

技术简讯

[1 / 63]

FA-CN-0001-B

用通用型QCPU替换高性能型QCPU的方法（详细篇）

■出版年月

2016年12月（2019年3月修订B版）

■相关机型

Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU、Q25HCPU、Q02UCPU、Q03UDCPU、Q03UDVCPU、Q03UDECPU、Q04UDHCPU、Q04UDVCPU、Q04UDEHCPU、Q06UDHCPU、Q06UDVCPU、Q06UDEHCPU、Q10UDHCPU、Q10UDEHCPU、Q13UDHCPU、Q13UDVCPU、Q13UDEHCPU、Q20UDHCPU、Q20UDEHCPU、Q26UDHCPU、Q26UDVCPU、Q26UDEHCPU、Q50UDEHCPU、Q100UDEHCPU

感谢您继续支持三菱电机MELSEC-Q 系列可编程控制器。

本技术简讯对用通用型QCPU替换高性能型QCPU的方法进行详细说明。

在考虑替换时，应在阅读本简讯前参照技术简讯“用通用型QCPU替换高性能型QCPU的方法（导入篇）（FA-CN-0209）”的基础上确认需要替换的设备和功能。

此外，关于用通用型QCPU替换基本型QCPU的方法，应参照技术简讯“FA-CN-0054”的最新版本。

注意，本技术简讯所述的参照手册或参照目标为截至2019年3月的信息。

目 录

1 本技术简讯中使用的总称	2
2 替换时的注意事项	3
3 适用设备和软件	10
4 指令	14
4.1 通用型 QCPU 不支持的指令和替代方法	14
4.2 使用多 CPU 间通信专用指令的程序替换	16
4.3 程序替换示例	16
5 功能	29
5.1 浮点运算指令	29
5.2 浮点数据比较指令的错误检查处理（不包括通用高速型 QCPU）	36
5.3 变址修饰的软元件的范围检查处理	40
5.4 软元件锁存功能	44
5.5 文件使用方法设置	47
5.6 参数有效驱动器和引导文件设置	49
5.7 外部输入输出强制 ON/OFF 功能	51
5.8 MELSECNET/H 简单双结构网络的替代方法	54
6 特殊继电器和特殊寄存器	60
6.1 特殊继电器一览表	60
6.2 特殊寄存器一览表	62
修订记录	63

1 本技术简讯中使用的总称

本技术简讯中，除了特别标注以外，均使用以下用语进行说明。

总称	说明
高性能型QCPU	Q02CPU、Q02HCPU、Q06HCPU、Q12HCPU和Q25HCPU的总称
通用型QCPU	Q02UCPU、Q03UDCPU、Q03UDVCPU、Q03UDECPU、Q04UDHCPU、Q04UDVCPU、Q04UDEHCPU、Q06UDHCPU、Q06UDVCPU、Q06UDEHCPU、Q10UDHCPU、Q10UDEHCPU、Q13UDHCPU、Q13UDVCPU、Q13UDEHCPU、Q20UDHCPU、Q20UDEHCPU、Q26UDHCPU、Q26UDVCPU、Q26UDEHCPU和Q100UDEHCPU的总称
内置以太网端口QCPU	Q03UDVCPU、Q03UDECPU、Q04UDVCPU、Q04UDEHCPU、Q06UDVCPU、Q06UDEHCPU、Q10UDEHCPU、Q13UDVCPU、Q13UDEHCPU、Q20UDEHCPU、Q26UDVCPU、Q26UDEHCPU、Q50UDEHCPU和Q100UDEHCPU的总称
通用高速型QCPU	Q03UDVCPU、Q04UDVCPU、Q06UDVCPU、Q13UDVCPU和Q26UDVCPU的总称
QnUD (H) CPU	Q03UDCPU、Q04UDHCPU、Q06UDHCPU、Q10UDHCPU、Q13UDHCPU、Q20UDHCPU和Q26UDHCPU的总称
QnUDE (H) CPU	Q03UDECPU、Q04UDEHCPU、Q06UDEHCPU、Q10UDEHCPU、Q13UDEHCPU、Q20UDEHCPU、Q26UDEHCPU、Q50UDEHCPU和Q100UDEHCPU的总称
QnUD (E) (H) CPU	Q03UDCPU、Q03UDECPU、Q04UDHCPU、Q04UDEHCPU、Q06UDHCPU、Q06UDEHCPU、Q10UDHCPU、Q10UDEHCPU、Q13UDHCPU、Q13UDEHCPU、Q20UDHCPU、Q20UDEHCPU、Q26UDHCPU、Q26UDEHCPU、Q50UDEHCPU和Q100UDEHCPU的总称
QnUDVCPU	Q03UDVCPU、Q04UDVCPU、Q06UDVCPU、Q13UDVCPU和Q26UDVCPU的总称

2 替换时的注意事项

本章介绍了用通用型QCPU替换高性能型QCPU时的注意事项和替换方法。

系统构成

■ 替换时的注意事项和替换方法（系统构成）

编号	项目	注意事项	替换方法	参照
1	AnS/A系列模块的使用	必须使用序列号（前5位）为“13102”或以后版本的通用型QCPU。由于序列号（前5位）为“13101”或更早版本的通用型QCPU无法安装AnS/A系列模块，应考虑使用Q系列模块构成系统。	—	—
2	GOT	无法连接GOT900系列。	使用GOT1000或GOT2000系列。	—
3	编程工具连接	适用USB电缆有所不同。 <ul style="list-style-type: none">高性能型QCPU: A-B型通用型QCPU: A-miniB型	使用A-miniB型的USB电缆。或使用B-miniB型的USB转接适配器进行连接。	—
4	适用设备和软件	必须使用兼容通用型QCPU的设备和软件。	应确认为兼容通用型QCPU需要替换的设备和需要升级的软件。	<ul style="list-style-type: none">10页 为了兼容通用型QCPU需要更换的设备12页 可与通用型QCPU构成多CPU系统的CPU模块
5	多CPU系统	要构成多CPU系统，必须使用兼容通用型QCPU的CPU模块。	应确认兼容通用型QCPU的CPU模块。	12页 可与通用型QCPU构成多CPU系统的CPU模块
		在使用运动CPU的多CPU系统中，以往的自动刷新区和用户设置区无法用于与运动CPU的数据通信。	对于与运动CPU的数据通信，应使用多CPU间高速通信区内的自动刷新区和用户设置区。	QCPU用户手册（多CPU系统篇）中的第4章
6	冗余电源系统	在冗余电源系统中，要使用SM1780至SM1783/SD1780至SD1783或系统监视窗口确认电源模块的状态时，必须使用序列号（前5位）为“10042”或以后版本的通用型QCPU。如果使用序列号（前5位）为“10041”或更早版本的通用型QCPU，应通过电源模块前部的LED确认电源模块的状态。（冗余电源系统中电源模块的状态无法存储于SM1780至SM1783/SD1780至SD1783，也不可显示在系统监视窗口中。）	—	QCPU用户手册（硬件设计/维护点检篇）中的第7.1节
7	MELSECNET/H	无简单双结构网络的特殊继电器。	如果使用了简单双结构网络，应修改程序和参数。	<ul style="list-style-type: none">Q系列MELSECNET/H网络系统技术参考手册（主站、从站）中的第7.7节54页 MELSECNET/H简单双结构网络的替代方法
8	MELSECNET/H、CC-Link IE控制网络	数据链接间传送中数据传送的时机不同。	添加握手程序至数据的发送端和接收端，使模块不会在发送数据时接收数据。	Q系列MELSECNET/H网络系统技术参考手册（主站、从站）中的第6.2节 MELSEC-Q CC-Link IE Controller Network Reference Manual中的第4.1节

程序

■ 替换时的注意事项和替换方法（程序）

编号	项目	注意事项	替换方法	参照
1	语言和指令	不支持某些指令。	使用了通用型QCPU不支持的指令时，应通过替代方法进行替换。	14页 通用型QCPU不支持的指令和替代方法
2	浮点运算	以往的浮点运算指令以单精度进行运算。	通用型QCPU中增加了用于双精度运算的浮点运算指令。 如果需要进行双精度的浮点运算，就应替换为双精度浮点运算指令。	• QnUCPU用户手册（功能解说/程序基础篇）中的附录4.4 • 29页 浮点运算指令
		使用浮点数据比较指令的LDE□、ANDE□、ORE□、LDED□、ANDED□、ORED□时，如果比较源数据为-0、非数值、非规范化数或±∞时，会检测到“运算错误”（错误代码：4101）。*2 （□表示以下一种运算符：=、<>、<=、>=、<、>）	使用浮点数据比较指令时，应修改程序。	36页 浮点数据比较指令的错误检查处理（不包括通用高速型QCPU）
3	变址修饰的软元件范围检查	当软元件号因变址修饰超出软元件范围时，会检测到“运算错误”（错误代码：4101）。	取消勾选PLC参数对话框PLC RAS选项卡中的“进行已变址修饰的软元件的范围检查”，以不执行检查。	• QnUCPU用户手册（功能解说/程序基础篇）中的第3.17节 • 40页 变址修饰的软元件的范围检查处理
4	程序执行类型	不支持低速执行型程序。	应使用扫描执行型程序或定时执行型程序。	QnUCPU用户手册（功能解说/程序基础篇）中的第2.10节
		无法通过远程操作更改程序执行类型。 但应注意，序列号（前5位）为“18112”或以后版本的QnUDVCPU、QnUDPVCPU的情况下，程序执行类型为扫描或待机时，可通过远程操作进行程序执行类型的更改。	切换程序执行类型时，应使用程序执行类型切换指令，如PSTOP、POFF和PSCAN。	QnUCPU用户手册（功能解说/程序基础篇）中的第2.10.5节
5	锁存设置	如果设置内部用户软元件的锁存范围，处理时间会与锁存的软元件点数成比例增加。 (例如，如果8K点被锁存用于QnUD(E)(H)CPU的锁存继电器(L)，处理时间为28.6μs。)	通用型QCPU中，如下增强了锁存功能。 <ul style="list-style-type: none">大容量文件寄存器(R、ZR)写入/读取软元件数据至标准ROM(SP.DEVST/S(P).DEVLD指令)内部用户软元件的锁存范围指定锁存间隔设置的“时间设置”*3 对于通过内部用户软元件锁存的内容，根据应用更改为通过以上方法锁存。	• QnUCPU用户手册（功能解说/程序基础篇）中的第3.3和3.3(5)(b)节 • 44页 软元件锁存功能
6	中断程序	不支持用于高速中断功能的中断指针(I49)。*2	考虑能否使用定时中断引起的中断指针(I28至I31)。	QnUCPU用户手册（功能解说/程序基础篇）中的第3.13.2节
		不支持中断计数器。	在中断程序监视列表窗口中确认中断程序执行次数。	
		不支持因发生错误引起的中断指针(I32至I40)。	—	QnUCPU用户手册（功能解说/程序基础篇）中的第4.11节
7	SCJ指令	在通用型QCPU中使用SCJ指令时，需要在SCJ指令前插入AND SM400（或NOP指令）。*2	使用SCJ指令时，在SCJ指令前插入AND SM400（或NOP指令）。	MELSEC-Q/L 编程手册（公共指令篇）中的第6.5节
8	ZPUSH指令	通用型QCPU的变址寄存器数增加至20个。同时也增大了使用ZPUSH指令存储变址寄存器中数据时的区域。	根据需要，增大用于ZPUSH指令的存储区。	MELSEC-Q/L 编程手册（公共指令篇）中的第7.19节
9	各程序的文件使用方法设置	以下各程序用的文件使用方法设置不可用。 ^{*1} <ul style="list-style-type: none">文件寄存器软元件初始值注释	设定文件使用方法时，应修改程序。	• QnUCPU用户手册（功能解说/程序基础篇）中的第2.10节 • 47页 文件使用方法设置
10	各程序的输入/输出刷新设置	各程序的输入/输出刷新设置不可用。	如果需要各程序的输入/输出刷新设置，应使用RFS指令。	MELSEC-Q/L 编程手册（公共指令篇）

编号	项目	注意事项	替换方法	参照
11	SM/SD	部分特殊继电器和特殊寄存器的使用方法有所不同。	按照替代方法替换相应的特殊继电器和特殊寄存器。	• 60页 特殊继电器一览表 • 62页 特殊寄存器一览表
		要使用A系列兼容的特殊继电器/特殊寄存器（SM1000至SM1255/SD1000至SD1255），必须使用序列号（前5位）为“10102”或以后版本的通用型QCPU。 如果使用了序列号（前5位）为“10101”或更早版本的通用型QCPU，应使用编程工具的转换功能替换为适用于通用型QCPU的特殊继电器/特殊寄存器。但应注意，不兼容通用型QCPU的会被替换为SM1255/SD1255，应根据需要修改程序。	—	QCPU用户手册（硬件设计/维护点检篇）
12	处理时间	扫描时间和各种处理时间有所不同。		确认处理时间的同时，根据需要修改程序。
13	步数	每符合以下条件时，程序步数将增加一步：*4 • 执行变址修饰。 • 使用上升或下降沿执行指令。 • 使用K1、K2、K3、K5、K6或K7指定了位数，或指定的软元件号不是16的倍数。		在替换对象程序中频繁使用如左所示的软元件修饰时，如果更改可编程控制器类型，将超出存储容量。因此，更改了可编程控制器类型后，应使用存储器大小计算功能确认程序大小。如果程序大小超过存储容量，请采取以下措施或更换为程序存储器更大的CPU模块。 • 移动参数和软元件注释至标准ROM。 • 减小用于运行中写入的保留区。 • 每使用64K字以内的文件寄存器、扩展数据寄存器和扩展链接寄存器的其中一个时，程序步数将减少一步，因此进行调整以确保在64K字以内使用。

*1 如果是序列号（前5位）为“10011”或更早版本的通用型QCPU，局部软元件文件使用方法也无法设置。

*2 用通用高速型QCPU替换高性能型QCPU时不适用。

*3 仅通用高速型QCPU支持此设置。

*4 仅在用通用高速型QCPU替换高性能型QCPU时适用。

驱动器和文件

■ 替换时的注意事项和替换方法（驱动器和文件）

编号	项目	注意事项	替换方法	参照
1	引导文件设置	标准ROM中的文件无法被引导至程序存储器。	通用型QCPU在电池电压下降时保留程序存储器中的数据，因而无需引导文件设置。 移动引导文件（从标准ROM至程序存储器）至程序存储器。	• QnUCPU用户手册（功能解说/程序基础篇）中的第2.11节 • 49页 参数有效驱动器和引导文件设置
		引导动作有所不同。	在高性能型QCPU中设置参数有效驱动器和引导文件设置时，应替换设置。	
		无法将存储卡（SRAM卡、ATA卡或闪存卡）指定为传送源（引导源）。*1	应指定SD存储卡为传送源（引导源）。	
2	从存储卡向标准ROM自动写入全部数据	该功能的设置方法有所不同。	在PLC参数对话框的引导文件设置选项卡中，选择“标准ROM”作为传送目标。但应注意，程序的传送目标应设置为“程序存储器”。（无需通过拨码开关设置。）	QnUCPU用户手册（功能解说/程序基础篇）中的第2.11节
3	软元件注释	无法将软元件注释文件存储于SRAM卡。*1	应将该文件存储于标准RAM。	—
		无法将软元件注释文件存储于ATA卡或闪存卡。 *1	应将该文件存储于SD存储卡。	—
4	软元件初始值	无法将软元件初始值文件存储于SRAM卡。*1	应将该文件存储于标准RAM或标准ROM。	QnUCPU用户手册（功能解说/程序基础篇）中的第3.25节
		无法将软元件初始值文件存储于ATA卡或闪存卡。 *1	应将该文件存储于SD存储卡。	
5	局部软元件	无法将局部软元件文件存储于SRAM卡。*1	• 应将该文件存储于标准RAM。 • 如果局部软元件文件的合计大小超出标准RAM容量，应考虑使用扩展SRAM卡盒扩展标准RAM的容量。	QnUCPU用户手册（功能解说/程序基础篇）中的第6.2节
6	文件寄存器	无法将文件寄存器文件存储于SRAM卡。*1	• 应将该文件存储于标准RAM。 • 如果文件寄存器文件的合计大小超出标准RAM容量，应考虑使用扩展SRAM卡盒扩展标准RAM的容量。	QnUCPU用户手册（功能解说/程序基础篇）中的第4.7.1节
		无法将文件寄存器文件存储于闪存卡。（顺控程序仅可读取闪存卡中的文件寄存器数据。）*1	使用SD存储卡中的软元件初始值文件或FREAD/FWRITE指令。	• QnUCPU用户手册（功能解说/程序基础篇）中的第3.25节 • MELSEC-Q/L 编程手册（公共指令篇）
7	采样跟踪	无法将采样跟踪文件存储于SRAM卡。*1	• 应将该文件存储于标准RAM。 • 如果采样跟踪文件的合计大小超出标准RAM容量，应考虑使用扩展SRAM卡盒扩展标准RAM的容量。	QnUCPU用户手册（功能解说/程序基础篇）中的第3.14（2）节
8	存储卡的CPU模块更换功能	无法将存储卡指定为备份目标或恢复源。*1	应指定SD存储卡为备份目标或恢复源。	QnUCPU用户手册（功能解说/程序基础篇）中的第3.31节

*1 用通用高速型QCPU替换高性能型QCPU时适用。

外部通信

■ 替换时的注意事项和替换方法（外部通信）

编号	项目	注意事项	替换方法	参照
1	模块服务间隔读取	无法读取模块服务间隔时间。	—	QnUCPU用户手册（功能解说/程序基础篇）中的第3.24.1节
2	MC协议	要使用A兼容的1C帧或A兼容的1E帧访问CPU模块，必须使用序列号（前5位）为“10102”或以后版本的通用型QCPU。如果使用序列号为“10101”或更早版本的通用型QCPU，应使用以下帧类型。 <ul style="list-style-type: none"> • QnA兼容的2C/3C/4C帧 • QnA兼容的3E帧 • 4E帧 	—	MELSEC 通信协议参考手册
		以下命令无法指定监视条件。 <ul style="list-style-type: none"> • 以字为单位随机读取数据（命令：0403） • 软元件存储器监视（命令：0801） 适用的帧类型如下： <ul style="list-style-type: none"> • QnA兼容的3C/4C帧 • QnA兼容的3E帧 • 4E帧 	—	

诊断功能

■ 替换时的注意事项和替换方法（诊断功能）

编号	项目	注意事项	替换方法	参照
1	错误履历	错误履历数据无法存储于存储卡。	通用型QCPU可以在内置存储器中存储以往存储卡可存储的件数（100件）。	QnUCPU用户手册（功能解说/程序基础篇）中的第3.18节
2	LED显示优先级设置	无法设置LED显示优先级。仅支持出错时的LED显示有无设置。	—	QnUCPU用户手册（功能解说/程序基础篇）中的第3.20.2节

调试

■ 替换时的注意事项和替换方法（调试）

编号	项目	注意事项	替换方法	参照
1	监视条件设置	要使用监视条件设置功能，必须使用序列号（前5位）为“10042”或以后版本的通用型QCPU。如果使用序列号为“10041”或更早版本的通用型QCPU，应使用采样跟踪功能确认在指定条件下的软元件数据。	—	QnUCPU用户手册（功能解说/程序基础篇）中的第3.11.1节和3.14节
2	扫描时间测定	要使用扫描时间测定功能，必须使用序列号（前5位）为“10042”或以后版本的通用型QCPU。 ^{*1} 如果使用序列号为“10041”或更早版本的通用型QCPU，应使用手册中所述的指令处理时间计算时间。	—	• QnUCPU用户手册（功能解说/程序基础篇）中的第3.13.3节 • MELSEC-Q/L 编程手册（公共指令篇）中的附录1
3	外部输入输出强制ON/OFF	要使用外部输入输出强制ON/OFF功能，必须使用序列号（前5位）为“10042”或以后版本的通用型QCPU。 ^{*2} 如果使用序列号为“10041”或更早版本的通用型QCPU，可用第5.7节中的替代程序替换该功能。但应注意，第5.7节中所述的替换方法不适用以下情况： • 使用直接输入软元件（DX）和直接输出软元件（DY）引用或更改用于强制ON/OFF的输入和输出。 • 在中断程序中引用或更改用于强制ON/OFF的输入和输出。	—	• QnUCPU用户手册（功能解说/程序基础篇）中的第3.11.3节 • 51页 外部输入输出强制ON/OFF功能

*1 可在程序监视列表窗口上确认各程序的扫描时间。

*2 可用序列号（前5位）为“10041”或更早版本的CPU模块（Q02UCPU、Q03UDCPU、Q04UDHCPU、Q06UDHCPU、Q13UDHCPU、Q26UDHCPU）执行软元件测试。

CPU模块前部的开关

■ 替换时的注意事项和替换方法（CPU模块前部的开关）

编号	项目	注意事项	替换方法	参照
1	CPU模块前部的开关	更改了用RESET/RUN/STOP开关的操作方法。	通用型QCPU的RESET/STOP/RUN开关可用于CPU模块的复位操作以及STOP和RUN状态间的切换。	QCPU用户手册（硬件设计/维护点检篇）中的第6.1.3节
		无法通过开关清除锁存数据。	要清除锁存数据，应执行远程锁存清除操作。	QnUCPU用户手册（功能解说/程序基础篇）中的第2.7(4)和3.6.4节
		无法通过开关设定系统保护。	可通过为各文件设置密码来防止文件改写。可使用编程工具注册各文件的密码。	QnUCPU用户手册（功能解说/程序基础篇）中的第3.19节
		无需参数有效驱动器设置。	通用型QCPU自动判定参数有效驱动器。当参数有效驱动器在高性能型QCPU中被设为非程序存储器时，应更改设置。	• QnUCPU用户手册（功能解说/程序基础篇）中的第2.1.2节 • 49页 参数有效驱动器和引导文件设置

SFC

■ 替换时的注意事项和替换方法 (SFC)

编号	项目	注意事项	替换方法	参照
1	步转移监视定时器	不支持步转移监视定时器。	按MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中附录3所述修改程序。	MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的第4.6节和附录3
2	SFC动作模式设置	不支持定期执行块设置。	按MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中附录3所述修改程序。	MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的第4.7节和附录3
		要在双块START上选择运行模式, 必须使用序列号 (前5位) 为“12052”或以后版本的通用型QCPU。如果使用序列号 (前5位) 为“12051”或更早版本的通用型QCPU, 双块START时的运行模式被固定为“WAIT”。	—	MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的第4.7节
		无法选择转移至活动步 (双步START时) 的运行模式。 (固定为转移。)	考虑使用转移至活动步“转移”的运行模式 (双步START时) 执行SFC程序。	MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的第4.7节
3	用于程序执行管理的SFC程序	不支持用于程序执行管理的SFC程序。	考虑是否可以仅使用一般SFC程序 (1个) 应对。	MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的第5.3节
4	SFC控制指令	不支持某些SFC控制指令。	—	<ul style="list-style-type: none"> • MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的第4.4节 • 15页 通用型QCPU不支持的SFC控制指令和替代方法
5	SFC注释读取指令	要执行以下SFC注释读取指令, 必须使用序列号 (前5位) 为“12052”或以后版本的通用型QCPU。 <ul style="list-style-type: none"> • S(P).SFCSCOMR (SFC步注释读取指令) • S(P).SFCTCOMR (SFC转移条件注释读取指令) 	—	MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的第4.8节
6	SFC程序更改的方法	不支持SFC程序文件的运行中写入。 (SFC图中的程序可通过运行中写入进行更改。)	<ul style="list-style-type: none"> • 在将通用型QCPU状态更改为STOP后写入程序数据至CPU模块。 • 可通过非活动块的运行中写入更改SFC程序中的非活动块。^{*1} 	MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的第6.6节

*1 该操作可用于除Q02UCPU外序列号 (前5位) 为“12052”或以后版本的通用型QCPU。

3 适用设备和软件

为了兼容通用型QCPU需要更换的设备

下表列出了为了兼容通用型QCPU需要更换的设备。（无需更换下表未列出的设备。）

■需替换的设备（信息模块）

产品	型号	兼容通用型QCPU的模块版本的序列号（前5位）*2			
		使用Q02U/Q03UD/ Q04UDH/Q06UDHCPU 时	使用Q13UDH/ Q26UDHCPU时	使用Q10UDH/ Q20UDHCPU或 QnUDE (H) CPU时	使用通用高速型 QCPU时
Web服务器模块*1	• QJ71WS96	“09042”或以后版 本	“10011”或以后版 本	“10012”或以后版 本	“14122”或以后版 本
MES接口模块	• QJ71MES96				
高速数据记录器模块	• QD81DL96	无限制	无限制	无限制	“14122”或以后版 本

*1 当使用安装了GX RemoteService-I或MX MESInterface-WS Version 1的Web服务器模块时，通用型QCPU无法正常运行。

*2 当使用不兼容的模块版本时，通用型QCPU无法正常运行。

■需替换的设备（PC接口插板）

产品	型号	兼容通用型QCPU的专用软件包版本*1			
		使用Q02U/Q03UD/ Q04UDH/Q06UDHCPU 时	使用Q13UDH/ Q26UDHCPU时	使用Q10UDH/ Q20UDHCPU或 QnUDE (H) CPU时	使用通用高速型 QCPU时
CC-Link IE现场网络接口插板	• Q81BD-J71GF11-T2	无限制	无限制	无限制	1.03D或以后版本
CC-Link IE控制网络接口插板	• Q81BD-J71GP21-SX • Q81BD-J71GP21S-SX • Q80BD-J71GP21-SX • Q80BD-J71GP21S-SX	无限制	1.03D或以后版本	1.06G或以后版本	1.15R或以后版本
MELSECNET/H接 口插板	SI/QSI/H-PCF光 缆	• Q80BD-J71LP21-25 • Q80BD-J71LP21S-25 • Q81BD-J71LP21-25	15R或以后版本	18U或以后版本	20W或以后版本
	GI光缆	• Q80BD-J71LP21G	19V或以后版本	19V或以后版本	
	同轴电缆	• Q80BD-J71BR11	15R或以后版本	18U或以后版本	
		• Q80BD-J61BT11N • Q81BD-J61BT11	1.02C或以后版本 1.06G或以后版本	1.05F或以后版本 1.06G或以后版本	
CC-Link系统主站/本地站接口插板			1.07H或以后版本	1.12N或以后版本	

*1 插板本身无限制。此外，关于专用软件包的最新版本，请向当地三菱电机代理店咨询。

■需替换的设备 (GOT)

产品	型号	兼容通用型QCPU的GT Designer2 OS版本*1					兼容通用型QCPU的GT Works3 OS版本*1
		使用Q02U/ Q03UD/Q04UDH/ Q06UDHCPU时	使用Q13UDH/ Q26UDHCPU时	使用Q10UDH/ Q20UDHCPU时	使用Q03UDE/ Q04UDEH/ Q06UDEH/ Q13UDEH/ Q26UDEHCPU时	使用Q10UDEH/ Q20UDEHCPU时	
GOT1000	GT16□-□	无限制	无限制	2.91V或以后版本	无限制	2.91V或以后版本	1.64S或以后版本
	GT15□-□	2.60N或以后版本	2.76E或以后版本	2.91V或以后版本	2.81K或以后版本	2.91V或以后版本	1.64S或以后版本
	GT14□-□	无限制	无限制	无限制	无限制	无限制	1.64S或以后版本
	GT12□-□	无限制	无限制	无限制	无限制	无限制	1.67V或以后版本
	GT11□-□	2.60N或以后版本	2.76E或以后版本	2.91V或以后版本	2.81K或以后版本	2.91V或以后版本	1.64S或以后版本
	GT10□-□	2.76E或以后版本	2.76E或以后版本	2.91V或以后版本	2.81K或以后版本	2.91V或以后版本	1.64S或以后版本

*1 GOT本身无限制。此外，关于GT Designer2或GT Works3的最新版本，请向当地三菱电机代理店咨询。

■需替换的设备 (网络模块和串行通信模块)

产品	型号	兼容通用型QCPU的模块版本		
		使用Q02U/Q03UD/Q04UDH/ Q06UDH/Q10UDH/Q13UDH/ Q20UDH/Q26UDHCPU时	使用QnUDE (H) CPU时	使用通用高速型QCPU时
MELSECNET/H模块	<ul style="list-style-type: none"> • QJ71LP21-25 • QJ71LP21S-25 • QJ71LP21G • QJ71LP21GE • QJ71BR11 	无限制	某些限制因使用条件而异*1	
串行通信模块	<ul style="list-style-type: none"> • QJ71C24N • QJ71C24N-R2 • QJ71C24N-R4 		序列号（前5位）“10042”或以后版本	无限制

*1 如果满足下述所有条件，MELSECNET/H模块的序列号（前5位）必须为“10042”或以后版本。

·配置了包含内置以太网端口QCPU的多CPU系统。

·编程工具或GOT连接至内置以太网端口QCPU的以太网端口。

·编程工具或GOT通过其他CPU控制的MELSECNET/H模块访问其他站上的CPU模块。

·其他站上的访问目标为A/QnA系列CPU模块。

可与通用型QCPU构成多CPU系统的CPU模块

下表列出了可与通用型QCPU构成多CPU系统的CPU模块。

■QnUD (H) CPU或内置以太网端口QCPU时

- 可与QnUD (H) CPU或内置以太网端口QCPU构成多CPU系统的CPU模块

CPU模块	型号	适用版本				限制
		用Q03UD/Q04UDH/ Q06UDHCPU构成	用Q13UDH/ Q26UDH/Q03UDE/ Q04UDEH/ Q06UDEH/ Q13UDEH/ Q26UDEHCPU构成	用Q10UDH/ Q20UDH/Q10UDEH/ Q20UDEHCPU构成	与通用高速型 QCPU结合使用	
运动CPU	<ul style="list-style-type: none"> • Q172DCPU • Q173DCPU • Q172DCPU-S1 • Q173DCPU-S1 • Q172DSCPU • Q173DSCPU 	无限制				仅使用多CPU间 高速主基板模块 (Q3□DB) 时可 构成。
PC CPU模块	• PPC-CPU852 (MS)	驱动程序 软件 (PPC-DRV- 02) 版本1.01或以 后版本	驱动程序 软件 (PPC-DRV- 02) 版本1.02或以 后版本	驱动程序 软件 (PPC-DRV- 02) 版本1.03或以 后版本	不可用	—
C语言控制器模块	<ul style="list-style-type: none"> • Q06CCPU-V • Q06CCPU-V-B 	无限制	序列号 (前5 位) “10012” 或以 后版本	序列号 (前5 位) “10102” 或以 后版本	不可用	—
	<ul style="list-style-type: none"> • Q12DCCPU-V • Q24DHCCPU-V 	无限制				序列号 (前5 位) “14122” 或以 后版本
高性能型QCPU	<ul style="list-style-type: none"> • Q02CPU • Q02HCPU • Q06HCPU • Q12HCPU • Q25HCPU 	功能版本B或以后版本				—
过程CPU	<ul style="list-style-type: none"> • Q02PHCPU • Q06PHCPU • Q12PHCPU • Q25PHCPU 	无限制				

■Q02UCPU时

- 可与Q02UCPU构成多CPU系统的CPU模块

CPU模块	型号	适用版本	限制
运动CPU	<ul style="list-style-type: none"> • Q172CPUN (-T) • Q173CPUN (-T) • Q172HCPU (-T) • Q173HCPU (-T) 	无限制	无法使用多CPU间高速主基板模块 (Q3□DB)。
PC CPU模块	• PPC-CPU852 (MS)	驱动程序软件 (PPC-DRV-02) 版本1.01或以后版本	—
C语言控制器模块	<ul style="list-style-type: none"> • Q06CCPU-V • Q06CCPU-V-B 	无限制	—
	<ul style="list-style-type: none"> • Q12DCCPU-V • Q24DHCCPU-V 	无限制	—

为了兼容通用型QCPU需要升级的软件

下表列出了为与通用型QCPU通信需要升级的软件。（无需对下表未列出的软件进行版本升级。）

■需升级的软件

产品	型号	兼容通用型QCPU的版本				
		使用Q02U/Q03UD/ Q04UDH/ Q06UDHCPU时	使用Q13UDH/ Q26UDHCPU时	使用Q03UDE/ Q04UDEH/ Q06UDEH/ Q13UDEH/ Q26UDEHCPU时	使用Q10UDH/ Q20UDH/Q10UDEH/ Q20UDEHCPU时	使用通用高速型 QCPU时
GX Works2	• SW1DND-GXW2-E • SW1DNC-GXW2-E	无限制				1. 98C或以后版本
GX Developer	SW8D5C-GPPW-E	8. 48A或以后版本	8. 62Q或以后版本	8. 68W或以后版本	8. 78G或以后版本	不可用
GX Configurator-AD	SW2D5C-QADU-E	2. 05F或以后版本 ^{*1}	2. 05F或以后版本 ^{*2}	2. 05F或以后版本 ^{*3}	2. 05F或以后版本 ^{*4}	不可用
GX Configurator-DA	SW2D5C-QDAU-E	2. 06G或以后版本 ^{*1}	2. 06G或以后版本 ^{*2}	2. 06G或以后版本 ^{*3}	2. 06G或以后版本 ^{*4}	不可用
GX Configurator-SC	SW2D5C-QSCU-E	2. 12N或以后版本 ^{*1}	2. 12N或以后版本 ^{*2}	2. 17T或以后版本 ^{*3}	2. 17T或以后版本 ^{*4}	不可用
GX Configurator-CT	SW0D5C-QCTU-E	1. 25AB或以后版本 ^{*1}	1. 25AB或以后版本 ^{*2}	1. 25AB或以后版本 ^{*3}	1. 25AB或以后版本 ^{*4}	不可用
GX Configurator-TI	SW1D5C-QTIU-E	1. 24AA或以后版本 ^{*1}	1. 24AA或以后版本 ^{*2}	1. 24AA或以后版本 ^{*3}	1. 24AA或以后版本 ^{*4}	不可用
GX Configurator-TC	SW0D5C-QTCU-E	1. 23Z或以后版本 ^{*1}	1. 23Z或以后版本 ^{*2}	1. 23Z或以后版本 ^{*3}	1. 23Z或以后版本 ^{*4}	不可用
GX Configurator-FL	SW0D5C-QFLU-E	1. 23Z或以后版本 ^{*1}	1. 23Z或以后版本 ^{*2}	1. 23Z或以后版本 ^{*3}	1. 23Z或以后版本 ^{*4}	不可用
GX Configurator-QP	SW2D5C-QD75P-E	2. 25B或以后版本	2. 29F或以后版本	2. 30G或以后版本 ^{*5}	2. 32J或以后版本	不可用
GX Configurator-PT	SW1D5C-QPTU-E	1. 23Z或以后版本 ^{*1}	1. 23Z或以后版本 ^{*2}	1. 23Z或以后版本 ^{*3}	1. 23Z或以后版本 ^{*4}	不可用
GX Configurator-AS	SW1D5C-QASU-E	1. 21X或以后版本 ^{*1}	1. 21X或以后版本 ^{*2}	1. 21X或以后版本 ^{*3}	1. 21X或以后版本 ^{*4}	不可用
GX Configurator-MB	SW1D5C-QMBU-E	1. 08J或以后版本 ^{*1}	1. 08J或以后版本 ^{*2}	1. 08J或以后版本 ^{*3}	1. 08J或以后版本 ^{*4}	不可用
MX Component	SW3D5C-ACT-E	3. 09K或以后版本	3. 10L或以后版本	3. 11M或以后版本	3. 12N或以后版本	4. 02C或以后版本
GX Simulator	SW7D5C-LLT-E	7. 23Z或以后版本 ^{*4}	7. 23Z或以后版本 ^{*4}	7. 23Z或以后版本 ^{*4}	7. 23Z或以后版本 ^{*4}	不可用
MESInterface IT	VN-SWMIT1-E	无限制				1. 12N或以后版本

*1 可通过安装GX Developer版本8. 48A或以后版本使用该软件。

*2 可通过安装GX Developer版本8. 62Q或以后版本使用该软件。

*3 可通过安装GX Developer版本8. 68W或以后版本使用该软件。

*4 可通过安装GX Developer版本8. 78G或以后版本使用该软件。

*5 仅限通过USB连接时也可使用GX Configurator-QP版本2. 29F。

通用型QCPU不支持的软件

下表显示了通用型QCPU不支持的软件。

■通用型QCPU不支持的软件

产品	型号
GX Explorer	SW□D5C-EXP-E
GX Converter	SW□D5C-CNWW-E

4 指令

4.1 通用型QCPU不支持的指令和替代方法

通用型QCPU不支持的指令，应通过如下所示的替代方法进行替换。（如果未使用如下所示的指令，请跳过本节。）

通用型QCPU不支持的指令和替代方法

指令符号	指令名称	替代方法	参照目标
IX	整个梯形图的变址修饰	可使用替代程序替换指令。	16页 IX和IXEND指令的替换示例
IXEND			
IXDEV	整个梯形图变址修饰中的修饰值指定	变更程序，以使用MOV指令等将通过IXSET指令指定的软元件偏置值设定至变址修饰表。	18页 IXDEV和IXSET指令的替换示例
IXSET			
PR	ASCII码打印指令	<ul style="list-style-type: none">建议使用GOT作为ASCII码显示装置。存储于软元件中的ASCII码在GOT上直接显示为字符。可使用替代程序替换指令。	20页 PR指令的替换示例
PRC	注释打印指令	<ul style="list-style-type: none">建议使用GOT作为注释显示装置。软元件注释可显示在GOT上。在使用读取软元件注释数据指令（COMRD(P)）读取数据后，可通过PR指令的替代程序输出至显示装置。	
CHKST	特定格式故障检查指令	可使用替代程序替换指令。	23页 CHKST和CHK指令的替换示例
CHK			
CHKCIR	检查指令的检查格式更改指令	通过更改替代程序中的故障检测梯形图应对。	
CHKEND			
PLOW	程序低速执行注册指令	<ul style="list-style-type: none">用扫描执行型程序替换低速执行型程序时，使用PSCAN指令替代本指令。如果用定时执行型程序替换低速执行型程序，则无可使用的指令。	—
PCHK	程序执行状态检查指令	在程序监视列表窗口上确认程序的执行状态。详情请参照QnUCPU用户手册（功能解说/程序基础篇）中的第3.13.1节	—
KEY	数字键输入指令	<ul style="list-style-type: none">建议使用GOT作为数字输入装置。可使用替代程序替换指令。	26页 KEY指令的替换示例
PLOADP	从存储卡加载程序	在程序存储器中存储要执行的所有程序。通用型QCPU不可在运行期间添加程序至程序存储器，也不可用其他程序更改它们。	—
PUNLOADP	从存储卡卸载程序	如果程序存储器的容量不足，应将程序存储器中的参数、软元件注释和软元件初始值存储至标准ROM或存储卡。	
PSWAPP	加载+卸载		

通用型QCPU不支持的SFC控制指令和替代方法

指令符号	指令名称	替代方法
LD TRn	强制转移检查指令	更改了可编程控制器类型时，这些指令会被转换至SM1255。应根据需要修改程序。
AND TRn		
OR TRn		
LDI TRn		
ANDI TRn		
ORI TRn		
LD BLm\TRn		
AND BLm\TRn		
OR BLm\TRn		
LDI BLm\TRn		
ANDI BLm\TRn	活动步更改指令	参照MELSEC-Q/L/QnA 编程手册（SFC控制指令篇）中的附录3“关于基本型QCPU、通用型QCPU、LCPU的限制和替代方法”。
ORI BLm\TRn		
SCHG (D)	转移控制指令	参照MELSEC-Q/L/QnA 编程手册（SFC控制指令篇）中的附录3“关于基本型QCPU、通用型QCPU、LCPU的限制和替代方法”。
SET TRn		
SET BLm\TRn		
RST TRn		
RST BLm\TRn	块切换指令	更改了可编程控制器类型时，这些指令会被转换至SM1255。应根据需要修改程序。
BRSET (S)*1		

*1 序列号（前5位）为“13102”或以后版本的通用型CPU中可以使用。

4.2 使用多CPU间通信专用指令的程序替换

用QnUD (H) CPU或内置以太网端口QCPU替换模块

如果使用了以下指令，应使用表中的替代指令替换它们。

有关各指令的规格，请参照运动CPU的手册。

■QnUD (H) CPU和内置以太网端口QCPU不支持的指令和替代指令

指令符号	指令说明	替代指令的指令符号
S(P). DDWR	将本机CPU模块的软元件数据写入至其他号机CPU模块的软元件	D(P). DDWR
S(P). DDRD	读取其他号机CPU软元件数据至本机CPU模块的软元件	D(P). DDRD
S(P). SFCS	运动SFC程序启动请求	D(P). SFCS
S(P). SVST	伺服程序启动请求	D(P). SVST
S(P). CHGA	停止轴/同步编码器/凸轮轴的当前值更改	D(P). CHGA
S(P). CHGV	定位和JOG运行时的轴速度更改	D(P). CHGV
S(P). CHGT	实模式时，运行/停止中的转矩控制值更改	D(P). CHGT
S(P). GINT	其他号机CPU中断程序启动请求	D(P). GINT

用Q02UCPU替换模块

Q02UCPU中可以使用与基本型QCPU相同的多CPU间通信专用指令。

QnUD (H) CPU和内置以太网端口QCPU中的替代指令不可用于Q02UCPU。关于QnUD (H) CPU和内置以太网端口QCPU中的替代指令，应参照下述内容。

☞ 16页 QnUD (H) CPU和内置以太网端口QCPU不支持的指令和替代指令

4.3 程序替换示例

关于通用型QCPU不支持的指令中有替代程序的指令，以下介绍程序的替换示例。（如果未使用通用型QCPU不支持的指令，请跳过本节。）关于通用型QCPU不支持的指令，应参照下述内容。

☞ 14页 通用型QCPU不支持的指令和替代方法

IX和IXEND指令的替换示例

使用IX和IXEND指令程序的替换示例如下。

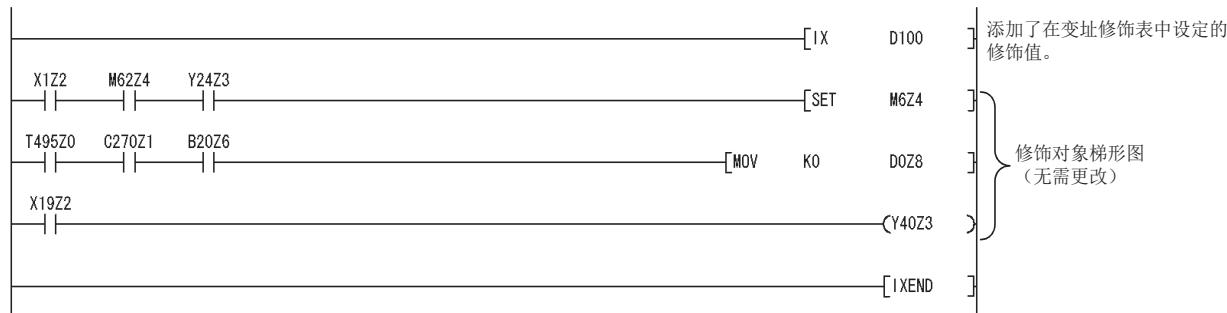
为了使用ZPUSH指令保存变址寄存器数据，需要设置23个字的变址寄存器保存区。

■软元件配置示例

(替换前)		(替换后)	
应用	软元件	应用	软元件
变址修饰表	D100至D115	变址修饰表	D100至D115
		变址寄存器保存区	D200至D222

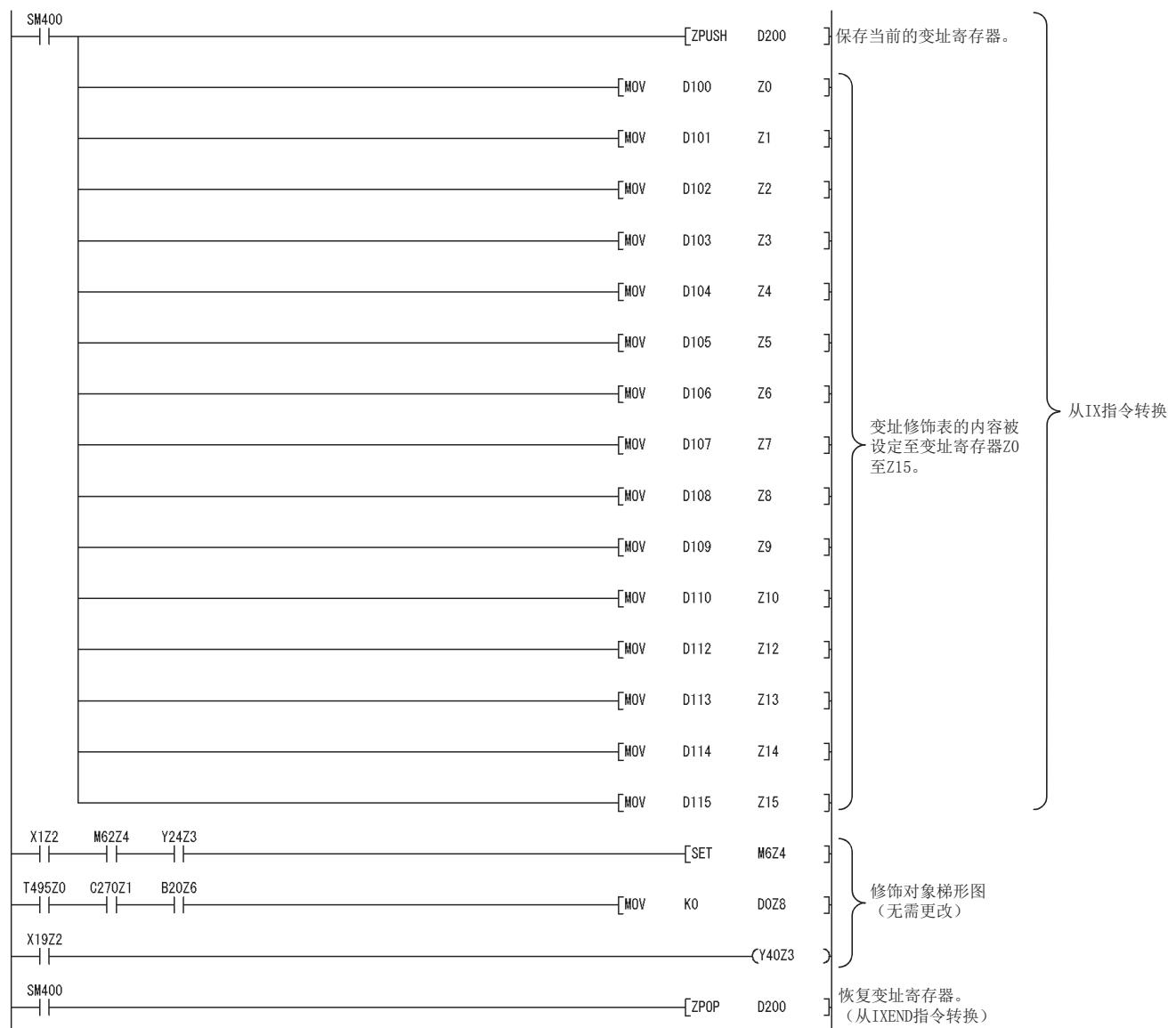
如果上述示例中的软元件号已用于其他应用，应分配未使用的软元件号。

■ 替换前程序



■ 替换后程序

- 用ZPUSH指令和将变址修饰表内容设定至变址寄存器中的处理替换IX指令。
- 用ZPOP指令替换IXEND指令。



IXDEV和IXSET指令的替换示例

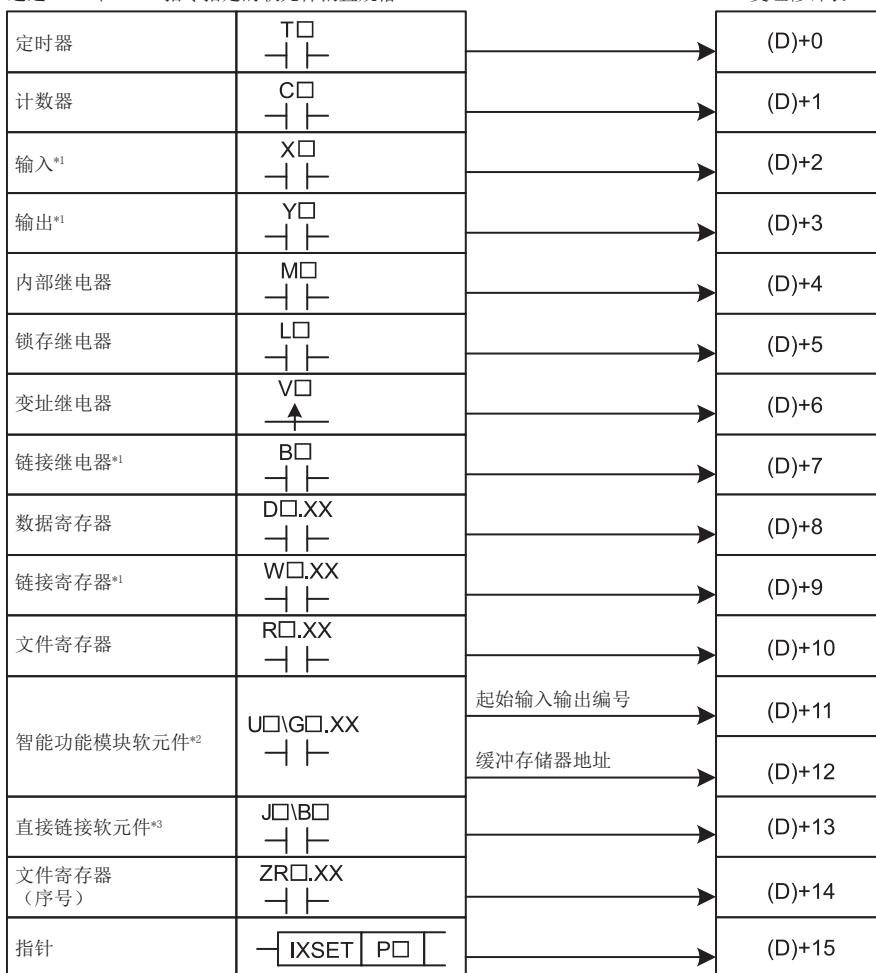
使用IXDEV和IXSET指令的程序的替换示例如下所示。

变更程序，以使用MOV指令将通过IXDEV和IXSET指令间触点指定的软元件偏置值直接设定至变址修饰表。

对于未通过IXDEV和IXSET指令指定软元件偏置值的软元件，应在替换后的程序中设定该软元件偏置值为0。

下图显示了通过IXDEV和IXSET指令的软元件偏置值指定与变址修饰表的对应关系。

通过IXDEV和IXSET指令指定的软元件偏置规格

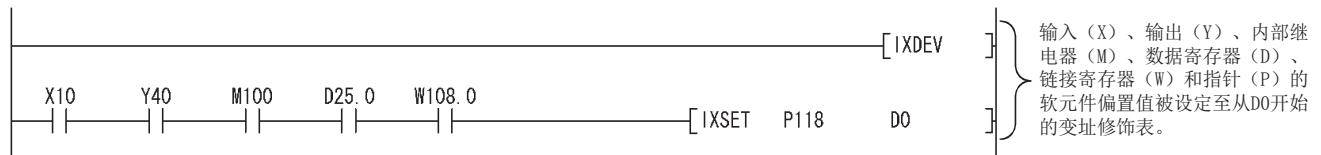
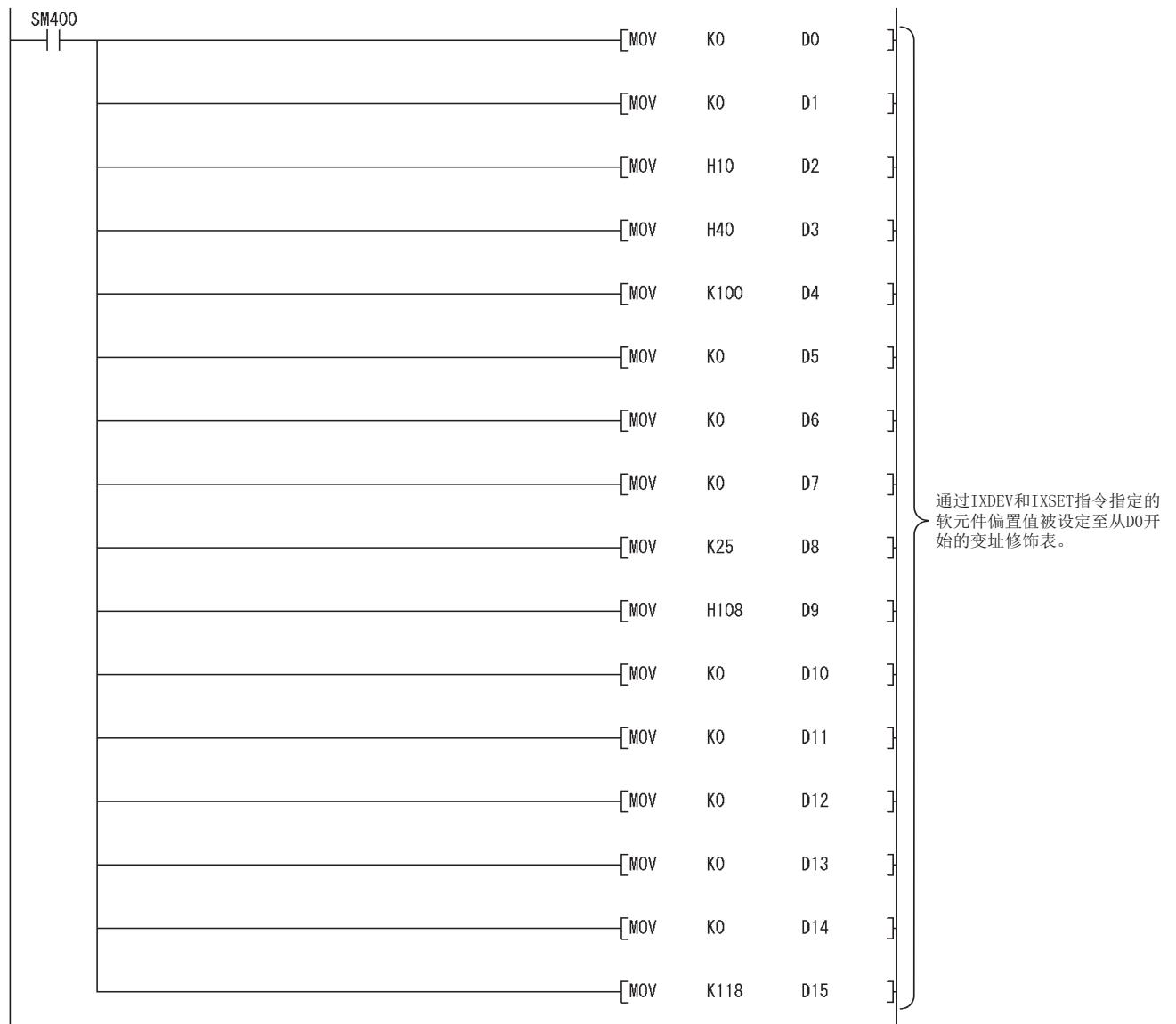


*1 软元件号以十六进制数表示。在变址修饰表中设定值时，应使用十六进制数常量 (H□)。

*2 起始输入输出编号 (U□) 以十六进制数表示。在变址修饰表中设定值时，应使用十六进制数常量 (H□)。

*3 可在J□\后指定软元件B、W、X或Y。在变址修饰表中将B、W、X和Y的软元件号设为各软元件的软元件偏置值。

例如，如果通过IXDEV或IXSET指令指定‘J10\Y220’，应在替换程序中的(D)+13中设定‘K10’，在(D)+3中设定‘H220’。(D)表示变址修饰表中的起始软元件。)

■ 替换前程序**■ 替换后程序**

PR指令的替换示例

使用PR指令的程序的替换示例如下所示。

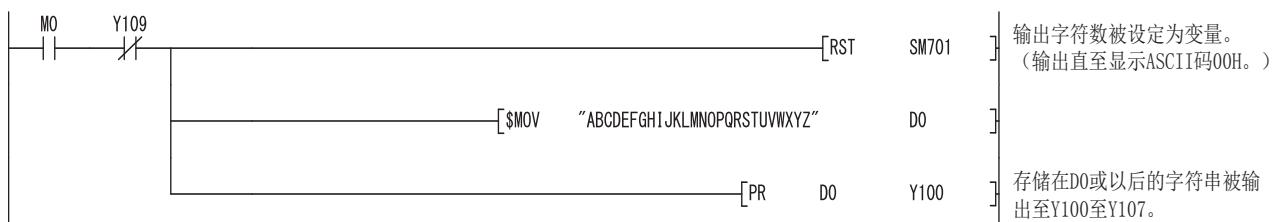
通过SM701可以切换输出字符数。

■软元件配置示例

(替换前)		(替换后)	
应用	软元件	应用	软元件
输出字符串	D0至D13	输出字符串	D0至D13
ASCII码输出信号	Y100至Y107	ASCII码输出信号	Y100至Y107
选通信号	Y108	选通信号	Y108
在执行标志	Y109	在执行标志	Y109
		输出字符串存储地址 (BIN32)	D20至D21
		输出字符串存储地址 (BIN32) (用于子程序和中断程序)	D200至D201
		输出字符数	D202
		输出模块起始Y号	D203
		字符提取位置	D204
		提取的字符数	D205
		字符串输出状态值	D206
		通过MIDR指令提取字符串的结果	D207
		字符串输出在执行标志	M200
		用于变址修饰	Z0

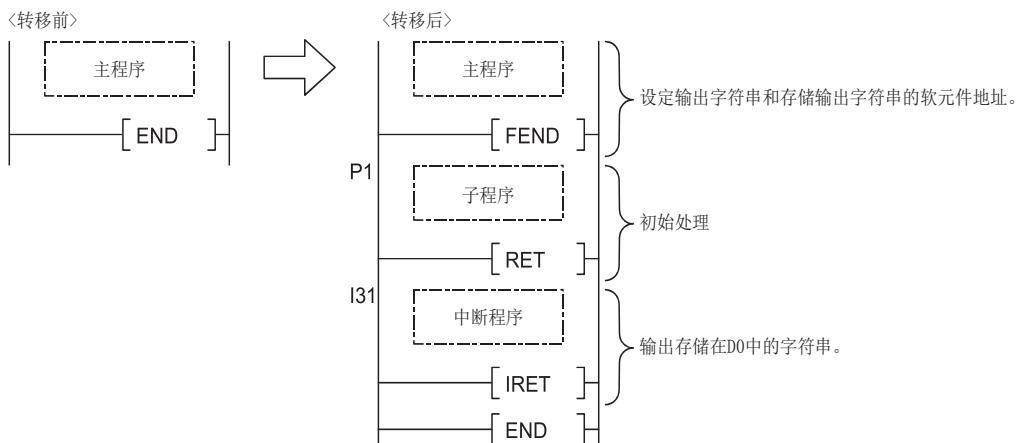
如果上述示例中的软元件号已用于其他应用，应分配未使用的软元件号。

■替换前程序



■替换后程序

在替换后的程序中，需要以下三个程序。

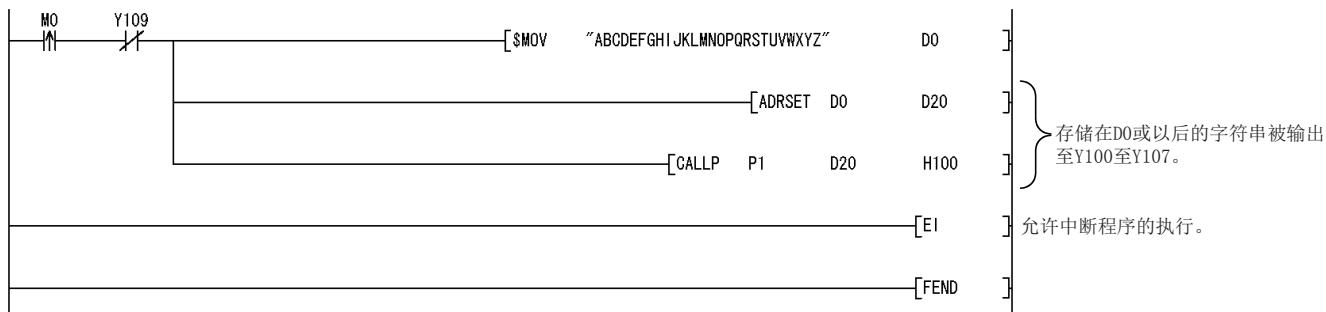


- 主程序

用使用CALL指令调用的子程序替换PR指令。

使用CALL指令无法直接指定输出字符串存储软元件（以下程序中的‘DO’）。使用ADRSET指令获取输出字符串存储软元件的间接地址，并使用CALL指令指定间接地址。不过，使用CALL指令无法直接将Y软元件（替换前程序中的‘Y100’）指定为输出Y号。以整数值指定输出Y号。

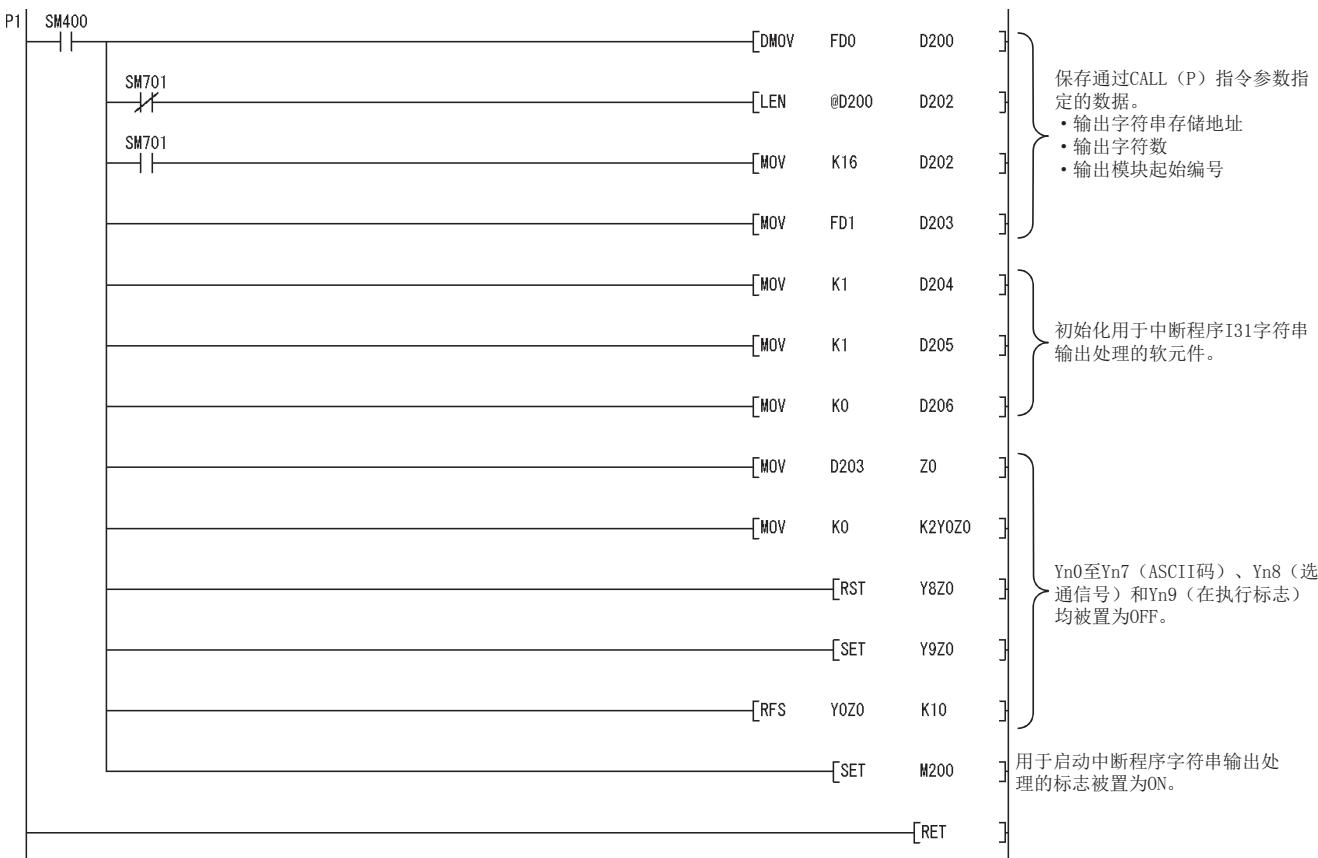
此外，中断程序用于通过输出模块输出字符代码。使用EI指令允许中断程序的执行。



- 子程序

在子程序中，用于在每10ms的定时中断程序输出ASCII码的数据被设定至工作软元件。并且，用于启动定时中断程序中的处理的标志被置为ON。此外，子程序中指定以下参数。

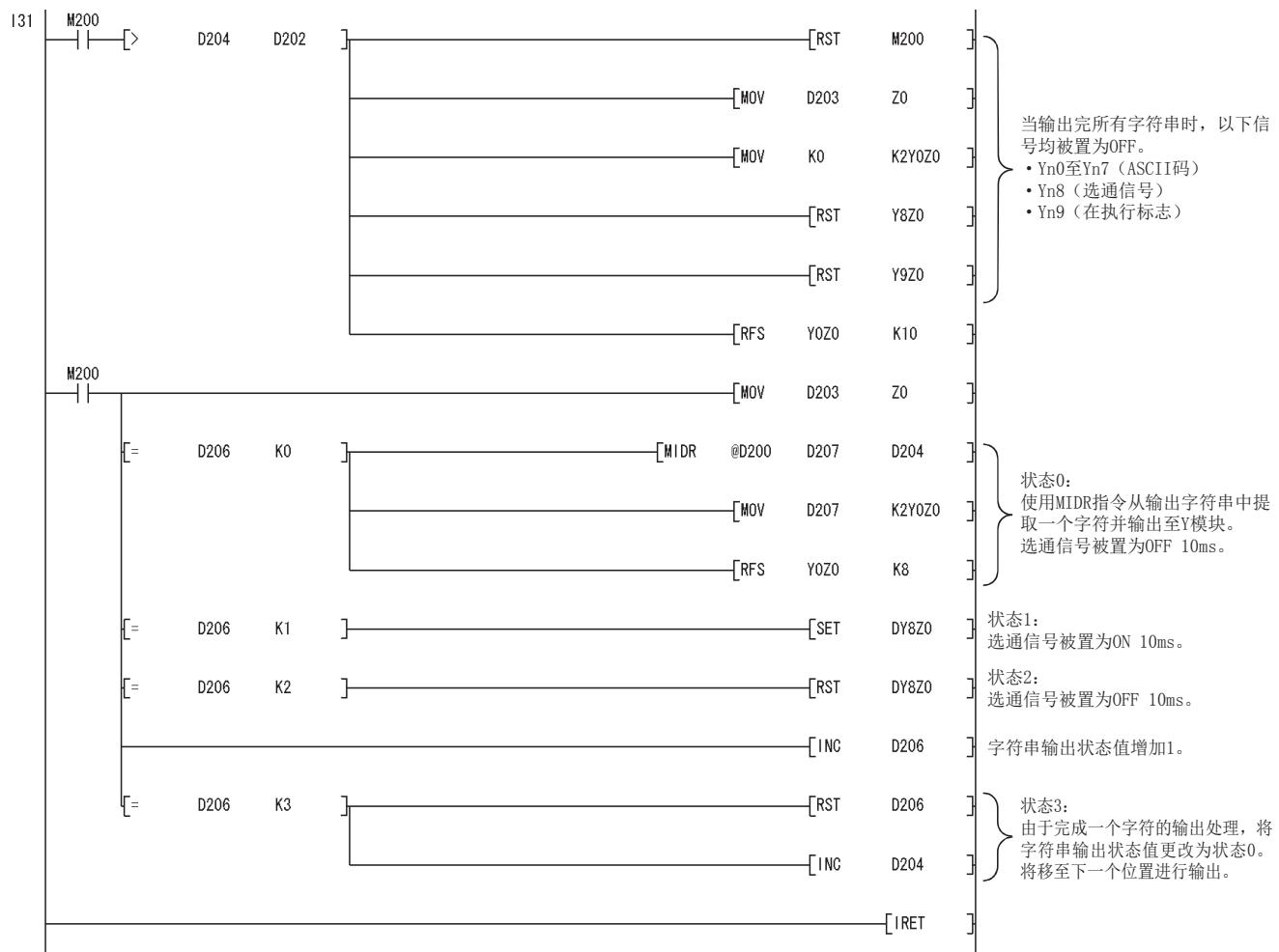
第一参数	输出字符串存储地址	(输入)
第二参数	输出模块起始Y号	(输入)



FA-CN-0001-B

• 中断程序

在每10ms的定时中断程序中添加以下处理。定时中断程序从输出模块输出ASCII码并控制选通信号。



CHKST和CHK指令的替换示例

使用CHKST和CHK指令的程序的替换示例如下所示。

在以下替换示例中，如果通过CHKST和CHK指令的替代程序检测到故障，故障号（触点号+线圈号）会被存储于D200，并且报警器F200被置为ON。

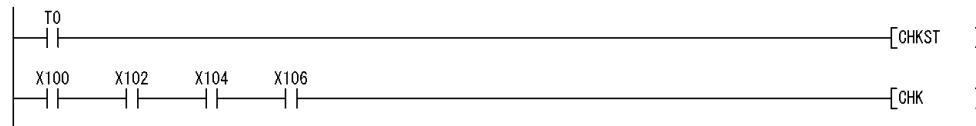
■软元件配置示例

(替换前)		(替换后)	
应用	软元件	应用	软元件
前进端检测传感器输入1	X100	前进端检测传感器输入1	X100
后进端检测传感器输入1	X101	后进端检测传感器输入1	X101
前进端检测传感器输入2	X102	前进端检测传感器输入2	X102
后进端检测传感器输入2	X103	后进端检测传感器输入2	X103
前进端检测传感器输入3	X104	前进端检测传感器输入3	X104
后进端检测传感器输入3	X105	后进端检测传感器输入3	X105
前进端检测传感器输入4	X106	前进端检测传感器输入4	X106
后进端检测传感器输入4	X107	后进端检测传感器输入4	X107
故障检测输出1	Y100	故障检测输出1	Y100
故障检测输出2	Y102	故障检测输出2	Y102
故障检测输出3	Y104	故障检测输出3	Y104
故障检测输出4	Y106	故障检测输出4	Y106
		线圈号（检测到的故障类型）	D100
		触点号	D101
		故障号	D200
		故障检测显示	F200
		用于变址修饰	Z0

如果上述示例中的软元件号已用于其他应用，应分配未使用的软元件号。

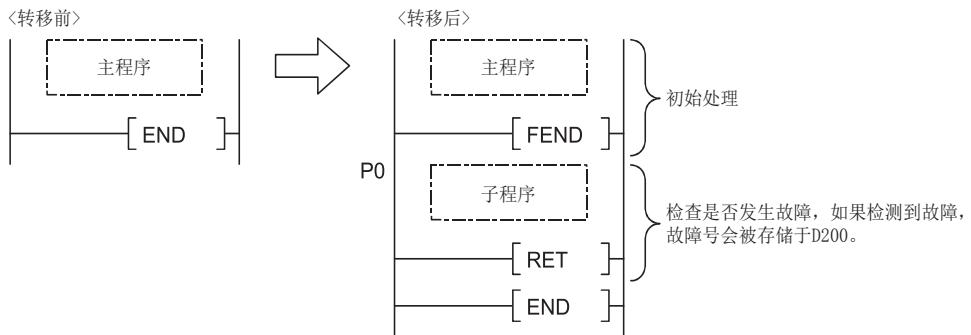
如果前进端检测传感器输入执行Xn的故障检测，应按如下所述为后进端检测传感器输入和故障检测输出分配软元件号。

前进端检测传感器输入	Xn
后进端检测传感器输入	Xn+1
故障检测输出	Yn

■替换前程序

■替换后程序

在替换后的程序中，需要以下两个程序。



- 主程序

用使用CALL指令调用的子程序替换CHKST和CHK指令。

对于每个被指定为CHK指令前检查条件的软元件，需要一个CALL指令。（替换前程序中，由于有4个CHK指令前的检查条件，需要添加4个CALL指令。）

X软元件的软元件号和触点号（检查条件）在各CALL指令中指定。此外，触点号用于在检测到故障时显示故障号。

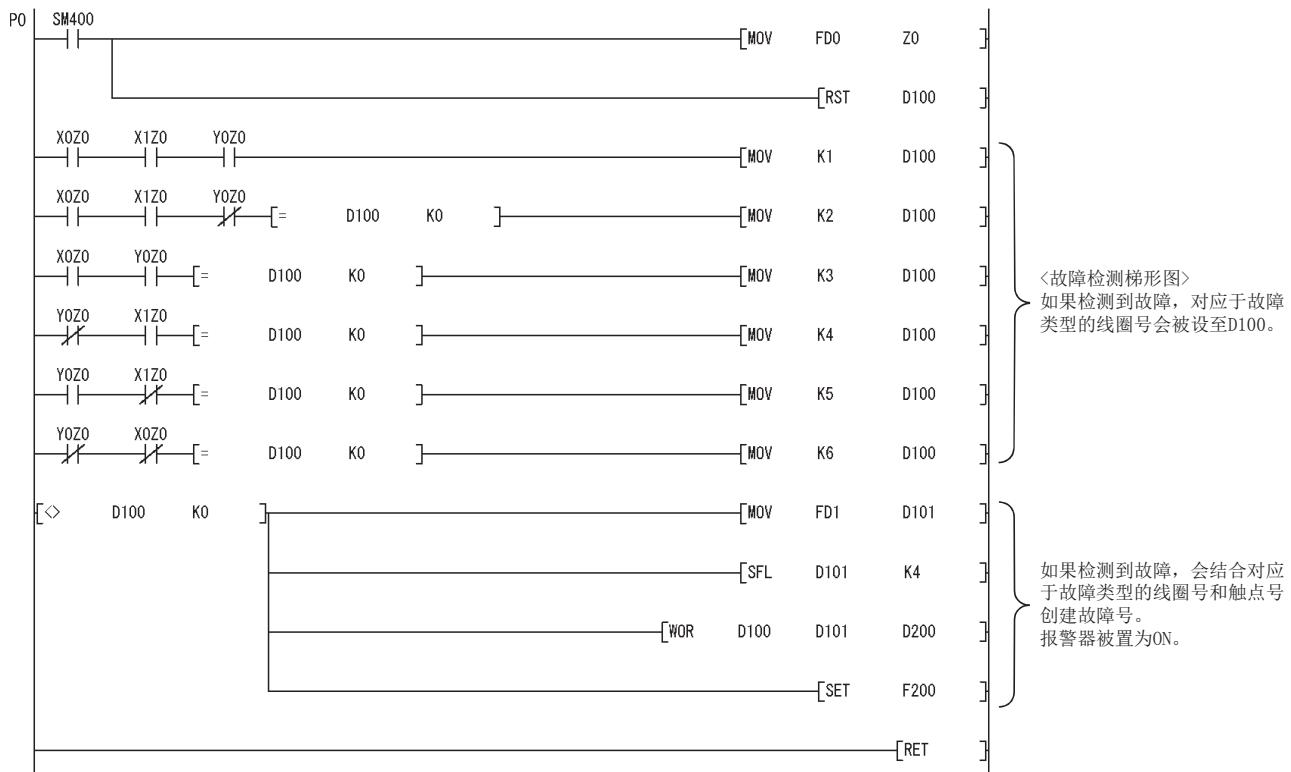


FA-CN-0001-B

• 子程序

在子程序中，使用故障检测梯形图检查是否发生故障。如果检测到故障，故障号会被存储于D200，并且报警器F200被置为ON。此外，子程序中指定以下参数。

第一参数	故障检查目标的X软元件的软元件号	(输入)
第二参数	故障检查目标的X软元件的触点号	(输入)



■通过CHKCIR和CHKEND指令更改了故障检测梯形图时的替换方法

可通过更改替换后程序的子程序内的故障检测梯形图，更改故障检测模式。

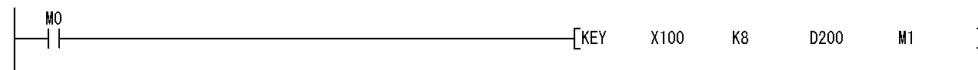
KEY指令的替换示例

使用KEY指令的程序的替换示例如下所示。

■软元件配置示例

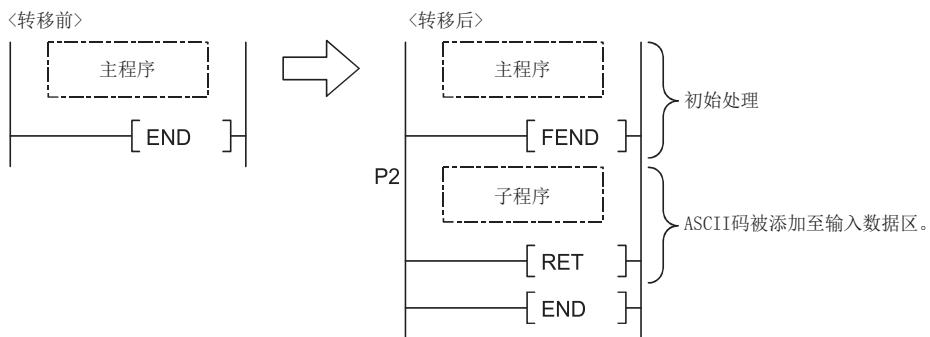
(替换前)		(替换后)	
应用	软元件	应用	软元件
数字输入执行指示	M0	数字输入执行指示	M0
输入完成标志	M1	输入完成标志	M1
输入数据区	D200至D203	输入数据区	D200至D203
ASCII码输入信号	X100至X107	ASCII码输入信号	X100至X107
选通信号	X108	选通信号	X108
		输入数据区地址 (BIN32)	D210至D211
		(输入数据区+0) 地址 (BIN32)	D212至D213
		(输入数据区+1) 地址 (BIN32)	D214至D215
		(输入数据区+2) 地址 (BIN32)	D216至D217
		输入数据移动用区域	D218
		输入数据转换用区域	D219至D220

如果上述示例中的软元件号已用于其他应用，应分配未使用的软元件号。

■替换前程序

■替换后程序

在替换后的程序中，需要以下两个程序。



- 主程序

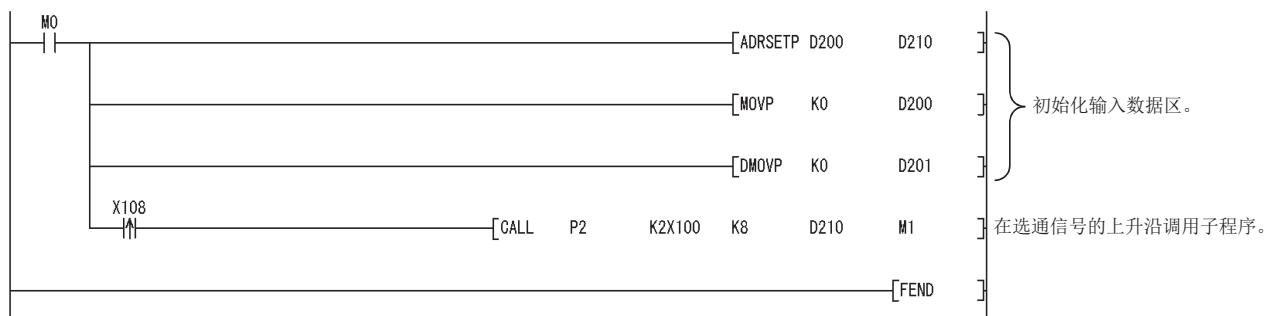
在执行指令（以下程序中的‘M0’）的上升沿上的输入数据区中设定‘0’并初始化程序。

在选通信号（以下程序中的‘X108’）的每一上升沿上执行CALL指令以调用子程序。

在子程序中，进行将输入代码添加至输入数据区的处理与输入完成的判断。

此外，在执行CALL指令时传递以下数据至子程序。

- 自输入模块的ASCII码输入值（Xn0至Xn7）
- 输入位数
- 输入数据区的间接地址（使用ADRSET指令获取输入数据区的间接地址。）
- 完成输入时为ON的位软元件

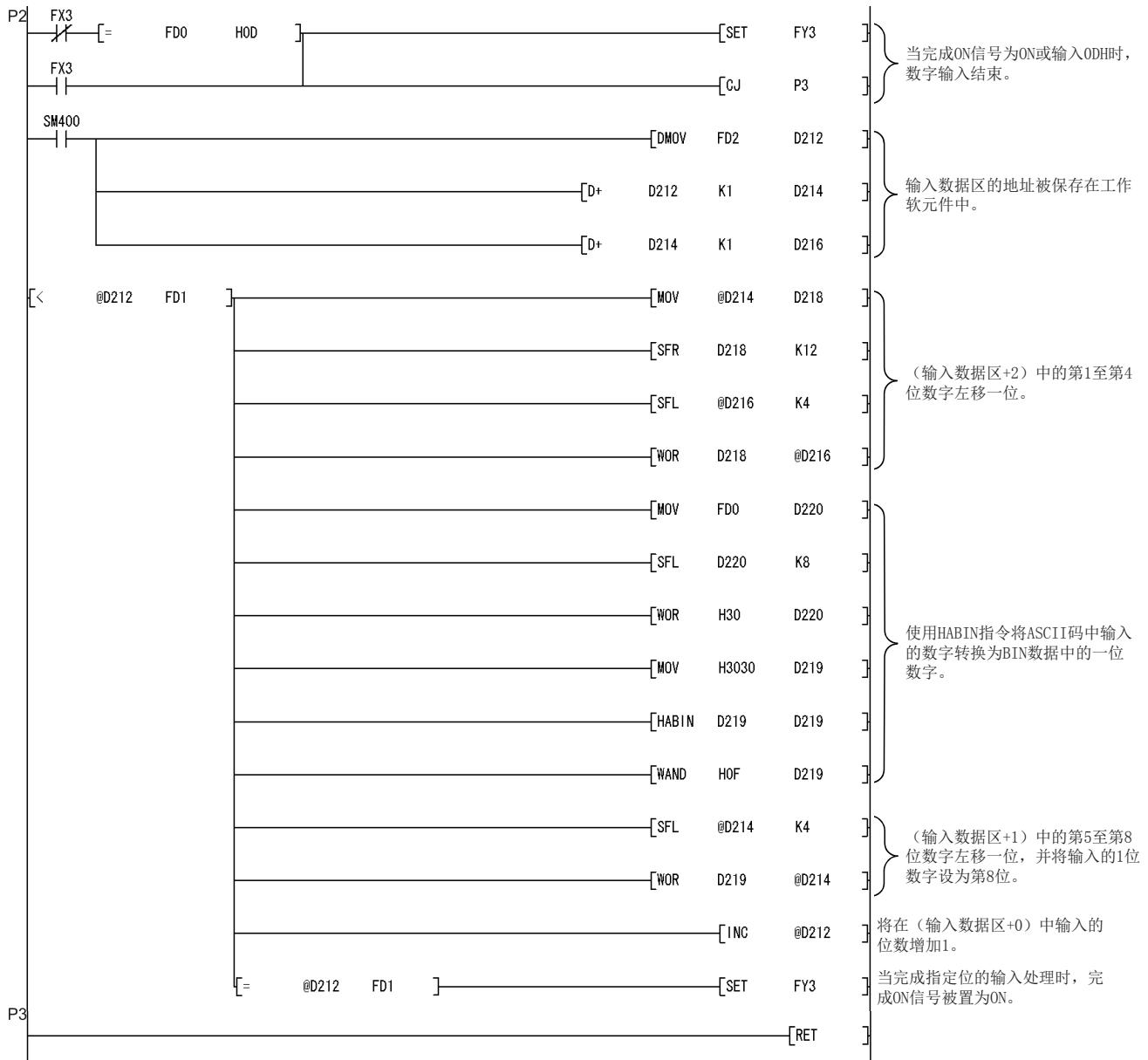


FA-CN-0001-B

• 子程序

在子程序中，通过参数指定的ASCII码被添加至输入数据区并判断输入完成条件是否成立。此外，子程序中指定以下参数。

第一参数	自输入模块的ASCII码输入 (K2Xn)	(输入)
第二参数	输入位数	(输入)
第三参数	输入数据区的间接地址	(输入)
第四参数	完成输入时为ON的位软元件	(输出)



5 功能

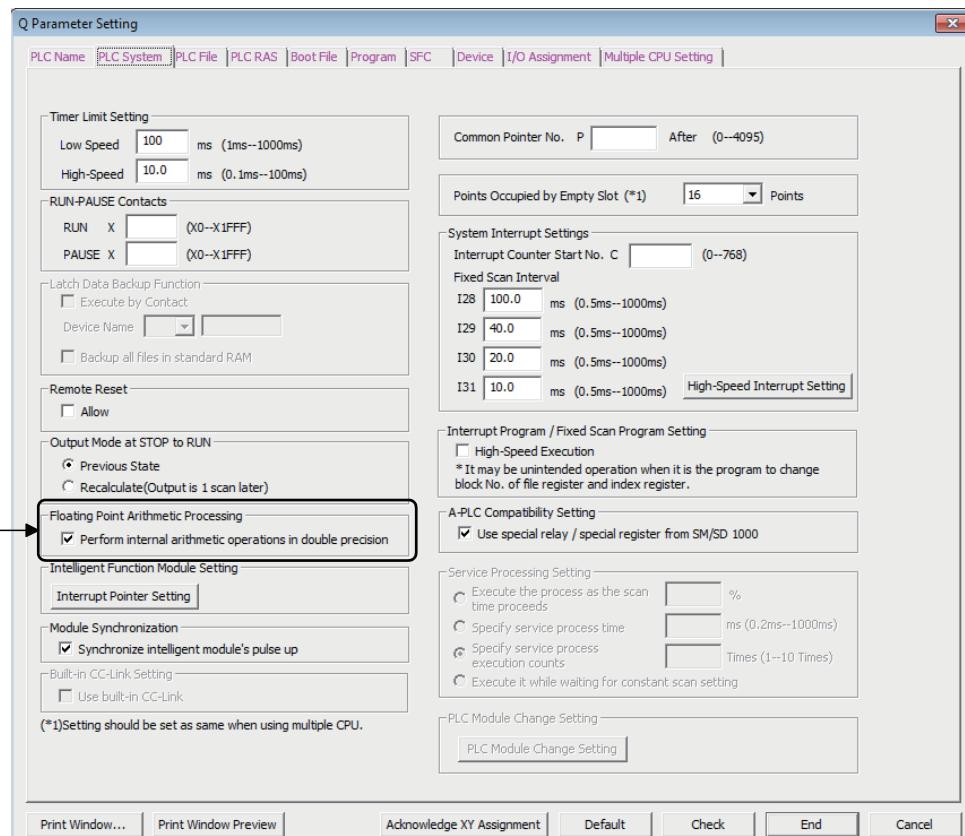
5.1 浮点运算指令

高性能型QCPU和通用型QCPU间的不同点

■高性能型QCPU

高性能型QCPU仅可执行单精度浮点运算指令。

但应注意，通过选择以下所示项目（默认：有勾选）可以双精度执行内部运算处理。



■通用型QCPU

通用型QCPU支持双精度浮点运算指令。

根据处理的数值，可区分使用单精度或双精度的运算。

因此，无法选择PLC参数对话框PLC系统选项卡中的“以双精度执行内部运算处理”项目。

因该新功能的原因，如果在高性能型QCPU中选择了“以双精度执行内部运算处理”，高性能型QCPU和通用型QCPU之间的运算结果（以单精度和双精度）会略有不同。

如果浮点运算需要更高的精度，应用如下所示的替代程序替换浮点运算指令。

32页 用双精度浮点运算指令替换所有单精度浮点运算指令

此外，作为能否将高性能型QCPU的单精度浮点运算直接使用的判断基准，如果浮点运算指令所需精度为有效位数6位或以下的情况下，则无需替换。

不替换指令时，确保其不会造成任何系统故障。

通用型QCPU的浮点运算指令

通用型QCPU的浮点运算指令列表如下所示。

此外，单精度浮点运算指令的规格与高性能型QCPU兼容。

■通用型QCPU中支持的浮点运算指令列表

指令名称		指令符号		备注
		单精度浮点数据	双精度浮点数据	
比较	浮点数据比较	LDE□	LDED□	□表示以下一种运算符： ◇、=、<、>、<=、>=
		ANDE□	ANDED□	
		ORE□	ORED□	
数据传送	浮点数据传送	EMOV (P)	EDMOV (P)	—
四则运算	浮点数据加法	E+ (P)	ED+ (P)	—
	浮点数据减法	E- (P)	ED- (P)	
	浮点数据乘法	E* (P)	ED* (P)	
	浮点数据除法	E/ (P)	ED/ (P)	
数据转换	将BIN 16位数据 →转换为浮点数据	FLT (P)	FLTD (P)	—
	将BIN 32位数据 →转换为浮点数据	DFLT (P)	DFLTD (P)	
	将浮点数据 →转换为BIN16位数据	INT (P)	INTD (P)	
	将浮点数据 →转换为BIN32位数据	DINT (P)	DINTD (P)	
	浮点符号反转	ENEG (P)	EDNEG (P)	
特殊函数	SIN运算	SIN (P)	SIND (P)	—
	COS运算	COS (P)	COSD (P)	
	TAN运算	TAN (P)	TAND (P)	
	SIN ⁻¹ 运算	ASIN (P)	ASIND (P)	
	COS ⁻¹ 运算	ACOS (P)	ACOSD (P)	
	TAN ⁻¹ 运算	ATAN (P)	ATAND (P)	
	角度到弧度的转换	RAD (P)	RADD (P)	
	弧度到角度的转换	DEG (P)	DEGD (P)	
	平方根	SQR (P)	SQRD (P)	
	指数运算	EXP (P)	EXPD (P)	
	自然对数运算	LOG (P)	LOGD (P)	

此外，使用以下指令可在单精度和双精度之间相互转换浮点数据。

指令名称	指令符号
单精度到双精度的转换	ECON (P)
双精度到单精度的转换	EDCON (P)

使用通用型QCPU的双精度浮点数据的优点和缺点

在通用型QCPU中执行双精度浮点运算指令的优点和缺点如下所示。

此外，如果在浮点运算中需要更高的精度，建议将浮点运算替换为双精度浮点运算指令。

■ 使用双精度浮点运算指令的优点和缺点

优点	缺点
精度较单精度浮点运算指令更高。	指令处理速度较单精度浮点运算指令更慢。 ^{*1} 双精度浮点运算数据使用两倍于单精度浮点运算数据的字元件点数。

*1 通用型QCPU中的双精度浮点运算指令的处理速度较在高性能型QCPU中使用内部双精度运算的浮点运算指令更高。

单精度和双精度浮点数据之间的比较如下所示。

项目		单精度浮点数据	双精度浮点数据
数据保持需要的字点数		2个字	4个字
设置范围		$-2^{128} \leq N \leq 2^{128}$, 0, $2^{-126} \leq N < 2^{128}$	$-2^{1024} \leq N \leq 2^{1024}$, 0, $2^{-1022} \leq N < 2^{1024}$
精度 (位数)	尾数部分	23位	52位
	指数部分	8位	11位
	符号部分	1位	1位
指令处理速度 (Q04UDHCPU/ Q06UDHCPU) (最小)	数据比较 (导通状态) (LDE>=/LDDE>=)	0.0285 μ s	3.6 μ s
	数据传送 (EMOV/EDMOV)	0.019 μ s	1.7 μ s
	加 (3个软元件) (E+/ED+)	0.0665 μ s	4.8 μ s
	SIN运算 (SIN/SIND)	4.1 μ s	8.5 μ s
指令处理速度 (通用高速型 QCPU) (最小)	数据比较 (导通状态) (LDE>=/LDDE>=)	0.0098 μ s	1.8 μ s
	数据传送 (EMOV/EDMOV)	0.0039 μ s	0.0078 μ s
	加 (3个软元件) (E+/ED+)	0.015 μ s	1.9 μ s
	SIN运算 (SIN/SIND)	1.6 μ s	2.6 μ s

用通用型QCPU替换高性能型QCPU的方法

■用双精度浮点运算指令替换所有单精度浮点运算指令

单精度浮点数据每个数据占用字元件的2个点。另一方面，每个双精度浮点数据需要占用4个点。因此，需要重新分配用于存储浮点数据的所有软元件号。

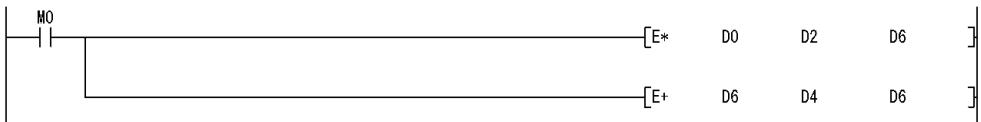
例

替换浮点运算 ‘A×B+C’ （将所有浮点数据更改为双精度。）

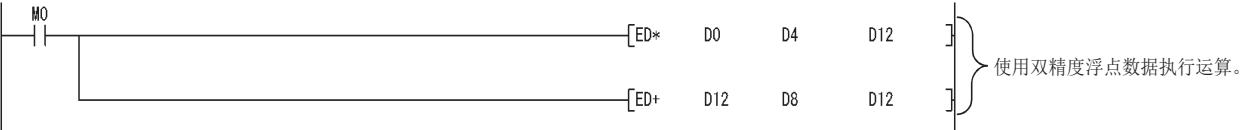
- 软元件配置

(替换前)			(替换后)		
应用	软元件	数据类型	应用	软元件	数据类型
数据A	D0至D1	浮点数据（单精度）	数据A(D)	D0至D3	浮点数据（双精度）
数据B	D2至D3		数据B(D)	D4至D7	
数据C	D4至D5		数据C(D)	D8至D11	
运算结果	D6至D7		运算结果(D)	D12至D15	

- 替换前程序



- 替换后程序



■用双精度浮点运算指令替换部分浮点运算指令

仅需要高精度的运算用双精度浮点运算指令替换。使用ECON和EDCON指令，在单精度和双精度之间相互转换浮点数据。将部分浮点运算以双精度运算时的程序流程如下：

- 使用ECON指令将执行双精度运算所需的数据从单精度转换为双精度。
- 使用双精度浮点运算指令以双精度执行运算。
- 使用EDCON指令将运算结果从双精度转换为单精度。

运算前后在单精度和双精度之间相互转换浮点数据的程序示例如下所示。

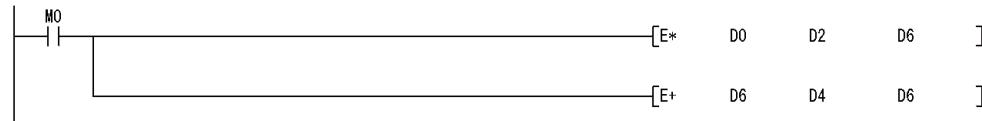
例

替换浮点运算 ‘A×B+C’ （使用ECON和EDCON指令）

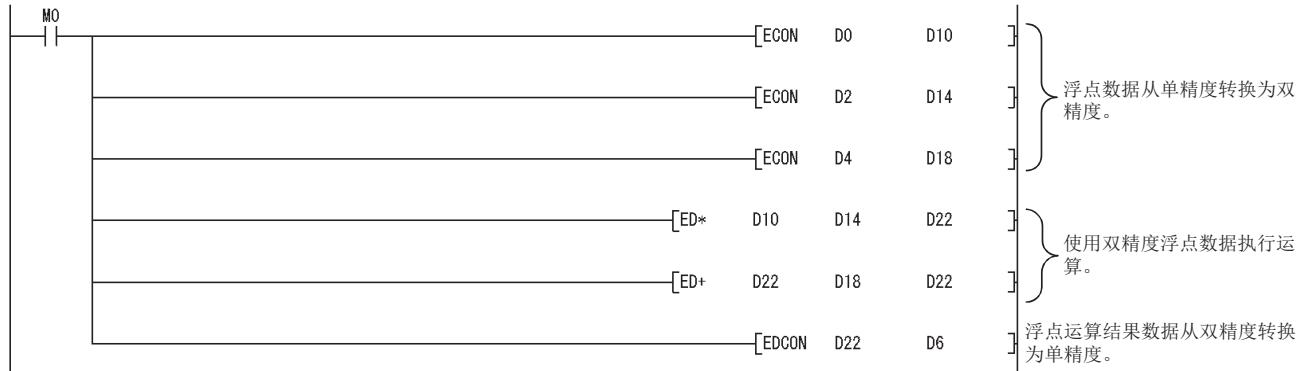
- 软元件配置

(替换前)			(替换后)		
应用	软元件	数据类型	应用	软元件	数据类型
数据A	D0至D1	浮点数据 (单精度)	数据A	D0至D1	浮点数据 (单精度)
数据B	D2至D3		数据B	D2至D3	
数据C	D4至D5		数据C	D4至D5	
运算结果	D6至D7		运算结果	D6至D7	
			数据A(D)	D10至D13	浮点数据 (双精度)
			数据B(D)	D14至D17	
			数据C(D)	D18至D21	
			运算结果(D)	D22至D25	

- 替换前程序



- 替换后程序



■使用子程序用双精度浮点运算指令替换部分浮点运算指令

将上述“用双精度浮点运算指令替换部分浮点运算指令”的程序的流程，视为一个子程序的方法如下所示。

首先为各浮点运算指令创建子程序，

然后用CALL (P) 指令替换原浮点运算指令以调用相应的子程序。使用该方法可最小程度上变更程序，但调用子程序的处理会增大扫描时间。

此外，本方法由于每个浮点运算指令执行从双精度到单精度的转换，运算时产生的化整误差会较以下方法更大。

☞ 33页 用双精度浮点运算指令替换部分浮点运算指令

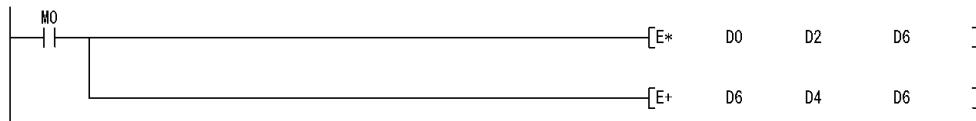
例

替换浮点运算 ‘A×B+C’ （使用子程序）

- 软元件配置

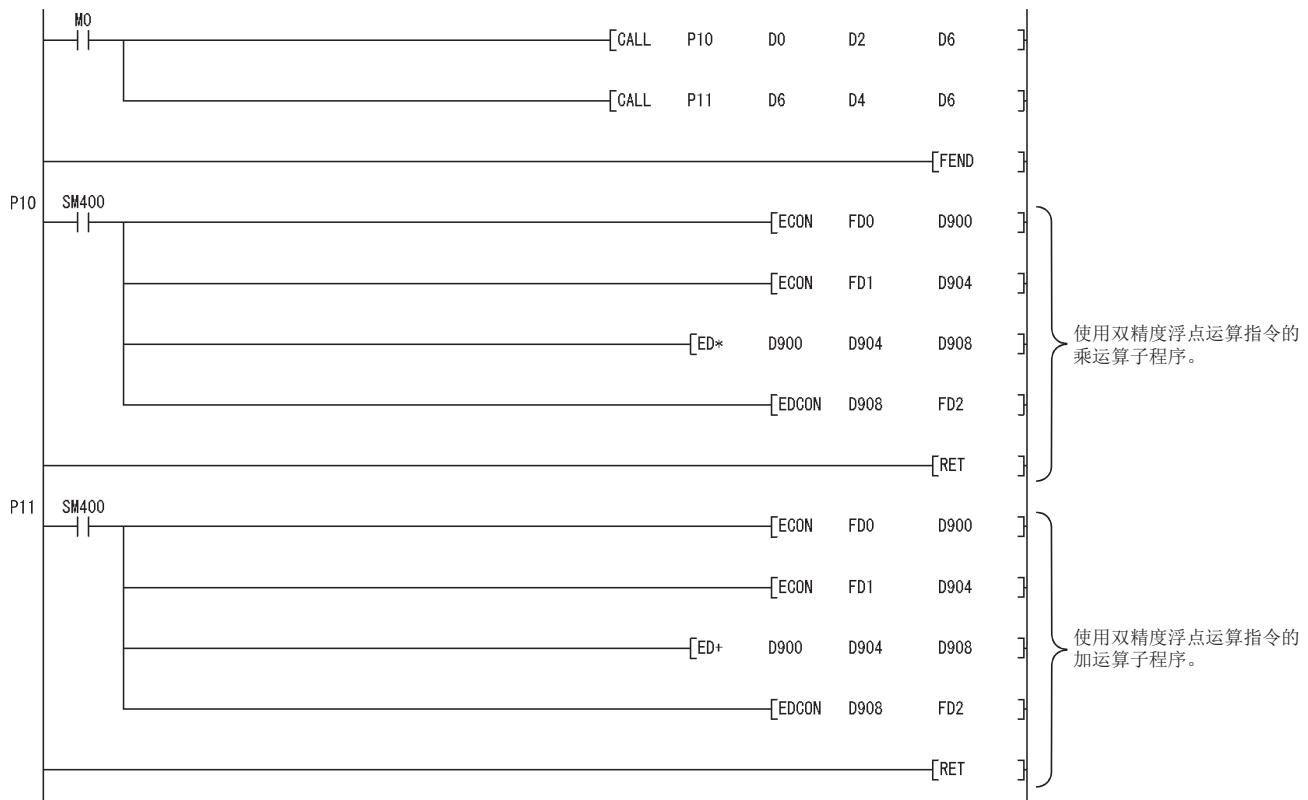
(替换前)			(替换后)		
应用	软元件	数据类型	应用	软元件	数据类型
数据A	D0至D1	浮点数据 (单精度)	数据A	D0至D1	浮点数据 (单精度)
数据B	D2至D3		数据B	D2至D3	
数据C	D4至D5		数据C	D4至D5	
运算结果	D6至D7		运算结果	D6至D7	
			子程序输入数据1	D900至D903	浮点数据 (双精度)
			子程序输入数据2	D904至D907	
			子程序运算结果	D908至D911	

- 替换前程序



FA-CN-0001-B

• 替换后程序



5.2 浮点数据比较指令的错误检查处理（不包括通用高速型QCPU）

浮点数据比较指令的输入数据检查

通用型QCPU的浮点数据比较指令的错误检查处理得到了加强。

会检查“特殊值”（-0、非数值、非规范化数或 $\pm\infty$ ），如果输入任何特殊值，CPU模块会检测“运算错误”（错误代码：4140）。

使用LDE□、ANDE□、ORE□、LDED□、ANDED□或ORED□指令（□表示以下一种运算符：=、 \diamond 、<、>、 \leq 、 \geq ）时，即使在指令前设为以显示浮点数据有效性的信号（有效数据标志）进行互锁，但是如果存在无效浮点数据，也将会检测到“运算错误”（错误代码：4140）。

根据通用型QCPU中执行的浮点运算结果，无效浮点数据不会被存储。

存储无效浮点数据的原因有以下可能：

- 相同的软元件用于存储浮点数据和其他数据，如二进制值、BCD值和字符串。

→使用不同的软元件存储浮点数据和非浮点数据。

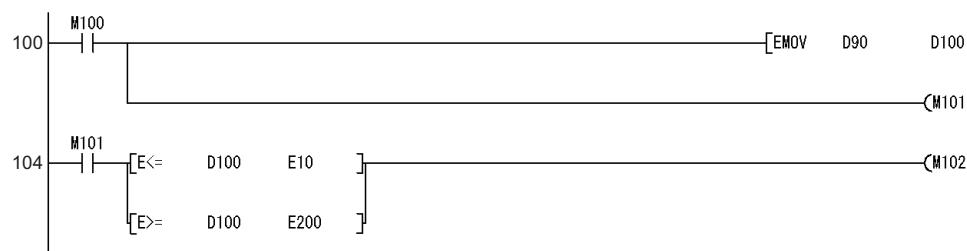
- 外部写入的浮点数据无效。

→对外部来源采取措施以写入有效数据。

如果浮点数据比较指令中发生错误，应采取措施以消除以上错误原因。

示例1) 使用LDE□指令检测“运算错误”（错误代码：4140）

[梯形图模式]



[列表模式]

100	LD	M100	
101	EMOV	D90	D100
103	OUT	M101	
104	LD	M101	
105	LDE \leq	D100	E10
109	ORE \geq	D100	E200
113	ANB		
114	OUT	M102	

在从步104开始的梯形图块中，当M101（有效数据标志）为OFF时，步105和109的浮点数据比较指令不执行。

但是，无论步104的LD指令在上述程序中的执行结果如何，都会执行步105的LDE \leq 指令和步109的ORE \geq 指令。因此，即使当M101为OFF时，如果在D100中存储了‘特殊值’，也会在步105的LDE \leq 指令中检测到“运算错误”（错误代码：4140）。

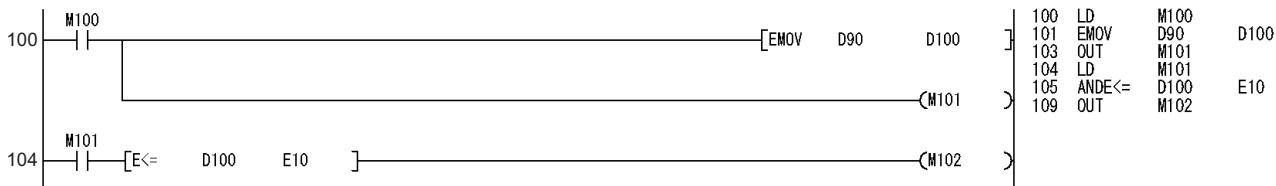
有关避免“运算错误”的方法，应参照下述内容。

38页 在浮点数据比较指令中避免“运算错误”（错误代码：4140）的方法

FA-CN-0001-B

示例2) 使用ANDE \leq 指令不检测“运算错误”（错误代码：4140）

[梯形图模式]



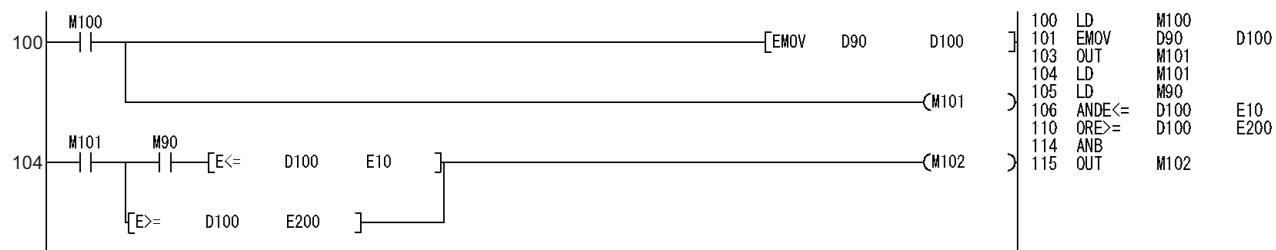
在从步104开始的梯形图块中，当M101（有效数据标志）为OFF时，不会执行步105的ANDE \leq 指令。

当M101在上述程序中的步104的LD指令中为OFF时，不会执行步105的ANDE \leq 指令。

因此，当M101为OFF时，即使在D100中存储了‘特殊值’，也将不会检测“运算错误”（错误代码：4140）。

示例3) 使用ANDE \leq 指令检测“运算错误”（错误代码：4140）

[梯形图模式]



在从步104开始的梯形图块中，当M101（有效数据标志）为OFF时，不会执行步106的ANDE \leq 指令和步110的OR \geq 指令。

但是，如果M90在步105的LD指令中为ON，则会执行步106的ANDE \leq 指令。

因此，即使当M101为OFF时，如果M90为ON并在D100中存储了‘特殊值’，将会在步106的ANDE \leq 指令中检测到“运算错误”（错误代码：4140）。

有关避免“运算错误”的方法，应参照下述内容。

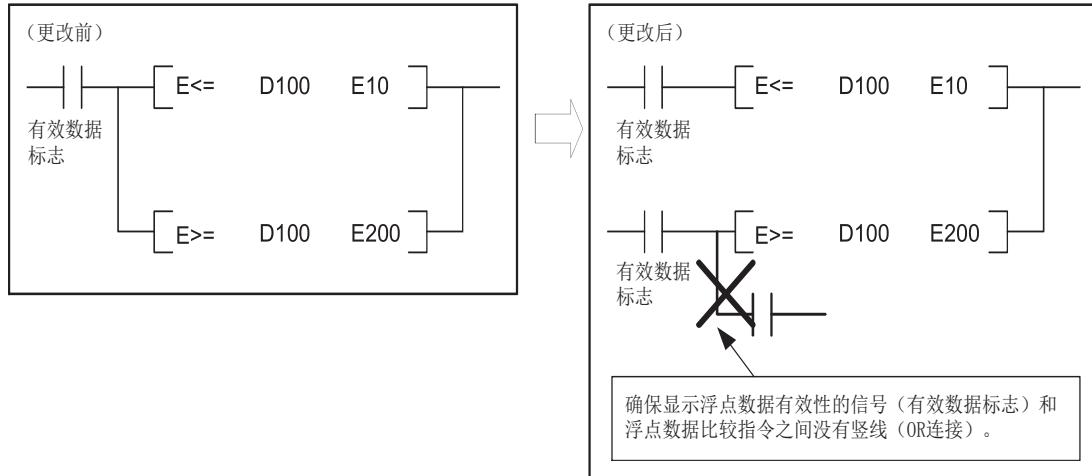
38页 在浮点数据比较指令中避免“运算错误”（错误代码：4140）的方法

在浮点数据比较指令中避免“运算错误”（错误代码：4140）的方法

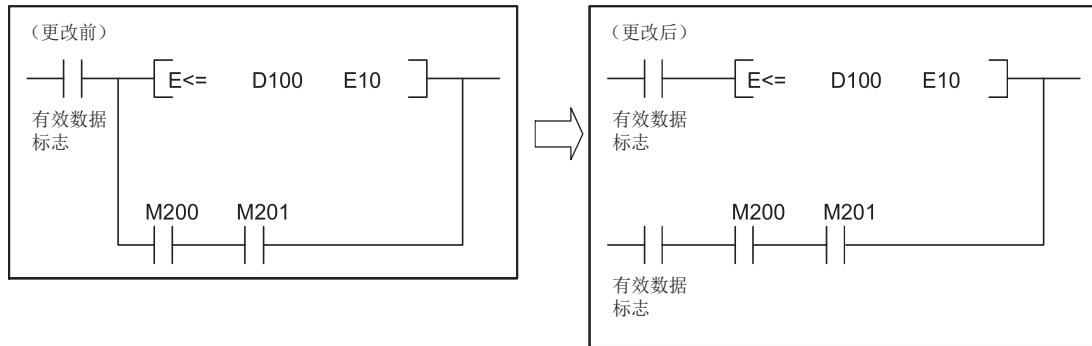
如以下更改示例1、2中所示，串联连接有效数据标志触点至浮点数据比较指令。（连接有效数据标志触点和浮点数据比较指令时使用AND连接。）

此时应确保有效数据标志触点和浮点数据比较指令之间没有竖线（OR连接）。

〈更改示例1〉



〈更改示例2〉

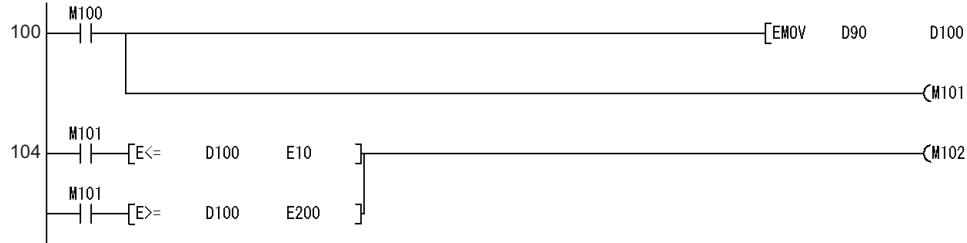


FA-CN-0001-B

对应上述示例1) 和示例3) 的更改示例如示例4) 、示例5) 所示。

示例4) 更改的程序 (示例1) (“运算错误”) (错误代码: 4140) 不再被检测。)

[梯形图模式]

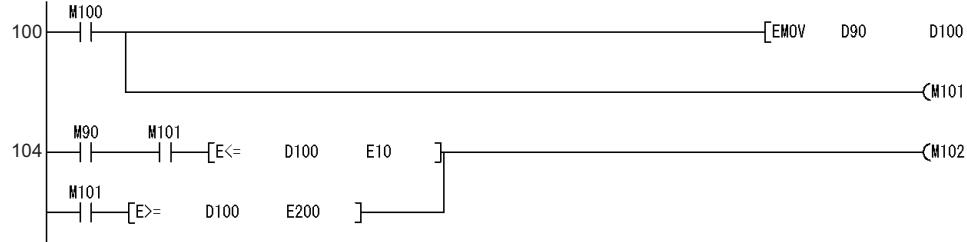


[列表模式]

100	LD	M100	D100
101	EMOV	D90	
103	OUT	M101	
104	LD	M101	
105	ANDE<=	D100	E10
109	LD	M101	
110	ANDE>=	D100	E200
114	ORB		
115	OUT	M102	

示例5) 更改的程序 (示例3) (“运算错误”) (错误代码: 4140) 不再被检测。)

[梯形图模式]



[列表模式]

100	LD	M100	D100
101	EMOV	D90	
103	OUT	M101	
104	LD	M90	
105	AND	M101	
106	ANDE<=	D100	E10
110	LD	M101	
111	ANDE>=	D100	E200
115	ORB		
116	OUT	M102	

5.3 变址修饰的软元件的范围检查处理

变址修饰的软元件范围检查

通用型QCPU在软元件变址修饰时的错误检查处理得到了加强。

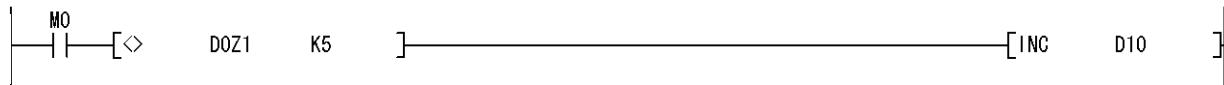
会检查每个变址修饰的软元件范围，如果检查目标软元件在变址修饰前超出软元件范围，CPU模块会检测“运算错误”（错误代码：4101）。

要点

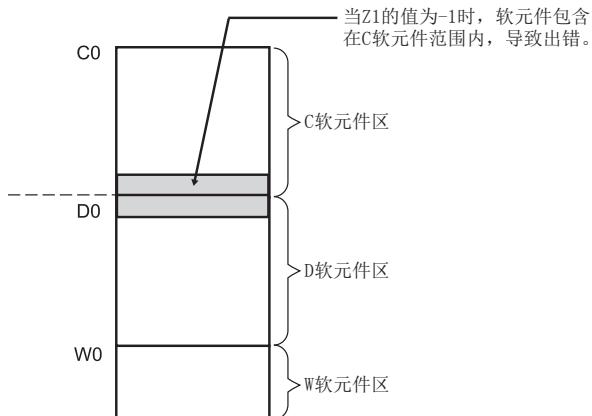
有关变址修饰的软元件范围检查，请参照以下：

『MELSEC-Q/L 编程手册（公共指令篇）』

示例1) 通过在软元件变址修饰时进行错误检查处理检测“运算错误”（错误代码：4101）。



在示例1) 中，当触点（M0）为ON并在Z1中指定了值-1或以下时，如下图所示，软元件D0Z1包含在C软元件范围，超出D软元件范围。因此，将会检测到“运算错误”（错误代码：4101）。



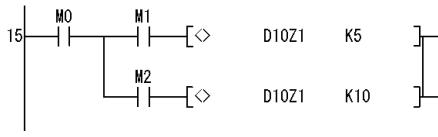
检测到错误时，应检查变址修饰值（上述示例中Z1的值）并排除出错原因。

检测到和未检测到错误情形的示例如下所示。

FA-CN-0001-B

示例2) 检测“运算错误”(错误代码: 4101)

[梯形图模式]



[列表模式]

15	LD	M0
16	LD	M1
17	AND<>	D10Z1 K5
20	LD	M2
21	AND<>	D10Z1 K10
24	ORB	
25	ANB	
26	MOV	D0 D1
28	INC	D0 D2

在示例2中，在从步15开始的梯形图块中，当M0（有效数据标志）为OFF时，步骤17或21的AND <>指令将不会执行。

但是，由于在步16和20中使用了始终执行的LD指令，当M1或M2为ON时，无论步15中的LD指令的执行结果如何，步17或21的AND <>指令都会执行。

因此，即使当M0为OFF，如果D10Z1值超出D软元件范围，将会在步17的AND <>指令中检测到“运算错误”(错误代码: 4101)。

注意，步26 (MOV D0 D1) 和步28 (INC D2) 不会执行。

有关为避免“运算错误”(错误代码: 4101) 所要采取的措施，应参照下述内容。

☞ 42页 为避免变址修饰时的“运算错误”(错误代码: 4101) 所需采取的措施

示例3) 检测“运算错误”(错误代码: 4101)



在示例3中，即使当步15中的M0（有效数据标志）为OFF，下一步（步16）中的AND指令也将执行。因此，如果X10Z1值超出X软元件范围，将会在步16的AND指令中

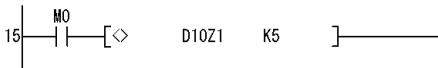
检测到“运算错误”(错误代码: 4101)。

有关为避免“运算错误”(错误代码: 4101) 所要采取的措施，应参照下述内容。

☞ 42页 为避免变址修饰时的“运算错误”(错误代码: 4101) 所需采取的措施

示例4) 不检测“运算错误”(错误代码: 4101)

[梯形图模式]



[列表模式]

15	LD	M0
16	AND<>	D10Z1 K5
19	MOV	D0 D1

在示例4中，当步15的M0（有效数据标志）为OFF时，步16的AND <>指令不会执行。

因此，无论D10Z1值如何，都不会检测到“运算错误”(错误代码: 4101)。

为避免变址修饰时的“运算错误”（错误代码：4101）所需采取的措施

如果无需检查变址修饰的软元件范围，应采取下述①中的措施。

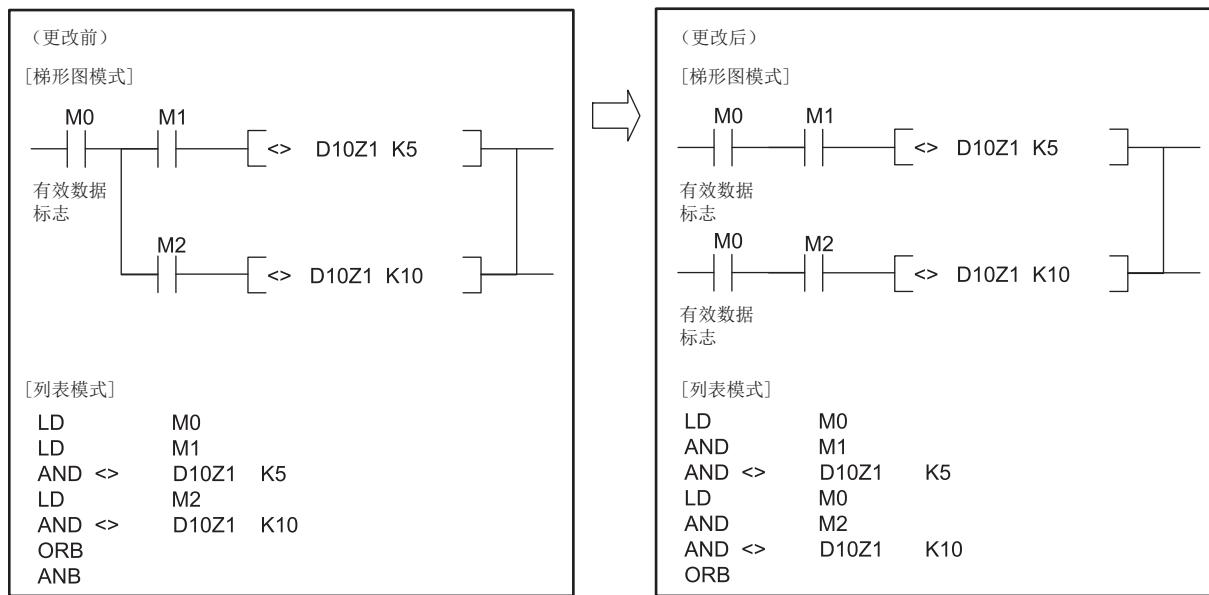
如果需要检查变址修饰的软元件范围，但要避免上述示例2) 和示例3) 中所示的错误检测时，应采取②至④中的措施。

☞ 40页 变址修饰的软元件范围检查

①取消选择PLC参数对话框PLC RAS选项卡中的“检查变址修饰的软元件范围”，以不检查变址修饰的软元件范围。

②如以下更改示例中所示，为检查变址修饰的软元件范围的每个指令串联连接有效数据标志的触点。（这不适用于通用高速型QCPU。）

<更改示例>



在更改前程序（左侧）中，正位于AND <>指令前的指令被视为LD指令。但在更改后程序（右侧）中，相同的指令将被视为AND指令。

在更改后程序中，仅当M0和M1（或M2）的触点都打开时，才会执行AND <>指令。因此，在变址修饰的软元件范围检查处理期间不会检测到错误。

③使用变址寄存器作为局部软元件

对于执行多个程序的工程，如果单独执行检测到“运算错误”（错误代码：4101）的程序且未出错，应使用变址寄存器作为局部软元件。

这可使各程序独立使用变址寄存器。即使其他程序用“导致变址修饰的软元件超出软元件范围的值”改写了变址寄存器，也将不会影响出错程序中使用的变址寄存器的值。因此，将不会检测到“运算错误”（错误代码：4101）。

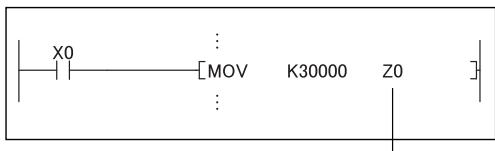
注意，由于保存和恢复局部软元件文件的时间增加，扫描时间也会增加。关于使用局部软元件时的设置，应参照下述手册。

☞ QnUCPU用户手册（功能解说/程序基础篇）

例

当变址寄存器用作局部软元件时

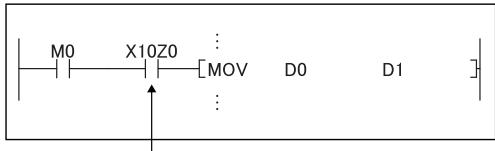
程序A



标准RAM/存储卡 (SRAM)



程序B

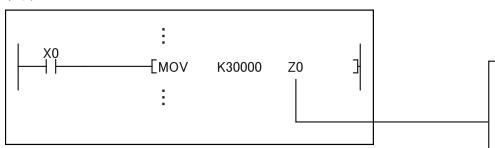


即使程序A使用30000值改写变址寄存器Z0的值，也不会改写程序B使用的变址寄存器Z0的值。只要X10Z0不超出X软元件范围，就不会发生错误。

例

当变址寄存器未用作局部软元件时

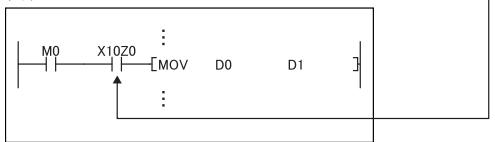
程序A



软元件存储器



程序B

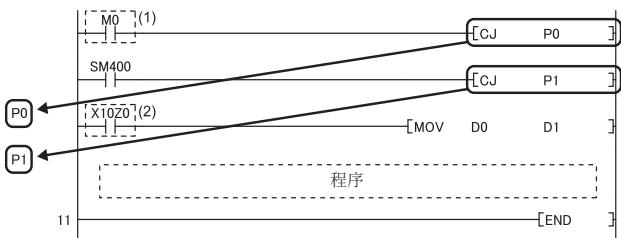


当程序A使用30000值改写变址寄存器Z0时，程序B使用的变址寄存器Z0的值也会更改。当X10Z0超出X软元件范围时，会发生错误。

④ 使用CJ指令

当按如下所示使用CJ指令并且前一个条件（下图中的“(1) LD M0”）为OFF时，应避免执行使用变址寄存器的触点指令（下图中的“(2) LD X10Z0”）。当下图中的(1)为OFF时，下图中的指令(2)不被执行，并且不读取用作触点的软元件的值。因此，软元件范围检查处理不会检测“运算错误”（错误代码：4101）。

注意，使用CJ指令会增加相应扫描时间。



5.4 软元件锁存功能

概述

与高性能型QCPU相比，通用型QCPU的软元件锁存功能^{*1}得到了进一步的加强。

本节介绍有关通用型QCPU中锁存功能的使用方法的要点。

*1 锁存功能用于在CPU模块断电或复位时保持软元件数据。

锁存功能的类型

通用型QCPU的软元件数据可通过以下方法锁存：

- 使用大容量文件寄存器 (R、ZR) ^{*1}
- 写入软元件数据至标准ROM，或从标准ROM读取软元件数据（使用SP. DEVST/S(P). DEVLD指令）
- 指定内部用户软元件的锁存范围
- 在锁存间隔设置参数中设置间隔（“时间设置”）^{*2}

*1 包括扩展数据寄存器 (D) 和扩展链接寄存器 (W)。

*2 仅通用高速型QCPU支持该设置。

锁存功能的详情

■大容量文件寄存器 (R、ZR)

文件寄存器是可通过电池进行锁存的软元件。

与高性能型QCPU相比，在通用型QCPU中，文件寄存器容量更大，处理速度更高。

锁存大量数据（多软元件点数）时，使用文件寄存器较为有效。

下表列出了各CPU模块中的文件寄存器容量。

- 各CPU模块中的文件寄存器容量

型号	标准RAM中文件寄存器 (R、ZR) 容量
Q02UCPU	64K点
Q03UD(E)CPU	96K点
Q03UDVCPU	不带扩展SRAM卡盒
	带扩展SRAM卡盒 (1M)
	带扩展SRAM卡盒 (2M)
	带扩展SRAM卡盒 (4M)
	带扩展SRAM卡盒 (8M)
Q04UD(E)CPU	128K点
Q04UDVCPU	不带扩展SRAM卡盒
	带扩展SRAM卡盒 (1M)
	带扩展SRAM卡盒 (2M)
	带扩展SRAM卡盒 (4M)
	带扩展SRAM卡盒 (8M)
Q06UD(E)CPU	384K点
Q06UDVCPU	不带扩展SRAM卡盒
	带扩展SRAM卡盒 (1M)
	带扩展SRAM卡盒 (2M)
	带扩展SRAM卡盒 (4M)
	带扩展SRAM卡盒 (8M)
Q10UD(E)HCPU、Q13UD(E)HCPU	512K点
Q13UDVCPU	不带扩展SRAM卡盒
	带扩展SRAM卡盒 (1M)
	带扩展SRAM卡盒 (2M)
	带扩展SRAM卡盒 (4M)
	带扩展SRAM卡盒 (8M)
Q20UD(E)HCPU、Q26UD(E)HCPU	640K点

型号	标准RAM中文件寄存器 (R、ZR) 容量
Q26UDVCPU	不带扩展SRAM卡盒 640K点
	带扩展SRAM卡盒 (1M) 1152K点
	带扩展SRAM卡盒 (2M) 1664K点
	带扩展SRAM卡盒 (4M) 2688K点
	带扩展SRAM卡盒 (8M) 4736K点
Q50UDEHCPU	768K点
Q100UDEHCPU	896K点

■写入软元件数据至标准ROM, 或从标准ROM读取软元件数据 (SP.DEVST/S(P).DEVLD指令)

使用SP.DEVST/S(P).DEVLD指令 (用于写入数据至标准ROM/从标准ROM读取数据的指令) 可以锁存通用型QCPU的软元件数据。

利用标准ROM可无需电池进行数据备份。该方法可有效锁存不经常更新的数据。

■指定内部用户软元件的锁存范围

可按照与高性能型QCPU相同的方式, 通过指定内部用户软元件的锁存范围来锁存通用型QCPU的软元件数据。

可在PLC参数对话框的软元件选项卡中设定该范围。

可以锁存的内部用户软元件如下:

- 锁存继电器 (L)
- 链接继电器 (B)
- 报警器 (F)
- 变址继电器 (V)
- 定时器 (T)
- 累积定时器 (ST)
- 计数器 (C)
- 数据寄存器 (D)
- 链接寄存器 (W)

要点

- 如果在通用型QCPU中设定内部用户软元件的锁存范围, 处理时间将与被锁存的软元件点数成比例增加。(例如, 如果8K点被锁存用于QnUD(E)(H)CPU的锁存继电器 (L), 处理时间为28.6 μs。) 要缩短扫描时间, 应删除不必要的锁存软元件点以最大程度地减小锁存范围。
- 即使指定文件寄存器 (R、ZR) 为锁存范围, 扫描时间也不会延长。

如何缩短扫描时间

当被锁存的数据存储于文件寄存器 (R或ZR) 时, 处理时间会较锁存内部用户软元件更短。

例

将数据寄存器 (D) 的锁存点从8K点减少至2K点, 并替代使用文件寄存器 (ZR) 时 (使用Q06UDVCPU时)

■数据寄存器 (D) 的锁存范围替换至文件寄存器 (ZR) 时的差异

项目		更改前	更改后
数据寄存器 (D) 的锁存点		8192 (8K) 点	2048 (2K) 点 (6K点被移至文件寄存器。)
程序中的软元件使用数	数据寄存器 (D) (锁存范围)	400	100
	文件寄存器 (ZR) (使用标准RAM)	0	300
增加的扫描时间		0.37ms	0.11ms*1
增加的步数		—	300步

*1 表示当文件寄存器数据被存储于标准RAM时额外需要的时间。

要点

通用高速型QCPU中, 在锁存间隔设置选择“时间设置”时, 可更改为在指定时间经过后的END处理锁存的动作。在此情况下, 由于锁存处理的执行与顺控程序不同步, 会减小扫描时间的增幅。

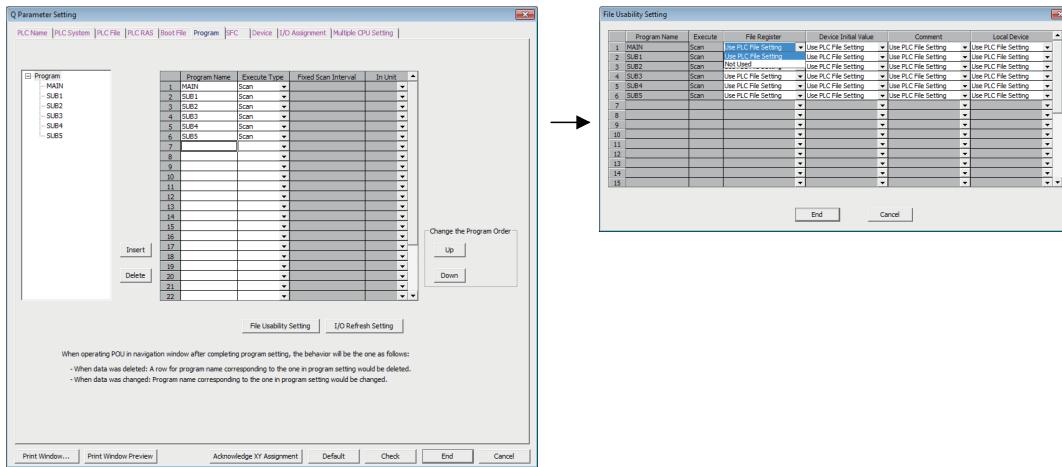
5.5 文件使用方法设置

高性能型QCPU和通用型QCPU之间的差异

■高性能型QCPU

在高性能型QCPU中，可通过在PLC参数对话框的程序选项卡上单击“文件使用方法设置”按钮打开窗口，并在窗口中为各程序设置以下文件的文件使用方法（“使用PLC文件设置”或“不使用”）。

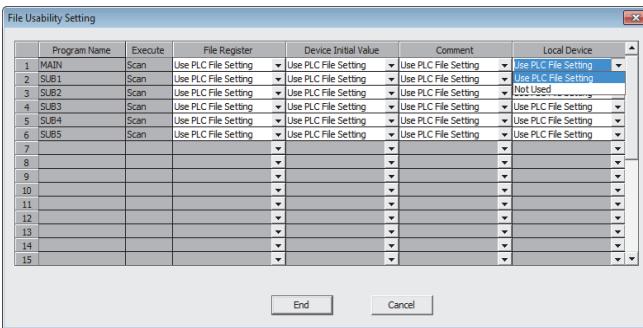
- 文件寄存器
- 软元件初始值
- 注释
- 局部软元件



■通用型QCPU

在通用型QCPU中，无法通过在PLC参数对话框的程序选项卡上单击“文件使用方法设置”按钮打开窗口，并在窗口中为各程序设置以下文件的文件使用方法。^{*1}

- 文件寄存器
- 软元件初始值
- 注释



*1 如果序列号（前5位）为“10011”或更早版本，局部软元件文件使用方法设置也不可用于通用型QCPU。

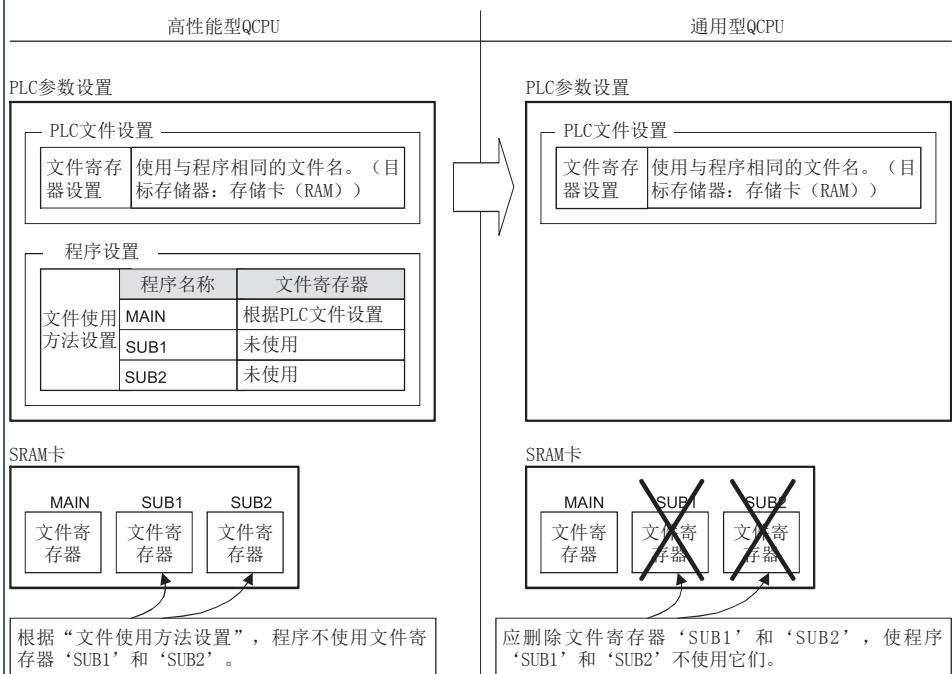
如果在高性能型QCPU中PLC参数对话框的PLC文件选项卡中设定为使用局部软元件时，用通用型QCPU替换后，所有程序都使用局部软元件。

当在高性能型QCPU中设定了文件使用方法设置时，应按下页所述进行替换。

用通用型QCPU替换高性能型QCPU的方法

替换方法因PLC参数对话框PLC文件选项卡中的设置而异。

■至通用型QCPU的替换方法

PLC文件选项卡中的设置	至通用型QCPU的替换方法
选择了“不使用”。	<p>无需更改参数设置。</p> <p>无论高性能型QCPU中的文件使用方法设置如何，通用型QCPU的动作均保持相同。</p>
选择了“使用和程序名相同的文件名”。	<p>当文件使用方法在高性能型QCPU中被设为“不使用”时，应从目标存储器删除使用与程序相同名称的相应程序文件（文件寄存器、软元件初始值或注释）。</p> <p>在执行程序时，如目标存储器中没有与程序相同名称的程序文件，通用型QCPU会执行程序而不使用程序文件。</p> 
选择了“使用以下文件”时。	<p>无需更改参数设置。</p> <p>无论高性能型QCPU中的文件使用方法设置如何，通用型QCPU的动作均保持相同。</p>

5.6 参数有效驱动器和引导文件设置

高性能型QCPU和通用型QCPU之间的差异

■高性能型QCPU

高性能型QCPU通过CPU模块前部的开关指定参数有效驱动器。

■通用型QCPU

通用型QCPU根据各驱动器（程序存储器、存储卡、SD存储卡或标准ROM）中有无参数自动判定参数有效驱动器。因此，在用通用型QCPU替换高性能型QCPU时，可能需要更改参数的引导文件设置或移动文件至其他驱动器。

用通用型QCPU替换高性能型QCPU时，应按如下所示进行替换。

用通用型QCPU替换高性能型QCPU

■在高性能型QCPU中当参数有效驱动器被设置到标准ROM时

- 当参数有效驱动器被设置到标准ROM时

高性能型QCPU中的设置	通用型QCPU中的设置
PCL参数对话框引导文件选项卡中的设置	
无引导文件设置。	更改设置，使通用型QCPU可参照标准ROM中的参数。 <ul style="list-style-type: none">无需更改参数设置。删除程序存储器、存储卡或SD存储卡中存在的参数。^{*1}
如下进行了引导文件设置。（无参数的引导文件设置） <ul style="list-style-type: none">数据类型：程序传送源：标准ROM传送至：程序存储器 或 <ul style="list-style-type: none">数据类型：参数传送源：标准ROM传送至：程序存储器	更改设置，使程序预先存储在程序存储器，而非从标准ROM引导。 <ul style="list-style-type: none">删除PLC参数对话框引导文件选项卡中所有的参数设置。删除程序存储器、存储卡或SD存储卡中存在的参数。^{*2}将带有引导设置的程序从标准ROM移至程序存储器。^{*1}
如下进行了引导文件设置。 <ul style="list-style-type: none">数据类型：程序传送源：标准ROM传送至：程序存储器 或 <ul style="list-style-type: none">数据类型：参数传送源：标准ROM传送至：程序存储器	更改设置，使程序和参数预先存储在程序存储器，而非从标准ROM引导。 <ul style="list-style-type: none">将带有引导设置的程序和参数从标准ROM移至程序存储器。^{*1}删除PLC参数对话框引导文件选项卡中所有的参数设置。
如下进行了引导文件设置。（无参数的引导文件设置） <ul style="list-style-type: none">数据类型：程序传送源：存储卡传送至：程序存储器 或	更改设置，使通用型QCPU可使用存储卡或SD存储卡中的参数，并从存储卡或SD存储卡引导程序至程序存储器。 <ul style="list-style-type: none">移动标准ROM中的参数至存储卡或SD存储卡。在PLC参数对话框引导文件选项卡中进行设置，使程序从存储卡或SD存储卡引导至程序存储器。^{*3}
如下进行了引导文件设置。 <ul style="list-style-type: none">数据类型：程序传送源：标准ROM传送至：程序存储器 或 <ul style="list-style-type: none">数据类型：参数传送源：标准ROM传送至：程序存储器	更改设置，使通用型QCPU可使用存储卡或SD存储卡中的参数，并从存储卡或SD存储卡引导程序和参数至程序存储器。 <ul style="list-style-type: none">移动标准ROM中的参数至存储卡或SD存储卡。在PLC参数对话框引导文件选项卡中进行设置，使程序和参数从存储卡或SD存储卡引导至程序存储器。^{*3}
如下进行了引导文件设置。 <ul style="list-style-type: none">数据类型：非程序和参数的数据传送源：存储卡传送至：程序存储器 或 <ul style="list-style-type: none">数据类型：非程序和参数的数据传送源：标准ROM传送至：程序存储器 <p>（非程序和参数的数据表示软元件初始值、软元件注释和标签程序。）</p>	在引导文件设置中删除非程序和参数数据的所有设置。由于这些数据虽未存储于程序存储器中但仍可使用，无需将它们引导至程序存储器。或者，更改设置使它们预先存储于程序存储器。 <ul style="list-style-type: none">删除PLC参数对话框引导文件选项卡中非程序和参数数据的引导设置。根据需要移动非程序和参数数据至程序存储器。

*1 通用型QCPU在电池电压下降时仍可保持程序存储器中的数据，因而无需引导文件设置。

*2 通用型QCPU按程序存储器→存储卡或SD存储卡→标准ROM的顺序搜索参数。然后，模块使用最先查找到的参数。如果参数存在于程序存储器、存储卡或SD存储卡中，通用型QCPU不使用标准ROM中的参数。

*3 通用型QCPU忽略标准ROM中参数的引导文件设置。

■在高性能型QCPU中当参数有效驱动器被设置到存储卡（RAM）或存储卡（ROM）时

- 当参数有效驱动器被设置到存储卡（RAM）或存储卡（ROM）时

高性能型QCPU中的设置	通用型QCPU中的设置
PCL参数对话框引导文件选项卡中的设置 无引导文件设置。	更改设置，使通用型QCPU可参照存储卡或SD存储卡中的参数。 <ul style="list-style-type: none">无需更改参数设置。删除程序存储器中存在的参数。^{*2}
如下进行了引导文件设置。（无参数的引导文件设置） <ul style="list-style-type: none">数据类型：程序传送源：存储卡传送至：程序存储器	更改设置，使通用型QCPU可参照存储卡或SD存储卡中的参数。 <ul style="list-style-type: none">无需更改参数设置。删除程序存储器中存在的参数。^{*2}
如下进行了引导文件设置。 <ul style="list-style-type: none">数据类型：程序传送源：标准ROM传送至：存储卡 或 <ul style="list-style-type: none">数据类型：参数传送源：存储卡传送至：程序存储器	无需更改。
如下进行了引导文件设置。（无参数的引导文件设置） <ul style="list-style-type: none">数据类型：程序传送源：标准ROM传送至：程序存储器	更改设置，使程序预先存储在程序存储器，而非从标准ROM引导。 <ul style="list-style-type: none">从标准ROM将用于引导的程序移动至程序存储器。^{*1}删除PLC参数对话框引导文件选项卡中所有的程序设置。删除程序存储器中的存在参数。^{*2}
如下进行了引导文件设置。 <ul style="list-style-type: none">数据类型：程序传送源：标准ROM传送至：程序存储器 或 <ul style="list-style-type: none">数据类型：参数传送源：存储卡传送至：程序存储器	更改设置，使程序预先存储在程序存储器，而非从标准ROM引导。 <ul style="list-style-type: none">从标准ROM将用于引导的程序移动至程序存储器。^{*1}删除PLC参数对话框引导文件选项卡中所有的程序引导设置。
如下进行了引导文件设置。 <ul style="list-style-type: none">数据类型：非程序和参数的数据传送源：存储卡传送至：程序存储器 或 <ul style="list-style-type: none">数据类型：非程序和参数的数据传送源：标准ROM传送至：程序存储器 <p>（非程序和参数的数据表示软元件初始值、软元件注释和标签程序。）</p>	在引导文件设置中删除非程序和参数数据的所有设置。由于这些数据虽未存储于程序存储器中但仍可使用，无需将它们引导至程序存储器。或者，更改设置使它们预先存储于程序存储器。 <ul style="list-style-type: none">删除PLC参数对话框引导文件选项卡中非程序和参数数据的引导设置。根据需要移动非程序和参数数据至程序存储器。

*1 通用型QCPU在电池电压下降时仍可保持程序存储器中的数据，因而无需引导文件设置。

*2 通用型QCPU按程序存储器→存储卡或SD存储卡→标准ROM的顺序搜索参数。然后，模块使用最先查找到的参数。如果参数存在于程序存储器、存储卡或SD存储卡中，通用型QCPU不使用标准ROM中的参数。

5.7 外部输入输出强制ON/OFF功能

高性能型QCPU和通用型QCPU之间的差异

■高性能型QCPU

通过在编程工具中选择[Online]（在线）⇒[Debug]（调试）⇒[Forced input output registration/cancellation]（强制输入输出注册/取消）而打开的窗口中，可强制ON/OFF外部输入输出。

■通用型QCPU

如果序列号（前5位）为“10041”或更早版本，则无法使用外部输入输出强制ON/OFF功能。

可通过使用下述替代程序强制ON/OFF外部输入输出。

用通用型QCPU替换高性能型QCPU的方法

如下图所示，在PLC参数对话框的程序选项卡中添加程序名称“SETX”和“SETY”。

〈替换前〉

〈替换后〉



	Program Name	Execute Type	Fixed Scan Interval	In Unit
1	MAIN	Scan	▼	▼
2	SUB	Scan	▼	▼
3		▼	▼	▼

	Program Name	Execute Type	Fixed Scan Interval	In Unit
1	SETX	Scan	▼	▼
2	MAIN	Scan	▼	▼
3	SUB	Scan	▼	▼
4	SETY	Scan	▼	▼
5		▼	▼	▼

下表显示了“SETX”和“SETY”的程序设置。

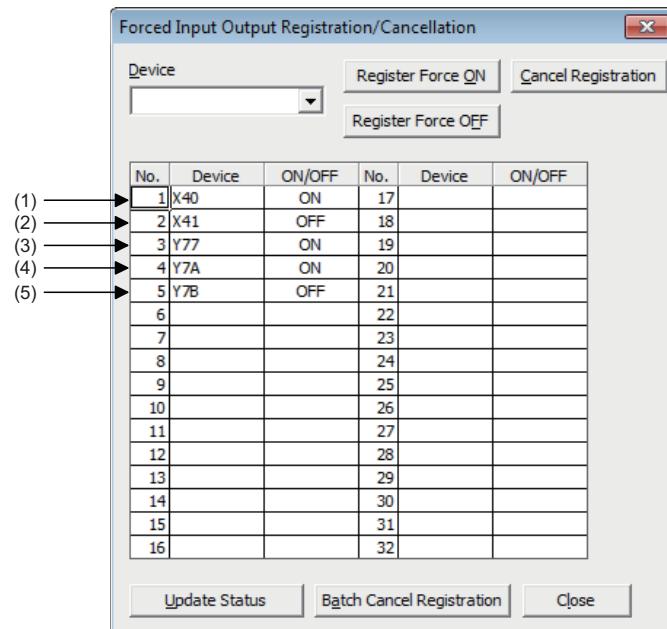
程序名称	执行类型	程序的添加位置
SETX	扫描	程序设置起始（编号1）
SETY	扫描	程序设置末尾

例

将X40、X77 和X7A强制ON，将X41 和Y7B强制OFF。

程序“SETX”和“SETY”使用SET和RST指令在每一次扫描对X和Y进行ON或OFF，该软元件已在外部输入输出强制ON/OFF功能注册为强制ON/OFF。

高性能型QCPU



通用型QCPU

- “SETX”的程序示例



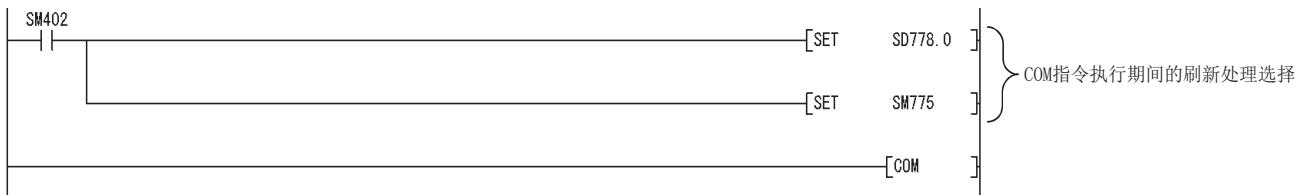
- “SETY”的程序示例



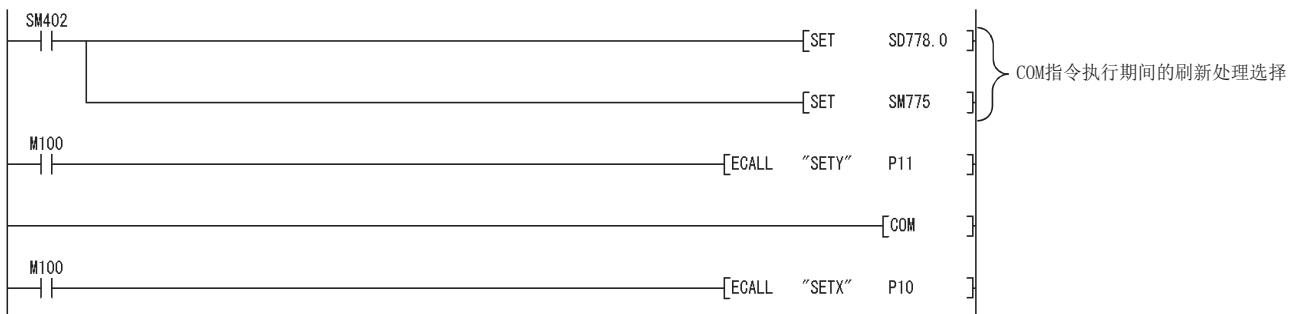
替换COM指令

如果使用COM指令，应在COM指令前和后添加调用P10和P11的子程序。（P10和P11为第5.7节中“用通用型QCPU替换高性能型QCPU的方法”的程序示例中所示的指针。）当SM775为ON（执行通过SD778设置的刷新）并且SD778的0位为OFF（不执行输入输出刷新）时，无需替换指令。

■替换前程序



■替换后程序



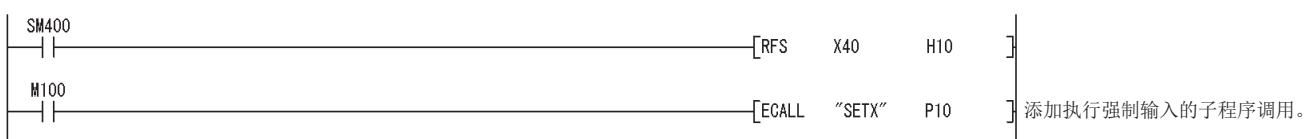
替换RFS指令

如果用于强制ON/OFF的任何输入输出编号包含在通过RFS指令指定的部分刷新范围内，应在RFS指令前和后添加调用P10和P11的子程序。（P10和P11为第5.7节中“用通用型QCPU替换高性能型QCPU的方法”的程序示例中所示的指针。）

如果不包含用于强制ON/OFF的输入输出编号，无需添加调用P10和P11的子程序。

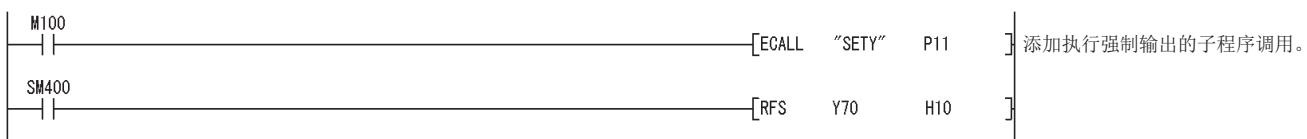
■当输入（X）的部分刷新通过RFS指令执行时

在RFS指令后添加执行强制输入的子程序调用。



■当输出（Y）的部分刷新通过RFS指令执行时

在RFS指令前添加执行强制输出的子程序调用。



限制事项

本章中记载的上述3种替换方法不适用于以下情形：

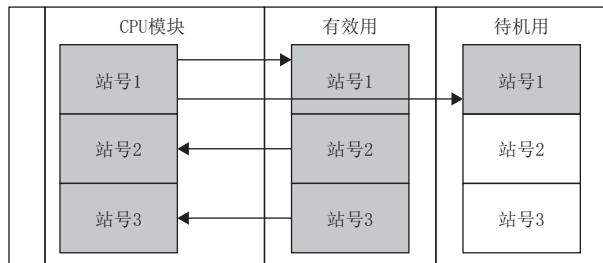
- 使用直接输入软元件（DX）/直接输出软元件（DY）参照或更改用于强制ON/OFF的输入和输出。
- 在中断程序中参照或更改用于强制ON/OFF的输入和输出。

5.8 MELSECNET/H简单双结构网络的替代方法

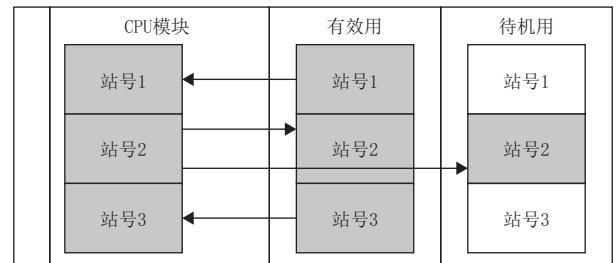
MELSECNET/H简单双结构网络的替换示例如下所示。

系统构成

站号1

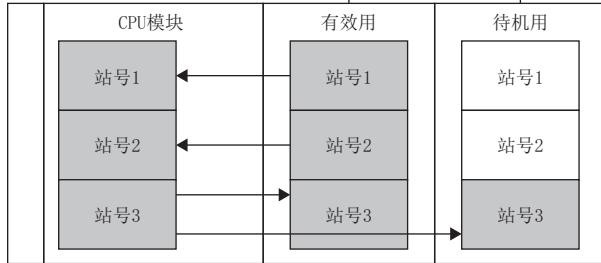


站号2



网络No. 1
网络No. 2

站号3



网络参数

按如下所示设置网络范围分配。

Station No.	Send Range for each Station			Send Range for each Station		
	LB			LW		
	Points	Start	End	Points	Start	End
1	16	0000	000F	16	0000	000F
2	16	0010	001F	16	0010	001F
3	16	0020	002F	16	0020	002F

替换前程序



- (1) 检查有效网络正常时的循环传送状态。
- (2) 检查待机网络正常时的循环传送状态。
- (3) 待机网络异常时切换为有效网络。(RUN後第1次扫描, 强制使用有效网络。)
- (4) 有效网络异常时切换为待机网络。

各网络的刷新设置软元件 (SM) 如下所示。

项目	模块1	模块2	模块3	模块4
有效/待机的区分 (OFF: 有效, ON: 待机)	SM255	SM260	SM265	SM270
CPU←网络模块的刷新 (OFF: 刷新, ON: 不刷新)	SM256	SM261	SM266	SM271
CPU→网络模块的刷新 (OFF: 刷新, ON: 不刷新)	SM257	SM262	SM267	SM272

关于详细内容, 应参照下述手册中的第7.7节。

Q系列MELSECNET/H网络系统技术参考手册 (主站、从站)

替换后程序

替代MELSECNET/H简单双结构网络的特殊继电器，通过程序检测各网络的异常，不使用网络的刷新参数，应以通过使用直接链接软元件（J□\B□、J□\W□）的指令进行的刷新替代。

此外，由于不支持循环数据的站单位块保证，因此在替换前的系统中执行循环数据的站单位块保证时，需要按每个链接数据配置互锁程序。关于互锁程序的详细内容，应参照下述手册中的第6.2.3项“互锁程序示例”。

Q系列MELSECNET/H网络系统技术参考手册（主站、从站）

■刷新参数

删除有关链接特殊继电器（SB）/链接特殊寄存器（SW）以外的刷新参数。

	Link Side				▲
	Dev. Name	Points	Start	End	
Transfer SB	SB	32	0000	001F	↔
Transfer SW	SW	32	0000	001F	↔
Random Cyclic	LB				↔
Random Cyclic	LW				↔
Transfer 1	▼				↔
Transfer 2	▼				↔

	PLC Side				▲
	Dev. Name	Points	Start	End	
	SB	32	0000	001F	↔
	SW	32	0000	001F	↔
	▼				▼
	▼				▼
	▼				▼
	▼				▼

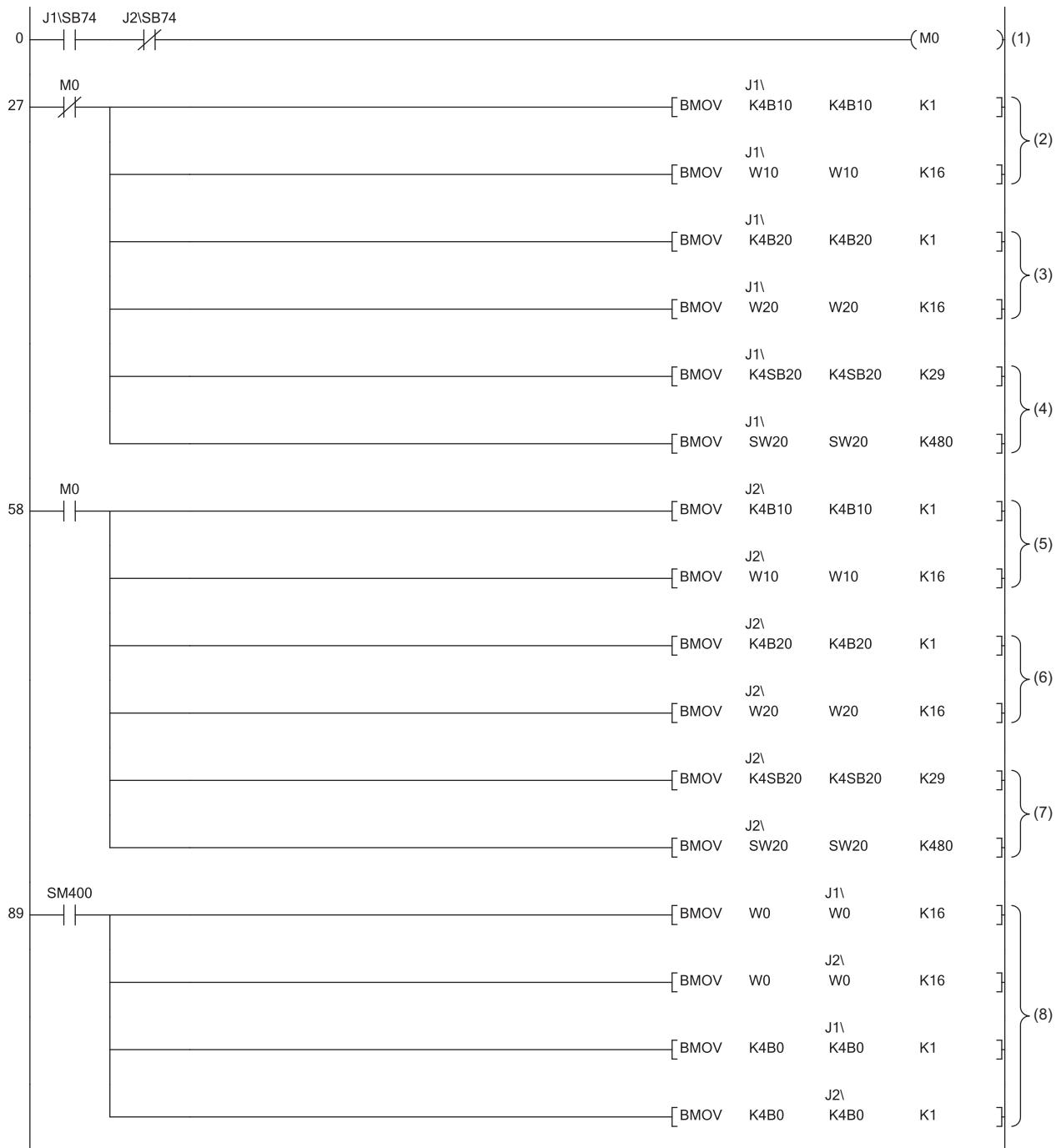
- 仅对SB0至SB1F、SW0至SW1F（CPU→网络的范围）进行刷新参数的设置。

- 删除SB/SW以外的刷新参数。

■程序

各站号的程序如下所示。

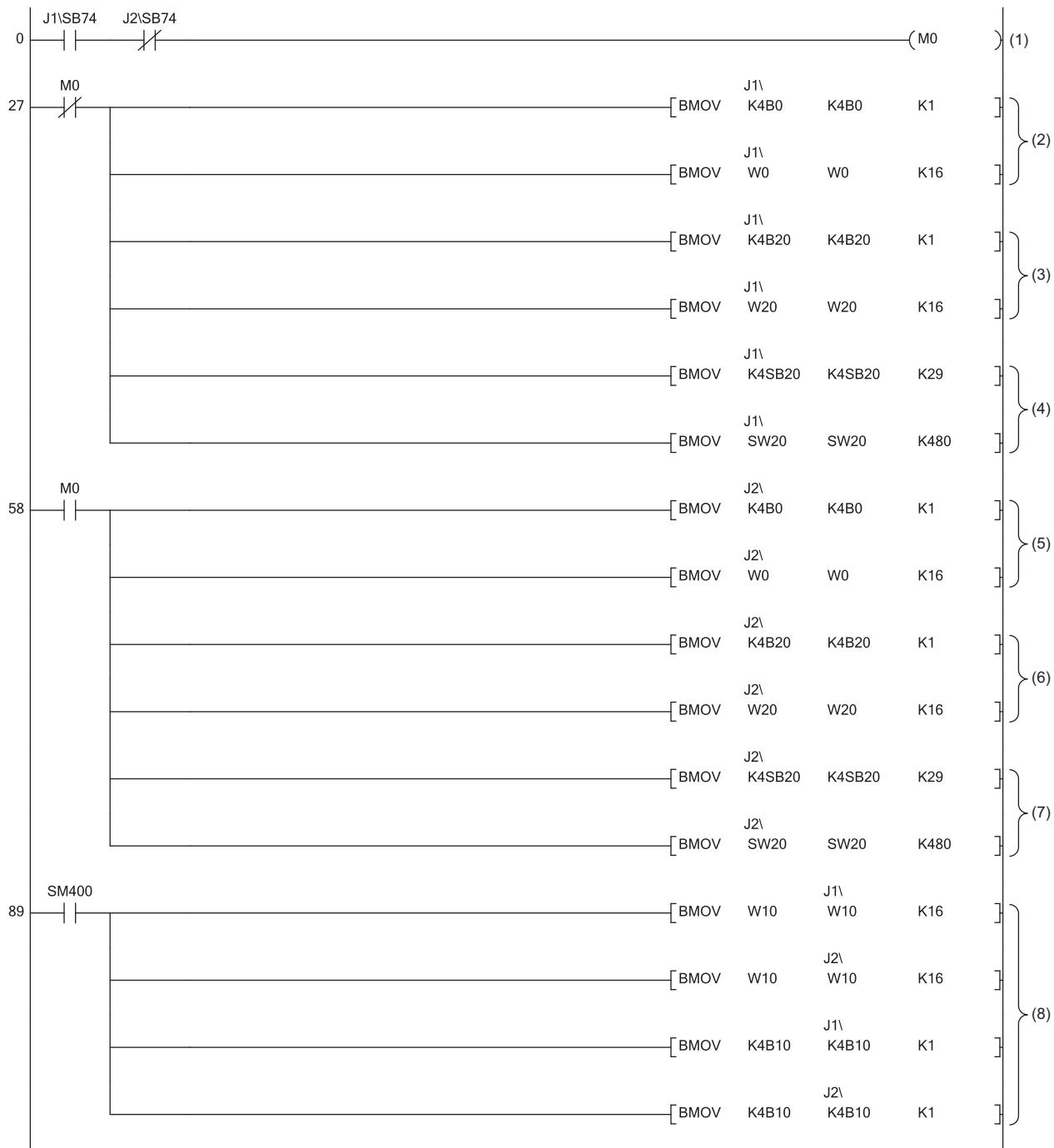
- 站号1的程序



- (1) 有效网络异常、待机网络正常时，将待机网络作为CPU←网络的刷新对象 (M0: ON)。除此以外，将有效网络作为CPU←网络的刷新对象 (M0: OFF)。
- (2) 从有效网络的站号2的发送区域获取至站号1的CPU模块的软元件存储器中。
- (3) 从有效网络的站号3的发送区域获取至站号1的CPU模块的软元件存储器中。
- (4) 从有效网络的SB20至SB1FF、SW20至SW1FF获取至站号1的SB20至SB1FF、SW20至SW1FF中。
- (5) 从待机网络的站号2的发送区域获取至站号1的CPU模块的软元件存储器中。
- (6) 从待机网络的站号3的发送区域获取至站号1的CPU模块的软元件存储器中。
- (7) 从待机网络的SB20至SB1FF、SW20至SW1FF获取至站号1的SB20至SB1FF、SW20至SW1FF中。
- (8) 将发送数据写入至控制网络/待机网络的站号1的发送区域。

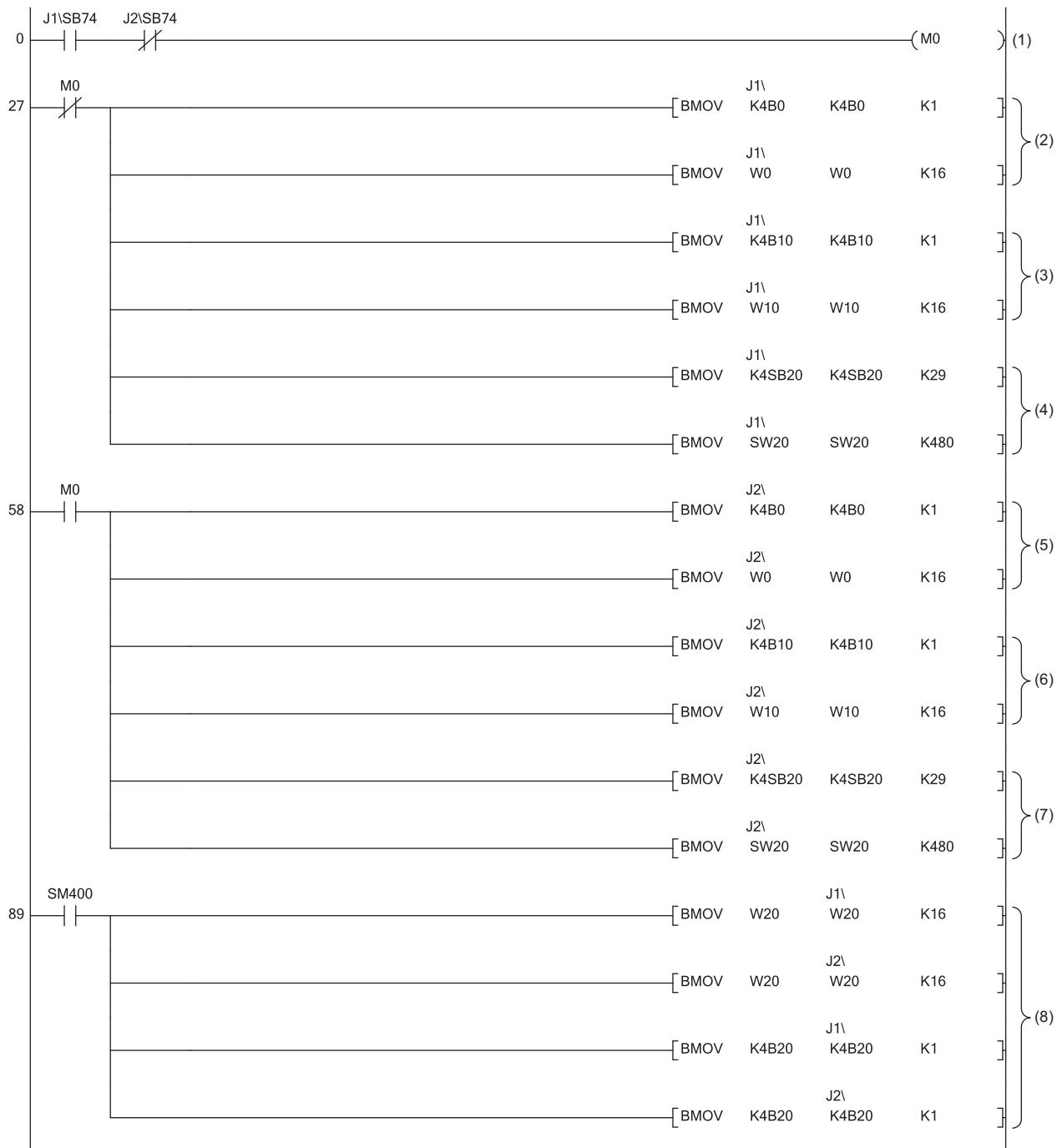
FA-CN-0001-B

• 站号2的程序



- (1) 有效网络异常、待机网络正常时，将待机网络作为CPU←网络的刷新对象 (M0: ON)。除此以外，将有效网络作为CPU←网络的刷新对象 (M0: OFF)。
- (2) 从有效网络的站号1的发送区域获取至站号2的CPU模块的软元件存储器中。
- (3) 从有效网络的站号3的发送区域获取至站号2的CPU模块的软元件存储器中。
- (4) 从有效网络的SB20至SB1FF、SW20至SW1FF获取至站号2的SB20至SB1FF、SW20至SW1FF中。
- (5) 从待机网络的站号1的发送区域获取至站号2的CPU模块的软元件存储器中。
- (6) 从待机网络的站号3的发送区域获取至站号2的CPU模块的软元件存储器中。
- (7) 从待机网络的SB20至SB1FF、SW20至SW1FF获取至站号2的SB20至SB1FF、SW20至SW1FF中。
- (8) 将发送数据写入至控制网络/待机网络的站号2的发送区域。

• 站号3的程序



(1) 有效网络异常、待机网络正常时，将待机网络作为CPU←网络的刷新对象 (M0: ON)。除此以外，将有效网络作为CPU←网络的刷新对象 (M0: OFF)。

(2) 从有效网络的站号1的发送区域获取至站号3的CPU模块的软元件存储器中。

(3) 从有效网络的站号2的发送区域获取至站号3的CPU模块的软元件存储器中。

(4) 从有效网络的SB20至SB1FF、SW20至SW1FF获取至站号3的SB20至SB1FF、SW20至SW1FF中。

(5) 从待机网络的站号1的发送区域获取至站号3的CPU模块的软元件存储器中。

(6) 从待机网络的站号2的发送区域获取至站号3的CPU模块的软元件存储器中。

(7) 从待机网络的SB20至SB1FF、SW20至SW1FF获取至站号3的SB20至SB1FF、SW20至SW1FF中。

(8) 将发送数据写入至控制网络/待机网络的站号3的发送区域。

6 特殊继电器和特殊寄存器

通用型QCPU不支持如下所示的特殊继电器和特殊寄存器。

☞ 60页 特殊继电器一览表、☞ 62页 特殊寄存器一览表

应使用下表所述的方法替换它们或删除相应的部分。

6.1 特殊继电器一览表

通用型QCPU不支持的特殊继电器和应对方法如下所示。

通用型QCPU不支持的特殊继电器和应对方法

编号	名称/说明		应对方法
SM80	CHK检测		通用型QCPU不支持CHK指令。 关于CHK指令的替代方法, 应参照下述内容。 ☞ 16页 程序替换示例
SM91	步转移监视定时器启动		通用型QCPU不支持步转移监视定时器功能。 关于该功能的替代方法, 应参照MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的附录3 “关于基本型QCPU、通用型QCPU、LCPU的限制和替代方法”。
SM92			
SM93			
SM94			
SM95			
SM96			
SM97			
SM98			
SM99			
SM250	最大的安装输入输出编号读取		无需SD250操作。通用型QCPU始终在SD250中存储最大的安装输入输出编号。 删除相应的部分。
SM255	MELSECNET/H 模块1信息	表示有效网络或待机网络	这些是用于简单双结构网络的特殊继电器。
SM256		在从链接模块至CPU的刷新中, 选择是否从链接模块读取数据。	通用型QCPU中无简单双结构网络的特殊继电器。有关特殊继电器的替代方法, 应参照下述内容。 ☞ 54页 MELSECNET/H简单双结构网络的替代方法
SM257		在CPU至链接模块的刷新中, 选择是否写入数据至链接模块。	
SM260	MELSECNET/H 模块2信息	表示有效网络或待机网络	
SM261		在从链接模块至CPU的刷新中, 选择是否从链接模块读取数据。	
SM262		在CPU至链接模块的刷新中, 选择是否写入数据至链接模块。	
SM265	MELSECNET/H 模块3信息	表示有效网络或待机网络	
SM266		在从链接模块至CPU的刷新中, 选择是否从链接模块读取数据。	
SM267		在CPU至链接模块的刷新中, 选择是否写入数据至链接模块。	
SM270	MELSECNET/H 模块4信息	表示有效网络或待机网络	
SM271		在从链接模块至CPU的刷新中, 选择是否从链接模块读取数据。	
SM272		在CPU至链接模块的刷新中, 选择是否写入数据至链接模块。	
SM280	CC-Link错误		用安装的CC-Link模块的输入输出信号 (Xn0、Xn1和XnF) 替换继电器。
SM330	低速执行型程序的动作模式		通用型QCPU不支持低速执行型程序。 删除相应的部分。
SM331	一般SFC程序执行状态		通用型QCPU仅支持一般SFC程序。删除使用SM331和SM332的互锁, 或用SM321替换它们。
SM332	程序执行管理SFC程序执行状态		
SM390	访问执行标志		根据各模块手册中所述的示例程序修改模块就绪信号 (Xn) 被用作互锁的程序。
SM404	低速执行型程序RUN后, 仅1次扫描为ON		通用型QCPU不支持低速执行型程序。 删除相应的部分或使用用于扫描执行型程序的特殊继电器 (SM402和SM403) 替换它们。
SM405	低速执行型程序RUN后, 仅1次扫描为OFF		

编号	名称/说明	应对方法
SM430	用户时间时钟编号5（用于低速执行型程序）	通用型QCPU不支持低速执行型程序。删除相应的部分或使用用于扫描执行型程序的特殊继电器（SM420至SM424）替换它们。
SM431	用户时间时钟编号6（用于低速执行型程序）	
SM432	用户时间时钟编号7（用于低速执行型程序）	
SM433	用户时间时钟编号8（用于低速执行型程序）	
SM434	用户时间时钟编号9（用于低速执行型程序）	
SM510	低速执行型程序执行标志	通用型QCPU不支持低速执行型程序。 删除相应的部分。
SM551	模块服务间隔读取	通用型QCPU不支持服务间隔测量功能。 删除相应的部分。
SM672	存储卡文件寄存器访问范围标志	当访问存储卡中超出文件寄存器的范围时，通用型QCPU检测“运算错误”（错误代码：4101）。 无需使用该特殊继电器进行检测错误的编程。 删除相应的部分。
SM710	CHK指令优先级标志	通用型QCPU不支持CHK指令。 关于CHK指令的替代方法，应参照下述内容。 ☞ 16页 程序替换示例
SM734	XCALL指令执行条件指定	通用型QCPU也在执行条件的上升沿执行XCALL指令。无需通过该特殊继电器设置执行条件上升沿时的动作。 删除相应的部分。
SM735 ^{*1}	SFC注释读取指令执行中标志	通用型QCPU不支持以下指令： • SFC步注释读取指令（S(P).SFCSOMR） • SFC转移条件注释读取指令（S(P).SFCTCOMR） 删除相应的部分。
SM1780 ^{*2}	电源OFF检测标志	通用型QCPU不在SM1780至SM1783中存储冗余电源系统信息。 删除相应的部分。
SM1781 ^{*2}	电源故障检测标志	
SM1782 ^{*2}	电源1的瞬时停电检测标志	
SM1783 ^{*2}	电源2的瞬时停电检测标志	

*1 如果通用型QCPU的序列号（前5位）为“12052”或以后版本，可使用该特殊继电器。

*2 如果通用型QCPU的序列号（前5位）为“10042”或以后版本，可使用该特殊继电器。

6.2 特殊寄存器一览表

通用型QCPU不支持的特殊寄存器和应对方法如下所示。

通用型QCPU不支持的特殊寄存器和应对方法

编号	名称/说明	应对方法
SD80	CHK编号	通用型QCPU不支持CHK指令。 关于CHK指令的替代方法, 应参照下述内容。 ☞ 16页 程序替换示例
SD90	步转移监视定时器设置值	通用型QCPU不支持步转移监视定时器功能。 关于该功能的替代方法, 应参照MELSEC-Q/L/QnA 编程手册 (SFC控制指令篇) 中的附录3 “关于基本型QCPU、通用型QCPU、LCPU的限制和替代方法”。
SD91		
SD92		
SD93		
SD94		
SD95		
SD96		
SD97		
SD98		
SD99		
SD280	CC-Link错误	用安装的CC-Link模块的输入输出信号 (Xn0、Xn1和XnF) 替换这些寄存器。
SD281		
SD315	确保用于通信处理的时间	通用型QCPU中可以在PLC参数对话框PLC系统设置选项卡上进行服务处理设置。 为服务处理设置参数选择“指定服务处理时间”并设置服务处理时间。也可选择其他设置方法。
SD430	低速扫描计数器	通用型QCPU不支持低速执行型程序。 删除相应的部分或使用用于扫描执行型程序的特殊寄存器 (SD420) 替换。
SD510	低速执行型程序编号	通用型QCPU不支持低速执行型程序。 删除相应的部分或使用用于扫描执行型程序的特殊寄存器 (SD500) 替换。
SD528	低速执行型程序的当前扫描时间	通用型QCPU不支持低速执行型程序。 删除相应的部分或使用用于扫描执行型程序的特殊寄存器 (SD520和SD521) 替换它们。
SD529		
SD532	低速执行型程序的最小扫描时间	通用型QCPU不支持低速执行型程序。 删除相应的部分或使用用于扫描执行型程序的特殊寄存器 (SD524至SD527) 替换它们。
SD533		
SD534	低速执行型程序的最大扫描时间	
SD535		
SD544	低速执行型程序的累积执行时间	通用型QCPU不支持低速执行型程序。 删除相应的部分。
SD545		
SD546	低速执行型程序的执行时间	通用型QCPU不支持低速执行型程序。 删除相应的部分。
SD547		
SD550	服务间隔测量模块	通用型QCPU不支持服务间隔测量功能。 删除相应的部分。
SD551	服务间隔时间	
SD552		
SD720	PLOADP指令的程序编号指定	通用型QCPU不支持PLOADP指令。 删除相应的部分。
SD1780 ^{*1}	电源OFF检测状态	通用型QCPU不在SD1780至SD1783中存储冗余电源系统信息。 删除相应的部分。
SD1781 ^{*1}	电源故障检测状态	(SD1780至SD1783常闭。)
SD1782 ^{*1}	电源1的瞬时停电检测计数器	
SD1783 ^{*1}	电源2的瞬时停电检测计数器	

*1 如果通用型QCPU的序列号 (前5位) 为“10042”或以后版本, 可使用该特殊寄存器。

修订记录

副编号	修订年月	修订内容
—	2016年12月	第一版
A	2017年8月	修改了第2章、第3章、第6.1节的内容，并添加了第5.8节。
B	2019年3月	已支持e-Manual。