager(属性管理器)]*1菜单。

*1 根据开发设置及自定义，菜单的配置有可能不同。

2. 右击用户属性表，选择快捷菜单的[Properties(属性)]。

将显示“Property Pages(属性页面)”画面。

3. 在“Property Pages(属性页面)”画面中，选择“Common Properties(通用属性)”⇒“Linker(链接器)”⇒“Advanced(详细设置)”。



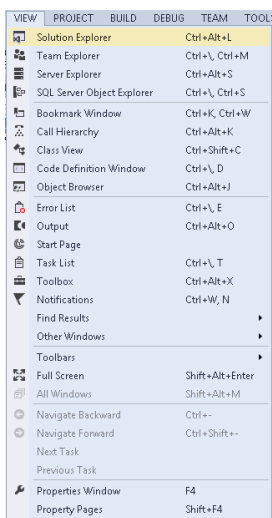
4. 选择“Image Has Safe Exception Handlers(包括安全例外处理程序的图像)”，已设置了选项的情况下应删除。

使用Visual Studio 2013、Visual Studio 2015及Visual Studio 2017的Visual C++的情况下

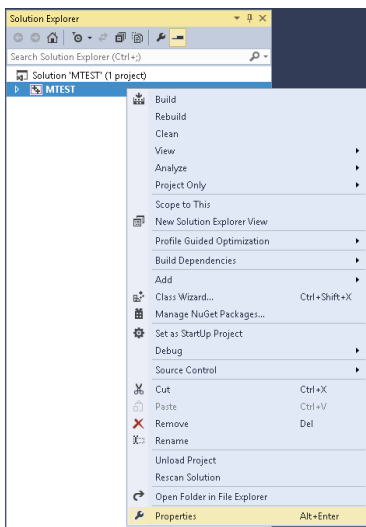
要点

- 本项的说明使用在Windows 8 Professional (x64)中使用Visual C++ 2013，转换样本程序“MTEST (VC)”后打开情况下的画面。
- 在Visual Studio 2015、Visual Studio 2017中，如果标准安装Visual Studio，有可能无法安装必要功能，样本程序的工程转换中有可能发生出错。应通过自定义选择安装“Microsoft Foundation Classes for C++”。关于安装时的功能的选择方法，或安装后的功能的添加方法，请参阅Microsoft的主页。

进行包含文件设置的情况下

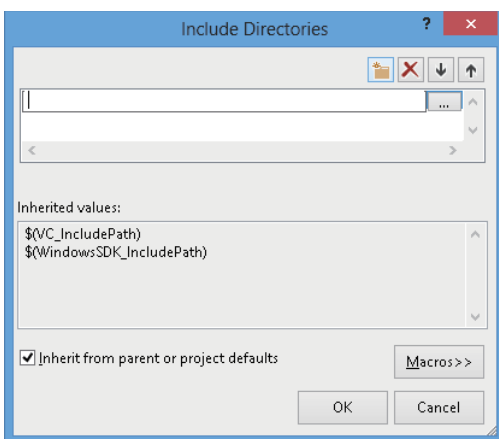
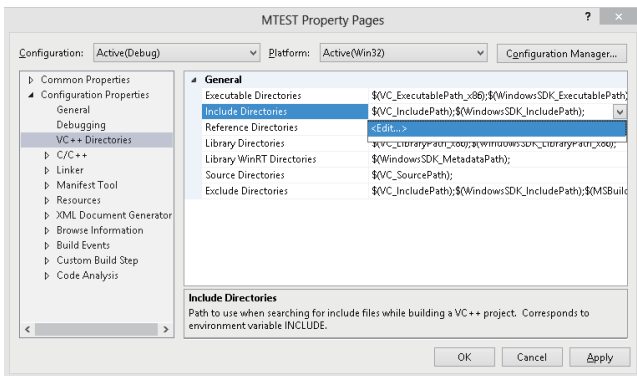
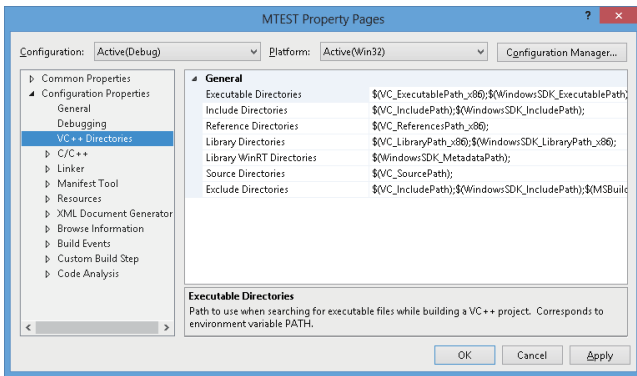
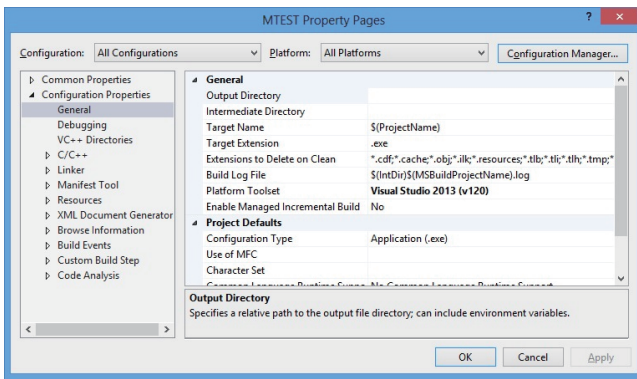


1. 打开创建的工程，选择[View(显示)]⇒[Solution Explorer(解决方案资源管理器)]菜单。



2. 在解决方案资源管理器中，右击工程，选择快捷菜单的[Properties(属性)]。
将显示“Property Pages(属性页面)”画面。





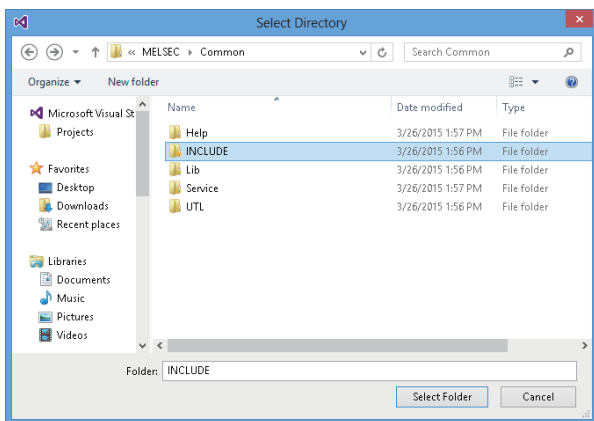
3. 在配置及平台中，选择更改设置的配置及平台。有多个的情况下，可以通过选择下述内容进行一次设置。

- 配置：“All Configurations(所有配置)”
- 平台：“All Platforms(所有平台)”

4. 在“Property Pages(属性页面)”画面中，选择 [Configuration Properties(配置属性)] => “VC++ Directories(VC++目录)”。

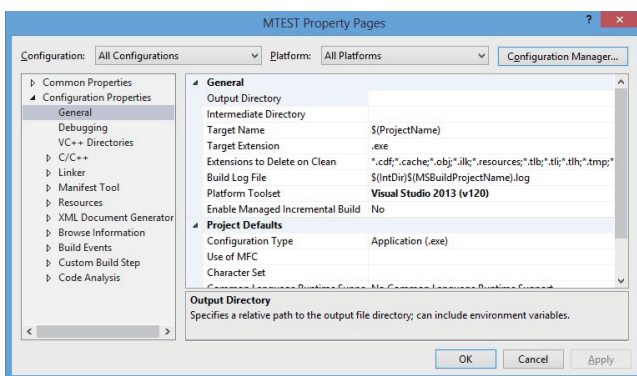
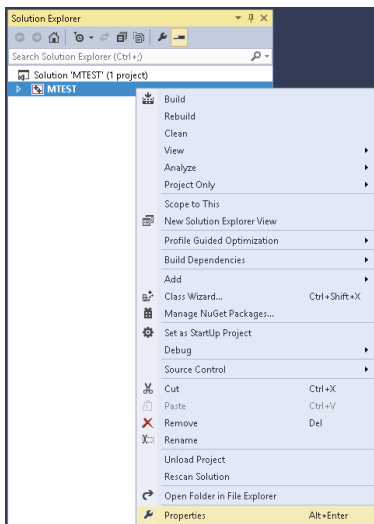
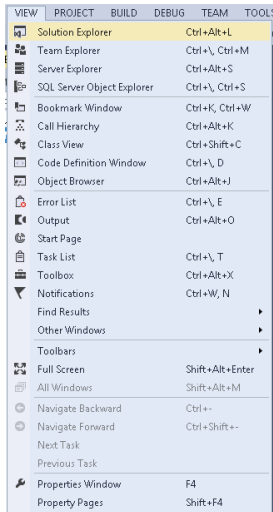
5. 选择包含目录后， 点击按钮，选择“<Edit... (编辑...)>”。

6. 在“Include Directories(包含目录)”画面中， 点击，接着 选。



7. 在“Select Directory(目录的选择)”画面中，选择存储了包含文件的文件夹。
 - “Mdfunc.h”存储在安装的文件夹的“COMMON\INCLUDE”中。
8. 在程序的起始处添加#include<Mdfunc.h>。

进行库文件设置的情况下

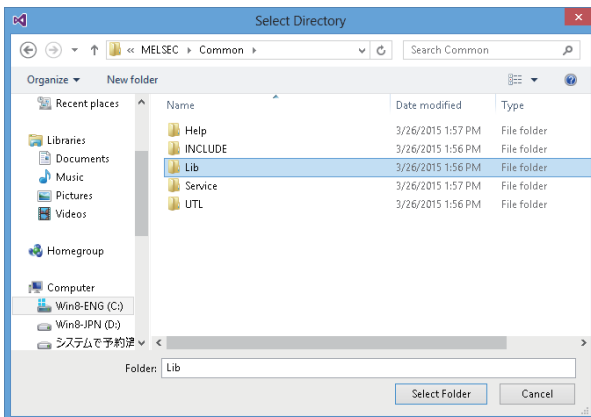
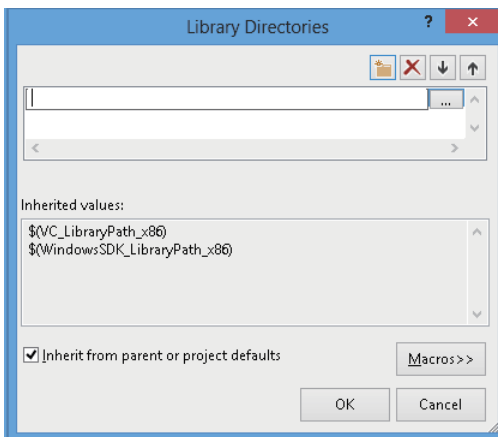
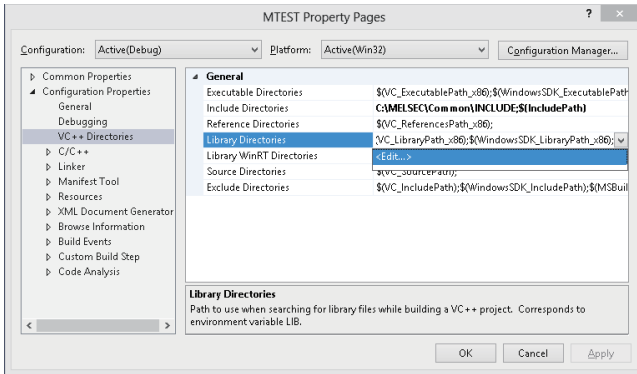
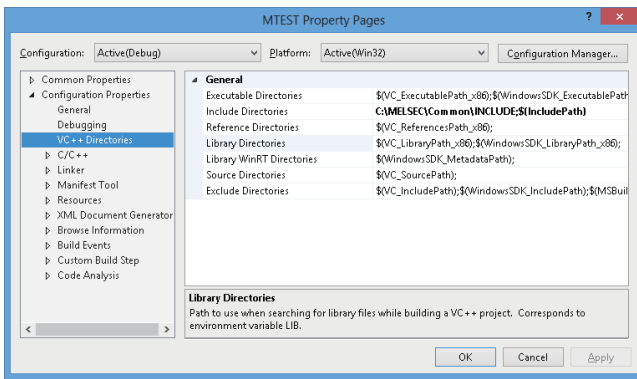


1. 打开创建的工程，选择[View(显示)]⇒[Solution Explorer(解决方案资源管理器)]菜单。


2. 在解决方案资源管理器中，右击工程，选择快捷菜单的[Properties(属性)]。将显示“Property Pages(属性页面)”画面。



3. 在配置及平台中，选择更改设置的配置及平台。有多个的情况下，可以通过选择下述内容进行一次设置。

- 配置：“All Configurations(所有配置)”
- 平台：“All Platforms(所有平台)”



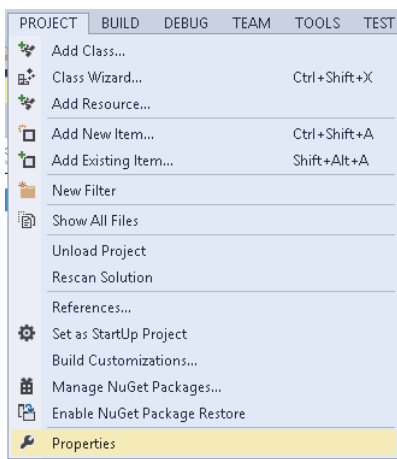
4. 在“Property Pages (属性页面)”画面中，选择 [Configuration Properties (配置属性)] ⇒ “VC++ Directories (VC++目录)”。

5. 选择“Library Directories (库目录)”。 点击按钮，选择“<Edit... (编辑...)>”。

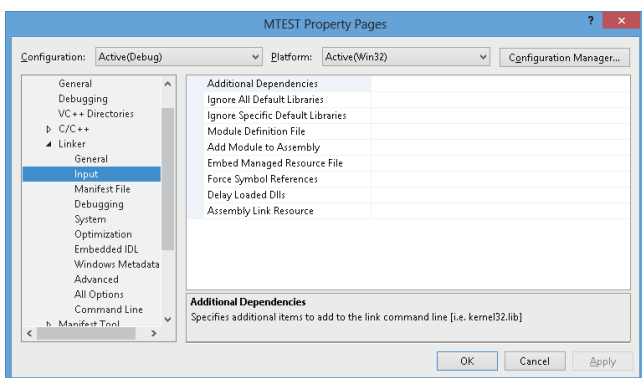
6. 在“Library Directories (库目录)”画面中， 点击按钮，接着  点击。

7. 在“Select Directory (目录的选择)”画面中，浏览存储了库文件的文件夹。

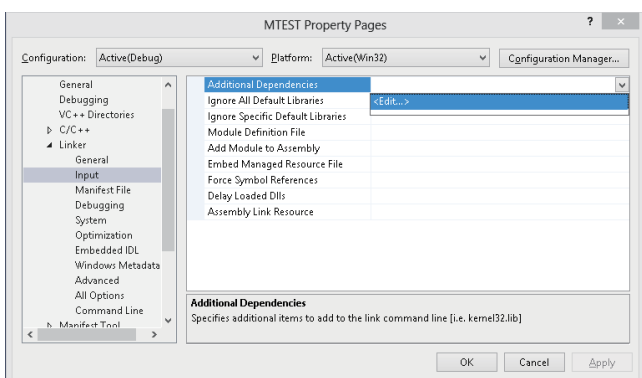
- 32位版用户应用程序创建用的情况下：“MdFunc32.lib”存储在安装的文件夹的“COMMON\LIB”中。
- 64位版用户应用程序创建用的情况下：“MdFunc32.lib”存储在安装的文件夹的“COMMON\LIB\x64”中。



8. 打开创建的工程，选择[Project (工程)]⇒[Properties (属性)]菜单。

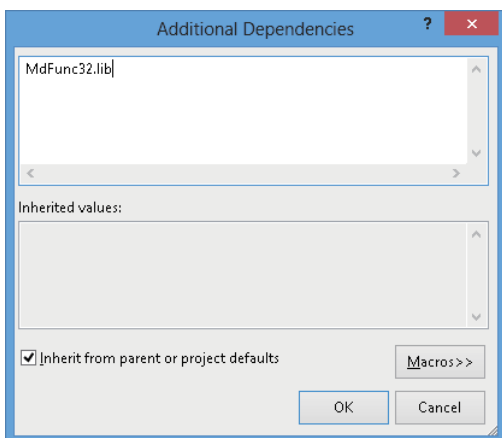


9. 在“Property Pages (属性页面)”画面中，选择“Configuration Properties (配置属性)”⇒“Linker (链接器)”⇒“Input (输入)”。



10. 选择“Additional Dependencies (添加的依存文件)”。

点击按钮，选择“<Edit... (编辑...)>”。

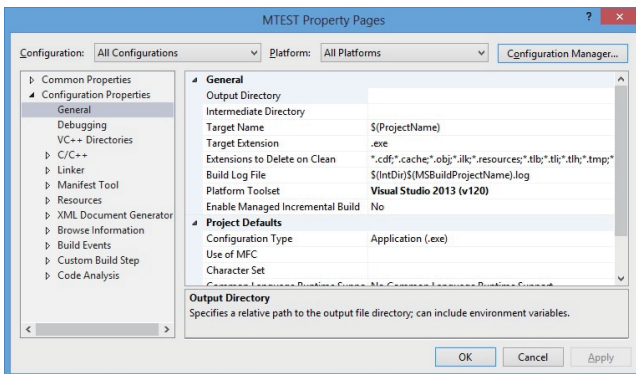
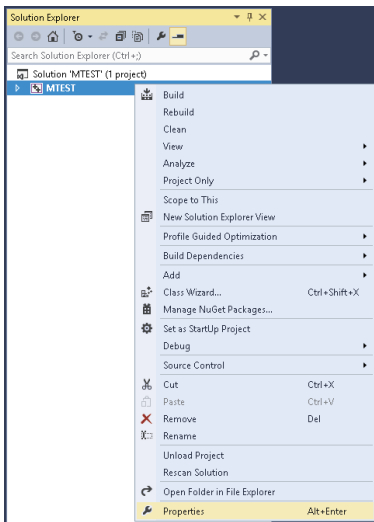
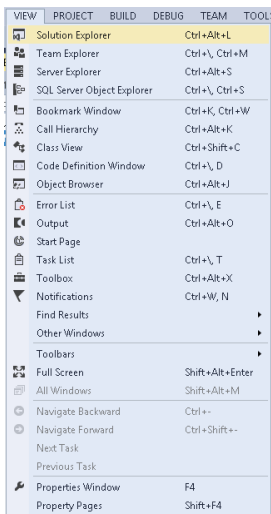


11. 在“Additional Dependencies (添加的依存文件)”画面中，输入“MdFunc32.lib”。

平台设置及不需要的选项的删除

通过1~5的操作，进行平台的设置。

此外，通过6~9的操作，删除不需要的选项。

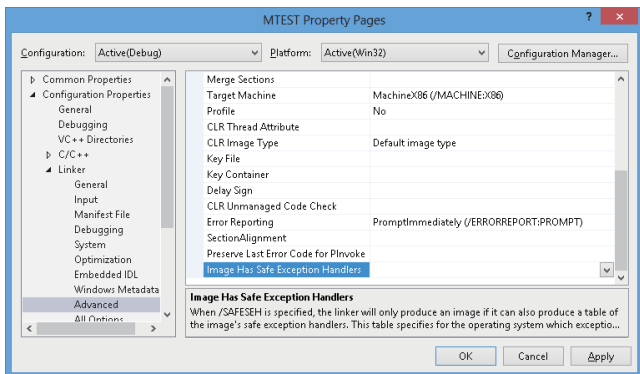
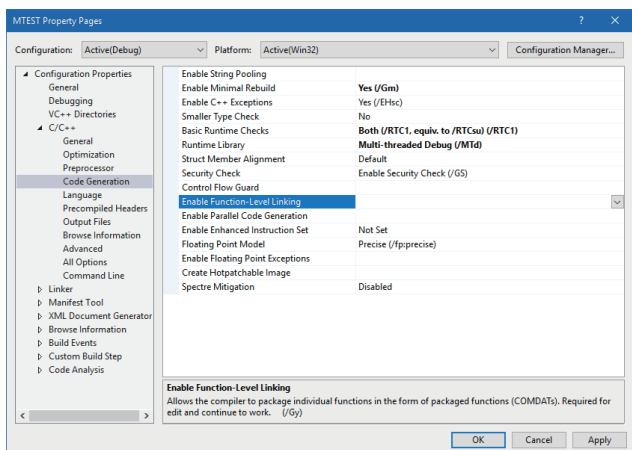
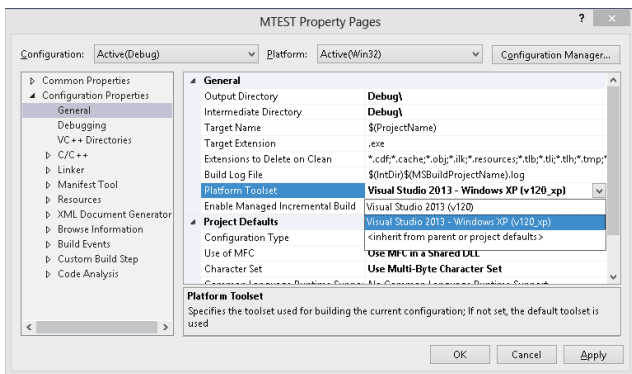
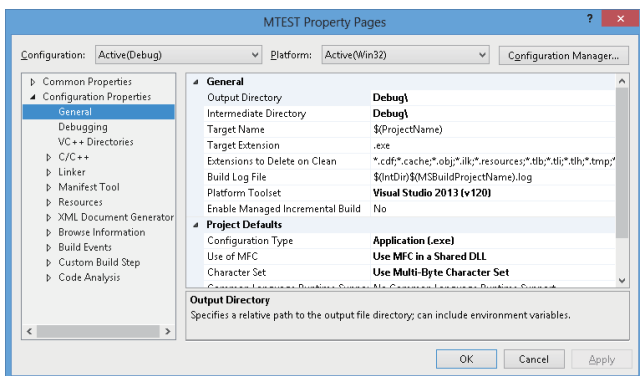


1. 打开创建的工程，选择[View(显示)]⇒[Solution Explorer(解决方案资源管理器)]菜单。

2. 在解决方案资源管理器中，右击工程，选择快捷菜单的[Properties(属性)]。
将显示“Property Pages(属性页面)”画面。

3. 在配置及平台中，选择更改设置的配置及平台。
有多个的情况下，可以通过选择下述内容进行一次设置。

- 配置：“All Configurations(所有配置)”
- 平台：“All Platforms(所有平台)”



4. 在“Property Pages(属性页面)”画面中,选择 [Configuration Properties(配置属性)]⇒[General(常规)]。

5. 选择平台工具设置。点击按钮,选择下述之一。

- “Visual Studio 2013 - Windows XP (v120_xp)”
- “Visual Studio 2015 - Windows XP (v140_xp)”
- “Visual Studio 2017 - Windows XP (v141_xp)”

配置Windows XP用程序的情况下需要。

6. 在“Property Pages(属性页面)”画面中,选择 “Configuration Properties(配置属性)” ⇒ “C/C++” ⇒ “Code Generation(代码生成)”。^{*1}

^{*1} 左述画面为Windows Server 2017。

7. 在“Enable Function-Level Linking(启用函数级链接)”中,确认未设置选项。已设置的情况下,应删除选项。

如果设置了选项,更新样本程序时有可能发生指令行出错D8016。

8. 在“Property Pages(属性页面)”画面中,选择 “Configuration Properties(配置属性)” ⇒ “Linker(链接器)” ⇒ “Advanced(详细设置)”。

9. 在“Image Has Safe Exception Handlers(包括安全例外处理程序的图像)”中,确认未设置选项。已设置的情况下,应删除选项。

正在使用“Image Has Safe Exception Handlers(包括安全例外处理程序的图像)”选项的情况下,将无法正常编译。

3 可访问软元件及范围

以下介绍使用了各板的通信时可访问软元件及可访问范围。

3.1 访问目标

可进行其它站访问的访问目标如下所示。

项目		型号*1	
MELSEC A系列	CPU模块	ACPU*2	A0J2HCPU、A1SCPU、A1SJCPU、A1SHCPU、A1SJHCPU、A1NCP、A2CCPU、A2CJCPU、A2NCP、A2NCP-S1、A2SCPU、A2SHCPU、A3NCP、A2ACPU、A2ACPU-S1、A3ACPU、A2UCPU、A2UCPU-S1、A2USCPU、A2USCPU-S1、A2USHCPU-S1、A3UCPU、A4UCPU
		QCPU(A模式)	Q02CPU-A、Q02HCPU-A、Q06HCPU-A
MELSEC QnA系列	CPU模块	QnACPU	Q2ACPU、Q2ASCPU、Q2ASHCPU、Q2ACPU-S1、Q2ASCPU-S1、Q2ASHCPU-S1、Q3ACPU、Q4ACPU、Q4ARCPU
MELSEC Q系列	CPU模块	QCPU(Q模式)	<ul style="list-style-type: none"> ■基本型QCPU Q00JCPU、Q00CPU、Q01CPU ■高性能型QCPU Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU ■过程CPU Q02PHCPU、Q06PHCPU、Q12PHCPU、Q25PHCPU ■冗余CPU Q12PRHCPU、Q25PRHCPU ■通用型QCPU Q00UCPU、Q00UJVCPU、Q01UCPU、Q02UCPU、Q03UDCPU、Q03UDECPU、Q03UDVCPU、Q04UDHCPU、Q04UDEHCPU、Q04UDVCPU、Q04UDPVCPU、Q06UDHCPU、Q06UDEHCPU、Q06UDVCPU、Q06UDPVCPU、Q10UDHCPU、Q10UDEHCPU、Q13UDHCPU、Q13UDEHCPU、Q13UDVCPU、Q13UDPVCPU、Q20UDHCPU、Q20UDEHCPU、Q26UDHCPU、Q26UDEHCPU、Q26UDVCPU、Q26UDPVCPU、Q50UDEHCPU、Q100UDEHCPU
MELSEC L系列	CPU模块	LCPU	L02SCPU、L02SCPU-P、L02CPU、L02CPU-P、L06CPU、L06CPU-P、L26CPU、L26CPU-P、L26CPU-BT、L26CPU-PBT
	起始模块		LJ72GF15-T2
MELSEC iQ-R系列	CPU模块	RCPU	R04CPU、R04ENCPU、R08CPU、R08ENCPU、R16CPU、R16ENCPU、R32CPU、R32ENCPU、R120CPU、R120ENCPU
以太网适配器模块			NZ2GF-ETB
智能设备站			AJ65BT-R2等(可进行瞬时传送的CC-Link系统上的从站)
CC-Link IE现场网络远程设备站			NZ2GF2B1-16D等(具有缓冲存储器的三菱电机生产CC-Link IE现场网络远程设备站)
计算机用接口板	CC-Link Ver.2板		Q80BD-J61BT11N、Q81BD-J61BT11
	MELSECNET/H板		Q80BD-J71LP21-25、Q81BD-J71LP21-25、Q80BD-J71LP21S-25、Q80BD-J71LP21G、Q80BD-J71LP21GE、Q80BD-J71BR11
	CC-Link IE控制网络板		Q80BD-J71GP21-SX、Q80BD-J71GP21S-SX、Q81BD-J71GP21-SX、Q81BD-J71GP21S-SX
	CC-Link IE现场网络板		Q80BD-J71GF11-T2、Q81BD-J71GF11-T2

*1 不支持各网络的型号或版本的产品不能访问。

关于各产品与网络的对应，请参阅各产品的手册。

*2 通过CC-Link IE现场网络板的情况下，只能访问A2UCPU、A2UCPU-S1、A2USCPU、A2USCPU-S1、A2USHCPU-S1、A3UCPU、A4UCPU。

要点

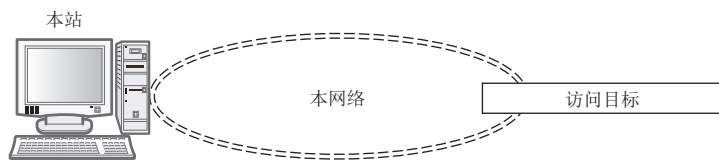
访问QCPU(Q模式)、RCPU及CC-Link IE现场网络远程设备站的情况下可以使用64位版用户应用程序。访问上述以外的情况下，应使用32位版用户应用程序。

3.2 可访问范围

以下介绍使用了各板的通信中的可访问范围。

本网络内的访问目标

在连接了各板的网络内，可以访问本站与下述模块以及板。



○：可以访问；×：不可访问

本站	本网络	访问目标*1						
		MELSEC iQ-R系列CPU模块	LCPU、起始模块	以太网适配器模块	QCPU(Q模式)、计算机用接口板	ACPU、QnACPU、QCPU(A模式)	智能设备站	CC-Link IE现场网络远程设备站
CC-Link Ver. 2板*2	CC-Link	○	○	×	○	○	○	×
MELSECNET/H板	MELSECNET/H	○	×	×	○	×	×	×
	MELSECNET/10	×	×	×	○	○	×	×
CC-Link IE控制网络板	CC-Link IE控制网络	○	×	×	○	×	×	×
CC-Link IE现场网络板	CC-Link IE现场网络	○	○	○	○	×	×	○

*1 35页 访问目标

*2 本站的站号为64的情况下，不能访问其它站。只能访问本站。

关于各访问目标的可访问软元件，请参阅下述内容。

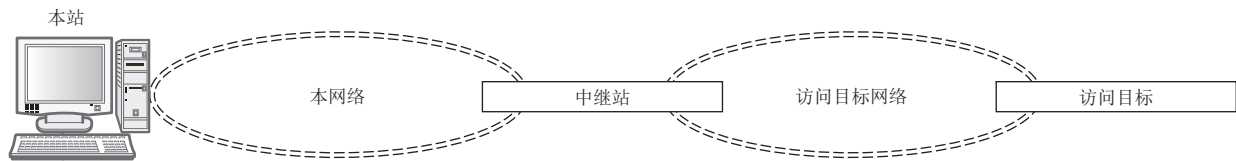
访问目标	参阅
至本站的连接软元件、缓冲存储器的访问	计算机用接口板 38页 至本站的连接软元件、缓冲存储器的访问
至其它站模块的软元件的访问	MELSEC iQ-R系列CPU模块 39页 至MELSEC iQ-R系列模块的软元件的访问
	QnACPU、QCPU(Q模式)、LCPU 40页 至MELSEC QnA、Q、L系列模块的软元件的访问
	ACPU、QCPU(A模式) 41页 至MELSEC A系列模块的软元件的访问
	以太网适配器模块、起始模块 42页 至以太网适配器模块、起始模块、CC-Link IE现场网络远程设备站的访问
至CC-Link的其它站缓冲存储器的访问	CC-Link的网络模块、CC-Link Ver. 2板等 42页 至CC-Link的其它站缓冲存储器的访问
通过SEND功能、RECV功能的访问	QnACPU、QCPU(Q模式)、LCPU、MELSEC iQ-R系列CPU模块、计算机用接口板 43页 通过SEND功能、RECV功能的访问

经由了网络情况下的访问目标

经由了多个网络情况下的访问可否如下所示。

下述内容中未记载的组合不能访问。

CC-Link不能进行经由了多个网络的访问。



○：可以访问；×：不可访问

本网络	中继站	访问目标网络	访问目标*1				
			MELSEC iQ-R系列CPU模块	LCPU、起始模块、以太网适配器模块	QCPU(Q模式)、计算机用接口板	ACPU*3、QnACPU、QCPU(A模式)	CC-Link IE现场网络远程设备站
CC-Link IE控制网络、CC-Link IE现场网络	RCPU	CC-Link IE控制网络	○	×	○	×	×
		CC-Link IE现场网络	○	○	○	×	×
	QCPU(Q模式)*2	MELSECNET/H	×	×	○	×	×
		MELSECNET/10	×	×	○	○	×
		CC-Link IE控制网络	○	×	○	×	×
CC-Link IE现场网络	○	○	○	×	×		
MELSECNET/H、MELSECNET/10	QCPU(Q模式)*2	MELSECNET/H	×	×	○	×	×
		MELSECNET/10	×	×	○	○	×
		CC-Link IE控制网络	×	×	○	×	×
		CC-Link IE现场网络	×	○	○	×	×
CC-Link IE现场网络	LCPU	CC-Link IE现场网络	○	○	○	×	×
MELSECNET/10	QnACPU、ACPU、QCPU(A模式)	MELSECNET/10	×	×	○	○	×

*1 35页 访问目标

*2 网络模块的安装个数为1个的CPU模块不能设置为中继站。

*3 通过CC-Link IE现场网络板的情况下，只能访问A2UCPU、A2UCPU-S1、A2USCPU、A2USCPU-S1、A2USHCPU-S1、A3UCPU、A4UCPU。

关于各访问目标的访问软件，请参阅下述内容。

访问目标	参阅	
至其它站模块的软件的访问	MELSEC iQ-R系列CPU模块	39页 至MELSEC iQ-R系列模块的软件的访问
	QnACPU、QCPU(Q模式)、LCPU	40页 至MELSEC QnA、Q、L系列模块的软件的访问
	ACPU、QCPU(A模式)	41页 至MELSEC A系列模块的软件的访问
	以太网适配器模块、起始模块	42页 以太网适配器模块、起始模块、CC-Link IE现场网络远程设备站的访问
通过SEND功能、RECV功能的访问	QnACPU、QCPU(Q模式)、LCPU、MELSEC iQ-R系列CPU模块、计算机用接口板	43页 通过SEND功能、RECV功能的访问

3.3 可访问软元件

各板的可访问软元件如下所示。

在本节的表中，将访问中使用的函数分为“Batch(批量)”及“Random(随机)”的2种类型表示能否访问。

访问方法	内容
批量	批量写入(mdSend、mdSendEx)
	批量读取(mdReceive、mdReceiveEx)
随机	随机写入(mdRandW、mdRandWEx)
	随机读取(mdRandR、mdRandREx)
	位设置(mdDevSet、mdDevSetEx)
	位复位(mdDevRst、mdDevRstEx)

至本站的链接软元件、缓冲存储器的访问

访问板(本站)情况下的可访问软元件如下所示。

○：可以访问；×：不可访问

软元件		访问方法	访问目标			
			CC-Link Ver.2板	MELSECNET/H板	CC-Link IE控制网络板	CC-Link IE现场网络板
链接输入	LX	批量/随机	×	○	○	×
链接输出	LY	批量/随机	×	○	○	×
链接继电器	LB	批量/随机	×	○	○	×
链接寄存器	LW	批量/随机	×	○	○	×
链接特殊继电器	SB	批量/随机	○	○	○	○
链接特殊寄存器	SW	批量/随机	○	○	○	○
远程输入	RX	批量/随机	○	×	×	○
远程输出	RY	批量/随机	○	×	×	○
远程寄存器(发送用)	RWw	批量/随机	○	×	×	○
远程寄存器(接收用)	RWr	批量/随机	○	×	×	○
缓冲存储器	—	批量/随机	○	×	○	○
随机访问缓冲	—	批量/随机	○	×	×	×

至MELSEC iQ-R系列模块的软元件的访问

至安装了其它站模块的MELSEC iQ-R系列CPU模块的软元件的访问可否如下所示。

○：可以访问；×：不可访问

软元件		访问方法	访问目标*1	
			RCPU	
输入	X	批量/随机	○	
输出	Y	批量/随机	○	
内部继电器	M	批量/随机	○	
锁存继电器	L	批量/随机	○	
链接继电器	B	批量/随机	○	
报警器	F	批量/随机	○	
链接特殊继电器	SB	批量/随机	○	
变址继电器	V	批量/随机	○	
定时器	触点、线圈、当前值	T	批量/随机	○
累计定时器	触点、线圈、当前值	ST	批量/随机	○
超长定时器	触点、线圈、当前值	LT	批量/随机	○*2
超长累计定时器	触点、线圈、当前值	LST	批量/随机	○*2
计数器	触点、线圈、当前值	C	批量/随机	○
超长计数器	触点、线圈、当前值	LC	批量/随机	○*2
数据寄存器		D	批量/随机	○
链接寄存器		W	批量/随机	○
链接特殊寄存器		SW	批量/随机	○
特殊继电器		SM	批量/随机	○
特殊寄存器		SD	批量/随机	○
链接直接软元件	链接输入 链接输出 链接继电器 链接特殊继电器 链接寄存器 链接特殊寄存器	Jn\X Jn\Y Jn\B Jn\SB Jn\W Jn\SW	批量/随机	○
模块访问软元件		Un\G	批量/随机	○
变址寄存器		Z	批量/随机	○
超长变址寄存器		LZ	批量/随机	○*2
文件寄存器		R*3、ZR	批量/随机	○
刷新数据寄存器		RD	批量/随机	○*2

*1 ④ 35页 访问目标

*2 无法通过MELSECNET/H板访问。

*3 使用软元件类型DevER0~256时，可以指定块No. 进行访问。

至MELSEC QnA、Q、L系列模块的软元件的访问

至安装了其它站模块的QnACPU、QCPU(Q模式)或LCPUs的软元件的访问可否如下所示。

○：可以访问；×：不可访问

软元件		访问方法	访问目标*1
			QnACPU、QCPU、LCPUs
输入	X	批量/随机	○
输出	Y	批量/随机	○
内部继电器	M	批量/随机	○
锁存继电器	L	批量/随机	○
报警器	F	批量/随机	○
变址继电器	V	批量/随机	○
链接继电器	B	批量/随机	○
链接特殊继电器	SB	批量/随机	○
定时器	触点、线圈、当前值	T	批量/随机
累计定时器	触点、线圈、当前值	ST	批量/随机
计数器	触点、线圈、当前值	C	批量/随机
数据寄存器	D	批量/随机	○
链接寄存器	W	批量/随机	○
链接特殊寄存器	SW	批量/随机	○
特殊继电器	SM	批量/随机	○
特殊寄存器	SD	批量/随机	○
链接直接软元件	链接输入 链接输出 链接继电器 链接寄存器 链接特殊继电器 链接特殊寄存器	Jn\X Jn\Y Jn\B Jn\W Jn\SB Jn\SW	批量/随机 ○
智能功能模块软元件	Un\G	批量/随机	○
变址寄存器	Z	批量/随机	○
文件寄存器	R*2、ZR	批量/随机	○*3

*1 35页 访问目标

*2 使用软元件类型DevER0~256时，可以指定块No. 进行访问。

*3 Q00CPU的情况下，不能访问。

至MELSEC A系列模块的软元件的访问

至安装了其它站模块的ACPU或QCPU(A模式)的软元件的访问可否如下所示。

○：可以访问；×：不可访问

软元件		访问方法	访问目标					
			A1NCPU* ¹	A0J2HCPU、 A1SCPU、 A1SJCPU、 A1SHCPU、 A1SJHCPU、 A2CCPU、 A2CJCPU、 A2NCPU、 A2NCPU-S1、 A2SCPU、 A2SHCPU、 A2ACPU、 A2ACPU-S1* ¹	A3NCPU、 A3ACPU* ¹	A2UCPU、A2UCPU-S1、 A2USCPU、 A2USHCPU-S1、 A3UCPU、QCPU(A模式)	A4UCPU	
输入	X	批量/随机	○	○	○	○	○	
输出	Y	批量/随机	○	○	○	○	○	
锁存继电器	L	批量/随机	○	○	○	○	○	
内部继电器	M	批量/随机	○	○	○	○	○	
特殊继电器	M9000~	批量/随机	○	○	○	○	○	
数据寄存器	D	批量/随机	○	○	○	○	○	
特殊寄存器	D9000~	批量/随机	○	○	○	○	○	
报警器	F	批量/随机	○	○	○	○	○	
定时器	触点、线圈、当前值 设置值主 设置值子1 设置值子2、子3	T	批量/随机	○	○	○	○	○
		批量	○	○	○	○	○	
		随机	×	×	×	×	×	
		批量	×	×	○	○	○	
		随机	×	×	×	×	×	
		批量	×	×	×	×	○	
计数器	触点、线圈、当前值 设置值主 设置值子1 设置值子2、子3	C	批量/随机	○	○	○	○	○
		批量	○	○	○	○	○	
		随机	×	×	×	×	×	
		批量	×	×	○	○	○	
		随机	×	×	×	×	×	
		批量	×	×	×	×	○	
累加器	A	批量/随机	○	○	○	○	○	
变址寄存器	Z、V	批量/随机	○	○	○	○	○	
文件寄存器 扩展文件寄存器* ²	R	批量/随机	×	○	○	○	○	
链接继电器	B	批量/随机	○	○	○	○	○	
链接寄存器	W	批量/随机	○	○	○	○	○	

*1 无法通过CC-Link IE现场网络访问。

*2 使用软元件类型DevER0~64时，可以指定块No. 进行访问。

至以太网适配器模块、起始模块、CC-Link IE现场网络远程设备站的访问

至经由了CC-Link及CC-Link IE现场网络的以太网适配器模块、起始模块或CC-Link IE现场网络远程设备站的访问可否如下所示。

○：可以访问；×：不可访问

软元件		访问目标	访问目标		
			以太网适配器模块	起始模块	CC-Link IE现场网络远程设备站
输入	X	批量/随机	○	○	×
输出	Y	批量/随机	○	○	×
特殊继电器	SM	批量/随机	○	○	×
特殊寄存器	SD	批量/随机	○	○	×
链接寄存器	W	批量/随机	○	○	×
链接特殊继电器	SB	批量/随机	○	○	×
链接特殊寄存器	SW	批量/随机	○	○	×
智能功能模块软元件	Un\G	批量/随机	×	○	×
缓冲存储器					○*1
智能功能模块访问软元件					×

*1 经由CC-Link IE现场网络时，只能通过mdRemBufWriteEx函数、mdRemBufReadEx函数进行缓冲存储器访问。

至CC-Link的其它站缓冲存储器的访问

至CC-Link的网络模块及其它CC-Link Ver. 2板的缓冲存储器的访问可否如下所示。

不能访问多CPU系统(逻辑站指定时)。

○：可以访问；×：不可访问

软元件		访问方法	访问可否
链接特殊继电器	SB	批量	○
链接特殊寄存器	SW	批量	○
远程输入	RX	批量	○
远程输出	RY	批量	○
远程寄存器	RW	批量	○
缓冲存储器	—	批量	○
随机访问缓冲	—	批量	○

要点

本站的站号为64的情况下，不能访问其它站。只能访问本站。

通过SEND功能、RECV功能的访问

在SEND功能、RECV功能中，与软元件访问一样，指定各功能对应的软元件类型，进行批量写入 (mdSend、mdSendEx) 或批量读取 (mdReceive、mdReceiveEx)。

○：可以访问；—：不是对象

软元件	访问方法	访问目标	
		QnACPU、QCPU(Q模式)、LCP、MELSEC iQ-R系列CPU模块	计算机用接口板
RECV功能	批量	—	○(本站)
SEND功能(有到达确认)	批量	○	○
SEND功能(无到达确认)			

要点

- 在CC-Link中，不能使用SEND功能、RECV功能。
- SW1DNC-MNETG-B为版本1.08J及以上产品，支持SEND功能、RECV功能。
- MELSEC A系列的CPU模块、以太网适配器模块及起始模块不支持SEND功能、RECV功能。

4 函数

以下介绍MELSEC数据链接库的各函数。

4.1 函数一览

附加在软件包中的MELSEC数据链接库的一览如下所示。

函数名	功能	备注	参阅
mdOpen	通信线路的打开	—	55页 mdOpen(通信线路的打开)
mdClose	通信线路的关闭	—	56页 mdClose(通信线路的关闭)
mdSendEx	软元件的批量写入	扩展函数*3	57页 mdSendEx(扩展软元件批量写入)
	数据发送(SEND功能)*1、*2	扩展函数*3	59页 mdSendEx(SEND功能)
mdReceiveEx	软元件的批量读取	扩展函数*3	61页 mdReceiveEx(扩展软元件批量读取)
	数据接收(RECV功能)*1、*2	扩展函数*3	63页 mdReceiveEx(RECV功能)
mdRandWEx	软元件的随机写入	扩展函数*3	65页 mdRandWEx(扩展软元件随机写入)
mdRandREx	软元件的随机读取	扩展函数*3	68页 mdRandREx(扩展软元件随机读取)
mdDevSetEx	位软元件设置	扩展函数*3	71页 mdDevSetEx(扩展位软元件设置)
mdDevRstEx	位软元件复位	扩展函数*3	72页 mdDevRstEx(扩展位软元件复位)
mdRemBufWriteEx	远程设备站的缓冲存储器写入*4	扩展函数*3	73页 mdRemBufWriteEx(远程设备站的缓冲存储器写入)
mdRemBufReadEx	远程设备站的缓冲存储器读取*4	扩展函数*3	74页 mdRemBufReadEx(远程设备站的缓冲存储器读取)
mdTypeRead	可编程控制器CPU类型的读取	—	75页 mdTypeRead(CPU型号读取)
mdControl	可编程控制器CPU的远程操作(RUN/STOP/PAUSE)	—	79页 mdControl(远程RUN/STOP/PAUSE)
mdWaitBdEvent	事件发生等待*1	—	80页 mdWaitBdEvent(事件等待)
mdBdRst	板的复位	—	83页 mdBdRst(板复位)
mdBdModSet	板的模式的设置	—	84页 mdBdModSet(板模式设置)
mdBdModRead	板的模式的读取	—	86页 mdBdModRead(板模式读取)
mdBdLedRead	板的LED信息的读取	—	87页 mdBdLedRead(板LED读取)
mdBdSwRead	板的开关状态的读取	—	93页 mdBdSwRead(板开关状态读取)
mdBdVerRead	板的版本信息的读取	—	94页 mdBdVerRead(板版本读取)
mdInit	可编程控制器软元件信息表的初始化	—	96页 mdInit(可编程控制器软元件信息表的初始化)

*1 在CC-Link IE控制网络板或CC-Link IE现场网络板中可以使用。

*2 在Version1.08J及以上的SW1DNC-MNETG-B中支持CC-Link IE控制网络板。

*3 是随着访问目标的软元件点数扩展，扩展访问范围的函数，可以访问所有的软元件No。
创建新程序的情况下，应使用扩展函数。

*4 在序列号的前5位数为15102及以上的CC-Link IE现场网络板、Version1.06G及以上的SW1DNC-CCIEF-B中可以使用。
64位版用户应用程序的情况下，应使用Version1.12N及以上的SW1DNC-CCIEF-B。

以往兼容函数一览

以往兼容函数是用于使用以往创建的程序的函数。

函数名	功能	备注	参阅
mdSend	软元件的批量写入	—	97页 mdSend(软元件批量写入)
	数据发送(SEND功能)*1、*2	—	99页 mdSend(SEND功能)
mdReceive	软元件的批量读取	—	101页 mdReceive(软元件批量读取)
	数据接收(RECV功能)*1、*2	—	103页 mdReceive(RECV功能)
mdRandW	软元件的随机写入	—	105页 mdRandW(软元件随机写入)
mdRandR	软元件的随机读取	—	107页 mdRandR(软元件随机读取)
mdDevSet	位软元件设置	—	108页 mdDevSet(位软元件设置)
mdDevRst	位软元件复位	—	109页 mdDevRst(位软元件复位)

*1 在MELSECNET/H板、CC-Link IE控制网络板或CC-Link IE现场网络板中可以使用。

*2 在Version1.08J及以上产品中支持CC-Link IE控制网络板。

要点

使用以往兼容函数的情况下，可访问软元件No. 为0~32767。

4.2 函数通用规格

以下介绍MELSEC数据链接库的函数中共同使用的参数的定义。

通道No. 指定

MELSEC数据链接库中使用的通道如下所示。

通道No.	通道名	内容
51	MELSECNET/H(第1块)	MELSECNET/H板
52	MELSECNET/H(第2块)	通道No. 通过MELSECNET/H实用程序设置。
53	MELSECNET/H(第3块)	
54	MELSECNET/H(第4块)	
81	CC-Link(第1块)	
82	CC-Link(第2块)	CC-Link Ver.2板 通道No. 通过通道No. 设置开关设置。 根据SW1、2的设置, 其设置如下所示。 • 81: OFF, OFF • 82: ON, OFF • 83: OFF, ON • 84: ON, ON
83	CC-Link(第3块)	
84	CC-Link(第4块)	
151	CC-Link IE控制网络(通道No. 151)	
152	CC-Link IE控制网络(通道No. 152)	CC-Link IE控制网络板 通道No. 通过CC IE Control实用程序设置。
153	CC-Link IE控制网络(通道No. 153)	
154	CC-Link IE控制网络(通道No. 154)	
181	CC-Link IE现场网络(通道No. 181)	
182	CC-Link IE现场网络(通道No. 182)	CC-Link IE现场网络板 通道No. 通过CC IE Field实用程序设置。
183	CC-Link IE现场网络(通道No. 183)	
184	CC-Link IE现场网络(通道No. 184)	

站号指定

MELSEC数据链接库中指定的站号如下所示。

关于扩展函数的网络No. 及站号的指定，请参阅下述章节。

(☞ 49页 扩展函数的网络No. 及站号指定)

CC-Link Ver. 2板的情况下

指定方法	站号
本站	255 (FFH)
其它站	0 (00H) ~ 63 (3FH)*1
通过实用程序设置的逻辑站号	65 (41H) ~ 239 (EFH)

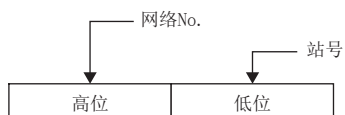
- *1 在CC-Link Ver. 2板中，不能进行站号64的指定。
本站的站号为64的情况下，不能访问其它站。只能访问本站。

MELSECNET/H板的情况下

指定方法		站号	
		高位字节	低位字节
本站		255 (FFH)	
其它站	站号	网络No. *3 1 (01H) ~ 239 (EFH)	0 (00H)*4
			1 (01H) ~ 120 (78H)*5
			125 (7DH)*4
	组No. 1 ~ 32*1、*2		129 (81H) ~ 160 (A0H)
	全站*1		240 (F0H)
通过实用程序设置的逻辑站号		65 (41H) ~ 239 (EFH)	

- *1 只有在使用无到达确认的SEND功能 (mdSend函数) 时启用全站指定、组No. 指定。
*2 MELSECNET/10模式的情况下，只能指定组No. 1 ~ 9 (129 (81H) ~ 137 (89H))。
*3 指定其它站的情况下，在站号的高位1字节中设置网络No.。

〈其它站指定时的站号设置〉



- *4 访问网络No. 中指定的网络的指定管理站或主站 (站号0)。
访问当前管理站 (实际作为管理站动作的站) 及主站动作站 (使用副主站功能时作为主站动作的站) 的情况下，应指定站号进行访问。
*5 站号65 (41H) ~ 120 (78H) 的指定在使用CC-Link IE控制网络时可以指定。

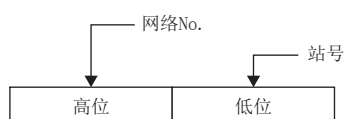
CC-Link IE控制网络板的情况下

指定方法		站号	
		高位字节	低位字节
本站		255 (FFH)	
其它站	站号	网络No. *2 1 (01H) ~ 239 (EFH)	0 (00H) *3
			1 (01H) ~ 120 (78H)
			125 (7DH) *3
	组No. 1 ~ 32 *1		129 (81H) ~ 160 (A0H)
	全站 *1		240 (F0H)
通过实用程序设置的逻辑站号		0 (00H) ~ 239 (EFH)	

*1 只有在使用无到达确认的SEND功能(mdSend函数)时启用全站指定、组No. 指定。

*2 指定其它站的情况下，在站号的高位1字节中设置网络No.。

〈其它站指定时的站号设置〉



*3 访问网络No. 中指定的网络的指定管理站或主站(站号0)。

访问当前管理站(实际作为管理站动作的站)及主站动作站(使用副主站功能时作为主站动作的站)的情况下，应指定站号进行访问。

CC-Link IE现场网络板的情况下

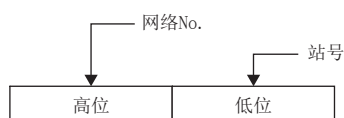
指定方法		站号	
		高位字节	低位字节
本站		255 (FFH)	
其它站	站号	网络No. *3 1 (01H) ~ 239 (EFH)	0 (00H) *4 ~ 120 (78H)
			125 (7DH) *4
			129 (81H) ~ 160 (A0H)
	组No. 1 ~ 32 *1、*2		240 (F0H)
	全站 *1		
通过实用程序设置的逻辑站号		0 (00H) ~ 239 (EFH)	

*1 只有在使用无到达确认的SEND功能(mdSend函数)时启用全站指定、组No. 指定。

*2 组No. 的指定在使用MELSECNET/H网络、CC-Link IE控制网络时可以指定。

*3 指定其它站的情况下，在站号的高位1字节中设置网络No.。

〈其它站指定时的站号设置〉



*4 访问网络No. 中指定的网络的指定管理站或主站(站号0)。

访问当前管理站(实际作为管理站动作的站)及主站动作站(使用副主站功能时作为主站动作的站)的情况下，应指定站号进行访问。

扩展函数的网络No. 及站号指定

MELSEC数据链接库的扩展函数中使用的网络No. 及站号如下所示。

关于扩展函数以外的网络No. 及站号的指定，请参阅下述章节。

(☞ 47页 站号指定)

CC-Link Ver. 2板的情况下

指定方法	网络No.	站号
本站	0 (00H)	255 (FFH)
其它站		0 (00H) ~ 63 (3FH) *1
通过实用程序设置的逻辑站号		65 (41H) ~ 239 (EFH)

- *1 在CC-Link Ver. 2板中，不能进行站号64的指定。
本站的站号为64的情况下，不能访问其它站。只能访问本站。

MELSECNET/H板的情况下

指定方法	网络No.	站号
本站	0 (00H)	255 (FFH)
其它站 站号	1 (01H) ~ 239 (EFH)	0 (00H) *1
		1 (01H) ~ 120 (78H) *2
		125 (7DH) *1
通过实用程序设置的逻辑站号	0 (00H)	65 (41H) ~ 239 (EFH)

- *1 访问网络No. 中指定的网络的指定管理站或主站(站号0)。
访问当前管理站(实际作为管理站动作的站)及主站动作站(使用副主站功能时作为主站动作的站)的情况下，应指定站号进行访问。
- *2 站号65(41H) ~ 120(78H)的指定在使用CC-Link IE控制网络时可以指定。

CC-Link IE控制网络板的情况下

指定方法		网络No.	站号
本站		0 (00H)	255 (FFH)
其它站	站号	1 (01H) ~ 239 (EFH)	0 (00H)*2
			1 (01H) ~ 120 (78H)
			125 (7DH)*2
	组No. 1~32*1		129 (81H) ~ 160 (A0H)
	全站*1		240 (F0H)
通过实用程序设置的逻辑站号		0 (00H)	0 (00H) ~ 239 (EFH)

*1 只有在使用无到达确认的SEND功能 (mdSendEx函数) 时启用全站指定、组No. 指定。

*2 访问网络No. 中指定的网络的指定管理站或主站 (站号0)。

访问当前管理站 (实际作为管理站动作的站) 及主站动作站 (使用副主站功能时作为主站动作的站) 的情况下, 应指定站号进行访问。

CC-Link IE现场网络板的情况下

指定方法		网络No.	站号
本站		0 (00H)	255 (FFH)
其它站	站号	1 (01H) ~ 239 (EFH)	0 (00H)*3 ~ 120 (78H)
			125 (7DH)*3
			129 (81H) ~ 160 (A0H)
	组No. 1~32*1、*2		240 (F0H)
	全站*1		
通过实用程序设置的逻辑站号		0 (00H)	0 (00H) ~ 239 (EFH)

*1 只有在使用无到达确认的SEND功能 (mdSendEx函数) 时启用全站指定、组No. 指定。

*2 组No. 的指定在使用MELSECNET/H网络、CC-Link IE控制网络时可以指定。

*3 访问网络No. 中指定的网络的指定管理站或主站 (站号0)。

访问当前管理站 (实际作为管理站动作的站) 及主站动作站 (使用副主站功能时作为主站动作的站) 的情况下, 应指定站号进行访问。

软元件类型指定

MELSEC数据链接库及软元件监视实用程序中使用的软元件类型如下所示。

在MELSEC数据链接库的函数中，应通过代码指定或软元件名指定指定软元件类型。

在软元件监视实用程序中，应设置表中的“软元件监视时的软元件类型”中记载的软元件类型。

每软元件1点的数据容量

以软元件类型指定的软元件根据每软元件1点中可处理的数据容量有位、字及双字的类型。

各软元件的类型中可处理的每软元件1点的数据容量如下所示。

类型	数据容量	
	位	字节
位	1	—
字	16	2
双字	32	4

本站的链接软元件、缓冲存储器

软元件			函数中指定的软元件类型			软元件监视时的软元件类型
软元件名	符号	类型	代码指定		软元件名指定	
			10进制数	16进制数		
链接输入	LX	位	1	0001H	DevX	X
远程输入	RX	位				
链接输出	LY	位	2	0002H	DevY	Y
远程输出	RY	位				
特殊继电器	SM	位	5	0005H	DevSM	SM
链接特殊继电器	SB	位				
特殊寄存器	SD	字	14	000EH	DevSD	SD
链接特殊寄存器	SW	字				
链接继电器	LB	位	23	0017H	DevB	B
链接寄存器	LW	字	24	0018H	DevW	W
远程寄存器	RW	字				
远程寄存器(发送用)	RWw	字	36	0024H	DevWw	Ww
远程寄存器(接收用)	RWr	字	37	0025H	DevWr	Wr
缓冲存储器	—	字	50	0032H	DevSPB	SPB
随机访问缓冲	—	字	33	0021H	DevMRB	MRB

其它站模块的软元件

软元件名		符号	类型	函数中指定的软元件类型		软元件名指定	软元件监视时的软元件类型
				代码指定			
				10进制数	16进制数		
输入		X	位	1	0001H	DevX	X
输出		Y	位	2	0002H	DevY	Y
锁存继电器		L	位	3	0003H	DevL	L
内部继电器		M	位	4	0004H	DevM	M
特殊继电器		SM	位	5	0005H	DevSM	SM
报警器		F	位	6	0006H	DevF	F
定时器	触点	T	位	7	0007H	DevTT	TT
	线圈		位	8	0008H	DevTC	TC
计数器	触点	C	位	9	0009H	DevCT	CT
	线圈		位	10	000AH	DevCC	CC
定时器	当前值	T	字	11	000BH	DevTN	TN
计数器	当前值	C	字	12	000CH	DevCN	CN
数据寄存器		D	字	13	000DH	DevD	D
特殊寄存器		SD	字	14	000EH	DevSD	SD
定时器	设置值主	T	字	15	000FH	DevTM	TM
	设置值子1		字	16	0010H	DevTS	TS
	设置值子2		字	16002	3E82H	DevTS2	TS2
	设置值子3		字	16003	3E83H	DevTS3	TS3
计数器	设置值主	C	字	17	0011H	DevCM	CM
	设置值子1		字	18	0012H	DevCS	CS
	设置值子2		字	18002	4652H	DevCS2	CS2
	设置值子3		字	18003	4653H	DevCS3	CS3
累加器		A	字	19	0013H	DevA	A
变址寄存器		Z	字	20	0014H	DevZ	Z
		V	字	21	0015H	DevV	V
文件寄存器 扩展文件寄存器 *1	块切换方式	R	字	22	0016H	DevR	R
			字	22000~22256*2	55F0H~56F0H*2	DevER0~256*2	ER
		ZR	字	220	00DCH	DevZR	ZR
链接继电器		B	位	23	0017H	DevB	B
链接寄存器		W	字	24	0018H	DevW	W
链接特殊继电器		SB	位	25	0019H	DevQSB	QSB
累计定时器	触点	ST	位	26	001AH	DevSTT	STT
	线圈		位	27	001BH	DevSTC	STC
链接特殊寄存器		SW	字	28	001CH	DevQSW	QSW
变址继电器		V	位	30	001EH	DevQV	QV
累计定时器	当前值	ST	字	35	0023H	DevSTN	STN
超长变址寄存器		LZ	双字	38	0026H	DevLZ	LZ
刷新数据寄存器		RD	字	39	0027H	DevRD	RD
超长定时器	触点	LT	位	41	0029H	DevLTT	LTT
	线圈		位	42	002AH	DevLTC	LTC
	当前值		双字	43	002BH	DevLTN	LTN
超长计数器	触点	LC	位	44	002CH	DevLCT	LCT
	线圈		位	45	002DH	DevLCC	LCC
	当前值		双字	46	002EH	DevLCN	LCN
超长累计定时器	触点	LST	位	47	002FH	DevLSTT	LSTT
	线圈		位	48	0030H	DevLSTC	LSTC
	当前值		双字	49	0031H	DevLSTN	LSTN

软元件				函数中指定的软元件类型			软元件监视时的软元件类型
软元件名	符号	类型	代码指定		软元件名指定		
			10进制数	16进制数			
链接直接软元件 *1、*3	链接输入	Jn\X	位	1001~1255	03E9H~04E7H	DevLX1~255	LX
	链接输出	Jn\Y	位	2001~2255	07D1H~08CFH	DevLY1~255	LY
	链接继电器	Jn\B	位	23001~23255	59D9H~5AD7H	DevLB1~255	LB
	链接寄存器	Jn\W	字	24001~24255	5DC1H~5EBFH	DevLW1~255	LW
	链接特殊继电器	Jn\SB	位	25001~25255	61A9H~62A7H	DevLSB1~255	LSB
	链接特殊寄存器	Jn\SW	字	28001~28255	6D61H~6E5FH	DevLSW1~255	LSW
智能功能模块软元件 缓冲存储器 模块访问软元件 *1、*4	Un\G	字	29000~29255	7148H~7247H	DevSPG0~255	SPG	

- *1 在随机读取(mdRandR、mdRandREx)函数中,即使指定实际不存在的软元件也有可能正常结束。(读取数据不正确。)
- *2 访问软元件类型中指定的块的文件寄存器。代码指定(10进制数)的后3位数及软元件名指定的数值中,应指定块No.(0~256)。
- *3 代码指定(10进制数)的后3位数及软元件名指定的数值中,应指定网络No.(1~255)。
- *4 代码指定(10进制数)的后3位数及软元件名指定的数值中,应指定(起始I/ONo./16)的值。

CC-Link的其它站缓冲存储器

通过CC-Link Ver.2板访问CC-Link的网络模块及其它CC-Link Ver.2板的缓冲存储器的情况下,应指定下述软元件类型。

软元件			函数中指定的软元件类型			软元件监视时的软元件类型
软元件名	符号	类型	代码指定		软元件名指定	
			10进制数	16进制数		
缓冲存储器*1	—	字	-32768	8000H	DevRBM	RBM
随机访问缓冲*1	—	字	-32736	8020H	DevRAB	RAB
远程输入*1	RX	位	-32735	8021H	DevRX	RX
远程输出*1	RY	位	-32734	8022H	DevRY	RY
远程寄存器*1	RWw、RWr	字	-32732	8024H	DevRW	RW
链接特殊继电器*1	SB	位	-32669	8063H	DevSB	SB
链接特殊寄存器*1	SW	字	-32668	8064H	DevSW	SW

- *1 该软元件在随机写入(mdRandW、mdRandWEx)、随机读取(mdRandR、mdRandREx)、位设置(mdDevSet、mdDevSetEx)、位复位(mdDevRst、mdDevRstEx)中不能使用。

SEND功能、RECV功能

在SEND功能及RECV功能中,与软元件访问一样,指定各功能对应的软元件类型,进行数据的发送(mdSend、mdSendEx)或数据的读取(mdReceive、mdReceiveEx)。

此外,本软元件类型是仅在SEND功能及RECV功能中可使用的特殊软元件类型,因此不能在软元件监视实用程序中指定。

功能	函数中指定的软元件类型		
	代码指定		软元件名指定
	10进制数	16进制数	
RECV功能	101	0065H	DevMAIL
SEND功能(有到达确认)			
SEND功能(无到达确认)	102	0066H	DevMAILNC

4.3 函数详细内容

MELSEC数据链接库的详细规格如下所示。
在本节中，以下述配置介绍各指令。

格式

各编程语言的记述方法如下所示。

■Visual C++

是Microsoft® Visual C++中的格式。

■Visual Basic .NET

是下述Visual Basic中的格式。

- Visual Studio .NET 2003 Visual Basic
- Visual Studio 2005 Visual Basic
- Visual Studio 2008 Visual Basic
- Visual Studio 2010 Visual Basic
- Visual Studio 2012 Visual Basic
- Visual Studio 2013 Visual Basic
- Visual Studio 2015 Visual Basic
- Visual Studio 2017 Visual Basic

■Visual Basic 5.0、Visual Basic 6.0

是下述Visual Basic中的格式。

- Visual Basic 5.0
- Visual Basic 6.0

详细规格

■参数

介绍函数的各参数有关内容。

■说明

介绍函数的功能及参数相关的详细内容。

■返回值

介绍返回值有关内容。

■关联函数

介绍编程时关联的函数。

mdOpen(通信线路的打开)

指定通信线路的通道No.，打开通信线路。

格式

■Visual C++

```
格式      ret = mdOpen(chan, mode, path);
参数      short      ret;          //返回值          OUT
          short      chan;       //通信线路的通道No.  IN
          short      mode;       //虚拟          IN
          long       *path;      //打开的线路的路径的指针  OUT
```

■Visual Basic .NET

```
格式      ret = mdOpen(chan, mode, path)
参数      Short      ret          ;返回值          OUT
          Short      chan        ;通信线路的通道No.  IN
          Short      mode        ;虚拟          IN
          Integer    path        ;打开的线路的路径的指针  OUT
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdOpen(chan%, mode%, path%)
参数      Integer    ret          ;返回值          OUT
          Integer    chan        ;通信线路的通道No.  IN
          Integer    mode        ;虚拟          IN
          Long       path        ;打开的线路的路径的指针  OUT
```

详细规格

■参数

参数	说明
chan	指定通信线路的通道No.。(☞ 46页 通道No. 指定)
mode	指定-1。
path	返回打开的线路的路径。

■说明

- 其它函数的参数中使用的通道的路径返回至打开的线路的路径的指针。
- 使用多个通信线路的情况下，需要对各通道No. 打开通信线路。

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdClose()

mdClose(通信线路的关闭)

指定通信线路的路径，关闭通信线路。

格式

■Visual C++

```
格式      ret = mdClose(path);
参数      short      ret;          //返回值          OUT
          long      path;        //通道的路径      IN
```

■Visual Basic .NET

```
格式      ret = mdClose(path)
参数      Short      ret          ;返回值          OUT
          Integer    path        ;通道的路径      IN
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdClose(path%)
参数      Integer    ret          ;返回值          OUT
          Long       path        ;通道的路径      IN
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)

■说明

使用多个通信线路的情况下，需要对各通道No. 关闭通信线路。

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(127页 出错代码)

■关联函数

mdOpen()

mdSendEx (扩展软元件批量写入)

对对象站，从起始软元件No. 开始将数据批量写入到相当于写入字节容量的软元件中。

格式

■Visual C++

格式	ret = mdSendEx (path, netno, stno, devtyp, devno, size, data);		
参数	long	ret;	//返回值 OUT
	long	path;	//通道的路径 IN
	long	netno;	//网络No. IN
	long	stno;	//站号 IN
	long	devtyp;	//软元件类型 IN
	long	devno;	//起始软元件No. IN
	long	*size;	//写入字节容量 IN/OUT
	short	data [];	//写入数据(单精度整数数组) IN

■Visual Basic .NET

格式	ret = mdSendEx (path, netno, stno, devtyp, devno, size, data(0))		
参数	Integer	ret	;返回值 OUT
	Integer	path	;通道的路径 IN
	Integer	netno	;网络No. IN
	Integer	stno	;站号 IN
	Integer	devtyp	;软元件类型 IN
	Integer	devno	起始软元件No. IN
	Integer	size	;写入字节容量 IN/OUT
	Short	data (n)	;写入数据(单精度整数数组) IN

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdSendEx (path&, netno%, stno%, devtyp%, devno%, size%, data%(0))		
参数	Long	ret	;返回值 OUT
	Long	path	;通道的路径 IN
	Long	netno	;网络No. IN
	Long	stno	;站号 IN
	Long	devtyp	;软元件类型 IN
	Long	devno	;起始软元件No. IN
	Long	size	;写入字节容量 IN/OUT
	Any	data (n)	;写入数据(单精度整数数组) IN

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞ 49页 扩展函数的网络No. 及站号指定)
stno	指定对象站的站号。(☞ 49页 扩展函数的网络No. 及站号指定)
devtyp	指定写入数据的软元件的软元件类型。(☞ 51页 软元件类型指定)
devno	指定写入数据的软元件的起始软元件No.。 <ul style="list-style-type: none"> 对于至位软元件的访问，应以8的倍数指定起始软元件No.。(但是，超长定时器及超长累计定时器的情况下，至触点、线圈的访问也可指定任意的起始软元件No.。) 对于至CC-Link其它站链接软元件的位软元件(RX、RY、SB)的访问，应以16的倍数(0、16、32...)指定起始软元件No.。
size	以偶数指定写入数据的字节容量。 <ul style="list-style-type: none"> 指定的写入字节容量超出软元件范围(-5: 容量出错)的情况下，size中将返回可写入容量。 devtyp中指定了双字软元件的情况下，应以4的倍数指定。
data	以单精度整数数组指定写入数据。

■说明

- 指定的写入字节容量超过瞬时传送最大发送容量的情况下，在函数内部分割执行。
- 其它站访问中，对分配了扩展注释的块(扩展文件寄存器)进行写入时，扩展注释信息将被清除。
- 其它站访问中，对与子2、子3程序设置区域重叠的块(扩展文件寄存器)进行写入时，子2、子3程序将被清除。
- devtyp中指定了双字软元件的情况下，应按下述方式将写入数据存储到data中。

例

devtyp为LZ，size为8的情况下

数组	值
data[0]	LZ0的低位字
data[1]	LZ0的高位字
data[2]	LZ1的低位字
data[3]	LZ1的高位字

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdSend()、mdReceive()、mdReceiveEx()

mdSendEx (SEND功能)

将数据发送至指定的对象站的通道No.。

格式

■Visual C++

格式	ret = mdSendEx (path, netno, stno, devtyp, devno, size, data);		
参数	long	ret;	//返回值 OUT
	long	path;	//通道的路径 IN
	long	netno;	//网络No. IN
	long	stno;	//站号 IN
	long	devtyp;	//软元件类型 IN
	long	devno;	//通道No. IN
	long	*size;	//发送字节容量 IN/OUT
	short	data [];	//发送数据(单精度整数数组) IN

■Visual Basic .NET

格式	ret = mdSendEx (path, netno, stno, devtyp, devno, size, data(0))		
参数	Integer	ret	;返回值 OUT
	Integer	path	;通道的路径 IN
	Integer	netno	;网络No. IN
	Integer	stno	;站号 IN
	Integer	devtyp	;软元件类型 IN
	Integer	devno	;通道No. IN
	Integer	size	;发送字节容量 IN/OUT
	Short	data (n)	;发送数据(单精度整数数组) IN

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdSendEx (path%, netno%, stno%, devtyp%, devno%, size%, data%(0))		
参数	Long	ret	;返回值 OUT
	Long	path	;通道的路径 IN
	Long	netno	;网络No. IN
	Long	stno	;站号 IN
	Long	devtyp	;软元件类型 IN
	Long	devno	;通道No. IN
	Long	size	;发送字节容量 IN/OUT
	Any	data (n)	;发送数据(单精度整数数组) IN

详细规格

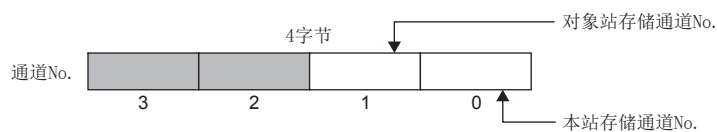
■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞49页 扩展函数的网络No.及站号指定)
stno	指定对象站的站号。(☞49页 扩展函数的网络No.及站号指定) • 不能进行逻辑站号的指定。
devtyp	指定是以“有到达确认”还是“无到达确认”进行数据发送。 • 有到达确认的情况下*1: 指定101(65H) • 无到达确认的情况下: 指定102(66H)
devno	指定本站使用通道及对象站存储通道。(☞60页 说明)
size	以偶数指定发送数据的字节容量。 • 应以2~1920的范围指定。 • 经由MELSECNET/10网络的情况下, 应以2~960的范围指定。
data	以单精度整数数组指定发送的数据。

*1 以有到达确认发送数据的情况下, 站号中请勿指定全站、组No.。

■说明

- 支持链接专用指令的SEND指令。关于功能详细内容，请参阅各板的手册。
- 通道No. 的指定方法如下所示。



项目	内容	
对象站存储通道	指定对象站存储通道No.。 • MELSECNET/H板、CC-Link IE控制网络板 • CC-Link IE现场网络板	— 1~8 1~2
本站使用通道	指定本站使用通道No.。 • CC-Link IE控制网络板 • CC-Link IE现场网络板	— 1~8 1~2

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(P.127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdSend()、mdReceive()、mdReceiveEx()

mdReceiveEx (扩展软元件批量读取)

从对象站中将起始软元件No. 中指定的软元件按读取字节容量进行批量读取。

格式

■Visual C++

格式	ret = mdReceiveEx(path, netno, stno, devtyp, devno, size, data);			
参数	long	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	long	netno;	//网络No.	IN
	long	stno;	//站号	IN
	long	devtyp;	//软元件类型	IN
	long	devno;	//起始软元件No.	IN
	long	*size;	//读取字节容量	IN/OUT
	short	data[];	//读取数据(单精度整数数组)	OUT

■Visual Basic .NET

格式	ret = mdReceiveEx(path, netno, stno, devtyp, devno, size, data(0))			
参数	Integer	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Integer	netno	;网络No.	IN
	Integer	stno	;站号	IN
	Integer	devtyp	;软元件类型	IN
	Integer	devno	;起始软元件No.	IN
	Integer	size	;读取字节容量	IN/OUT
	Short	data(n)	;读取数据(单精度整数数组)	OUT

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdReceiveEx(path&, netno%, stno%, devtyp%, devno%, size%, data%(0))			
参数	Long	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Long	netno	;网络No.	IN
	Long	stno	;站号	IN
	Long	devtyp	;软元件类型	IN
	Long	devno	;起始软元件No.	IN
	Long	size	;读取字节容量	IN/OUT
	Any	data(n)	;读取数据(单精度整数数组)	OUT

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞49页 扩展函数的网络No. 及站号指定)
stno	指定对象站的站号。(☞49页 扩展函数的网络No. 及站号指定)
devtyp	指定读取数据的软元件的软元件类型。(☞51页 软元件类型指定)
devno	指定读取软元件范围的起始软元件No.。 <ul style="list-style-type: none"> 对于至位软元件的访问，应以8的倍数指定起始软元件No.。(但是，超长定时器及超长累计定时器的情况下，至触点、线圈的访问也可指定任意的起始软元件No.。) 对于至CC-Link其它站链接软元件的位软元件(RX、RY、SB)的访问，应以16的倍数(0、16、32...)指定起始软元件No.。
size	以偶数指定读取数据的字节容量。 <ul style="list-style-type: none"> 指定的读取字节容量超出软元件范围(-5: 容量出错)的情况下，size中将返回可读取容量。 devtyp中指定了双字软元件的情况下，应以4的倍数指定。
data	存储读取的数据。

■说明

- 指定的读取字节容量超过瞬时传送最大发送容量的情况下，在函数内部分割执行。
- devtyp中指定了双字节元件的情况下，数据按下述方式存储到data中。

例

devtyp为LZ，size为8的情况下

数组	值
data[0]	LZ0的低位字
data[1]	LZ0的高位字
data[2]	LZ1的低位字
data[3]	LZ1的高位字

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdReceive()、mdSend()、mdSendEx()

mdReceiveEx (RECV功能)

从到达本站的数据中读取指定通道No. 的数据。

格式

■Visual C++

格式	ret = mdReceiveEx(path, netno, stno, devtyp, devno, size, data);			
参数	long	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	long	netno;	//网络No.	IN
	long	stno;	//站号	IN
	long	devtyp;	//软元件类型	IN
	long	devno;	//通道No.	IN
	long	*size;	//接收字节容量	IN/OUT
	short	data[];	//接收数据+发送源信息(单精度整数数组)	OUT

■Visual Basic .NET

格式	ret = mdReceiveEx(path, netno, stno, devtyp, devno, size, data(0))			
参数	Integer	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Integer	netno	;网络No.	IN
	Integer	stno	;站号	IN
	Integer	devtyp	;软元件类型	IN
	Integer	devno	;通道No.	IN
	Integer	size	;接收字节容量	IN/OUT
	Short	data(n)	;接收数据+发送源信息(单精度整数数组)	OUT

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdReceiveEx(path&, netno%, stno%, devtyp%, devno%, size%, data%(0))			
参数	Long	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Long	netno	;网络No.	IN
	Long	stno	;站号	IN
	Long	devtyp	;软元件类型	IN
	Long	devno	;通道No.	IN
	Long	size	;接收字节容量	IN/OUT
	Any	data(n)	;接收数据+发送源信息(单精度整数数组)	OUT

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定0(0H)。
stno	指定本站(255(FFH))。
devtyp	指定RECV功能(101(65H))。
devno	指定存储了接收信息的本站的通道No.。 <ul style="list-style-type: none"> • CC-Link IE控制网络板: 1~8 • CC-Link IE现场网络板: 1~2
size	以偶数指定接收数据的字节容量。 <ul style="list-style-type: none"> • 应以2~1920的范围指定。 • 通过CC-Link IE控制网络板、CC-Link IE现场网络板接收从MELSECNET/H板发送的数据的情况下, 应以2~960字节的范围指定。 返回接收的实际数据的容量。
data	以单精度整数数组存储接收的实际数据及发送源信息。 应确保相当于指定的接收字节容量+发送源信息(6字节)的区域。

■说明

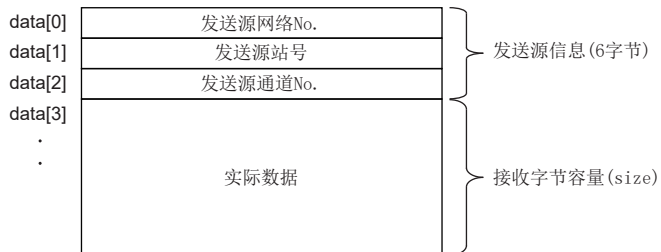
支持链接专用指令的RECV指令。

关于功能详细内容，请参阅各板的手册。

<关于接收数据+发送源信息(data)>

- 接收数据+发送源信息(data)中，存储在接收字节容量(size)中附加了发送源信息(6字节)的数据。接收数据+发送源信息(data)中，应确保相当于[接收字节容量(size)+发送源信息(6字节)]的区域。
- 接收数据+发送源信息(data)按下述方式存储。

接收数据+发送源信息(data)



- 接收的实际数据的容量大于指定的接收字节容量(size)的情况下，只存储相当于指定的接收字节容量(size)的实际数据。
- 发送源的站号为0的情况下，发送源站号中将存储125(7DH)。

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdReceive()、mdSend()、mdSendEx()

mdRandWEx (扩展软元件随机写入)

将数据写入到随机指定软元件中指定的对象站的软元件中。

格式

■Visual C++

格式	ret = mdRandWEx (path, netno, stno, dev, buf, bufsize);			
参数	long	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	long	netno;	//网络No.	IN
	long	stno;	//站号	IN
	long	dev[];	//随机指定软元件	IN
	short	buf[];	//写入数据(单精度整数数组)	IN
	long	bufsize;	//虚拟	IN

■Visual Basic .NET

格式	ret = mdRandWEx (path, netno, stno, dev (0), buf (0), bufsize)			
参数	Integer	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Integer	netno	;网络No.	IN
	Integer	stno	;站号	IN
	Integer	dev (n)	;随机指定软元件	IN
	Short	buf (n)	;写入数据(单精度整数数组)	IN
	Integer	bufsize	;虚拟	IN

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdRandWEx (path%, netno%, stno%, dev% (0), buf% (0), bufsize%)			
参数	Long	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Long	netno	;网络No.	IN
	Long	stno	;站号	IN
	Long	dev (n)	;随机指定软元件	IN
	Any	buf (n)	;写入数据(单精度整数数组)	IN
	Long	bufsize	;虚拟	IN

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞ 49页 扩展函数的网络No.及站号指定)
stno	指定对象站的站号。(☞ 49页 扩展函数的网络No.及站号指定)
dev	指定写入软元件的块数、软元件类型、起始软元件No.、点数。(☞ 67页 随机指定软元件(dev)的指定方法)
buf	指定buf[0]及以后的数组变量中写入的数据。(☞ 67页 随机指定软元件(dev)的指定方法)
bufsize	—

■说明

- 根据指定的随机指定软元件，函数内部进行的瞬时通信次数将变化。
- 随机写入与批量写入相比函数执行时间将变长。希望缩短函数执行时间的情况下，应讨论批量写入的使用。
- 其它站访问中，对分配了扩展注释的块(扩展文件寄存器)进行写入时，扩展注释信息将被清除。
- 其它站访问中，对与子2、子3程序设置区域重叠的块(扩展文件寄存器)进行写入时，子2、子3程序将被清除。
- 至MELSECNET (II)、MELSECNET/10的本站的B、W的随机写入时发生了出错的情况下，正确完成写入的块与发生出错的块有可能混在一起。
- 根据随机指定软元件的点数指定，写入的数据的容量对各软元件的类型有所不同。(☞ 51页 每软元件1点的数据容量)

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdRandREx()

随机指定软元件(dev)的指定方法

随机指定软元件(dev)的指定方法如下所示。

随机指定软元件	说明	
dev[0]	指定块数。(1~32767)	
dev[1]	软元件类型	指定块1的软元件。
dev[2]	起始软元件No.	
dev[3]	点数	
dev[4]	软元件类型	指定块2的软元件。
dev[5]	起始软元件No.	
dev[6]	点数	
:	(指定相当于指定的块数的软元件。)	

■写入多个软元件中的情况下

软元件中写入下述值情况下的示例如下所示。

块	指定软元件	软元件值
块1	M100~M115(位软元件16点)	全部OFF(0000H)
块2	D10~D13(字软元件4点)	D10: 10, D11: 200, D12: 300, D13: 400

• 随机指定软元件(dev)

程序示例	内容
dev[0]=2;	块数: 2
dev[1]=DevM; dev[2]=100; dev[3]=16;	块1 • 起始软元件: M100 • 点数: 16点
dev[4]=DevD; dev[5]=10; dev[6]=4;	块2 • 起始软元件: D10 • 点数: 4点

• 写入数据(buf)

数组	值	软元件
buf[0]	0	M100~M115
buf[1]	10	D10
buf[2]	200	D11
buf[3]	300	D12
buf[4]	400	D13

■写入双字软元件中的情况下

软元件中写入下述值情况下的示例如下所示。

块	指定软元件	软元件值
块1	LC100~LC101(双字软元件2点)	LC100(当前值)=1, LC101(当前值)=65536(10000H)

• 随机指定软元件(dev)

程序示例	内容
dev[0]=1;	块数: 1
dev[1]=DevLCN; dev[2]=100; dev[3]=2;	块1 • 起始软元件: LC100(当前值) • 点数: 2点

• 写入数据(buf)

数组	值	软元件
buf[0]	1	LC100(当前值)的低位字
buf[1]	0	LC100(当前值)的高位字
buf[2]	0	LC101(当前值)的低位字
buf[3]	1	LC101(当前值)的高位字

mdRandREx (扩展软元件随机读取)

从对象站中读取随机指定软元件中指定的软元件。

格式

Visual C++

格式	ret = mdRandREx (path, netno, stno, dev, buf, bufsize);			
参数	long	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	long	netno;	//网络No.	IN
	long	stno;	//站号	IN
	long	dev[];	//随机指定软元件	IN
	short	buf[];	//读取数据(单精度整数数组)	OUT
	long	bufsize;	//读取数据的字节数	IN

Visual Basic .NET

格式	ret = mdRandREx (path, netno, stno, dev (0), buf (0), bufsize)			
参数	Integer	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Integer	netno	;网络No.	IN
	Integer	stno	;站号	IN
	Integer	dev (n)	;随机指定软元件	IN
	Short	buf (n)	;读取数据(单精度整数数组)	OUT
	Integer	bufsize	;读取数据的字节数	IN

Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdRandREx (path%, netno%, stno%, dev%(0), buf%(0), bufsize%)			
参数	Long	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Long	netno	;网络No.	IN
	Long	stno	;站号	IN
	Long	dev (n)	;随机指定软元件	IN
	Any	buf (n)	;读取数据(单精度整数数组)	OUT
	Long	bufsize	;读取数据的字节数	IN

详细规格

参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞ 49页 扩展函数的网络No.及站号指定)
stno	指定对象站的站号。(☞ 49页 扩展函数的网络No.及站号指定)
dev	指定读取软元件的块数、软元件类型、起始软元件No.、点数。(☞ 69页 随机指定软元件(dev)的指定方法)
buf	读取数据被存储到buf[0]及以后的数组变量中。(☞ 69页 随机指定软元件(dev)的指定方法)
bufsize	指定读取数据的字节数。(☞ 69页 随机指定软元件(dev)的指定方法)

说明

- 根据指定的随机指定软元件，函数内部进行的瞬时通信次数将变化。
- 随机读取与批量读取相比函数执行时间将变长。希望缩短函数执行时间的情况下，应讨论批量读取的使用。
- 根据随机指定软元件的点数指定，读取的数据的容量对各软元件的类型有所不同。(☞ 51页 每软元件1点的数据容量)

返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

关联函数

mdOpen ()、mdClose ()、mdRandWEx ()

随机指定软元件(dev)的指定方法

随机指定软元件(dev)的指定方法如下所示。

随机指定软元件	说明	
dev[0]	指定块数。(1~32767)	
dev[1]	软元件类型	指定块1的软元件。
dev[2]	起始软元件No.	
dev[3]	点数	
dev[4]	软元件类型	指定块2的软元件。
dev[5]	起始软元件No.	
dev[6]	点数	
:	(指定相当于指定的块数的软元件。)	

■从多个软元件中读取的情况下

从软元件中读取下述值情况下的示例如下所示。

块	指定软元件	软元件值
块1	M100~M115(位软元件16点)	全部OFF(0000H)
块2	D10~D13(字软元件4点)	D10: 10, D11: 200, D12: 300, D13: 400
块3	M0~M13(位软元件14点)	全部ON(3FFFH)
块4	T10(当前值)(字软元件1点)	T10(当前值): 10

• 随机指定软元件(dev)

程序示例	内容
dev[0]=4;	块数: 4
dev[1]=DevM; dev[2]=100; dev[3]=16;	块1 • 起始软元件: M100 • 点数: 16点
dev[4]=DevD; dev[5]=10; dev[6]=4;	块2 • 起始软元件: D10 • 点数: 4点
dev[7]=DevM; dev[8]=0; dev[9]=14;	块3 • 起始软元件: M0 • 点数: 14点
dev[10]=DevTN; dev[11]=10; dev[12]=1;	块4 • 起始软元件: T10(当前值) • 点数: 1点

• 读取数据(buf)

数组	值	软元件
buf[0]	0	M100~M115
buf[1]	10	D10
buf[2]	200	D11
buf[3]	300	D12
buf[4]	400	D13
buf[5]	16383(3FFFH)	M0~M13
buf[6]	10	T10(当前值)

• 读取数据的字节数(bufsize)

指定存储读取数据的数组变量buf的字节数。

$(buf[0] \sim buf[6] = 7) \times 2 = 14$ 字节

在本例中指定“14”。

■从双字软元件中读取的情况下

从软元件中读取下述值情况下的示例如下所示。

块	指定软元件	软元件值
块1	LC100~LC101(双字软元件2点)	LC100(当前值)=1, LC101(当前值)=65536(10000H)

- 随机指定软元件(dev)

程序示例	内容
dev[0]=1;	块数: 1
dev[1]=DevLCN; dev[2]=100; dev[3]=2;	块1 • 起始软元件: LC100(当前值) • 点数: 2点

- 读取数据(buf)

数组	值	软元件
buf[0]	1	LC100(当前值)的低位字
buf[1]	0	LC100(当前值)的高位字
buf[2]	0	LC101(当前值)的低位字
buf[3]	1	LC101(当前值)的高位字

- 读取数据的字节数(bufsize)

指定存储读取数据的数组变量buf的字节数。

$(buf[0] \sim buf[3] = 4) \times 2 = 8$ 字节

在本例中指定“8”。

mdDevSetEx (扩展位软元件设置)

设置 (ON) 对象站的指定位软元件。

格式

■ Visual C++

```
格式      ret = mdDevSetEx(path, netno, stno, devtyp, devno);
参数      long      ret;          //返回值          OUT
          long      path;       //通道的路径    IN
          long      netno;      //网络No.      IN
          long      stno;       //站号         IN
          long      devtyp;     //软元件类型   IN
          long      devno;      //指定软元件No. IN
```

■ Visual Basic .NET

```
格式      ret = mdDevSetEx(path, netno, stno, devtyp, devno)
参数      Integer   ret        ;返回值          OUT
          Integer   path       ;通道的路径    IN
          Integer   netno      ;网络No.      IN
          Integer   stno       ;站号         IN
          Integer   devtyp     ;软元件类型   IN
          Integer   devno      ;指定软元件No. IN
```

■ Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdDevSetEx(path%, netno%, stno%, devtyp%, devno%)
参数      Long      ret        ;返回值          OUT
          Long      path       ;通道的路径    IN
          Long      netno      ;网络No.      IN
          Long      stno       ;站号         IN
          Long      devtyp     ;软元件类型   IN
          Long      devno      ;指定软元件No. IN
```

详细规格

■ 参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞ 49页 扩展函数的网络No.及站号指定)
stno	指定对象站的站号。(☞ 49页 扩展函数的网络No.及站号指定)
devtyp	指定设置(ON)的软元件的软元件类型。(☞ 51页 软元件类型指定)
devno	指定设置(ON)的软元件的软元件No.。

■ 说明

本函数是链接继电器(B)、内部继电器(M)等的位软元件专用的函数。

■ 返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■ 关联函数

mdOpen()、mdClose()、mdDevRstEx()

mdDevRstEx (扩展位软元件复位)

复位 (OFF) 对象站的指定位软元件。

格式

■ Visual C++

格式	ret = mdDevRstEx (path, netno, stno, devtyp, devno);			
参数	long	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	long	netno;	//网络No.	IN
	long	stno;	//站号	IN
	long	devtyp;	//软元件类型	IN
	long	devno;	//指定软元件No.	IN

■ Visual Basic .NET

格式	ret = mdDevRstEx (path, netno, stno, devtyp, devno)			
参数	Integer	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Integer	netno	;网络No.	IN
	Integer	stno	;站号	IN
	Integer	devtyp	;软元件类型	IN
	Integer	devno	;指定软元件No.	IN

■ Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdDevRstEx (path%, netno%, stno%, devtyp%, devno%)			
参数	Long	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Long	netno	;网络No.	IN
	Long	stno	;站号	IN
	Long	devtyp	;软元件类型	IN
	Long	devno	;指定软元件No.	IN

详细规格

■ 参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞ 49页 扩展函数的网络No.及站号指定)
stno	指定对象站的站号。(☞ 49页 扩展函数的网络No.及站号指定)
devtyp	指定复位(OFF)的软元件的软元件类型。(☞ 51页 软元件类型指定)
devno	指定复位(OFF)的软元件的软元件No.。

■ 说明

本函数是链接继电器(B)、内部继电器(M)等的位软元件专用的函数。

■ 返回值

正常结束时: 返回0。

异常结束时: 返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■ 关联函数

mdOpen()、mdClose()、mdDevSetEx()

mdRemBufWriteEx (远程设备站的缓冲存储器写入)

将数据写入到对象站(CC-Link IE现场网络的远程设备站)的缓冲存储器中。

格式

■Visual C++

格式	ret = mdRemBufWriteEx(path, netno, stno, offset, size, data);			
参数	long	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	long	netno;	//网络No.	IN
	long	stno;	//站号	IN
	long	offset;	//偏置	IN
	long	*size;	//写入字节容量	IN
	short	data [];	//写入数据(单精度整数数组)	IN

■Visual Basic .NET

格式	ret = mdRemBufWriteEx(path, netno, stno, offset, size, data(0))			
参数	Integer	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Integer	netno	;网络No.	IN
	Integer	stno	;站号	IN
	Integer	offset	;偏置	IN
	Integer	size	;写入字节容量	IN
	Short	data(n)	;写入数据(单精度整数数组)	IN

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞ 49页 扩展函数的网络No.及站号指定)
stno	指定对象站的站号。(☞ 49页 扩展函数的网络No.及站号指定)
offset	指定写入数据的远程设备站的缓冲存储器的起始地址。
size	以偶数(2~480)指定写入数据的字节容量。
data	以单精度整数数组指定写入数据。

■说明

- 本函数仅将数据写入到CC-Link IE现场网络的远程设备站的缓冲存储器中。无法将数据写入到CC-Link IE现场网络的起始模块管理的智能功能模块的缓冲存储器中。将数据写入到CC-Link IE现场网络的起始模块管理的智能功能模块的缓冲存储器中的情况下,应使用mdSendEx/mdSend函数。
- 应避免从偏置开始的写入字节容量超出远程设备站的缓冲存储器范围。超出缓冲存储器范围时有可能发生超时出错及远程设备站有可能动作不正常。关于远程设备站的缓冲存储器的范围,请参阅各远程设备站的手册。
- 64位版用户应用程序的情况下,应使用Version1.12N及以上的SWIDNC-CCIEF-B。

■返回值

正常结束时: 返回0。

异常结束时: 返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdRemBufReadEx()

mdRemBufReadEx (远程设备站的缓冲存储器读取)

从对象站(CC-Link IE现场网络的远程设备站)的缓冲存储器中读取数据。

格式

Visual C++

```

格式      ret = mdRemBufReadEx (path, netno, stno, offset, size, data);
参数      long      ret;           //返回值           OUT
          long      path;       //通道的路径       IN
          long      netno;      //网络No.          IN
          long      stno;       //站号             IN
          long      offset;     //偏置             IN
          long      *size;      //读取字节容量     IN
          short     data [ n];  //读取数据(单精度整数数组) OUT

```

Visual Basic .NET

```

格式      ret = mdRemBufReadEx (path, netno, stno, offset, size, data(0))
参数      Integer    ret           ;返回值           OUT
          Integer    path       ;通道的路径       IN
          Integer    netno      ;网络No.          IN
          Integer    stno       ;站号             IN
          Integer    offset     ;偏置             IN
          Integer    size       ;读取字节容量     IN
          Short     data (n)    ;读取数据(单精度整数数组) OUT

```

详细规格

参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
netno	指定对象站的网络No.。(☞ 49页 扩展函数的网络No.及站号指定)
stno	指定对象站的站号。(☞ 49页 扩展函数的网络No.及站号指定)
offset	指定读取数据的远程设备站的缓冲存储器的起始地址。
size	以偶数(2~480)指定读取数据的字节容量。
data	以单精度整数数组指定读取数据。

说明

- 本函数只能从CC-Link IE现场网络的远程设备站的缓冲存储器中读取数据。不能从CC-Link IE现场网络的起始模块管理的智能功能模块的缓冲存储器中读取数据。从CC-Link IE现场网络的起始模块管理的智能功能模块的缓冲存储器中读取数据的情况下，应使用mdReceiveEx/mdReceive函数。
- 应避免从偏置开始的读取字节容量超出远程设备站的缓冲存储器范围。超出缓冲存储器范围时，有可能发生超时出错及有可能读取不定值。关于远程设备站的缓冲存储器的范围，请参阅各远程设备站的手册。
- 64位版用户应用程序的情况下，应使用Version1.12N及以上的SW1DNC-CCIEF-B。

返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

关联函数

mdOpen()、mdClose()、mdRemBufWriteEx()

mdTypeRead(CPU型号读取)

读取对象站的CPU的型号代码。

格式

■Visual C++

```
格式      ret = mdTypeRead(path, stno, buf);
参数      short      ret;          //返回值          OUT
          long      path;        //通道的路径    IN
          short     stno;        //站号          IN
          short     *buf;        //型号代码      OUT
```

■Visual Basic .NET

```
格式      ret = mdTypeRead(path, stno, buf)
参数      Short      ret          ;返回值          OUT
          Integer    path        ;通道的路径    IN
          Short     stno        ;站号          IN
          Short     buf         ;型号代码      OUT
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdTypeRead(path&, stno%, buf%)
参数      Integer    ret          ;返回值          OUT
          Long      path        ;通道的路径    IN
          Integer    stno        ;站号          IN
          Integer    buf         ;型号代码      OUT
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定)
buf	返回读取的CPU的型号代码。(☞ 76页 型号代码及CPU型号一览)

■说明

关于读取的型号代码及CPU型号，请参阅一览。(☞ 76页 型号代码及CPU型号一览)

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()

型号代码及CPU型号一览

■MELSEC iQ-R系列

型号代码(16进制)	型号
4800H	R04CPU
4801H	R08CPU
4802H	R16CPU
4803H	R32CPU
4804H	R120CPU
4805H	R04ENCPU
4806H	R08ENCPU
4807H	R16ENCPU
4808H	R32ENCPU
4809H	R120ENCPU

■MELSEC-L系列

型号代码(16进制)	型号
0541H	L02CPU
0543H	L02SCPU
0544H	L06CPU
0545H	L26CPU
0548H	L26CPU-BT
0549H	L02CPU-P
054AH	L26CPU-PBT
054BH	L26CPU-P
054CH	L02SCPU-P
054DH	L06CPU-P
0641H	LJ72GF15-T2

■MELSEC-Q系列

型号代码(16进制)	型号
0041H	Q02CPU、Q02HCPU
0042H	Q06HCPU
0043H	Q12HCPU
0044H	Q25HCPU
0049H	Q12PHCPU
004AH	Q25PHCPU
004BH	Q12PRHCPU
004CH	Q25PRHCPU
004DH	Q02PHCPU
004EH	Q06PHCPU
0141H	Q02CPU(A模式)、Q02HCPU(A模式)
0142H	Q06HCPU(A模式)
0250H	Q00JCPU
0251H	Q00CPU
0252H	Q01CPU
0260H	Q00UJCPU
0261H	Q00UCPU
0262H	Q01UCPU
0263H	Q02UCPU
0266H	Q10UDHCPU
0267H	Q20UDHCPU
0268H	Q03UDCPU
0269H	Q04UDHCPU

型号代码(16进制)	型号
026AH	Q06UDHCPU
026BH	Q13UDHCPU
026CH	Q26UDHCPU
02E6H	Q10UDEHCPU
02E7H	Q20UDEHCPU
02E8H	Q03UDECPU
02E9H	Q04UDEHCPU
02EAH	Q06UDEHCPU
02EBH	Q13UDEHCPU
02ECH	Q26UDEHCPU
02EDH	Q50UDEHCPU
02EEH	Q100UDEHCPU
0362H	Q04UDPVCPU
0363H	Q06UDPVCPU
0364H	Q13UDPVCPU
0365H	Q26UDPVCPU
0366H	Q03UDVCPU
0367H	Q04UDVCPU
0368H	Q06UDVCPU
036AH	Q13UDVCPU
036CH	Q26UDVCPU
2010H	Q172CPU、Q172CPUN、Q172CPUN-T
2012H	Q172HCPU
2011H	Q173CPU、Q173CPUN、Q173CPUN-T
2013H	Q173HCPU
2014H	Q172DCPU
2015H	Q173DCPU
2018H	Q172DSCPU
2019H	Q173DSCPU

■MELSEC-QnA系列

型号代码(16进制)	型号
0021H	Q2ACPU、Q2AHCPU、Q2ASCPU、Q2ASHCPU
0022H	Q2ACPU-S1、Q2AHCPU-S1、Q2ASCPU-S1、Q2ASHCPU-S1
0023H	Q3ACPU
0024H	Q4ACPU、Q4ARCPU

■MELSEC-A系列

型号代码(16进制)	型号
0082H	A2UCPU、A2USCPU
0083H	A2UCPU-S1、A2USCPU-S1
0084H	A3UCPU、A2USHCPU-S1
0085H	A4UCPU
0092H	A2ACPU
0093H	A2ACPU-S1
0094H	A3ACPU
0098H	A0J2HCPU、A1SCPU、A1SJCPU
009AH	A2CCPU、A2CJCPU
00A0H	A0J2CPU
00A1H	A1CPU、A1NCPU
00A2H	A2CPU、A2NCPU、A2SCPU
00A3H	A3CPU、A3NCPU、A1SHCPU、A1SJHCPU、A2SHCPU、A2SH1CPU
00A4H	A3HCPU、A3MCPU

■其它

型号代码(16进制)	型号
0090H	Q80BD-J71GF11-T2、Q81BD-J71GF11-T2、Q80BD-J71GP21-SX、Q80BD-J71GP21S-SX、Q81BD-J71GP21-SX、Q81BD-J71GP21S-SX、Q80BD-J71LP21-25、Q81BD-J71LP21-25、Q80BD-J71LP21S-25、Q80BD-J71LP21G、Q80BD-J71LP21GE、Q80BD-J71BR11、Q80BD-J61BT11N、Q81BD-J61BT11
0642H	NZ2GF-ETB

mdControl (远程RUN/STOP/PAUSE)

对对象站的CPU，执行远程操作(远程RUN/STOP/PAUSE)。

格式

■Visual C++

```
格式      ret = mdControl (path, stno, buf);
参数      short      ret;           //返回值           OUT
          long      path;        //通道的路径     IN
          short     stno;        //站号           IN
          short     buf;         //指示代码       IN
```

■Visual Basic .NET

```
格式      ret = mdControl (path, stno, buf)
参数      Short      ret          ;返回值           OUT
          Integer    path        ;通道的路径     IN
          Short     stno        ;站号           IN
          Short     buf         ;指示代码       IN
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdControl (path%, stno%, buf%)
参数      Integer    ret          ;返回值           OUT
          Long      path        ;通道的路径     IN
          Integer    stno        ;站号           IN
          Integer    buf         ;指示代码       IN
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定)
buf	指定CPU的远程操作(远程RUN/STOP/PAUSE)的指示代码。(☞ 79页 说明)

■说明

指示代码及指示内容如下所示。

指示代码(16进制)	指示内容
0	远程RUN
1	远程STOP
2	远程PAUSE

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()

mdWaitBdEvent (事件等待)

等待事件的发生直至超时为止。

格式

■Visual C++

```
格式      ret = mdWaitBdEvent(path, eventno, timeout, signaledno, details);
参数      short      ret;           //返回值           OUT
          long      path;        //通道的路径     IN
          short     eventno [ ]; //等待的事件No.  IN
          long      timeout;     //超时时间       IN
          short     *signaledno; //发生的事件No.  OUT
          short     details[4];  //事件详细信息   OUT
```

■Visual Basic .NET

```
格式      ret = mdWaitBdEvent(path, eventno(0), timeout, signaledno, details(0))
参数      Short      ret           ;返回值           OUT
          Integer    path          ;通道的路径     IN
          Short      eventno(n)    ;等待的事件No.  IN
          Integer    timeout       ;超时时间       IN
          Short      signaledno    ;发生的事件No.  OUT
          Short      details(4)    ;事件详细信息   OUT
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdWaitBdEvent(path&, eventno%(0), timeout&, signaledno%, details%(0))
参数      Integer    ret           ;返回值           OUT
          Long       path          ;通道的路径     IN
          Integer    eventno(n)    ;等待的事件No.  IN
          Long       timeout       ;超时时间       IN
          Integer    signaledno    ;发生的事件No.  OUT
          Integer    details(4)    ;事件详细信息   OUT
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
eventno	指定等待的事件的事件No.。(使用实用程序中设置的事件No.。)
timeout	指定等待事件时的超时为止的时间。
signaledno	返回发生的事件No.。
details	存储条件成立的软元件的条件成立时的位模式或软元件值。必须确保4字的变量。

■说明

对指定的通道，对等待的事件No. 中指定的事件的发生按超时时间进行等待。

超时时间	内容
-1	无限等待
0	不等待
0~2147483647 (7FFFFFFFH)	ms单位等待

等待的事件No. (eventno)按下述方式指定。

等待的事件(eventno)	指定内容
eventno[0]	等待的事件的个数(1~64)
eventno[1]	等待的事件No. (0~63)……第1个
eventno[2]	等待的事件No. (0~63)……第2个
.	.
.	.
eventno[64]	等待的事件No. (0~63)……第64个

同时等待多个事件的情况下，其情况如下所示。

例

同时等待事件No. 1、No. 5、No. 12的情况下

等待的事件(eventno)	指定内容
eventno[0]=3;	等待的事件的个数(3)
eventno[1]=1;	等待的事件No. (1)……第1个
eventno[2]=5;	等待的事件No. (5)……第2个
eventno[3]=12;	等待的事件No. (12)……第3个

发生了No. 5的中断事件的情况下，signaledno中将返回5。

对于事件详细信息(details)中存储的值，发生的事件条件的指定软元件为位软元件的情况下与字软元件的情况下有所不同。请参阅下述章节。

☞ 82页 事件详细信息(details)中存储的值

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()

要点

- 未正常数据链接的情况下，不发生事件。
- 未使用网络范围分配中分配的软元件的情况下，不发生事件。
- 在调用了本函数的时刻事件已发生的情况下，将立即返回。
- 编程时应避免从多个过程指定相同通道No. 的相同事件No. 同时调用本函数。如果实施，随后调用本函数的一方将返回“事件No. 重复登录出错-63 (FFC1H)”的返回值。
- 对于事件的发生间隔，应延长至用户程序可充分处理的程度。
- 如果在未进行事件设置的状态下执行，本函数将返回“无事件设置出错-67 (FFBDH)”的返回值。
- 本函数执行中如果进行板复位或主站(管理站)的复位，本函数将返回“事件初始化出错-66 (FFBEH)”的返回值。此时，未获取的事件信息将被放弃。
- 同时发生了多个事件的情况下，首先检测出其中一个事件。再次执行了本函数的情况下检测出其它事件。
- 是CC-Link IE控制网络板、CC-Link IE现场网络板专用的函数。指定其它板的通道路径时将发生出错。(☞ 127页 出错代码)

事件详细信息(details)中存储的值

对于事件详细信息(details)中存储的值,发生的事件条件的指定软元件为位软元件的情况下与字软元件的情况下分别如下所示。

■位软元件的情况下

事件详细信息中,存储表示条件成立的软元件的位模式。

- 条件成立的软元件对应的位中将被设置1。
- 在CC-Link IE现场网络板中不能指定软元件点数,因此条件成立时第1点的位中将被设置1。
- 软元件点数不足64点的情况下,非对应的位中将被设置0。

事件详细信息(details)	从起始软元件开始的点数	存储的位模式
details[0]	第1点~第16点	位0: 第1点…位15: 第16点
details[1]	第17点~第32点	位0: 第17点…位15: 第32点
details[2]	第33点~第48点	位0: 第33点…位15: 第48点
details[3]	第49点~第64点	位0: 第49点…位15: 第64点

■字软元件的情况下

事件详细信息中存储条件成立时的软元件值。

事件详细信息(details)	存储的软元件值
details[0]	条件成立时的软元件值
details[1]	0
details[2]	0
details[3]	0

■RECV功能的情况下(仅CC-Link IE现场网络板)

事件详细信息中存储接收数据相关的信息。

事件详细信息(details)	存储的软元件值
details[0]	接收站使用通道(1~2)
details[1]	0
details[2]	0
details[3]	0

超时时间内未发生事件的情况下,返回值中将返回超时出错。

多次发生了同一事件No. 的事件的情况下

多次发生了同一事件No. 的事件的情况下,本函数的执行结果与details[]中存储的值的位软元件及字软元件中的情况分别如下所示。

■位软元件的情况下

- 本函数将正常结束。
- details[]中存储发生的各事件的位模式的逻辑和。

■字软元件的情况下

- 本函数将返回“事件发生重复出错-70(FFB4H)”的返回值。
- 此时, details[0]的值将存储最初发生的事件的条件成立时的软元件值。

mdBdRst (板复位)

对板进行复位。

格式

■ Visual C++

```
格式      ret = mdBdRst (path);
参数      short      ret;           //返回值           OUT
          long      path;         //通道的路径       IN
```

■ Visual Basic .NET

```
格式      ret = mdBdRst (path)
参数      Short      ret           ;返回值           OUT
          Integer    path         ;通道的路径       IN
```

■ Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdBdRst (path%)
参数      Integer    ret           ;返回值           OUT
          Long       path         ;通道的路径       IN
```

详细规格

■ 参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)

■ 说明

- 对板进行复位并置为实用程序功能中当前设置的状态。(进行与实用程序功能的板复位操作相同的处理。)
- 对于正在访问已复位的板的其它应用程序, 返回通知已进行了板复位的板复位出错(9922H)。

■ 返回值

正常结束时: 返回0。

异常结束时: 返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■ 关联函数

mdOpen()、mdClose()

mdBdModSet (板模式设置)

暂时更改板的模式。

格式

■Visual C++

```
格式      ret = mdBdModSet(path, mode);
参数      short      ret;           //返回值           OUT
          long      path;        //通道的路径       IN
          short     mode;        //模式             IN
```

■Visual Basic .NET

```
格式      ret = mdBdModSet(path, mode)
参数      Short      ret          ;返回值           OUT
          Integer    path        ;通道的路径       IN
          Short     mode        ;模式             IN
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdBdModSet(path%, mode%)
参数      Integer    ret          ;返回值           OUT
          Long      path        ;通道的路径       IN
          Integer    mode        ;模式             IN
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
mode	指定希望动作的模式对应的代码。(☞ 84页 说明)

■说明

- 执行本函数时，将执行板复位。
- 返回实用程序中设置的模式的情况下，应进行板复位或计算机的重新启动。
- 模式对应的代码的一览如下所示。

代码	内容			
	CC-Link Ver. 2	MELSECNET/H	CC-Link IE控制网络	CC-Link IE现场网络
0000H	在线(有自动恢复)	在线(有自动恢复)	在线	在线(标准模式)
0001H	—	—	—	在线(高速模式)*1
0002H	离线	离线	离线	离线
0003H	数据链接测试	正环路测试*2	—	—
0004H	远程站测试	副环路测试*2	—	—
0005H	—	站间测试(执行站)*2	站间测试*2	—
0006H	硬件测试	站间测试(被执行站)*2	线路测试*2	线路测试*1、*3
0007H	不可设置	自回送测试*2	自回送测试*2	自回送测试*3
0008H	—	自回送测试(内部)*2	—	—
0009H	—	H/W测试*2	H/W测试*2	H/W测试*3
000EH	—	—	总线I/F测试*2	总线I/F测试*4
0010H	—	—	—	存储器测试*4

*1 对象板为本站的情况下不能设置。

*2 关于测试方法及测试结果的确认方法，请参阅各板的手册。

*3 关于测试方法，请参阅各板的手册。对于测试结果，应通过各板的链接特殊继电器/链接特殊寄存器的值进行确认。

*4 对于测试结果，应通过返回值进行确认。

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdBdModRead()

mdBdModRead(板模式读取)

读取板当前动作的模式。

格式

Visual C++

```
格式      ret = mdBdModRead(path, mode);
参数      short      ret;           //返回值           OUT
          long      path;        //通道的路径       IN
          short     *mode;       //模式           OUT
```

Visual Basic .NET

```
格式      ret = mdBdModRead(path, mode)
参数      Short      ret          ;返回值           OUT
          Integer    path        ;通道的路径       IN
          Short      mode        ;模式           OUT
```

Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdBdModRead(path&, mode%)
参数      Integer    ret          ;返回值           OUT
          Long       path        ;通道的路径       IN
          Integer    mode        ;模式           OUT
```

详细规格

参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
mode	返回读取的模式对应的代码。(☞ 86页 说明)

说明

- 执行mdBdModSet()暂时更改板的模式的情况下，不是读取当前动作的模式，而是读取实用程序中设置的模式。
- 模式对应的代码的一览如下所示。

代码	内容			
	CC-Link Ver. 2	MELSECNET/H	CC-Link IE控制网络	CC-Link IE现场网络
0000H	在线(有自动恢复)	在线(有自动恢复)	在线	在线(标准模式)
0001H	—	—	—	在线(高速模式)
0002H	离线	离线	离线	离线
0003H	数据链接测试	正环路测试	—	—
0004H	远程站测试	副环路测试	—	—
0005H	—	站间测试(执行站)	站间测试	—
0006H	硬件测试	站间测试(被执行站)	线路测试	线路测试
0007H	不可设置	自回送测试	自回送测试	自回送测试
0008H	—	自回送测试(内部)	—	—
0009H	—	H/W测试	H/W测试	H/W测试
000EH	—	—	总线I/F测试	总线I/F测试
0010H	—	—	—	存储器测试

返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

关联函数

mdOpen()、mdClose()、mdBdModSet()

mdBdLedRead(板LED读取)

读取板的LED状态。

格式

■Visual C++

```
格式      ret = mdBdLedRead(path, buf);
参数      short      ret;           //返回值           OUT
          long      path;         //通道的路径       IN
          short     buf[ ];       //读取数据         OUT
```

■Visual Basic .NET

```
格式      ret = mdBdLedRead(path, buf(0))
参数      Short      ret           ;返回值           OUT
          Integer    path         ;通道的路径       IN
          Short      buf(n)       ;读取数据         OUT
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdBdLedRead(path&, buf%(0))
参数      Integer    ret           ;返回值           OUT
          Long       path         ;通道的路径       IN
          Integer    buf(n)       ;读取数据         OUT
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
buf	存储读取的LED信息。(☞ 88页 LED信息一览)

■说明

- 关于LED信息，请参阅LED信息一览。(☞ 88页 LED信息一览)
- LED闪烁时，存储读取的时刻的亮灯状态。关于LED的详细内容，请参阅各板的手册。

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()

LED信息一览

■CC-Link Ver. 2的情况下

元素	位	LED	内容	备注
buf[0]	b15-b9	保留	—	—
	b8	ERR	显示全站通信异常的检测状态。 1: 异常检测 0: 未检测	亮灯: 1 熄灯: 0
	b7-b1	保留	—	—
	b0	RUN	显示板动作状态。 1: 正常动作中 0: 板WDT出错	亮灯: 1 熄灯: 0
buf[1]	b15-b1	保留	—	—
	b0	SW	显示开关设置异常的检测状态。 1: 异常检测 0: 未检测	亮灯: 1 熄灯: 0
buf[2]	b15-b9	保留	—	—
	b8	PRM	显示参数出错的检测状态。 1: 出错检测 0: 未检测	亮灯: 1 熄灯: 0
	b7-b1	保留	—	—
	b0	M/S	显示主站重复出错的检测状态。 1: 出错检测 0: 未检测	亮灯: 1 熄灯: 0
buf[3]	b15-b9	保留	—	—
	b8	LINE	显示电缆断线出错的检测状态。 1: 出错检测 0: 未检测	亮灯: 1 熄灯: 0
	b7-b1	保留	—	—
	b0	TIME	显示数据链接监视定时器动作时的出错检测状态。 1: 出错检测 0: 未检测	亮灯: 1 熄灯: 0
buf[4]	b15-b9	保留	—	—
	b8	L. ERR	显示通信出错的检测状态。 1: 出错检测 0: 未检测	亮灯: 1 熄灯: 0
	b7-b1	保留	—	—
	b0	L. RUN	显示数据链接执行状态。 1: 数据链接实施中 0: 数据链接未实施	亮灯: 1 熄灯: 0
buf[5]	b15-b9	保留	—	—
	b8	RD	显示数据接收状态。 1: 数据接收中 0: 数据未接收	亮灯: 1 熄灯: 0
	b7-b1	保留	—	—
	b0	SD	显示数据发送状态。 1: 数据发送中 0: 数据未发送	亮灯: 1 熄灯: 0

■MELSECNET/H的情况下

元素	位	LED	内容	备注
buf[0]	b15	R. LOOP OVER	显示副环路侧数据获取延迟出错的检测状态。 1: 出错检测 0: 未检测	亮灯: 1 熄灯: 0
	b14	F. LOOP CRC	显示正环路侧代码检查出错的检测状态。 1: 出错检测 0: 未检测	
	b13	R. LOOP CRC	显示副环路侧代码检查出错的检测状态。 1: 出错检测 0: 未检测	
	b12	F. LOOP	显示正环路侧环路线路的出错检测状态。 1: 出错检测 0: 未检测	
	b11	R. LOOP	显示副环路侧环路线路的出错检测状态。 1: 出错检测 0: 未检测	
	b10	F. LOOP RD	显示正环路侧的数据接收状态。 1: 数据接收中 0: 数据未接收	
	b9	R. LOOP RD	显示副环路侧的数据接收状态。 1: 数据接收中 0: 数据未接收	
	b8	F. LOOP SD	显示正环路侧的数据发送状态。 1: 数据发送中 0: 数据未发送	
	b7	R. LOOP SD	显示副环路侧的数据发送状态。 1: 数据发送中 0: 数据未发送	
	b6	S. MNG	显示站类型。 1: 作为副管理站动作中 0: 副管理站以外	
	b5	MNG	显示站类型。 1: 作为管理站动作中 0: 管理站以外	
	b4	D. LINK	显示数据链接状态(循环传送状态)。 1: 数据链接实施中 0: 数据链接未实施	
	b3	T. PASS	显示令牌传递状态。 1: 令牌传递实施中 0: 令牌传递未实施	
	b2	PRM. E.	显示参数出错的检测状态。 1: 参数出错检测 0: 未检测	
	b1	SW. E.	显示开关设置异常检测状态。 1: 异常检测 0: 未检测	
	b0	RUN	显示板动作状态。 1: 正常动作中 0: 硬件异常或板WDT出错	

元素	位	LED	内容	备注
buf[1]	b15-b10	保留	—	—
	b9	M/S. E.	显示同一网络上站号或管理站的重复出错检测状态。 1: 出错检测 0: 未检测	亮灯: 1 熄灯: 0
	b8	F. LOOP UNDER	显示正环路侧发送数据的内部处理不是一定间隔时的出错检测状态。 1: 出错检测 0: 未检测	
	b7	R. LOOP UNDER	显示副环路侧发送数据的内部处理不是一定间隔时的出错检测状态。 1: 出错检测 0: 未检测	
	b6	F. LOOP DATA	显示正环路侧接收了2K字节及以上的异常数据时的出错检测状态。 1: 出错检测 0: 未检测	
	b5	R. LOOP DATA	显示副环路侧接收了2K字节及以上的异常数据时的出错检测状态。 1: 出错检测 0: 未检测	
	b4	F. LOOP TIME	显示正环路侧的数据链接监视定时器动作时的出错检测状态。 1: 出错检测 0: 未检测	
	b3	R. LOOP TIME	显示副环路侧的数据链接监视定时器动作时的出错检测状态。 1: 出错检测 0: 未检测	
	b2	F. LOOP AB. IF	显示正环路侧连续接收了规定以上的“1”时及接收数据长过短时的出错检测状态。 1: 出错检测 0: 未检测	
	b1	R. LOOP AB. IF	显示副环路侧连续接收了规定以上的“1”时及接收数据长过短时的出错检测状态。 1: 出错检测 0: 未检测	
b0	F. LOOP OVER	显示正环路侧数据获取延迟出错的检测状态。 1: 出错检测 0: 未检测		

■CC-Link IE控制网络的情况下

元素	位	LED	内容	备注
buf[0]	b15	EXT. PW	显示带外部供应电源功能CC-Link IE控制网络模块的外部电源供应状态。 1: 外部电源供应中 0: 外部电源未供应	亮灯: 1 熄灯: 0
	b14-b7	保留	—	—
	b6	RD	显示数据接收状态。 1: 数据接收中 0: 数据未接收	亮灯: 1 熄灯: 0
	b5	D. LNK	显示数据链接状态(正在进行循环传送的状态)。 1: 数据链接实施中 0: 数据链接未实施	
	b4	PRM	显示参数出错的检测状态。 1: 出错检测 0: 未检测	
	b3	ERR.	显示出错的检测状态。 1: 出错检测 0: 未检测	
	b2	SD	显示数据发送状态。 1: 数据发送中 0: 数据未发送	
	b1	MODE	显示动作模式。 1: 以在线模式动作中 0: 以离线模式动作中	
	b0	RUN	显示板动作状态。 1: 正常动作中 0: 硬件异常或板WDT出错	
buf[1]	b15-b0	保留	—	—

■CC-Link IE现场网络的情况下

元素	位	LED	内容	备注
buf[0]	b15-b12	保留	—	—
	b11	LNK2	显示PORT2的连接状态。 1: 链接中 0: 链接死机中	亮灯: 1 熄灯: 0
	b10	LINK1	显示PORT1的连接状态。 1: 链接中 0: 链接死机中	
	b9	LER2	显示PORT2的帧丢失状态。 1: 帧丢失发生中 0: 无帧丢失	
	b8	LER1	显示PORT1的帧丢失状态。 1: 帧丢失发生中 0: 无帧丢失	
	b7	MODE	显示动作模式。 1: 以在线模式动作中 0: 以离线模式动作中	
	b6	RD	显示网络接收状态。 1: 数据接收中 0: 数据未接收	
	b5	SD	显示网络发送状态。 1: 数据发送中 0: 数据未发送	
	b4	L. ERR	显示通信出错的检测状态。 1: 出错检测 0: 未检测	
	b3	ERR.	显示网络板的出错检测状态。 1: 出错检测 0: 未检测	
	b2	D. LINK	显示数据链接状态(正在进行循环传送的状态)。 1: 数据链接实施中 0: 数据链接未实施	
	b1	MST	显示站类型。 1: 作为主站动作中 0: 主站以外	
	b0	RUN	显示板动作状态。 1: 正常动作中 0: 板WDT出错发生中或板复位中	
buf[1]	b15-b0	保留	—	

mdBdSwRead(板开关状态读取)

读取板的开关状态(站号设置、板No. 设置、板识别、I/O地址设置信息等)。

格式

■Visual C++

```
格式      ret = mdBdSwRead(path, buf);
参数      short      ret;           //返回值           OUT
          long      path;        //通道的路径       IN
          short     buf[ ];      //读取数据         OUT
```

■Visual Basic .NET

```
格式      ret = mdBdSwRead(path, buf(0))
参数      Short      ret          ;返回值           OUT
          Integer    path        ;通道的路径       IN
          Short      buf(n)      ;读取数据         OUT
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdBdSwRead(path%, buf%(0))
参数      Integer    ret          ;返回值           OUT
          Long       path        ;通道的路径       IN
          Integer    buf(n)      ;读取数据         OUT
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
buf	存储读取的板的开关状态。(☞ 93页 说明)

■说明

读取板的开关状态(站号设置、板No. 设置、板识别、I/O地址设置信息等)。

元素	项目	内容			
		CC-Link Ver. 2	MELSECNET/H	CC-Link IE控制网络	CC-Link IE现场网络
buf[0]	站号的设置值	0~64 (0: 主站)	1~64	1~120	0~120
buf[1]	组No. 的设置值	固定为0	MNET/10模式时: 0~9 MNET/H模式时: 0~32	0~32	固定为0
buf[2]	网络No. 的设置值	固定为0	1~239	1~239	1~239
buf[3]	保留	固定为0	固定为0	固定为0	固定为0
buf[4]	保留	固定为0	固定为0	固定为0	固定为0
buf[5]	保留	固定为0	固定为0	固定为0	固定为0

■返回值

正常结束时: 返回0。

异常结束时: 返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()

mdBdVerRead(板版本读取)

读取板的版本信息。

格式

■Visual C++

```
格式      ret = mdBdVerRead(path, buf);
参数      short      ret;          //返回值          OUT
          long      path;        //通道的路径      IN
          short     buf[ ];      //读取数据          OUT
```

■Visual Basic .NET

```
格式      ret = mdBdVerRead(path, buf(0))
参数      Short      ret          ;返回值          OUT
          Integer    path        ;通道的路径      IN
          Short      buf(n)      ;读取数据          OUT
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdBdVerRead(path&, buf%(0))
参数      Integer    ret          ;返回值          OUT
          Long       path        ;通道的路径      IN
          Integer    buf(n)      ;读取数据          OUT
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
buf	存储读取的板的版本信息。(☞ 95页 说明)

■说明

版本详细信息的一览如下所示。

偏置 (字)	项目	内容			
		CC-Link Ver. 2	MELSECNET/H	CC-Link IE控制网络	CC-Link IE现场网络
00H	固定值	固定为‘S’、‘G’	固定为‘S’、‘G’	固定为‘S’、‘G’	固定为‘S’、‘G’
01H	校验和	02H~0FH的合计	02H~0FH的合计	02H~0FH的合计	02H~0FH的合计
02H	S/W版本	‘A’ ~ ‘ZZ’	‘0A’ ~ ‘9Z’	‘0A’ ~ ‘9Z’	‘0A’ ~ ‘9Z’
03H~05H	日期	日期年月日 例)2010年2月1日 ‘1’ ‘0’ ‘0’ ‘2’ ‘0’ ‘1’	日期年月日 例)2010年2月1日 ‘1’ ‘0’ ‘0’ ‘2’ ‘0’ ‘1’	日期年月日 例)2010年2月1日 ‘1’ ‘0’ ‘0’ ‘2’ ‘0’ ‘1’	日期年月日 例)2010年2月1日 ‘1’ ‘0’ ‘0’ ‘2’ ‘0’ ‘1’
06H~07H	保留区域(4字节)	固定为000H	固定为000H	固定为000H	固定为000H
08H~0FH	S/W型号(16字节)	‘QJ61BT11N’	‘QJ71LP21-25’、‘QJ71LP21S-2’、‘QJ71LP21G’、‘QJ71BR11’	‘QJ71GP21-SX’、‘QJ71GP21S-SX’	‘QJ71GF11-T2’
10H~17H	H/W型号(16字节)	‘Q80BD-J61BT11N’、‘Q81BD-J61BT11’	‘Q80BD-J71LP21-25’、‘Q81BD-J71LP21-25’、‘Q80BD-J71LP21S-25’、‘Q80BD-J71LP21G’、‘Q80BD-J71LP21GE’、‘Q80BD-J71BR11’	‘Q80BD-J71GP21’、‘Q80BD-J71GP21S’、‘Q81BD-J71GP21’、‘Q81BD-J71GP21S’	‘Q80BD-J71GF11-T2’、‘Q81BD-J71GF11-T2’
18H	2端口存储器占用容量	0200H(512K字节)	0080H(128K字节)	0180H(384K字节)	0080H(128K字节)
19H	2端口属性	固定为0080H	固定为0080H	固定为0080H	固定为0080H
1AH	可使用偏置	固定为0000H	固定为0000H	固定为0000H	固定为0000H
1BH(L)	机型类型(10字节)	功能版本 (‘A’、‘B’...)	固定为0000H	功能版本 (‘A’、‘B’...)	功能版本 (‘A’、‘B’...)
1BH(H)		CC-Link主要版本(0002H)		固定为0000H	固定为0000H
1CH(L)		CC-Link次要版本(0000H)			
1CH(H)~1FH		固定为0000H			

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(P.127页 出错代码)

■关联函数

mdOpen()、mdClose()

mdInit(可编程控制器软元件信息表的初始化)

刷新MELSEC数据链接库的内部数据的可编程控制器软元件地址表。

格式

■Visual C++

```
格式      ret = mdInit(path);
参数      short      ret;          //返回值          OUT
          long      path;        //通道的路径      IN
```

■Visual Basic .NET

```
格式      ret = mdInit(path)
参数      Short      ret          ;返回值          OUT
          Integer    path        ;通道的路径      IN
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdInit(path%)
参数      Integer    ret          ;返回值          OUT
          Long      path        ;通道的路径      IN
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)

■说明

- 放弃初次访问可编程控制器CPU时获取的可编程控制器软元件信息。
- 执行mdInit函数后，初次访问可编程控制器CPU时，再次进行可编程控制器软元件信息的获取。因此，执行mdInit函数后，初次的函数执行时间将变长。

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(127页 出错代码)

■关联函数

mdOpen()、mdClose()

mdSend(软元件批量写入)

对对象站，从起始软元件No. 开始将数据批量写入到相当于写入字节容量的软元件中。

格式

■Visual C++

格式	ret = mdSend(path, stno, devtyp, devno, size, data);			
参数	short	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	short	stno;	//站号	IN
	short	devtyp;	//软元件类型	IN
	short	devno;	//起始软元件No.	IN
	short	*size;	//写入字节容量	IN/OUT
	short	data[];	//写入数据(单精度整数数组)	IN

■Visual Basic .NET

格式	ret = mdSend(path, stno, devtyp, devno, size, data(0))			
参数	Short	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Short	stno	;站号	IN
	Short	devtyp	;软元件类型	IN
	Short	devno	;起始软元件No.	IN
	Short	size	;写入字节容量	IN/OUT
	Short	data(n)	;写入数据(单精度整数数组)	IN

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdSend(path%, stno%, devtyp%, devno%, size%, data%(0))			
参数	Integer	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Integer	stno	;站号	IN
	Integer	devtyp	;软元件类型	IN
	Integer	devno	;起始软元件No.	IN
	Integer	size	;写入字节容量	IN/OUT
	Integer	data(n)	;写入数据(单精度整数数组)	IN

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定)
devtyp	指定写入数据的软元件的软元件类型。(☞ 51页 软元件类型指定)
devno	指定写入数据的软元件的起始软元件No。 <ul style="list-style-type: none"> 对于至位软元件的访问，应以8的倍数指定起始软元件No。(但是，超长定时器及超长累计定时器的情况下，至触点、线圈的访问也可指定任意的起始软元件No。) 对于至CC-Link其它站链接软元件的位软元件(RX、RY、SB)的访问，应以16的倍数(0、16、32...)指定起始软元件No。
size	以偶数指定写入数据的字节容量。 <ul style="list-style-type: none"> 指定的写入字节容量超出软元件范围(-5: 容量出错)的情况下，size中将返回可写入容量。 devtyp中指定了双字软元件的情况下，应以4的倍数指定。
data	以单精度整数数组指定写入数据。

■说明

- 指定的写入字节容量超过瞬时传送最大发送容量的情况下，在函数内部分割执行。
- 其它站访问中，对分配了扩展注释的块(扩展文件寄存器)进行写入时，扩展注释信息将被清除。
- 其它站访问中，对与子2、子3程序设置区域重叠的块(扩展文件寄存器)进行写入时，子2、子3程序将被清除。
- devtyp中指定了双字软元件的情况下，应按下述方式将写入数据存储到data中。

例

devtyp为LZ，size为8的情况下

数组	值
data(0)	LZ0的低位字
data(1)	LZ0的高位字
data(2)	LZ1的低位字
data(3)	LZ1的高位字

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdSendEx()、mdReceive()、mdReceiveEx()

mdSend (SEND功能)

将数据发送至指定的对象站的通道No.。

格式

■Visual C++

格式	ret = mdSend(path, stno, devtyp, devno, size, data);			
参数	short	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	short	stno;	//站号	IN
	short	devtyp;	//软元件类型	IN
	short	devno;	//通道No.	IN
	short	*size;	//发送字节容量	IN/OUT
	short	data[];	//发送数据(单精度整数数组)	IN

■Visual Basic .NET

格式	ret = mdSend(path, stno, devtyp, devno, size, data(0))			
参数	Short	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Short	stno	;站号	IN
	Short	devtyp	;软元件类型	IN
	Short	devno	;通道No.	IN
	Short	size	;发送字节容量	IN/OUT
	Short	data(n)	;发送数据(单精度整数数组)	IN

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdSend(path&, stno%, devtyp%, devno%, size%, data%(0))			
参数	Integer	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Integer	stno	;站号	IN
	Integer	devtyp	;软元件类型	IN
	Integer	devno	;通道No.	IN
	Integer	size	;发送字节容量	IN/OUT
	Any	data(n)	;发送数据(单精度整数数组)	IN

详细规格

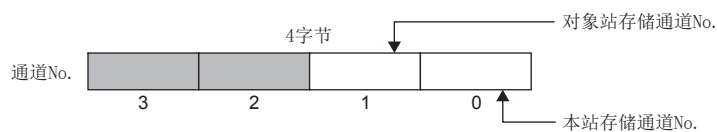
■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定) • 不能进行逻辑站号的指定。
devtyp	指定是以“有到达确认”还是“无到达确认”进行数据发送。 • 有到达确认的情况下*1: 指定101(65H) • 无到达确认的情况下: 指定102(66H)
devno	指定本站使用通道及对象站存储通道。(☞ 100页 说明)
size	以偶数指定发送数据的字节容量。 • 应以2~1920的范围指定。 • 从MELSECNET/H板向CC-Link IE控制网络、CC-Link IE现场网络发送数据的情况下, 应以2~960字节的范围指定。 • 经由MELSECNET/10网络的情况下, 应以2~960的范围指定。
data	以单精度整数数组指定发送的数据。

*1 以有到达确认发送数据的情况下, 站号中请勿指定全站、组No.。

■说明

- 支持链接专用指令的SEND指令。关于功能详细内容，请参阅各板的手册。
- 通道No. 的指定方法如下所示。



项目	内容	
对象站存储通道	指定对象站存储通道No。 • MELSECNET/H板、CC-Link IE控制网络板 • CC-Link IE现场网络板	— 1~8 1~2
本站使用通道	指定本站使用通道No。 • MELSECNET/H板、CC-Link IE控制网络板 • CC-Link IE现场网络板	— 1~8 1~2

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(P. 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdSendEx()、mdReceive()、mdReceiveEx()

mdReceive(软元件批量读取)

从对象站中将起始软元件No. 中指定的软元件按读取字节容量进行批量读取。

格式

■Visual C++

格式	ret = mdReceive(path, stno, devtyp, devno, size, data);			
参数	short	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	short	stno;	//站号	IN
	short	devtyp;	//软元件类型	IN
	short	devno;	//起始软元件No.	IN
	short	*size;	//读取字节容量	IN/OUT
	short	data[];	//读取数据(单精度整数数组)	OUT

■Visual Basic .NET

格式	ret = mdReceive(path, stno, devtyp, devno, size, data(0))			
参数	Short	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Short	stno	;站号	IN
	Short	devtyp	;软元件类型	IN
	Short	devno	;起始软元件No.	IN
	Short	size	;读取字节容量	IN/OUT
	Short	data(n)	;读取数据(单精度整数数组)	OUT

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdReceive(path%, stno%, devtyp%, devno%, size%, data%(0))			
参数	Integer	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Integer	stno	;站号	IN
	Integer	devtyp	;软元件类型	IN
	Integer	devno	;起始软元件No.	IN
	Integer	size	;读取字节容量	IN/OUT
	Integer	data(n)	;读取数据(单精度整数数组)	OUT

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定)
devtyp	指定读取数据的软元件的软元件类型。(☞ 51页 软元件类型指定)
devno	指定读取软元件范围的起始软元件No。 <ul style="list-style-type: none"> 对于至位软元件的访问，应以8的倍数指定起始软元件No。(但是，超长定时器及超长累计定时器的情况下，至触点、线圈的访问也可指定任意的起始软元件No。) 对于至CC-Link其它站链接软元件的位软元件(RX、RY、SB)的访问，应以16的倍数(0、16、32...)指定起始软元件No。
size	以偶数指定读取数据的字节容量。 <ul style="list-style-type: none"> 指定的读取字节容量超出软元件范围(-5: 容量出错)的情况下，size中将返回可读取容量。 devtyp中指定了双字软元件的情况下，应以4的倍数指定。
data	存储读取的数据。

■说明

- 指定的读取字节容量超过瞬时传送最大发送容量的情况下，在函数内部分割执行。
- devtyp中指定了双字节元件的情况下，数据按下述方式存储到data中。

例

devtyp为LZ，size为8的情况下

数组	值
data(0)	LZ0的低位字
data(1)	LZ0的高位字
data(2)	LZ1的低位字
data(3)	LZ1的高位字

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdReceiveEx()、mdSend()、mdSendEx()

mdReceive (RECV功能)

从到达本站的数据中读取指定通道No. 的数据。

格式

■Visual C++

格式	ret = mdReceive(path, stno, devtyp, devno, size, data);			
参数	short	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	short	stno;	//站号	IN
	short	devtyp;	//软元件类型	IN
	short	devno;	//通道No.	IN
	short	*size;	//接收字节容量	IN/OUT
	short	data[];	//接收数据+发送源信息(单精度整数数组)	OUT

■Visual Basic .NET

格式	ret = mdReceive(path, stno, devtyp, devno, size, data(0))			
参数	Short	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Short	stno	;站号	IN
	Short	devtyp	;软元件类型	IN
	Short	devno	;通道No.	IN
	Short	size	;接收字节容量	IN/OUT
	Short	data(n)	;接收数据+发送源信息(单精度整数数组)	OUT

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdReceive(path%, stno%, devtyp%, devno%, size%, data%(0))			
参数	Integer	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Integer	stno	;站号	IN
	Integer	devtyp	;软元件类型	IN
	Integer	devno	;通道No.	IN
	Integer	size	;接收字节容量	IN/OUT
	Integer	data(n)	;接收数据+发送源信息(单精度整数数组)	OUT

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定本站(255(FFH))。
devtyp	指定RECV功能(101(65H))。
devno	指定存储了接收信息的本站的通道No。 <ul style="list-style-type: none"> • MELSECNET/H 板、CC-Link IE控制网络板: 1~8 • CC-Link IE现场网络板: 1~2
size	以偶数指定接收数据的字节容量。 <ul style="list-style-type: none"> • 应以2~1920的范围指定。 • 通过CC-Link IE控制网络板、CC-Link IE现场网络板接收从MELSECNET/H板发送的数据的情况下, 应以2~960字节的范围指定。 • 经由MELSECNET/10网络的情况下, 应以2~960的范围指定。 返回接收的实际数据的容量。
data	以单精度整数数组存储接收的实际数据及发送源信息。 应确保相当于指定的接收字节容量+发送源信息(6字节)的区域。

■说明

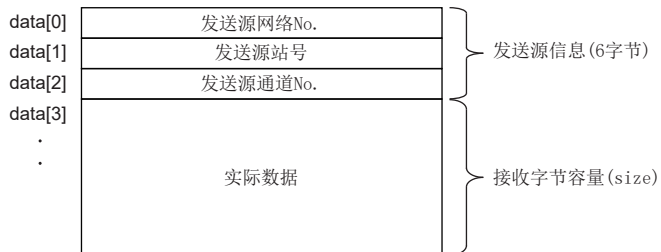
支持链接专用指令的RECV指令。

关于功能详细内容，请参阅各板的手册。

<关于接收数据+发送源信息 (data)>

- 接收数据+发送源信息 (data) 中，存储在接收字节容量 (size) 中附加了发送源信息 (6字节) 的数据。接收数据+发送源信息 (data) 中，应确保相当于 [接收字节容量 (size)+发送源信息 (6字节)] 的区域。
- 接收数据+发送源信息 (data) 按下述方式存储。

接收数据+发送源信息 (data)



- 接收的实际数据的容量大于指定的接收字节容量 (size) 的情况下，只存储相当于指定的接收字节容量 (size) 的实际数据。
- 发送源的站号为0的情况下，发送源站号中将存储125 (7DH)。

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdReceiveEx()、mdSend()、mdSendEx()

mdRandW(软元件随机写入)

将数据写入到随机指定软元件中指定的对象站的软元件中。

格式

■Visual C++

格式	ret = mdRandW(path, stno, dev, buf, bufsize);			
参数	short	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	short	stno;	//站号	IN
	short	dev[];	//随机指定软元件	IN
	short	buf[];	//写入数据(单精度整数数组)	IN
	short	bufsize;	//虚拟	IN

■Visual Basic .NET

格式	ret = mdRandW(path, stno, dev(0), buf(0), bufsize)			
参数	short	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	short	stno	;站号	IN
	short	dev(n)	;随机指定软元件	IN
	Short	buf(n)	;写入数据(单精度整数数组)	IN
	short	bufsize	;虚拟	IN

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdRandW(path%, stno%, dev%(0), buf%(0), bufsize%)			
参数	Integer	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Integer	stno	;站号	IN
	Integer	dev(n)	;随机指定软元件	IN
	Any	buf(n)	;写入数据(单精度整数数组)	IN
	Integer	bufsize	;虚拟	IN

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定)
dev	指定写入软元件的块数、软元件类型、起始软元件No.、点数。(☞ 67页 随机指定软元件(dev)的指定方法)
buf	以单精度整数数组指定写入数据。(☞ 67页 随机指定软元件(dev)的指定方法)
bufsize	—

■说明

- 根据指定的随机指定软元件，函数内部进行的瞬时通信次数将变化。
- 随机写入与批量写入相比函数执行时间将变长。希望缩短函数执行时间的情况下，应讨论批量写入的使用。
- 其它站访问中，对分配了扩展注释的块(扩展文件寄存器)进行写入时，扩展注释信息将被清除。
- 其它站访问中，对与子2、子3程序设置区域重叠的块(扩展文件寄存器)进行写入时，子2、子3程序将被清除。
- 至MELSECNET(II)、MELSECNET/10的本站的B、W的随机写入时发生了出错的情况下，正确完成写入的块与发生出错的块有可能混在一起。
- 根据随机指定软元件的点数指定，写入的数据的容量对各软元件的类型有所不同。(☞ 51页 每软元件1点的数据容量)

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdRandR()

mdRandR(软元件随机读取)

从对象站中读取随机指定软元件中指定的软元件。

格式

■Visual C++

格式	ret = mdRandR(path, stno, dev, buf, bufsize);			
参数	short	ret;	//返回值	OUT
	long	path;	//通道的路径	IN
	short	stno;	//站号	IN
	short	dev[];	//随机指定软元件	IN
	short	buf[];	//读取数据(单精度整数数组)	OUT
	short	bufsize;	//读取数据的字节数	IN

■Visual Basic .NET

格式	ret = mdRandR(path, stno, dev(0), buf(0), bufsize)			
参数	Short	ret	;返回值	OUT
	Integer	path	;通道的路径	IN
	Short	stno	;站号	IN
	Short	dev(n)	;随机指定软元件	IN
	Short	buf(n)	;读取数据(单精度整数数组)	OUT
	Short	bufsize	;读取数据的字节数	IN

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdRandR(path&, stno%, dev%(0), buf%(0), bufsize%)			
参数	Integer	ret	;返回值	OUT
	Long	path	;通道的路径	IN
	Integer	stno	;站号	IN
	Integer	dev(n)	;随机指定软元件	IN
	Any	buf(n)	;读取数据(单精度整数数组)	OUT
	Integer	bufsize	;读取数据的字节数	IN

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定)
dev	指定读取软元件的块数、软元件类型、起始软元件No.、点数。(☞ 69页 随机指定软元件(dev)的指定方法)
buf	存储读取的数据。(☞ 69页 随机指定软元件(dev)的指定方法)
bufsize	指定读取数据的字节数。(☞ 69页 随机指定软元件(dev)的指定方法)

■说明

- 根据指定的随机指定软元件，函数内部进行的瞬时通信次数将变化。
- 随机读取与批量读取相比函数执行时间将变长。希望缩短函数执行时间的情况下，应讨论批量读取的使用。
- 根据随机指定软元件的点数指定，读取的数据的容量对各软元件的类型有所不同。(☞ 51页 每软元件1点的数据容量)

■返回值

正常结束时：返回0。

异常结束时：返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdRandW()

mdDevSet (位软元件设置)

设置 (ON) 对象站的指定位软元件。

格式

■Visual C++

格式	ret = mdDevSet (path, stno, devtyp, devno);		
参数	short	ret;	//返回值 OUT
	long	path;	//通道的路径 IN
	short	stno;	//站号 IN
	short	devtyp;	//软元件类型 IN
	short	devno;	//指定软元件No. IN

■Visual Basic .NET

格式	ret = mdDevSet (path, stno, devtyp, devno)		
参数	Short	ret	;返回值 OUT
	Integer	path	;通道的路径 IN
	Short	stno	;站号 IN
	Short	devtyp	;软元件类型 IN
	Short	devno	;指定软元件No. IN

■Visual Basic 5.0、Visual Basic 6.0

格式	ret% = mdDevSet (path%, stno%, devtyp%, devno%)		
参数	Integer	ret	;返回值 OUT
	Long	path	;通道的路径 IN
	Integer	stno	;站号 IN
	Integer	devtyp	;软元件类型 IN
	Integer	devno	;指定软元件No. IN

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定)
devtyp	指定设置(ON)的软元件的软元件类型。(☞ 51页 软元件类型指定)
devno	指定设置(ON)的软元件的软元件No.。

■说明

本函数是链接继电器(B)、内部继电器(M)等的位软元件专用的函数。

■返回值

正常结束时: 返回0。

异常结束时: 返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdDevRst()

mdDevRst (位软元件复位)

复位 (OFF) 对象站的指定位软元件。

格式

■Visual C++

```
格式      ret = mdDevRst (path, stno, devtyp, devno);
参数      short      ret;           //返回值           OUT
          long      path;         //通道的路径       IN
          short     stno;         //站号             IN
          short     devtyp;       //软元件类型       IN
          short     devno;        //指定软元件No.   IN
```

■Visual Basic .NET

```
格式      ret = mdDevRst (path, stno, devtyp, devno)
参数      Short      ret           ;返回值           OUT
          Integer    path         ;通道的路径       IN
          Short      stno         ;站号             IN
          Short      devtyp       ;软元件类型       IN
          Short      devno        ;指定软元件No.   IN
```

■Visual Basic 5.0、Visual Basic 6.0

```
格式      ret% = mdDevRst (path%, stno%, devtyp%, devno%)
参数      Integer    ret           ;返回值           OUT
          Long       path         ;通道的路径       IN
          Integer    stno         ;站号             IN
          Integer    devtyp       ;软元件类型       IN
          Integer    devno        ;指定软元件No.   IN
```

详细规格

■参数

参数	说明
path	指定通信线路已打开的通道的路径。 (使用执行mdOpen时返回的路径。)
stno	指定对象站的站号。(☞ 47页 站号指定)
devtyp	指定复位(OFF)的软元件的软元件类型。(☞ 51页 软元件类型指定)
devno	指定复位(OFF)的软元件的软元件No.。

■说明

本函数是链接继电器(B)、内部继电器(M)等的位软元件专用的函数。

■返回值

正常结束时: 返回0。

异常结束时: 返回0以外。请参阅出错一览。(☞ 127页 出错代码)

■关联函数

mdOpen()、mdClose()、mdDevSet()

5 样本程序

以下介绍软件包中存储的样本程序。

样本程序登录在安装了实用程序的文件夹内。

类型	文件夹	内容
CC-Link Ver. 2板	\CCBD2\Sample\MDFunction	MELSEC数据链接库的样本程序
	\CCBD2\Sample\LocalStation	CC-Link Ver. 2板的数据链接确认用样本程序
	\CCBD2\Sample\MasterStation	
MELSECNET/H板	\MNETH\SAMPLES	MELSEC数据链接库的样本程序
CC-Link IE控制网络板	\MNETG\SAMPLES	MELSEC数据链接库的样本程序
CC-Link IE现场网络板	\CCIEF\SAMPLES	MELSEC数据链接库的样本程序
	\CCIEF\SAMPLES\RemoteDevice	远程I/O模块用样本程序

要点

附加的样本程序用于创建用户程序时的参考。
关于样本程序的使用，应由用户自担责任。

MELSEC数据链接库的样本程序

作为函数的使用示例存储了下述样本程序。

文件夹	内容	内容	
DEMO	软件数据读取	CC-Link Ver. 2板	读取主站的软件D0的样本程序
		MELSECNET/H板	读取网络No. 1、站号1的软件D0的样本程序
		CC-Link IE控制网络板	
		CC-Link IE现场网络板	
MTEST MTEST2	函数常规测试	MELSEC数据链接库的常规样本程序	

CC-Link Ver. 2板的数据链接确认用样本程序

作为确认主站与各站是否正常数据链接的程序示例，存储了下述样本程序。

文件夹	内容	内容
LocalStation	Datalink	主站与本地站的通信 主站↔本地站通信的样本程序
MasterStation	Datalink	主站与本地站的通信 主站↔本地站通信的样本程序
	PositioningSystem	主站与智能设备站(AJ65BT-D75P2-S3)的通信 进行AJ65BT-D75P2-S3的初始化、定位、原点回归、JOG运行的样本程序
	R2	主站与智能设备站(AJ65BT-R2)的通信 进行AJ65BT-R2用的初始化、发送、接收的样本程序
	RemoteDevice	主站与远程设备站的通信 进行AJ65BT-64DAV用的数字/模拟转换的样本程序
	RemoteIO	主站与远程I/O站的通信 进行远程I/O站用的远程I/O读写的样本程序

关于确认方法的详细内容，请参阅下述手册。

📖 Q80BD-J61BT11N/Q81BD-J61BT11型CC-Link系统主站/本地站接口板用户手册(SW1DNC-CCBD2-B系列)

CC-Link IE现场网络远程I/O模块用样本程序

存储了通过mdRemBufWriteEx函数及mdRemBufReadEx函数进行CC-Link IE现场网络的远程I/O模块(远程设备站)的初始设置的程序示例。

5.1 使用Visual Basic .NET的情况下

Visual Basic .NET用的样本程序存储在“NETVB”文件夹中。

样本程序为Visual Studio .NET 2003的工程格式。


使用Visual Studio .NET 2003以外的情况下，应进行工程转换。

要点

样本程序中未包含“Mdfunc.vb”。

应预先添加“Mdfunc.vb”之后再使用。

关于设置方法，请参阅下述章节。

 17页 使用Visual Basic .NET的情况下

MTEST及MTEST2

MELSEC数据链接库的函数的常规测试用样本程序根据所使用的编程语言存储在下列文件夹中。

文件夹	MTEST	MTEST2
编程语言	Visual Studio .NET 2003 Visual Studio 2005 Visual Studio 2008 Visual Studio 2010	Visual Studio 2012 Visual Studio 2013 Visual Studio 2015 Visual Studio 2017

要点

本样本程序不能作为64位版用户应用程序使用。将本样本程序在64位版OS上使用的情况下，应通过Visual Studio将平台选择为“x86”，作为32位版用户应用程序进行编译。

NETVB\MTEST

在本样本程序中使用MSFlexGrid。未安装MSFlexGrid的情况下，将发生警告，但对样本程序的动作无影响。

NETVB\MTEST2

在本样本程序中使用Visual Basic 6.0的控制器，因此虽然会发生警告，但对样本程序的动作无影响。

5.2 使用Visual Basic 5.0、Visual Basic 6.0的情况下


Visual Basic 5.0、Visual Basic 6.0用的样本程序存储在“VB”文件夹中。

要点

样本程序中未包含“Mdfunc.bas”。

应预先添加“Mdfunc.bas”之后再使用。

关于设置方法，请参阅下述章节。

 14页 使用Visual Basic 5.0、Visual Basic 6.0的情况下

5.3 使用Visual C++的情况下

Visual C++用的样本程序存储在“VC”文件夹中。

样本程序为下述Visual C++的工程格式。

使用不同格式的Visual C++的情况下，应进行工程转换。

软件包	样本程序的工程格式
SW1DNC-CCBD2-B	Visual C++ 5.0
SW0DNC-MNETH-B	Visual C++ 5.0
SW1DNC-MNETG-B	Visual C++ 6.0
SW1DNC-CCIEF-B	Visual C++ .NET2003

如果标准安装Visual Studio，有可能无法安装必要功能，样本程序的工程转换中有可能发生出错。

应通过自定义选择必要功能安装Visual Studio。

例)Visual Studio 2017的情况下，应勾选下述内容。

- 使用MFC的情况下：“Microsoft Foundation Classes for C++”
- 创建Windows XP中可执行的程序的情况下：“Windows XP Support for C++(支持C++相关的Windows XP)”、“Windows Universal CRT SDK”

关于安装时的功能的选择方法，或安装后的功能的添加方法，请参阅Microsoft的主页。

要点

样本程序未包含包含文件及库文件。

应预先添加包含文件及库文件之后再使用。

关于设置方法，请参阅下述章节。

-  15页 使用Visual C++ 5.0、Visual C++ 6.0的情况下
-  18页 使用Visual C++ .NET 2003、Visual C++ 2005、Visual C++ 2008的情况下
-  20页 使用Visual Studio 2010、Visual Studio 2012的Visual C++的情况下
-  27页 使用Visual Studio 2013、Visual Studio 2015及Visual Studio 2017的Visual C++的情况下

6 软元件监视实用程序

软元件监视实用程序是用于监视软元件，更改当前值的实用程序。

软元件监视实用程序使用MELSEC数据链接库监视软元件。

关于软元件监视实用程序的可访问软元件，请参阅下述章节。

☞ 38页 可访问软元件

软元件监视实用程序的安装

软元件监视实用程序包含在下述软件包中。同一计算机中已安装的情况下，将启用较新软件包的软元件监视实用程序。

关于动作环境及安装方法，请参阅各软件包的手册。

软件包	参阅
SW1DNC-CCBD2-B	☞ Q80BD-J61BT11N/Q81BD-J61BT11型CC-Link系统主站/本地站接口板用户手册 (SW1DNC-CCBD2-B系列)
SW0DNC-MNETH-B	☞ MELSECNET/H接口板用户手册 (SW0DNC-MNETH-B系列)
SW1DNC-MNETG-B	☞ CC-Link IE控制网络接口板用户手册 (SW1DNC-MNETG-B系列)
SW1DNC-CCIEF-B	☞ CC-Link IE现场网络接口板用户手册 (对应SW1DNC-CCIEF-B)

6.1 操作流程

监视时的基本操作流程如下所示。



要点

- 在软件监视实用程序中，不能监视CC-Link IE Field网络远程设备站的缓冲存储器。
- 软件监视实用程序中发生了出错的情况下，应通过出错代码一览确认发生的出错代码。(127页 出错代码)

6.2 菜单一览

以下介绍软件监视实用程序的功能有关内容。

[Menu(菜单)]		参阅
⇒[Batch monitor(批量监视)]	对1种类型的软元件进行批量监视。	119页 批量监视
⇒[16-point register monitor(16点登录监视)]	以16点单位同时监视多个软元件。	119页 16点登录监视
⇒[Exit(结束)]	结束软元件监视实用程序。	116页 启动及结束
[Setting(设置)]		参阅
⇒[Network setting(网络设置)]	设置监视的可编程控制器CPU或板。	120页 设置监视目标
⇒[Device setting(软元件设置)]	设置监视的软元件。	121页 设置监视的软元件
[Device Write(软元件写入)]		参阅
⇒[Data changing(数据更改)]	更改字软元件/双字软元件的值。	122页 更改字软元件/双字软元件值
⇒[Continuous change in data(数据连续更改)]	将连续的字软元件/双字软元件更改为相同的值。	
⇒[Bit device setting(位软元件设置)]	将位软元件置为ON。	124页 将位软元件置为ON、OFF
⇒[Bit device resetting(位软元件复位)]	将位软元件置为OFF。	
[Changing display(显示切换)]		参阅
⇒[Word device(字软元件)]⇒[(Display format(各显示格式))]	更改字软元件的显示格式。	125页 切换显示格式
⇒[Double Word device(双字软元件)]⇒[(Display format(各显示格式))]	更改双字软元件的显示格式。	
⇒[Bit device(位软元件)]⇒[(Display format(各显示格式))]	更改位软元件的显示格式。	
[Options(选项)]		参阅
⇒[Numerical Pad(数值输入板)]	对数值输入板的使用设置进行设置/解除。	126页 通过数值输入板输入
[Help(帮助)]		参阅
⇒[Help(帮助)]	显示帮助。	118页 帮助
⇒[About(版本信息)]	显示版本等的产品信息。	

6.3 画面配置及基本操作

以下介绍软元件监视实用程序的画面配置及基本操作有关内容。

启动及结束

以下介绍实用程序的启动/结束的操作方法有关内容。

启动

启动软元件监视实用程序。

可以通过各板用的实用程序的菜单或Windows的开始进行启动。

■通过CC-Link Ver.2实用程序、MNETH实用程序启动

操作步骤

1. 点击[Device Monitor (软元件监视)]按钮。

■通过CC IE Control实用程序启动

操作步骤

1. 在系统菜单中选择[Device Monitor Utility (软元件监视实用程序)]。

■通过CC IE Field实用程序启动

操作步骤

1. 选择[Online (在线)]⇒[Device Monitor Utility (软元件监视实用程序)]。

■通过Windows的开始启动

操作步骤

1. Windows的开始*1⇒选择[MELSEC]⇒[Device Monitor Utility (Board) (软元件监视实用程序 (Board))]

*1 [Start screen (开始画面)]⇒[All apps (所有应用)]或[Start (开始)]⇒[All Programs (所有程序)]

结束

结束软元件监视实用程序。

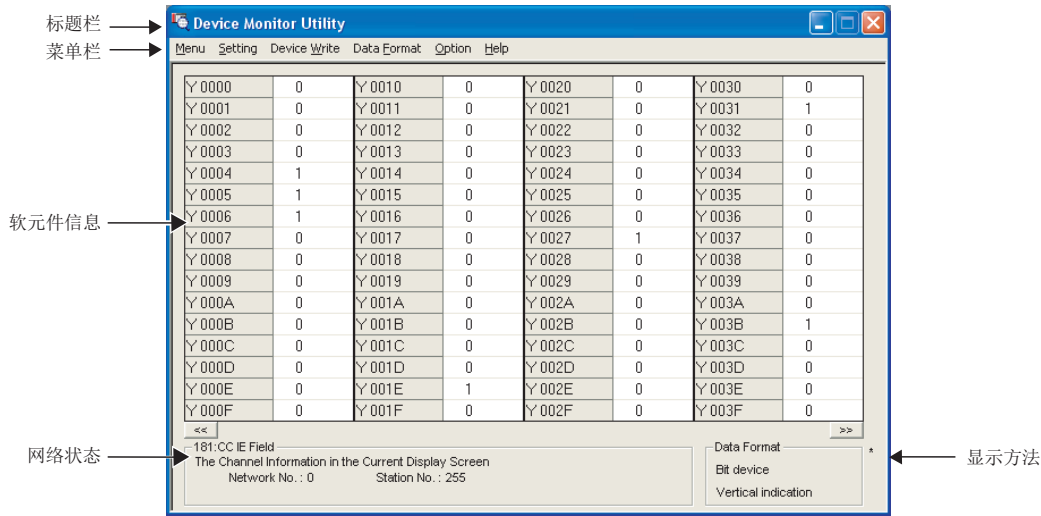
操作步骤

1. 选择[Menu (菜单)]⇒[Exit (结束)]。

总体画面配置及基本操作

软元件监视实用程序的总体画面的画面配置如下所示。

画面显示



显示内容

项目	内容
标题栏	显示工程名等。
菜单栏	显示执行各功能的菜单。
软元件信息	显示当前的软元件的状态。 更改监视方法的情况下，请参阅下述章节。 ☞ 119页 选择监视方法 更改当前值的情况下，请参阅下述章节。 ☞ 122页 更改当前值
网络状态	显示当前设置的网络的状态。 设置网络的情况下，请参阅下述章节。 ☞ 120页 设置监视目标
显示方法	显示正在显示的软元件的类型及显示格式。 更改软元件的类型的情况下，请参阅下述章节。 ☞ 121页 设置监视的软元件 更改显示格式的情况下，请参阅下述章节。 ☞ 125页 切换显示格式

要点

- 将本站设置为监视对象的情况下 网络状态中，网络No. 显示为“0”，站号显示为“255”。

帮助

显示帮助

显示软件元件监视实用程序的帮助画面。

操作步骤

1. 选择[Help(帮助)]⇒[Manual(手册)]。

确认版本

显示软件元件监视实用程序的软件版本等的信息。

操作步骤

1. 选择[Help(帮助)]⇒[About(版本信息)]。



6.4 选择监视方法

以下介绍软件监视实用程序中的监视方法的选择有关内容。

批量监视

批量监视时，指定1种类型的软元件进行监视。

画面显示

 [Menu(菜单)]⇒[Batch monitor(批量监视)]


<位软件的情况下>

Y 0000	0	Y 0010	0	Y 0020	0	Y 0030	0
Y 0001	0	Y 0011	0	Y 0021	0	Y 0031	1
Y 0002	0	Y 0012	0	Y 0022	0	Y 0032	0
Y 0003	0	Y 0013	0	Y 0023	0	Y 0033	0
Y 0004	1	Y 0014	0	Y 0024	0	Y 0034	0
Y 0005	1	Y 0015	0	Y 0025	0	Y 0035	0
Y 0006	1	Y 0016	0	Y 0026	0	Y 0036	0
Y 0007	0	Y 0017	0	Y 0027	1	Y 0037	0
Y 0008	0	Y 0018	0	Y 0028	0	Y 0038	0
Y 0009	0	Y 0019	0	Y 0029	0	Y 0039	0
Y 000A	0	Y 001A	0	Y 002A	0	Y 003A	0
Y 000B	0	Y 001B	0	Y 002B	0	Y 003B	1
Y 000C	0	Y 001C	0	Y 002C	0	Y 003C	0
Y 000D	0	Y 001D	0	Y 002D	0	Y 003D	0
Y 000E	0	Y 001E	1	Y 002E	0	Y 003E	0
Y 000F	0	Y 001F	0	Y 002F	0	Y 003F	0

<字软件的情况下>

Ww 0000	0	Ww 0010	0	Ww 0020	0	Ww 0030	0
Ww 0001	0	Ww 0011	0	Ww 0021	0	Ww 0031	0
Ww 0002	0	Ww 0012	0	Ww 0022	0	Ww 0032	0
Ww 0003	0	Ww 0013	0	Ww 0023	0	Ww 0033	0
Ww 0004	0	Ww 0014	0	Ww 0024	0	Ww 0034	0
Ww 0005	10	Ww 0015	0	Ww 0025	0	Ww 0035	0
Ww 0006	0	Ww 0016	0	Ww 0026	0	Ww 0036	0
Ww 0007	0	Ww 0017	0	Ww 0027	0	Ww 0037	0
Ww 0008	0	Ww 0018	0	Ww 0028	0	Ww 0038	0
Ww 0009	0	Ww 0019	255	Ww 0029	0	Ww 0039	0
Ww 000A	0	Ww 001A	0	Ww 002A	0	Ww 003A	0
Ww 000B	0	Ww 001B	0	Ww 002B	0	Ww 003B	0
Ww 000C	0	Ww 001C	0	Ww 002C	0	Ww 003C	0
Ww 000D	0	Ww 001D	0	Ww 002D	0	Ww 003D	0
Ww 000E	0	Ww 001E	0	Ww 002E	0	Ww 003E	0
Ww 000F	0	Ww 001F	0	Ww 002F	0	Ww 003F	0

显示内容

项目	内容
软元件信息*1	— 软元件名 监视结果
	显示当前的软元件的状态。 显示监视对象的软元件名。 显示各软元件的监视结果。 • 位软元件的情况下 0: OFF, 1: ON • 字软元件/双字软元件的情况下 设置的显示格式的值 ( 125页 切换显示格式)
[<<]/[>>]按钮	滚动显示范围。

*1 显示列数根据连接目标CPU及显示格式的设置而有所不同。


16点登录监视

16点登录监视时，指定不同类型的软元件，以16点单位进行监视。

最多可以同时监视5种位软元件、1种字软元件/双字软元件。

各软元件的显示内容与批量监视的情况下相同。

画面显示

 [Menu(菜单)]⇒[16-point register monitor(16点登录监视)]

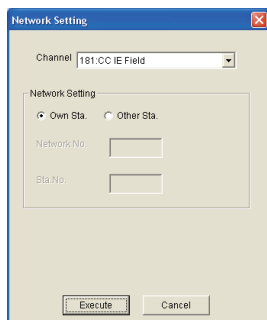
X 0000	0	X 0010	0	Y 0000	0	Y 0010	0	Ww 0000	0
X 0001	0	X 0011	0	Y 0001	0	Y 0011	0	Ww 0001	0
X 0002	0	X 0012	0	Y 0002	0	Y 0012	0	Ww 0002	0
X 0003	0	X 0013	0	Y 0003	0	Y 0013	0	Ww 0003	0
X 0004	0	X 0014	0	Y 0004	1	Y 0014	0	Ww 0004	0
X 0005	0	X 0015	0	Y 0005	1	Y 0015	0	Ww 0005	10
X 0006	0	X 0016	0	Y 0006	1	Y 0016	0	Ww 0006	0
X 0007	0	X 0017	0	Y 0007	0	Y 0017	0	Ww 0007	0
X 0008	0	X 0018	0	Y 0008	0	Y 0018	0	Ww 0008	0
X 0009	0	X 0019	0	Y 0009	0	Y 0019	0	Ww 0009	0
X 000A	0	X 001A	0	Y 000A	0	Y 001A	0	Ww 000A	0
X 000E	0	X 001B	0	Y 000B	0	Y 001B	0	Ww 000E	0
X 000C	0	X 001C	0	Y 000C	0	Y 001C	0	Ww 000C	0
X 000D	0	X 001D	0	Y 000D	0	Y 001D	0	Ww 000D	0
X 000E	0	X 001E	0	Y 000E	0	Y 001E	1	Ww 000E	0
X 000F	0	X 001F	0	Y 000F	0	Y 001F	0	Ww 000F	0

6.5 设置监视目标

设置进行监视的可编程控制器CPU或板。
应在启动软件监视实用程序时进行设置。

画面显示

 [Setting(设置)]⇒[Network setting(网络设置)]



操作步骤

1. 设置画面项目。

项目	内容
通道	设置使用的访问源的通道No.。
网络设置	设置监视对象。 访问目标为其它站的情况下应设置网络No. 及站号。

2. 点击[Execute(执行)]按钮。

要点

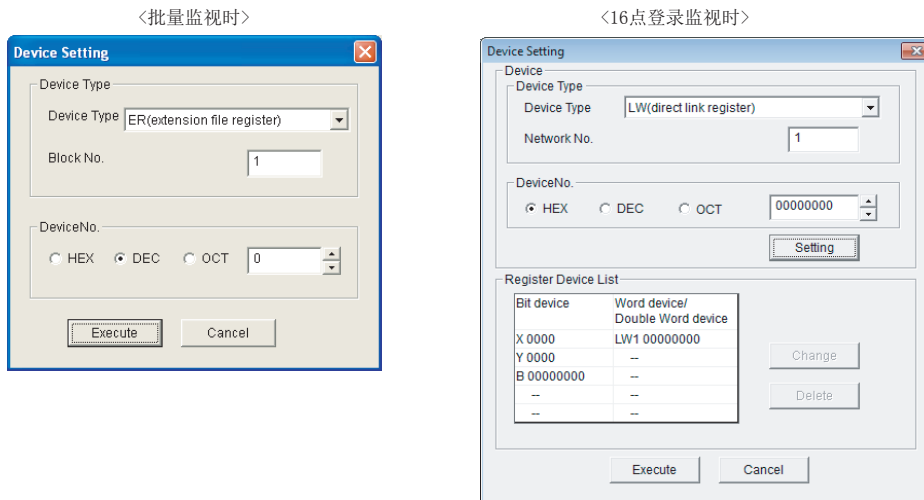
- 访问多CPU系统的情况下 应在网络No. 中输入“0”，在站号中输入各实用程序中设置的“逻辑站号”的值。
- 监视目标中请勿指定CC-Link的远程I/O站及智能设备站。指定时将发生出错。

6.6 设置监视的软元件

设置进行监视的软元件

画面显示

[Setting(设置)]⇒[Device setting(软元件设置)]



操作步骤

1. 设置画面项目。

项目	内容
软元件类型	设置进行监视的软元件的类型。 应指定MELSEC数据链接库的软元件类型。(☞ 51页 软元件类型指定) • ER的情况下，应输入块No.。 • LX、LY、LB、LW、LSB、LSW的情况下，应输入网络No.。 • SPG的情况下，应以10进制数输入起始I/O No. +16的值。
软元件No.	设置进行监视的软元件的起始编号。 (HEX: 16进制数; DEC: 10进制数; OCT: 8进制数)
登录软元件一览	对登录的软元件进行一览显示。
[Setting(设置)]按钮	将“Device Type(软元件类型)”、“Device No.(软元件No.)”中设置的内容添加到“Register Device List(登录软元件一览)”中。
[Change(更改)]按钮	将“Register Device List(登录软元件一览)”中选择的软元件的设置设置到“Device Type(软元件类型)”、“Device No.(软元件No.)”中。
[Delete(删除)]按钮	删除“Register Device List(登录软元件一览)”中选择的软元件的设置。

2. 点击[Execute(执行)]按钮。

要点

- 16点登录监视中可监视的软元件 16点登录监视中可监视的软元件仅为可随机访问的软元件。指定了不可随机访问的软元件的情况下，将发生软元件类型出错。关于各软元件的随机访问的可否，请参阅下述章节。

☞ 38页 可访问软元件

6.7 更改当前值

以下介绍软元件的当前值的更改方法有关内容。

注意事项

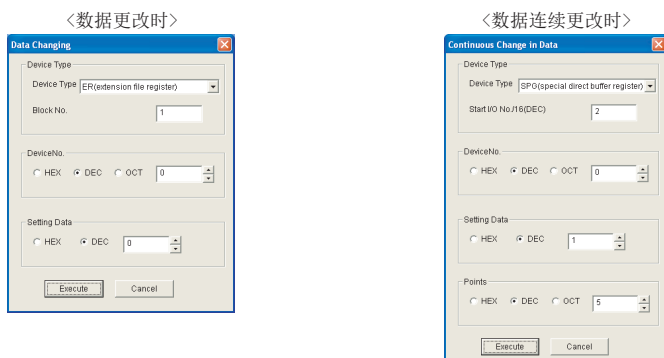
进行运行中的数据更改时，应在程序中配置互锁电路，确保整个系统始终都会安全运行。此外，应在计算机与CPU模块之间确定发生数据通信异常时系统方面的处理方法。

更改字软元件/双字软元件值

更改字软元件/双字软元件的值。在数据连续更改功能中，将连续的字软元件/双字软元件更改为相同的值。

画面显示

☞ [Device Write(软元件写入)]⇒[Data changing(数据更改)]/[Continuous change in data(数据连续更改)]



操作步骤

1. 设置当前值更改的对象。

项目	内容
软元件类型	设置更改值的软元件的类型。 应指定MELSEC数据链接库的软元件类型。(☞ 51页 软元件类型指定) <ul style="list-style-type: none"> ER的情况下，应输入块No。 LW、LSW的情况下，应输入网络No。 SPG的情况下，应以10进制数输入起始I/O No. ÷16的值。
软元件No.	设置更改值的软元件的起始编号。 (HEX: 16进制数; DEC: 10进制数; OCT: 8进制数)

2. 设置更改的值及点数。

项目	内容
设置数据	设置更改的值。 (HEX: 16进制数; DEC: 10进制数)*1

*1 选择了DEC(10进制数)时，只能指定10进制数有符号的格式。

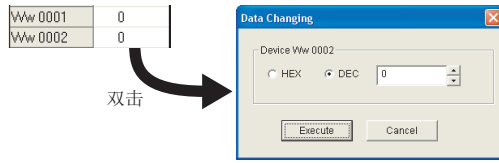
3. 数据连续更改的情况下，设置更改的软元件点数。

项目	内容
点数	设置更改值的软元件的点数。 (HEX: 16进制数; DEC: 10进制数; OCT: 8进制数)设置范围: 1~64点

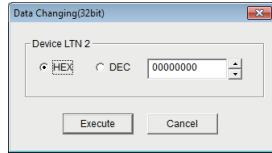
4. 点击[Execute(执行)]按钮。

要点

在监视画面中双击“软元件信息”的各软元件栏的情况下，也可进行字数据/双字数据的数据更改。




在双击后显示的数据更改画面中，显示格式为10进制显示(无16位符号)或10进制显示(无32位符号)时，可以设置与显示格式相同格式的值。

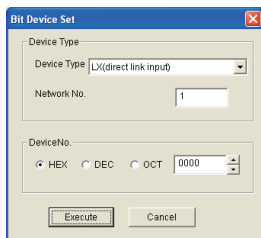


将位软元件置为ON、OFF

更改位软元件的ON/OFF。

画面显示

 [Device Write(软元件写入)]⇒[Bit device set(位软元件设置)]/[Bit device reset(位软元件复位)]



操作步骤

1. 设置当前值更改的对象。

项目	内容
软元件类型	设置更改值的软元件的类型。 应指定MELSEC数据链接库的软元件类型。(☞ 51页 软元件类型指定) • LX、LY、LB、LSB的情况下，应输入网络No.。
软元件No.	设置更改值的软元件的起始编号。 (HEX: 16进制数; DEC: 10进制数; OCT: 8进制数)

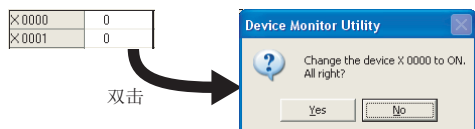
2. 点击[Execute(执行)]按钮。

要点

- 关于可更改的软元件 本功能中可更改的软元件仅为可随机访问的软元件。指定了不可随机访问的软元件的情况下，将发生软元件类型出错。关于各软元件的随机访问的可否，请参阅下述章节。

☞ 38页 可访问软元件

- 在监视画面中双击“软元件信息”的各软元件栏的情况下，也可进行位软元件的ON/OFF。



6.8 切换显示格式

切换监视时的软元件信息的显示格式(16进制、10进制等)。
批量监视及16点登录监视中可选择的菜单如下所示。

操作步骤

1. 选择[Changing display(显示切换)]⇒[Word device(字软元件)]/[Double Word device(双字软元件)]/[Bit device(位软元件)]⇒[(Display format(各显示格式))]

字软元件

○：可以显示；—：不可显示

显示格式	批量监视	16点登录监视
10进制显示(16位有符号)	○	○
10进制显示(16位无符号)	○	○
16进制显示(16位)	○	○
8进制显示(16位)	○	○
2进制显示(16位)	○	—
10进制显示(32位有符号)*1	○	○
10进制显示(32位无符号)*1	○	○
16进制显示(32位)*1	○	○
8进制显示(32位)*1	○	○
2进制显示(32位)*1	○	—

*1 显示格式为32位的情况下，将32位的数据(字软元件2点)显示为1点。

双字软元件

○：可以显示；—：不可显示

显示格式	批量监视	16点登录监视
10进制显示(32位有符号)	○	○
10进制显示(32位无符号)	○	○
16进制显示(32位)	○	○
8进制显示(32位)	○	○
2进制显示(32位)	○	—

位软元件

○：可以显示；—：不可显示

显示格式	批量监视	16点登录监视
纵向显示	○	—
横向显示(F-0)(10进制显示)	○	—
横向显示(F-0)(16进制显示)	○	—
横向显示(O-F)(10进制显示)	○	—
横向显示(O-F)(16进制显示)	○	—

6.9 通过数值输入板输入

以下介绍用于数值输入的选项功能有关内容。

使用数值输入板时，设置软元件值等的数值输入栏的情况下，只能通过鼠标操作输入数值。

对数值输入板的使用进行设置/设置解除

操作步骤

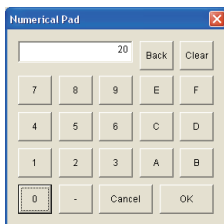
1. [Options(选项)]⇒[Numerical Pad(数值输入板)]

此后，点击数值输入栏时，将显示数值输入板画面。再次选择本菜单时，设置将被解除。

通过数值输入板输入数值

画面显示

 点击各画面的数值输入栏。



操作步骤

1. 点击数字按钮输入数值。

2. 点击[OK]按钮。

数值将被输入到数值输入栏中。

7 出错代码

出错代码对应的出错内容及处理方法如下所示。

发生了下述出错一览表中未记载的出错代码的出错的情况下，请向当地三菱电机代理店咨询。

出错代码		出错内容	处理方法
10进制	16进制		
1	0001H	■驱动器未启动出错 驱动器未启动。	<ul style="list-style-type: none"> • 确认通道No.。 • 通过设备管理器确认驱动器的动作状态。 • 通过事件查看器确认是否发生了出错。 • 确认板的设置。 • 重新安装软件包。
2	0002H	■超时出错(板响应出错) <ul style="list-style-type: none"> • 至板的请求在超时监视时间内未完成。 • 通过64位版用户应用程序对QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块执行了请求。 • 板与软件包不一致。 • 访问远程设备站的缓冲存储器时偏置或偏置+写入/读取字节指定了超出对象站的缓冲存储器的范围的值。 • CC-Link Ver. 2板中本站的站号为64站的情况下，对其它站执行了请求。 	<ul style="list-style-type: none"> • 确认网络状态、对象站的动作状态及模块的安装状态。 • 确认板的安装状态。 • 确认实用程序的对象目标设置。 • 确认实用程序的瞬时超时监视时间设置。 • 进行重试。 • 对QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块进行访问的情况下应使用32位版用户应用程序。 • CC-Link IE现场网络板中，执行RECV功能时发生了本出错的情况下，应使用SWIDNC-CCIEF-B的Version1.04E及以上。 • 确认偏置或偏置+写入/读取字节是否在对象站的缓冲存储器的范围内。 • 使用CC-Link Ver. 2板对其它站执行请求的情况下，应将本站的站号设置为64站以外。 • 确认是否发生了CRC出错等的线路异常。关于确认方法，请参阅所使用的网络系统的手册。
66	0042H	■已OPEN出错 指定的通道已被OPEN。	<ul style="list-style-type: none"> • 将打开设置为1次。 • 由于path中存储了正确的值，因此本出错有可能被作为正常处理。
68	0044H	■路径出错 <ul style="list-style-type: none"> • 指定的路径无效。 • 没有指定的路径的板。 	<ul style="list-style-type: none"> • 确认路径。 • 通过设备管理器确认驱动器的动作状态。 • 确认板的安装状态。
69	0045H	■不支持功能执行出错 <ul style="list-style-type: none"> • 对象站中执行了不支持的功能。 • 在指定的通道中执行了不支持的功能。 	<ul style="list-style-type: none"> • 确认通道的路径、网络No.、站号。 • 确认对象站中请求的功能是否支持。
70	0046H	■站号出错 <ul style="list-style-type: none"> • 对象站的站号有错误。 • 对本站请求了应对其它站请求的处理。或站号为本站(255(FFH))但网络No.不为0。 • 通过64位版用户应用程序对QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块执行了请求。 	<ul style="list-style-type: none"> • 确认网络No.、站号。 • 对QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块进行访问的情况下应使用32位版用户应用程序。
71	0047H	■无接收数据出错(RECV功能时) <ul style="list-style-type: none"> • 未接收数据。 • CC-Link IE控制板、CC-Link IE现场板的RECV功能中，接收了超出从MELSECNET/H板发送的960字节的数据。 	<ul style="list-style-type: none"> • 确认通道No.。 • 在SEND功能的发送站中，确认是否正在发送数据。 • 确认MELSECNET/H板的SEND功能中，发送的数据是否超过了960字节。 • 接收站为CC-Link IE控制板的情况下，重新启动接收站的计算机。 • 进行重试。
77	004DH	■存储器确保出错/资源存储器不足出错 未能确保存储器。	<ul style="list-style-type: none"> • 存储器有可能不足，应结束其它正在动作的应用程序。 • 结束程序后，重新启动计算机。 • 扩大计算机的最小工作集区域。(☞ 132页 扩大计算机的最小工作集区域的方法)
85	0055H	■SEND/RECV通道No. 出错 SEND/RECV功能中指定的通道有错误。	确认通道No.。
100	0064H	■板H/W资源繁忙 板上的资源不足变为无法执行下一个处理的状态。	<ul style="list-style-type: none"> • 进行重试。 • 连续发生出错的情况下，请向当地三菱电机代理店咨询。
101	0065H	■路由异常 路由参数未正确设置。	确认路由由参数的设置。 确认指定的网络No.。

出错代码		出错内容	处理方法
10进制	16进制		
102	0066H	<p>■板驱动器I/F出错</p> <ul style="list-style-type: none"> 至板驱动器的请求数据发送失败。 OS的系统资源不足。 <p>关于详细内容, 请参阅下述章节。  12页 OS及其它应用程序的影响</p>	<ul style="list-style-type: none"> 进行重试。 通过事件日志等确认Windows®是否正常工作。 结束程序后, 重新启动计算机。 确认板的安装状态。
103	0067H	<p>■板驱动器I/F出错</p> <ul style="list-style-type: none"> 从板驱动器的响应数据接收失败。 板与软件包不一致。 	<ul style="list-style-type: none"> 进行重试。 通过事件日志等确认Windows®是否正常工作。 结束程序后, 重新启动计算机。 CC-Link IE现场网络板中, 执行SEND功能时发生了本出错的情况下, 应使用SW1DNC-CCIEF-B的Version1.04E及以上。
133	0085H	<p>■参数出错</p> <p>板中设置的参数有错误。</p>	<ul style="list-style-type: none"> 进行板复位。 重新设置参数。
4096~16383	1000H~3FFFH	<p>■MELSEC数据链接库的内部检测出的出错</p>	<ul style="list-style-type: none"> 结束程序后, 重新启动计算机。 重新安装软件包。 请向当地三菱电机代理店咨询。
16384~20479	4000H~4FFFH	<p>■对象站的可编程控制器CPU中检测出的出错</p>	<p>请参阅对象站的可编程控制器CPU的用户手册。</p>
16432	4030H	<p>■软元件指定出错</p> <p>指定的软元件类型不存在。</p>	<p>指定软元件类型一览表中记载的软元件类型。</p> <p>确认对象站的可编程控制器CPU中, 指定软元件是否有效。</p>
16433	4031H	<p>■软元件指定出错</p> <ul style="list-style-type: none"> 指定的软元件No. 超出范围。 指定的软元件的起始I/ONo. 无效。 指定的软元件的块No. 无效。 	<ul style="list-style-type: none"> 确认软元件No.。 确认指定的软元件的块No. (软元件类型)、起始I/ONo.。 确认对象站的可编程控制器CPU中, 指定软元件及块No.、起始I/ONo. 是否有效。
16512	4080H	<p>■请求数据异常</p> <p>通过64位版用户应用程序对QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块执行了请求。</p>	<ul style="list-style-type: none"> 确认指定的请求数据内容。 对QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块进行访问的情况下应使用32位版用户应用程序。
18944	4A00H	<p>■链接关联出错</p> <ul style="list-style-type: none"> 路由参数中设置的网络不存在。 对象CPU不支持网络。 对象站的网络No.、站号有错误。 	<ul style="list-style-type: none"> 重新设置路由参数的设置。 更换为支持网络的CPU。 确认网络No.、站号。
18945	4A01H		
19202	4B02H	不是至CPU模块的请求。	对可实施指定功能的模块实施操作。
-1	FFFH (FFFFFFFH)	<p>■路径出错</p> <p>指定的路径无效。</p>	<p>指定通过mdOpen函数返回的路径。</p> <p>指定支持功能的通信线路的路径。</p>
-2	FFFEH (FFFFFFFEH)	<p>■起始软元件No. 出错</p> <ul style="list-style-type: none"> 指定的起始软元件No. 超出范围。 位软元件指定时, 起始软元件No. 不是8的倍数。 软元件随机读取/写入中指定的同一块的起始软元件No. + 点数超出软元件范围。 	<ul style="list-style-type: none"> 确认起始软元件No.。 位软元件指定时, 以8的倍数指定起始软元件No.。 确认起始软元件No. + 点数。 确认对象站的可编程控制器CPU中, 指定软元件是否有效。
-3	FFFDH (FFFFFFFDH)	<p>■软元件类型出错</p> <p>指定的软元件类型不存在。</p>	<ul style="list-style-type: none"> 指定软元件类型一览表中记载的软元件类型。 确认对象站的可编程控制器CPU中, 指定软元件是否有效。
-5	FFFBH (FFFFFFFBH)	<p>■容量出错</p> <ul style="list-style-type: none"> 起始软元件No. + 容量超出软元件范围。 以奇数字字节数进行了访问。 	<ul style="list-style-type: none"> 确认软元件容量。 确认起始软元件No. + 容量。 指定为偶数字字节数。
-6	FFFAH (FFFFFFFAH)	<p>■块数出错</p> <p>软元件随机读取/写入中指定的块数超出范围。</p>	<p>确认块数。</p>
-8	FFF8H (FFFFFFF8H)	<p>■通道No. 出错</p> <p>mdOpen函数中指定的通道No. 无效。</p>	<p>确认通道No.。</p>
-12	FFF4H (FFFFFFF4H)	<p>■块No. 出错</p> <p>指定的文件寄存器的块No. 无效。</p>	<ul style="list-style-type: none"> 确认文件寄存器的块No. (软元件类型)。 确认对象站的可编程控制器CPU中, 指定软元件是否有效。
-13	FFF3H (FFFFFFF3H)	<p>■写保护出错</p> <p>指定的扩展文件寄存器的块No. 与存储卡的写保护区域重复。</p>	<ul style="list-style-type: none"> 确认扩展文件寄存器的块No. (软元件类型)。 确认对象站的可编程控制器CPU中, 存储卡的写保护开关。
-16	FFF0H (FFFFFFF0H)	<p>■网络No.、站号出错</p> <p>指定的网络No.、站号超出范围。</p>	<p>确认网络No.、站号。</p>
-17	FFFEH (FFFFFFFEH)	<p>■全站、组No. 指定出错</p> <ul style="list-style-type: none"> 对SEND功能以外的功能进行了全站指定、组No. 指定。 全站指定、组No. 指定中使用SEND功能时, 将软元件类型指定为了“有到达确认”。 	<ul style="list-style-type: none"> 确认网络No.、站号。 确认是否为可进行全站指定、组No. 指定的功能。 全站指定、组No. 指定中使用SEND功能的情况下, 将软元件类型指定为“无到达确认”。
-18	FFEEH (FFFFFFEEH)	<p>■远程指示代码出错</p> <p>mdControl中指定了无效的指示代码。</p>	<p>确认指示代码。</p>

出错代码		出错内容	处理方法
10进制	16进制		
-19	FFEDH (FFFFFFEDH)	■SEND/RECV通道No. 出错 SEND/RECV功能中指定的通道No. 超出范围。	确认通道No.。
-31	FFE1H (FFFFFFE1H)	■DLL载入出错 用于执行函数所需的DLL的载入失败。	重新安装软件包。
-32	FFE0H (FFFFFFE0H)	■资源超时出错 • 强制结束了用户程序。 • 其它任务/线程占用资源, 瞬时超时监视时间内未能释放资源。	<ul style="list-style-type: none"> • 正确结束用户程序。 • 进行重试。 • 有可能存储器不足, 因此应结束其它正在动作的应用程序。 • 结束程序后, 重新启动计算机。
-33	FFDFH (FFFFFFDFH)	■通信目标不支持出错 网络No.、站号中指定的通信目标是不支持的机型。	<ul style="list-style-type: none"> • 确认网络No.、站号是否指定了不支持的通信目标。 • 确实用程序中设置的对象目标设置的内容。 • 更新软件包。
-34	FFDEH (FFFFFFDEH)	■注册表访问出错	重新安装软件包。
-35	FFDDH (FFFFFFDDH)		
-36	FFDCH (FFFFFFDCH)		
-37	FFDBH (FFFFFFDBH)	■通信初始化设置出错 用于进行通信的初始设置失败。	<ul style="list-style-type: none"> • 进行重试。 • 存储器有可能不足, 应结束其它正在动作的应用程序。 • 结束程序后, 重新启动计算机。 • 确认存储器的可用空间。
-42	FFD6H (FFFFFFD6H)	■关闭出错 无法关闭通信。	<ul style="list-style-type: none"> • 进行重试。 • 结束程序后, 重新启动计算机。
-43	FFD5H (FFFFFFD5H)	■ROM运行出错 对ROM运行中的CPU进行了TC设置值的写入。	在RAM运行中进行TC设置值的更改。
-61	FFC3H (FFFFFFC3H)	■事件数出错 mdWaitBdEvent函数中指定的等待事件的个数超出范围。	确认等待事件的个数。
-62	FFC2H (FFFFFFC2H)	■事件No. 出错 mdWaitBdEvent函数中指定的等待事件No. 超出范围。	确认等待事件No.。
-63	FFC1H (FFFFFFC1H)	■事件No. 登录重复出错 mdWaitBdEvent函数中指定的等待事件No. 重复。	指定为不重复的等待事件No.。
-64	FFC0H (FFFFFFC0H)	■超时时间出错 mdWaitBdEvent函数中指定的超时时间超出范围。	确认超时时间。
-65	FFBFH (FFFFFFBFH)	■事件等待超时出错 超时时间内未发生事件。	进行重试。
-66	FFBEH (FFFFFFBEH)	■事件初始化出错 mdWaitBdEvent函数执行中实施了板复位或主站(管理站)的复位。	进行重试。
-67	FFBDH (FFFFFFBDH)	■无事件设置出错 mdWaitBdEvent函数中指定的等待事件No. 的事件设置不存在。	通过实用程序设置等待事件No.。
-69	FFBBH (FFFFFFBBH)	■不支持功能执行出错 执行了软件包或驱动器不支持的功能。	确认是否为软件包或驱动器支持的功能。
-70	FFBAH (FFFFFFBAH)	■事件发生重复出错 同一事件No. 的事件发生了多次。	将事件发生间隔延长至用户程序可充分处理程度。
-71	FFB9H (FFFFFFB9H)	■远程设备站访问出错 远程设备站的缓冲存储器访问失败。	<ul style="list-style-type: none"> • 确认对象站是否为CC-Link IE现场网络的远程设备站。 • 确认网络No.、站号。 • 确认偏置或偏置+写入/读取字节是否在对象站的缓冲存储器的范围内。
-257~-4096	FEFFH~F000H (FFFFFFFFH)~ (FFF000H)	■MELSECNET/10、MELSECNET/H网络系统中检测出的出错	请参阅MELSECNET/10、MELSECNET/H网络系统的参考手册。
-2174	F782H (FFF782H)	■瞬时数据对象站号异常 • 对象站号未正确设置。 • 对象站号为0。	<ul style="list-style-type: none"> • 确认对象站号后, 再次执行。 • 上述处理后仍然异常的情况下, 请向当地三菱电机代理店咨询。
-4097~-8192	EFFFH~E000H (FFFEEFFH)~ (FFF000H)	■CC-Link IE控制网络系统中检测出的出错	发生了本表中未记载的出错的情况下, 请参阅CC-Link IE控制网络的手册。

出错代码		出错内容	处理方法
10进制	16进制		
-7656	E218H (FFFFE218H)	■瞬时数据对象站号异常 • 对象站号未正确设置。 • 对象站号为0。	• 确认对象站号后, 再次执行。 • 上述处理后仍然异常的情况下, 请向当地三菱电机代理店咨询。
-7672	E208H (FFFFE208H)		
-8193~-12288	DFFFH~D000H (FFFFDFFFH)~ (FFFFD000H)	■CC-Link IE现场网络系统中检测出的出错	发生了本表中未记载的出错的情况下, 请参阅CC-Link IE现场网络接口板及CC-Link IE现场网络主站本地站模块的用户手册。
-11683	D25DH (FFFFD25DH)	■瞬时数据不正确 从MELSECNET/H板对CC-Link IE现场网络板发送了超过960字节的数据。	确认MELSECNET/H板的SEND功能中, 是否发送了超过960字节的数据。
-11717	D23BH (FFFFD23BH)	■网络No. 出错 mdRemBufReadEx函数/mdRemBufWriteEx函数中, 指定了本网络以外的网络No.。	确认网络No. 是否指定了不支持的通信目标。
-11746	D21EH (FFFFD21EH)	■站号出错 • 对象站的站号有错误。 • 对本站请求了应对其它站请求的处理。或站号为本站(255(FFH))但网络No. 不为0。	确认网络No.、站号。
-12128	D0A0H (FFFFD0A0H)	■瞬时数据发送响应等待超时异常	• 确认本站或对象站是否被切断。 • 确认网络No.、站号是否指定了不支持的通信目标。
-12289~-16384	CFFFH~C000H (FFFFCFFFH)~ (FFFFC000H)	■以太网网络系统中检测出的出错	请参阅以太网接口模块的用户手册。
-16385~-20480	BFFFH~B000H (FFFFBFFFH)~ (FFFFB000H)	■CC-Link系统中检测出的出错	发生了本表中未记载的出错的情况下, 请参阅CC-Link系统主站本地站接口板及CC-Link系统主站本地站模块的用户手册。
-18560	B780H (FFFFB780H)	■模块模式设置出错 对远程I/O站发出了瞬时请求。	确认网络No.、站号。
-18572	B774H (FFFFB774H)	■瞬时不支持出错 对不是智能设备站的站发送了瞬时请求。	• 确认网络No.、站号。 • 指定智能设备站的站号。 • 确认软元件类型。
-25056	9E20H (FFFF9E20H)	■处理代码出错 • 设置了请求目标站无法处理的处理代码。(请求目标链接模块检查。) • 通过64位版用户应用程序对QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块执行了请求。	• 确认请求目标站号及处理代码。 • 请勿对其它站的板执行mdTypeRead函数以外的函数。 • 对QCPU(Q模式)、MELSEC iQ-R系列CPU模块以外的CPU模块进行访问的情况下应使用32位版用户应用程序。
-26334	9922H (FFFF9922H)	■复位出错 • 本站或其它站访问中, 对正在使用同一通道的其它任务进行了复位。 • 通过实用程序的监视中, 进行了复位操作。	进行重试。
-26336	9920H (FFFF9920H)	■至不支持路由功能站的路由请求出错 对不支持路由功能的站进行了至其它环路的路由请求。	确认路由参数的设置。
-27902	9302H (FFFF9302H)	■事件等待超时出错 mdWaitBdEvent函数中指定的超时的时间内未发生事件。	• 延长超时时间。 • 确认事件的发生条件。
-28138	9216H (FFFF9216H)	■不支持站单位块保证 将站单位块保证被设置为启用的参数写入到ROM版本为1A的CC-Link Ver. 2板中, 对板进行了启动及复位。	• 更换为ROM版本2B及以上的板。 • 将站单位块保证设置为禁用。
-28139	9215H (FFFF9215H)	■链接刷新出错 链接刷新处理未正常动作。	• 对板进行复位。 • 存储器有可能不足, 应结束其它正在动作的应用程序。 • 结束程序后, 重新启动计算机。 • 确认存储器的可用空间。 • 确认板的安装状态。 • 在其它计算机中正常动作的情况下, 可能是计算机的异常, 应修理或更换计算机。 • 在其它计算机中也发生同一出错的情况下, 应更换板。 • 请向当地三菱电机代理店咨询。
-28140	9214H (FFFF9214H)	■不正确模式设置出错 模式设置时, 指定了不正确的模式。	确认模式。
-28141	9213H (FFFF9213H)	■系统睡眠出错 检测到至睡眠、休眠的转移, 或通过快速启动的启动。	• 结束程序后, 重新启动计算机。 • 更改电源选项的设置, 避免发生至睡眠、休眠的转移, 或通过快速启动的启动。
-28142	9212H (FFFF9212H)	■模式出错 执行了当前设置的模式中无法使用的请求。	• 确认是否设置了参数。 • 确认当前设置的模式。

出错代码		出错内容	处理方法
10进制	16进制		
-28143	9211H (FFFF9211H)	■硬件自诊断出错 硬件自诊断中检测出异常。	<ul style="list-style-type: none"> • 确认事件查看器的系统日志，进行登录的出错的处理。 • 结束程序后，重新启动计算机。 • 采取计算机的防噪声对策。 • 将计算机的电源置为OFF之后再行连接器的插拔。 • 确认板的安装状态。 • 在其它计算机中正常动作的情况下，可能是计算机的异常，应修理或更换计算机。 • 在其它计算机中也发生同一出错的情况下，应更换板。 • 请向当地三菱电机代理店咨询。
-28144	9210H (FFFF9210H)		
-28150	920AH (FFFF920AH)	■数据链接中断站的软元件访问出错 对本站软元件RX、RY、Rww、Rwr的数据链接中断站或保留站中分配的软元件范围进行了访问。	<ul style="list-style-type: none"> • 确认指定的软元件起始No.、容量或主站的参数的软元件范围。 即使发生了本出错，也可进行数据的写入、读取，但不能保证数据。
-28151	9209H (FFFF9209H)	■异常数据接收出错 接收了异常的响应数据。	<ul style="list-style-type: none"> • 确认对象站及对象站的可编程控制器CPU中是否发生了出错。如果正常则再次发出请求。 • 参阅各产品的手册，确认网络的状态。
-28158	9202H (FFFF9202H)	■驱动器WDT出错 驱动器WDT出错发生中。 对于驱动器WDT出错，由于下述原因，系统负载有可能发生暂时上升。 <ul style="list-style-type: none"> • 计算机启动时的Windows®启动处理 • 图形板等的设备驱动程序的动作 • 其它软件等的动作 	<ul style="list-style-type: none"> • 进行板复位。 • 进行计算机的重新启动。 • 消除导致系统高负载的原因。 • 在实用程序中取消“Use driver WDT function(使用驱动器WDT功能)”的勾选，禁用WDT。或延长驱动器WDT监视时间。 由于图形板等的更改，有可能变为不发生驱动器WDT出错的环境。
-28622	9032H (FFFF9032H)	■专用指令通道使用中异常 本站使用通道或对象站存储通道正在被其它指令使用。	应稍等片刻后重试。 更改本站使用通道或对象站存储通道。
-28634	9026H (FFFF9026H)	■硬件自诊断出错 硬件自诊断中检测出异常。	<ul style="list-style-type: none"> • 确认事件查看器的系统日志，进行登录的出错的处理。 • 结束程序后，重新启动计算机。 • 确认板的安装状态。 • 在其它计算机中正常动作的情况下，可能是计算机的异常，应修理或更换计算机。 • 在其它计算机中也发生同一出错的情况下，应更换板。 • 请向当地三菱电机代理店咨询。
-28636	9024H (FFFF9024H)		

要点

作为扩展函数(☞ 44页 函数一览)的返回值返回的情况下
 在16进制表示中，出错代码-1~-28636变为表中的8位的值(FFFFFFFFH~FFFF9024H)。

附录

附1 扩大计算机的最小工作集区域的方法

作为MELSEC数据链接库函数的执行中发生了出错代码77 (004DH) 的出错时的处理，扩大计算机的最小工作集区域时的方法及样本程序如下所示。

计算机用板的驱动器使用通过用户程序确保的存储器区域的最小工作集区域执行动作。

根据用户程序，有可能使用较大的最小工作集区域。

由此，无法确保计算机用板的驱动器使用的最小工作集区域时，将返回出错代码77。

发生了此类状况的情况下，执行MELSEC数据链接库函数前应通过用户程序扩大最小工作集区域。(☞ 133页 样本程序)

计算机启动时，作为最小工作集区域确保200KB的区域。

将本手册中介绍的样本程序用于实际系统的情况下，应充分验证对象系统中不存在控制方面的问题。

样本程序的处理概要

1. 通过GetCurrentProcessId函数获取用户程序的ID。
2. 通过1. 中获取的用户程序的ID，通过OpenProcess函数获取用户程序的句柄。
3. 希望获取当前的最小工作集、最大工作集的值的情况下，执行GetProcessWorkingSetSize函数。
4. 设置大于3. 中获取的最小工作集的值，执行SetProcessWorkingSetSize函数。
5. 通过CloseHandle函数打开用户程序的句柄。

样本程序

通过Visual Basic进行设置的情况下

■最小工作集值1MB、最大工作集值3MB的示例

通过Visual Basic 5.0、Visual Basic 6.0编程时，应将变量(id、ph、wkmin、wkmax)的类型定义从Integer更改为Long。此处的设置值为参考标准。应根据使用的系统调整设置值。

程序示例

```
Dim id As Integer           '用户程序的ID用变量
Dim ph As Integer          '用户程序的句柄用变量
Dim wkmin As Integer       '最小工作集用变量
Dim wkmax As Integer       '最大工作集用变量
Dim bret As Boolean        '返回值

'获取用户程序的ID
id = GetCurrentProcessId()

'打开用户程序的句柄
PROCESS_SET_QUOTA = 256, PROCESS_QUERY_INFORMATION = 1024
ph = OpenProcess(256 + 1024, False, id)

'获取用户程序的最大工作集值及最小工作集值
bret = GetProcessWorkingSetSize(ph, wkmin, wkmax)

'将最小工作集值设置为1MB(1 * 1024 * 1024 = 1048576)
wkmin = 1048576

'将最大工作集值设置为3MB(3 * 1024 * 1024 = 3145728)
wkmax = 3145728

'更改用户程序的最大工作集值及最小工作集值
bret = SetProcessWorkingSetSize(ph, wkmin, wkmax)

'关闭用户程序的句柄
bret = CloseHandle(ph)
```

通过Visual C++进行设置的情况下

■最小工作集值1MB、最大工作集值3MB的示例

此处的设置值为参考标准。应根据使用的系统调整设置值。

程序示例

```
#define ERROR -1
short ChangeWorkingSetSize()
{
    DWORD dwProcessId;          /*用户程序的ID用变量*/
    HANDLE hProcess;           /*用户程序的句柄用变量*/
    DWORD dwMinimumWorkingSetSize; /*最小工作集用变量*/
    DWORD dwMaximumWorkingSetSize; /*最大工作集用变量*/
    /*获取用户程序的ID*/
    dwProcessId = GetCurrentProcessId();
    /*打开用户程序的句柄*/
    hProcess = OpenProcess(PROCESS_SET_QUOTA+PROCESS_QUERY_INFORMATION, FALSE, dwProcessId);
    if (hProcess == NULL) {
        /*出错结束*/
        return (ERROR);
    }
    /*获取用户程序的最大工作集值及最小工作集值*/
    if (GetProcessWorkingSetSize(hProcess, &dwMinimumWorkingSetSize, &dwMaximumWorkingSetSize)==0) {
        /*出错结束*/
        CloseHandle(hProcess);
        return (ERROR);
    }
    /*将最小工作集值设置为1MB*/
    dwMinimumWorkingSetSize = 1 * 1024 * 1024;
    /*将最大工作集值设置为3MB*/
    dwMaximumWorkingSetSize = 3 * 1024 * 1024;
    /*更改用户程序的最大工作集值及最小工作集值*/
    if (SetProcessWorkingSetSize(hProcess, dwMinimumWorkingSetSize, dwMaximumWorkingSetSize)==0) {
        /*出错结束*/
        CloseHandle(hProcess);
        return (ERROR);
    }
    /*关闭用户程序的句柄*/
    CloseHandle(hProcess);
    /*正常返回*/
    return (0);
}
```


函数索引

M

mdBdLedRead (板LED读取)	87
mdBdModRead (板模式读取)	86
mdBdModSet (板模式设置)	84
mdBdRst (板复位)	83
mdBdSwRead (板开关状态读取)	93
mdBdVerRead (板版本读取)	94
mdClose (通信线路的关闭)	56
mdControl (远程RUN/STOP/PAUSE)	79
mdDevRst (位软元件复位)	109
mdDevRstEx (扩展位软元件复位)	72
mdDevSet (位软元件设置)	108
mdDevSetEx (扩展位软元件设置)	71
mdInit (可编程控制器软元件信息表的初始化)	96
mdOpen (通信线路的打开)	55
mdRandR (软元件随机读取)	107
mdRandREx (扩展软元件随机读取)	68
mdRandW (软元件随机写入)	105
mdRandWEx (扩展软元件随机写入)	65
mdReceive (RECV功能)	103
mdReceive (软元件批量读取)	101
mdReceiveEx (扩展软元件批量读取)	61
mdReceiveEx (RECV功能)	63
mdRemBufReadEx (远程设备站的缓冲存储器读取)	74
mdRemBufWriteEx (远程设备站的缓冲存储器写入)	73
mdSend (软元件批量写入)	97
mdSend (SEND功能)	99
mdSendEx (扩展软元件批量写入)	57
mdSendEx (SEND功能)	59
mdTypeRead (CPU型号读取)	75
mdWaitBdEvent (事件等待)	80

修订记录

*本手册号在封底的左下角。

修订日期	*手册编号	修改内容
2019年9月	SH (NA)-082183CHN-A	第一版
2020年7月	SH (NA)-082183CHN-B	第二版
2022年2月	SH (NA)-082183CHN-C	第三版

日文手册编号：SH-081034-Q

本手册不授予工业产权或任何其它类型的权利，也不授予任何专利许可。三菱电机对由于使用了本手册中的内容而引起的涉及工业产权的任何问题不承担责任。

©2019 MITSUBISHI ELECTRIC CORPORATION

质保

使用之前请确认以下产品质保的详细说明。

1. 免费质保期限和免费质保范围

在免费质保期内使用本产品时如果出现任何属于三菱电机责任的故障或缺陷（以下称“故障”），则经销商或三菱电机服务公司负责免费维修。

但是如果需要在国内现场或海外维修时，则要收取派遣工程师的费用。对于涉及到更换故障模块后的任何再试运转、维护或现场测试，三菱电机将不负任何责任。

[免费质保期限]

免费质保期限为自购买日或交货的一年内。

注意产品从三菱电机生产并出货之后，最长分销时间为 6 个月，生产后最长的免费质保期为 18 个月。维修零部件的免费质保期不得超过修理前的免费质保期。

[免费质保范围]

(1) 范围局限于按照使用手册、用户手册及产品上的警示标签规定的使用状态、使用方法和使用环境正常使用的情况下。

(2) 以下情况下，即使在免费质保期内，也要收取维修费用。

- ① 因不当存储或搬运、用户过失或疏忽而引起的故障。因用户的硬件或软件设计而导致的故障。
- ② 因用户未经批准对产品进行改造而导致的故障等。
- ③ 对于装有三菱电机产品的用户设备，如果根据现有的法定安全措施或工业标准要求配备必需的功能或结构后本可以避免的故障。
- ④ 如果正确维护或更换了使用手册中指定的耗材（电池、背光灯、保险丝等）后本可以避免的故障。
- ⑤ 因火灾或异常电压等外部因素以及因地震、雷电、大风或水灾等不可抗力而导致的故障。
- ⑥ 根据从三菱电机出货时的科技标准还无法预知的原因而导致的故障。
- ⑦ 任何非三菱电机或用户责任而导致的故障。

2. 产品停产后的有偿维修期限

(1) 三菱电机在本产品停产后的 7 年内受理该产品的有偿维修。

停产的消息将以三菱电机技术公告等方式予以通告。

(2) 产品停产，将不再提供产品（包括维修零件）。

3. 海外服务

在海外，维修由三菱电机在当地的海外 FA 中心受理。注意各个 FA 中心的维修条件可能会不同。

4. 机会损失和间接损失不在质保责任范围内

无论是否在免费质保期内，凡以下事由三菱电机将不承担责任。

- (1) 任何非三菱电机责任原因而导致的损失。
- (2) 因三菱电机产品故障而引起的用户机会损失、利润损失。
- (3) 无论三菱电机能否预测，由特殊原因而导致的损失和间接损失、事故赔偿、以及三菱电机产品以外的损伤。
- (4) 对于用户更换设备、现场机械设备的再调试、运行测试及其它作业等的补偿。

5. 产品规格的改变

目录、手册或技术文档中的规格如有改变，恕不另行通知。

商标

Microsoft, Visual Basic, Visual C++, Visual Studio, Windows, and Windows XP are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or 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 '™' or '®' are not specified in this manual.

SH (NA) -082183CHN-C (2202) MEACH

MODEL: MD-FUNC-LIB-R-C

 **三菱电机自动化(中国)有限公司**

地址：上海市虹桥路1386号三菱电机自动化中心

邮编：200336

电话：021-23223030 传真：021-23223000

网址：<http://cn.MitsubishiElectric.com/fa/zh/>

技术支持热线 **400-821-3030**



扫描二维码,关注官方微博



扫描二维码,关注官方微信

内容如有更改 恕不另行通知