OMRON

机器自动化控制器

NJ/NX系列

CPU单元

用户手册
软件篇

NX701-17□□
NX701-16□□
NJ501-□5□□
NJ501-□4□□
NJ501-□3□□
NJ301-12□□
NJ301-11□□
NJ101-10□□
NJ101-90□□

CPU单元

SYSMAC always in control

SBCA-CN5-359P

声明

- 禁止誊写、复制或转载本手册的部分或全部内容。
- 本手册的内容可能因改良需要而有所变更，恕不另行通知，敬请谅解。
- 本手册的内容经严格审核，若发现不明之处或错误，请与本公司联系。
  联系时，请告知卷末记载的手册编号（Man.No.）。

商标

- Sysmac是欧姆龙株式会社FA机器产品在日本及其他国家的商标或注册商标。
- Microsoft及Windows、 Windows Vista、 Excel、 Visual Basic是美国Microsoft Corporation在美国、日本及其他国家的注册商标或商标。
- EtherCAT® 是德国倍福自动化有限公司提供的注册商标，是已获得专利的技术。
- ODVA、CIP、CompoNet、DeviceNet、EtherNet/IP是ODVA的商标。
- SD、SDHC标识是SD-3C, LLC的商标。

此外，记载的公司名称和产品名称为各公司的注册商标或商标。

著作权

Screen shot已取得Microsoft Corporation的使用许可。

前言

感谢您购买 NJ/NX 系列 CPU 单元。
本手册记载了使用 NJ/NX 系列 CPU 单元所必需的信息。请在使用前仔细阅读本手册，在充分理解功能和性能的基础上，灵活用于系统构建。
此外，请妥善保管本手册，阅读之后放于随时可取阅之处。

阅读对象

本手册提供给下列阅读对象。
具有电工专业知识的人员 （合格的电气工程师或具有同等知识的人员）：
- 引进 FA 设备的人员；
- 设计 FA 系统的人员；
- 安装和连接 FA 设备的人员；
- FA 现场管理人员。

此外，编程语言的对象为掌握国际标准规格 IEC 61131-3 或日本国内标准规格格 JIS B 3503 之规定内容的人员。

对象产品

本手册的对象为以下产品：
- NX 系列 CPU 单元
  - NX701-17 □□
  - NX701-16 □□

- NJ 系列 CPU 单元
  - NJ501- □ 5 □□
  - NJ501- □ 4 □□
  - NJ501- □ 3 □□
  - NJ301-12 □□
  - NJ301-11 □□
  - NJ101-10 □□
  - NJ101-90 □□

各产品的部分规格或限制事项可能在其他手册中有记载。📖请确认 「*分册构成* (P.2)」及 📖「*相关手册* (P.42)」。

分册构成

本产品的手册如下表所示分册。请根据不同目的选读，灵活运用本产品。

本产品的操作主要使用自动化软件 Sysmac Studio。关于 Sysmac Studio，请参考 📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

使用目的	手册											
	基本信息											
	NX 系列 CPU 单元 用户手册 硬件篇	NJ 系列 CPU 单元 用户手册 硬件篇	NJ/NX 系列 CPU 单元 用户手册 软件篇	NJ/NX 系列 指令基准手册 基本篇	NJ/NX 系列 CPU 单元 用户手册 运动控制篇	NJ/NX 系列 指令基准手册 运动篇	NJ/NX 系列 CPU 单元 内置 EtherCAT 端口 用户手册	NJ/NX 系列 CPU 单元 内置 EtherNet/IP 端口用户手册	NJ 系列 数据库连接 CPU 单元 用户手册	NJ 系列 配备 SECS/GEM 的 CPU 单元 用户手册	NJ 系列 NJ Robotics CPU 单元 用户手册	NJ/NX 系列 故障排除手册
了解 NX 系列的概要	●											
了解 NJ 系列的概要		●										
进行安装、设置和硬件设定	●	●										
进行运动控制时					●							
使用 EtherCAT 时							●					
使用 EtherNet/IP 时								●				
使用数据库连接服务时									●			
使用 GEM 服务时										●		
进行机器人控制时											●	
进行软件设定			●									
进行运动控制时					●							
使用 EtherCAT 时							●					
使用 EtherNet/IP 时								●				
使用数据库连接服务时									●			
使用 GEM 服务时										●		
进行机器人控制时											●	
创建用户程序			●	●								
进行运动控制时					●	●						
使用 EtherCAT 时							●					
使用 EtherNet/IP 时								●				
使用数据库连接服务时									●			
使用 GEM 服务时										●		
进行机器人控制时											●	
组合异常处理时												●
进行动作确认、调试时			●									
进行运动控制时					●							
使用 EtherCAT 时							●					
使用 EtherNet/IP 时								●				
使用数据库连接服务时									●			
使用 GEM 服务时										●		
进行机器人控制时											●	
了解异常管理功能或故障发生时的处理方法[*1]	△	△	△		△		△	△	△	△		●

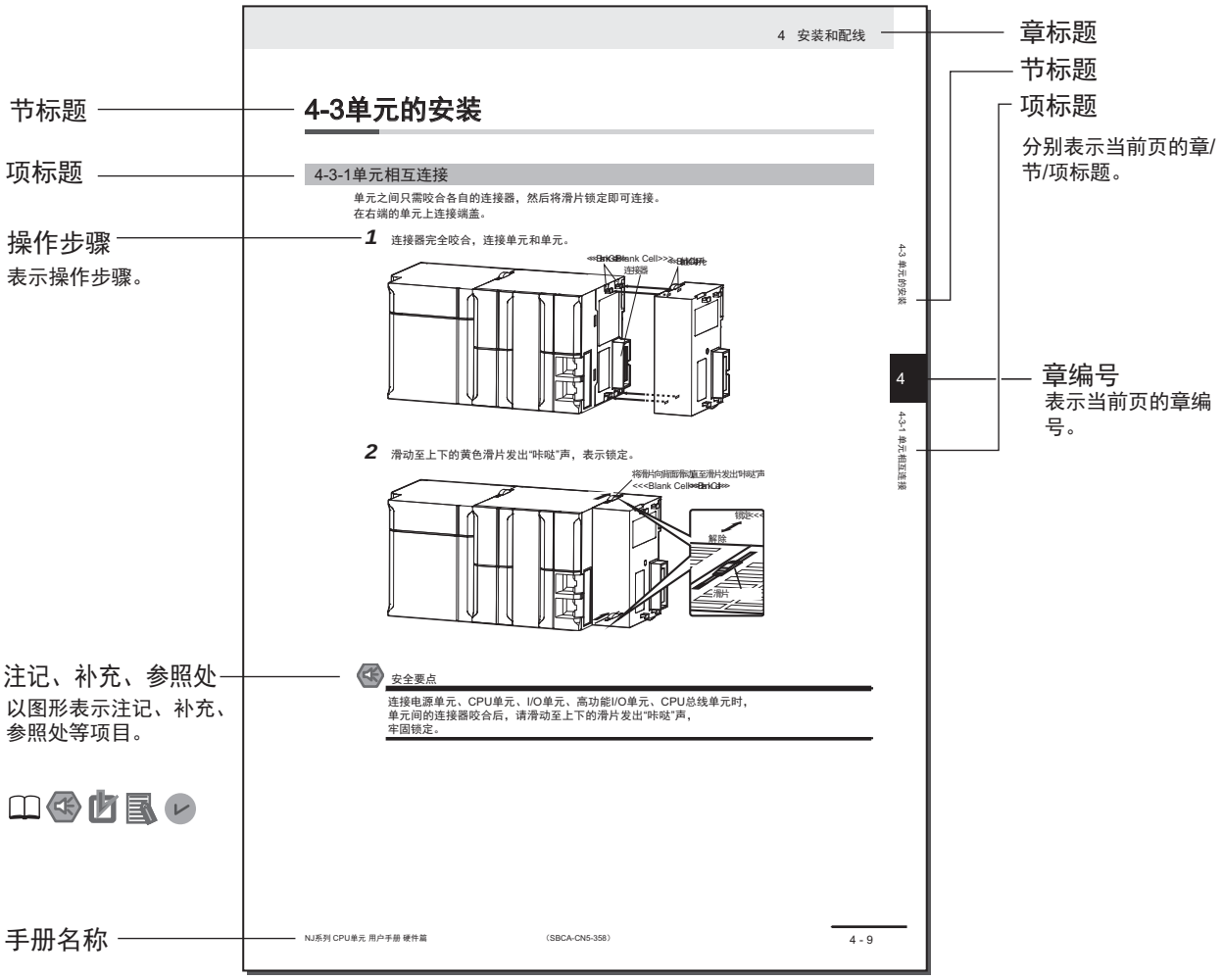
使用目的	手册											
	基本信息											
	NX 系列 CPU 单元 用户手册 硬件篇	NJ 系列 CPU 单元 用户手册 硬件篇	NJ/NX 系列 CPU 单元 用户手册 软件篇	NJ/NX 系列 指令基准手册 基本篇	NJ/NX 系列 CPU 单元 用户手册 运动控制篇	NJ/NX 系列 指令基准手册 运动篇	NJ/NX 系列 CPU 单元 内置 EtherCAT 端口 用户手册	NJ/NX 系列 CPU 单元 内置 EtherNet/IP 端口用户手册	NJ 系列 数据库连接 CPU 单元 用户手册	NJ 系列 配备 SECS/GEM 的 CPU 单元 用户手册	NJ 系列 NJ Robotics CPU 单元 用户手册	NJ/NX 系列 故障排除手册
了解维护作业	●	●										
进行运动控制时					●							
使用 EtherCAT 时							●					
使用 EtherNet/IP 时								●				

*1. 关于异常管理的说明和异常项目的概要，请参考📖《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。关于异常的详情，请根据异常的内容参考带△标志的手册。

手册说明

页面构成

本手册各页的构成如下。



本页为说明用样本。与实际内容不同。

图标

本手册中使用的图标含义如下。

安全要点
表示为了安全使用，应该实施或避免的行为。

使用注意事项
表示为了防止产品出现动作不良、误动作或严重影响其性能、功能，应该实施或避免的行为。

参考
需要时阅读的项目。
了解后有助于使用的信息以及使用时可参考的内容。

表示记载有详细信息、相关信息的位置。

版本相关信息
介绍不同版本的 CPU 单元或 Sysmac Studio 的性能和功能区别。

表述注意事项

本手册中，从 Sysmac Studio 向控制器实体传送数据记载为 “ 下载 ”，从控制器实体向 Sysmac Studio 传送数据记载为 “ 上传 ”。
在 Sysmac Studio 中，无论是 “ 下载 ” 还是 “ 上传 ”，都使用 “ 同步 ” 功能。 Sysmac Studio 的 “ 同步 ” 功能可以自动核对电脑上的数据和控制器实体上的数据，由用户选择传送方向。

目录结构

1 NJ/NX系列控制器的概要

2 CPU单元的动作

3 I/O单元和从站/单元构成

4 控制器设定

5 设计任务

6 编程

7 动作确认和运行

8 CPU内置功能

9 备份功能

10 通信设定

11 实际使用步骤示例

12 异常的确认和处理

A 附录

I 索引

目录

前言 1
阅读对象 1
对象产品 1

分册构成 2

手册说明 4
页面构成 4
图标 5
表述注意事项 5

目录结构 7

承诺事项 16

安全注意事项 18
安全信息的标识及其含义 18
图标说明 18
警告 19
注意 21

安全要点 23

使用注意事项 32

法规和标准 37
日本国外的使用 37
符合 EU 指令 37
KC 标准的适用 38
船舶标准的适用 38
软件许可和著作权 38

版本 39
版本的确认方法 39
CPU 单元的单元版本和 Sysmac Studio 的版本 41

相关手册 42

术语说明 45

手册修订记录 49

第 1 章　NJ/NX 系列控制器的概要

1-1　NJ/NX 系列控制器定义 1-2
1-1-1　特点 1-2
1-1-2　系统构成概要 1-5

1-2　规格概要 1-9

1-3　NJ/NX 系列的整体使用步骤 1-12
1-3-1　步骤概要 1-12
1-3-2　步骤详情 1-13

第 2 章　CPU 单元的动作

2-1　CPU 单元的动作概要 2-2
2-1-1　CPU 单元的原理概要 2-2
2-1-2　CPU 单元在各种状态下的动作概要 2-3
2-2　软件的原理 2-4
2-2-1　软件构成 2-4
2-2-2　软件的动作 2-5
2-3　与 I/O 的访问原理 2-10
2-3-1　变量的种类 2-10
2-3-2　变量和 I/O 的访问 2-13
2-4　顺序控制和运动控制的原理 2-19
2-4-1　控制的概要 2-19
2-4-2　顺序控制的原理 2-21
2-4-3　运动控制的原理 2-22
2-4-4　顺序控制和运动控制的同步 2-23
2-5　CPU 单元内数据的概要 2-25
2-6　CPU 单元在各种状态下的动作 2-26
2-6-1　CPU 单元的状态 2-26
2-6-2　各种状态下的动作 2-28
2-6-3　动作模式 2-29

第 3 章　I/O 单元和从站 / 单元构成

3-1　从站 / 单元构成的步骤 3-2
3-2　从站 / 单元构成的创建和核对 3-4
3-2-1　EtherCAT 从站构成的创建 3-4
3-2-2　单元构成的创建 3-6
3-2-3　单元构成的核对 3-6
3-3　I/O 端口和设备变量 3-8
3-3-1　I/O 端口 3-8
3-3-2　I/O 端口名称 3-9
3-3-3　设备变量 3-10
3-4　向单元分配变量 3-13
3-4-1　向单元分配变量的方法 3-13
3-4-2　单元中分配的变量的使用方法 3-15
3-5　“ 轴 ” 的创建和伺服驱动器 / 编码器输入从站的分配 3-17
3-5-1　概要 3-17
3-5-2　轴变量 / 轴组变量 3-17
3-5-3　“ 轴 ” 及轴变量的创建和使用方法 3-19

第 4 章　控制器设定

4-1　控制器设定的概要 4-2
4-2　PLC 功能模块的初始设定 4-4
4-2-1　概要 4-4
4-2-2　控制器设定 4-4
4-2-3　任务设置 4-6
4-2-4　单元构成和单元设定 4-12
4-3　高功能单元的初始设定 4-13
4-4　运动控制功能模块的初始设定 4-15
4-4-1　概要 4-15
4-4-2　设定方法 4-16
4-5　EtherCAT 主机功能模块的初始设定 4-17
4-6　EtherNet/IP 功能模块的初始设定 4-18

第 5 章　设计任务

5-1　任务设计步骤的概要 5-3

5-2　任务的概要 5-5
5-2-1　任务定义 5-5
5-2-2　任务相关的指令 5-7
5-2-3　任务相关的系统定义变量 5-7

5-3　NX 系列的任务规格和基本动作 5-9
5-3-1　NX 系列的任务规格 5-9
5-3-2　NX 系列的任务分配说明 5-10
5-3-3　NX 系列的任务基本动作 5-10
5-3-4　NX 系列事件任务的执行条件 5-19
5-3-5　NX 系列事件任务的执行时序 5-23
5-3-6　事件任务完成前，执行条件再次成立时的动作 5-27

5-4　NJ 系列的任务规格和基本动作 5-28
5-4-1　NJ 系列的任务规格 5-28
5-4-2　NJ 系列的任务分配说明 5-29
5-4-3　NJ 系列的任务基本动作 5-29
5-4-4　NJ 系列事件任务的执行条件 5-37
5-4-5　NJ 系列事件任务的执行时序 5-41
5-4-6　事件任务完成前，执行条件再次成立时的动作 5-46

5-5　标签数据链接服务和系统服务 5-47
5-5-1　标签数据链接服务和系统服务的执行优先度和执行顺序 5-48
5-5-2　标签数据链接服务的处理内容和执行时序 5-50
5-5-3　系统服务的处理内容和执行时序 5-54

5-6　任务相关的分配和设定 5-57
5-6-1　分配到 I/O 刷新的任务中 5-57
5-6-2　将程序分配到任务中 5-65
5-6-3　任务的设定项目 5-66

5-7　确保变量的同时性 5-69
5-7-1　确保任务间变量值同时性的方法 5-69
5-7-2　从控制器外部进行的变量访问 5-75

5-8　任务相关的异常 5-79

5-9　任务执行状态、任务执行时间的监视 5-82

5-10　任务设计方法及输入输出响应时间 5-87
5-10-1　任务执行时间的确认 5-87
5-10-2　任务设计示例 5-90
5-10-3　系统输入输出响应时间 5-91

第 6 章　编程

6-1　编程步骤概要 6-3

6-2　POU （程序构成单位） 6-5
6-2-1　POU 定义 6-5
6-2-2　三个类型 POU 的概要 6-6
6-2-3　程序、功能块、函数的区别 6-7
6-2-4　程序的详情 6-8
6-2-5　功能块 （FB）的详情 6-9
6-2-6　函数 （FUN）的详情 6-18
6-2-7　功能块 （FB）、函数 （FUN）共通动作 6-24
6-2-8　POU 的限制事项 6-26

6-3　变量 6-28
6-3-1　变量定义 6-28
6-3-2　变量的种类 6-28
6-3-3　POU 对应的用户定义变量的种类 6-29
6-3-4　变量的属性 6-30
6-3-5　数据类型 6-31
6-3-6　派生数据型 6-41
6-3-7　数据类型的排列 / 范围指定 6-51

6-3-8 变量的各属性说明 6-57
6-3-9 与状态变化对应的变量值的变化 6-64
6-3-10 功能块 (FB) 的实例 6-76
6-3-11 变量值的监视 6-76
6-3-12 变量名称等程序相关名称的限制 6-77

6-4 常数 （文本） 6-79
6-4-1 常数定义 6-79
6-4-2 各数据类型的记述形式 6-79

6-5 编程语言 6-84
6-5-1 编程语言定义 6-84
6-5-2 梯形图语言 （LD） 6-84
6-5-3 ST 语言 6-90

6-6 指令 6-128
6-6-1 指令定义 6-128
6-6-2 指令的基本知识 6-128
6-6-3 指令错误 6-131

6-7 名称空间 6-137
6-7-1 名称空间定义 6-137
6-7-2 名称空间的规格 6-138
6-7-3 名称空间的使用步骤 6-141

6-8 库 6-142
6-8-1 库的概要 6-142
6-8-2 库的规格 6-143
6-8-3 库部件的规格 6-144
6-8-4 库的使用步骤 6-145

6-9 创建用户程序时的注意事项 6-146
6-9-1 输入变量 / 输出变量 / 输入输出变量为排列指定的变量时 6-146
6-9-2 输入变量 / 输出变量 / 输入输出变量为结构体型变量时 6-146
6-9-3 主控 6-147

第 7 章 动作确认和运行

7-1 动作确认和运行的步骤概要 7-2

7-2 离线调试 7-3
7-2-1 模拟功能的特点 7-3
7-2-2 模拟功能的执行 7-3
7-2-3 模拟功能的设定 7-6

7-3 上机动作确认和运行 7-8
7-3-1 从上机动作确认到运行开始之间的步骤 7-8
7-3-2 项目的传送 7-9
7-3-3 I/O 的配线确认 7-9
7-3-4 MC 调试 7-9
7-3-5 用户程序的动作确认 7-10
7-3-6 正式运行的开始 7-10

第 8 章 CPU 内置功能

8-1 数据管理、时钟、运行相关功能 8-3
8-1-1 存储器全部清除 8-3
8-1-2 时钟功能 8-3
8-1-3 运行中输出功能 8-6

8-2 CJ 单元相关管理功能 8-7
8-2-1 基本 I/O 单元相关 8-7
8-2-2 高功能单元相关 8-8

8-3 SD 存储卡功能 8-10
8-3-1 SD 存储卡功能一览 8-10
8-3-2 支持的 SD 存储卡及文件夹 / 文件规格 8-11
8-3-3 SD 存储卡操作指令 8-13
8-3-4 FTP 客户端通信指令 8-13

8-3-5 FTP 服务器功能 8-14
8-3-6 通过 Sysmac Studio 进行的文件操作 8-14
8-3-7 SD 存储卡的寿命检测功能 8-14
8-3-8 SD 存储卡相关的系统定义变量一览 8-15
8-3-9 SD 存储卡的自检功能 8-16
8-3-10 SD 存储卡中文件访问的排他控制 8-18

8-4 安全功能 8-19
8-4-1 用户程序执行用 ID 认证功能 8-20
8-4-2 无用户程序复原信息传送功能 8-23
8-4-3 项目文件整体的保护功能 8-24
8-4-4 数据保护功能 8-25
8-4-5 操作权限的认证功能 8-27
8-4-6 CPU 单元的写入保护功能 8-29
8-4-7 CPU 单元名称功能及序列 ID 功能 8-30

8-5 调试功能 8-33
8-5-1 强制值刷新 8-33
8-5-2 当前值变更 8-37
8-5-3 在线编辑 8-39
8-5-4 数据追踪 8-41
8-5-5 微分监控 8-46

8-6 事件日志功能 8-51
8-6-1 概要 8-51
8-6-2 事件日志功能的详情 8-52
8-6-3 控制器事件 （控制器异常 / 信息） 8-57
8-6-4 用户事件 （用户异常 / 信息） 8-58

8-7 事件重要程度变更功能 8-64
8-7-1 事件重要程度变更的用途 8-64
8-7-2 可变更事件重要程度的事件 8-64
8-7-3 事件重要程度变更的操作方法 8-65

第 9 章 备份功能

9-1 备份功能的概要 9-3
9-1-1 备份功能的用途 9-3
9-1-2 备份功能的操作步骤示例 9-4
9-1-3 备份功能的对象数据 9-6
9-1-4 备份功能的种类 9-7
9-1-5 备份功能的种类和数据表的关系 9-9
9-1-6 备份功能的对象范围 9-10

9-2 SD 存储卡的备份功能 9-12
9-2-1 备份 （从控制器到 SD 存储卡） 9-13
9-2-2 恢复 （从 SD 存储卡到控制器） 9-18
9-2-3 核对 （控制器和 SD 存储卡之间） 9-20

9-3 SD 存储卡备份禁止功能 9-25

9-4 SD 存储卡自动传送功能 9-26

9-5 SD 存储卡程序传送功能 9-28

9-6 Sysmac Studio 控制器备份功能 9-34
9-6-1 备份 （从控制器到计算机） 9-35
9-6-2 恢复 （从计算机到控制器） 9-36
9-6-3 核对 （控制器和计算机之间） 9-37

9-7 Sysmac Studio 备份文件的导入 / 导出功能 9-38

9-8 Sysmac Studio 变量 / 存储器的备份功能 9-39
9-8-1 Sysmac Studio 变量 / 存储器备份功能的对象数据 9-39
9-8-2 Sysmac Studio 变量 / 存储器备份功能的操作方法 9-39
9-8-3 CPU 单元型号间的兼容性 9-40

9-9 连接 EtherCAT 从站时的备份功能 9-42
9-9-1 EtherCAT 从站的备份对象数据 9-42
9-9-2 控制器在各种状态下的备份功能执行可否 9-43
9-9-3 EtherCAT 从站的恢复执行条件 9-44
9-9-4 备份功能的对象 EtherCAT 从站 9-45

9-10 连接 EtherCAT 从站终端时的备份功能 9-48
9-10-1 EtherCAT 从站终端的备份对象数据 9-48
9-10-2 EtherCAT 从站终端在各状态下的备份功能执行可否 9-49
9-10-3 EtherCAT 从站终端的恢复执行条件 9-49
9-11 连接 CJ 单元时的备份功能 9-50
9-11-1 CJ 单元的备份对象数据 9-50
9-11-2 控制器在各种状态下的备份功能执行可否 9-50
9-11-3 CJ 单元的恢复执行条件 9-51
9-12 备份相关文件 9-52
9-12-1 备份相关文件的种类 9-52
9-12-2 备份文件的规格 9-53
9-12-3 恢复指令文件的规格 9-54
9-12-4 自动传送指令文件的规格 9-56
9-12-5 控制器核对结果文件的规格 9-58
9-12-6 EtherCAT 核对结果文件的规格 9-59
9-12-7 EtherCAT 从站终端核对结果文件的规格 9-60
9-12-8 CJ 单元核对结果文件的规格 9-61
9-13 备份功能的文件间兼容性 9-63
9-13-1 备份功能间的兼容性 9-63
9-13-2 CPU 单元型号间的兼容性 9-64
9-13-3 CPU 单元的单元版本兼容性 9-65
9-14 无法与备份功能同时执行的功能 9-66

第 10 章 通信设定

10-1 通信系统的概要 10-2
10-1-1 概要 10-3
10-2 Sysmac Studio 的连接 10-5
10-2-1 可在线连接的构成 10-5
10-2-2 无法在线连接的构成 10-6
10-3 与其他控制器或从站连接 10-8
10-3-1 控制器与控制器之间的连接构成 10-8
10-3-2 控制器与从站的连接构成 10-11
10-4 与显示器和串行通信设备连接 10-12
10-4-1 与显示器的连接 10-12
10-4-2 与串行通信设备连接 10-13

第 11 章 实际使用步骤示例

11-1 实际使用示例 11-2
11-1-1 系统构成 11-2
11-1-2 动作 11-2
11-2 操作顺序示例的概要 11-3
11-2-1 配线和设定 11-3
11-2-2 软件设计 11-3
11-2-3 Sysmac Studio 上的软件设定 11-4
11-2-4 Sysmac Studio 上的编程 11-8
11-2-5 Sysmac Studio 上的模拟 11-9
11-2-6 上机确认动作和运行 11-9

第 12 章 异常的确认和处理

12-1 异常状态的动作 12-2
12-1-1 NJ/NX 系列的状态概要 12-2
12-1-2 CPU 单元无法动作的异常 12-4
12-1-3 CPU 单元可动作的异常 12-6
12-2 故障排除 12-13
12-2-1 确认 CPU 单元是否正在动作 12-13

12-2-2 CPU 单元可动作的异常时的处理流程 12-15
12-2-3 异常一览 12-16
12-2-4 异常内容 12-30
12-2-5 CPU 单元以外的异常及其处理 12-75

附录

A-1 规格 A-3
A-1-1 一般规格 A-3
A-1-2 性能规格 A-4
A-1-3 功能规格 A-7

A-2 NX 系列中任务实际处理时间的计算标准 A-13
A-2-1 任务实际处理时间平均值的计算方法 A-14
A-2-2 任务实际处理时间平均值的计算和任务周期的设定示例 A-21

A-3 NJ 系列中任务实际处理时间的计算标准 A-24
A-3-1 任务实际处理时间平均值的计算方法 A-25
A-3-2 任务实际处理时间平均值的计算和任务周期的设定示例 A-35

A-4 系统定义变量一览 A-38
A-4-1 NJ/NX 系列控制器的系统整体 （类别名称：无） A-38
A-4-2 PLC 功能模块 （类别名称：_PLC） A-46
A-4-3 PLC 功能模块 （类别名称：_CJB） A-47
A-4-4 运动控制功能模块 （类别名称：_MC） A-49
A-4-5 EtherCAT 主机功能模块 （类别名称：_EC） A-52
A-4-6 EtherNet/IP 功能模块 （类别名称：_EIP） A-56
A-4-7 异常状态中各个位的含义 A-78

A-5 系统定义变量的个别规格 A-79
A-5-1 NJ/NX 系列控制器的系统整体 （类别名称：无） A-80
A-5-2 PLC 功能模块 （类别名称：_PLC） A-93
A-5-3 PLC 功能模块 （类别名称：_CJB） A-95
A-5-4 运动控制功能模块 （类别名称：_MC） A-99
A-5-5 EtherCAT 主机功能模块 （类别名称：_EC） A-102
A-5-6 EtherNet/IP 功能模块 （类别名称：_EIP） A-110

A-6 CPU 单元内数据的属性 A-132

A-7 CJ 单元用存储器的详情 A-135
A-7-1 CIO 区域 A-135
A-7-2 内部辅助继电器 A-138
A-7-3 保持继电器 A-138
A-7-4 数据存储器 A-138
A-7-5 扩展数据存储器 A-139

A-8 变量的存储器确保方法 A-140
A-8-1 变量的存储器确保规则 A-140
A-8-2 应注意的案例 A-149

A-9 CX-Designer 的变量表登录方法 A-153

A-10 EtherCAT 从站或轴的有效 / 无效切换 A-156
A-10-1 使用 EtherCAT 从站和轴时的项目设定 A-156
A-10-2 执行 EtherCAT 从站或轴的有效 / 无效切换的指令 A-156
A-10-3 表示 EtherCAT 从站或轴状态的系统定义变量 A-157
A-10-4 程序的执行 / 停止指令 A-158
A-10-5 确认程序的执行 / 停止 A-158
A-10-6 Sysmac Studio 上的设定 A-159
A-10-7 EtherCAT 从站或轴的有效 / 无效切换应用示例 A-160

A-11 用户程序的大小相关的限制 A-163
A-11-1 用户程序的各部件限制 A-163
A-11-2 用户程序的各部件数量 A-167

A-12 Ver.1.02 以下的 CPU 单元更换方法 A-169
A-12-1 CPU 单元内数据的上传 A-169
A-12-2 连接新的 CPU 单元 A-172

A-12-3 将数据下载到 CPU 单元 A-173

A-13 **NX 系列的版本相关信息 A-177**

A-13-1 单元版本和 Sysmac Studio 版本之间的关系 A-177

A-13-2 通过更新单元版本追加和变更支持功能 A-179

A-14 **NJ 系列的版本相关信息 A-180**

A-14-1 单元版本和 Sysmac Studio 版本之间的关系 A-180

A-14-2 硬件修订版本和 Sysmac Studio 版本之间的关系 A-181

A-14-3 通过更新单元版本追加和变更支持功能 A-182

A-14-4 通过单元版本更新改善性能 A-185

索引

承诺事项

关于 “ 本产品 ”，若无特殊协议，无论客户从何处购买，均适用本承诺事项中的条件。

● 定义

本承诺事项中术语的定义如下所示。

- “ 本公司产品 ”：“ 本公司 ” 的 FA 系统设备、通用控制设备、传感设备、电子和机械零件
- “ 产品样本等 ”：与 “ 本公司产品 ” 相关的欧姆龙工控设备、电子和机械零件综合样本、其他产品样本、规格书、使用说明书、手册等，还包括通过电磁介质提供的资料。
- “ 使用条件等 ”：“ 产品样本等 ” 中的 “ 本公司产品 ” 的使用条件、额定值、性能、运行环境、使用方法、使用注意事项、禁止事项等
- “ 用户用途 ”：用户使用 “ 本公司产品 ” 的方法，包括直接使用或将 “ 本公司产品 ” 装入用户制造的零件、印刷电路板、机械、设备或系统等。
- “ 适用性等 ”：“ 用户用途 ” 中 “ 本公司产品 ” 的 （a）适用性、（b）动作、（c）不侵犯第三方知识产权、（d）遵守法律以及 （e）遵守各种标准

● 记载内容的注意事项

关于 “ 产品样本等 ” 中的内容，请注意以下几点。

- 额定值和性能值是在各条件下进行单独试验后获取的值，并不保证在复合条件下可获取各额定值和性能值。
- 参考数据仅供参考，并不保证在该范围内始终正常运行。
- 使用实例仅供参考，“ 本公司 ” 不保证 “ 适用性等 ”。
- “ 本公司 ” 可能会因产品改良、本公司的原因而中止 “ 本产品 ” 的生产或变更 “ 本产品 ” 的规格。

● 使用注意事项

使用时，请注意以下几点。

- 使用时请符合额定值、性能以及 “ 使用条件等 ”。
- 请用户自行确认 “ 适用性等 ”，判断是否可使用 “ 本公司产品 ”。“ 本公司 ” 对 “ 适用性等 ” 不作任何保证。
- 用户将 “ 本公司产品 ” 用于整个系统时，请务必事先自行确认配电、设置是否恰当。
- 使用 “ 本公司产品 ” 时，请注意以下各事项。（i）使用 “ 本公司产品 ” 时，应在额定值和性能方面留有余量，采用冗余设计等安全设计，（ii）采用安全设计，即使 “ 本公司产品 ” 发生故障，也可将 “ 用户用途 ” 造成的危险降至最低程度，（iii）对整个系统采取安全措施，以便向使用者告知危险，（iv）定期维护 “ 本公司产品 ” 及 “ 用户用途 ”。
- 本公司设计并制造面向一般工业产品的通用产品。但是，不可用于以下用途。如果用户将 “ 本公司产品 ” 用于以下用途，则 “ 本公司 ” 不对 “ 本公司产品 ” 作任何保证。但如果属于本公司许可的特别产品用途或与本公司签订特殊协议的场合除外。
  - (a) 需高安全性的用途 （例：原子能控制设备、燃烧设备、航空航天设备、铁路设备、起重设备、娱乐设备、医疗设备、安全装置以及其他危及生命、健康的用途）

(b) 需高可靠性的用途 （例：煤气、自来水、电力等供应系统、 24 小时连续运行的系统、支付系统等涉及权利、财产的用途等）

(c) 用于严格条件或环境下 （例：需设置在室外的设备、会受化学污染的设备、会受电磁波干扰的设备、会受振动、冲击影响的设备等）

(d) 在 “ 产品样本等 ” 中未记载的条件或环境下使用

- 上述 （a）～ （d）以及 “ 本产品样本等中记载的产品 ” 不可用于汽车 （含两轮车。下同）。请勿装入汽车进行使用。关于可装入汽车的产品，请咨询本公司销售负责人。

● 保修条件

“ 本产品 ” 的保修条件如下所述。

- 保修期为购买本产品后的 1 年内。
  （“ 产品样本等 ” 中另有记载的情况除外。）
- 保修内容　对发生故障的 “ 本公司产品 ”，经 “ 本公司 ” 判断后提供以下任一服务。

  (a) 发生故障的 “ 本公司产品 ” 可在本公司维修服务网点免费维修
  （不提供电子和机械零件的维修服务。）

  (b) 免费提供与发生故障的 “ 本公司产品 ” 数量相同的替代品
- 非保修范围　如果因以下任一原因造成故障，则不在保修范围内。

  (a) 用于非 “ 本公司产品 ” 原本用途的用途时

  (b) 未按 “ 使用条件等 ” 进行使用

  (c) 违反本承诺事项中的 “ 使用注意事项 ” 进行使用

  (d) 改造或维修未经 “ 本公司 ”

  (e) 使用的软件程序非由 “ 本公司 ” 人员编制

  (f) 因以出厂时的科学技术水平无法预见的原因

  (g) 除上述以外，因 “ 本公司 ” 或 “ 本公司产品 ” 以外的原因 （包括自然灾害等不可抗力）

● 责任免除

本承诺事项中的保修即与 “ 本公司产品 ” 相关的保修的所有内容。
对因 “ 本公司产品 ” 造成的损害， “ 本公司 ” 及 “ 本公司产品 ” 的销售店概不负责。

● 出口管理

出口 “ 本公司产品 ” 或技术资料或向非居民的人员提供时，应遵守日本及各国安全保障贸易管理相关的法律法规。如果用户违反上述法律法规，则可能无法向其提供 “ 本公司产品 ” 或技术资料。

安全注意事项

安全信息的标识及其含义

为了安全使用 NJ/NX 系列 CPU 单元，本手册使用下列标识及图标说明注意事项。这里所记载的注意事项均为与安全有重大相关的内容。请务必遵守。
标识及含义如下所示。

⚠ 警告	操作不当时可能导致操作人员轻度、中度受伤，严重时可致重伤或死亡。此外还有可能引发重大财产损失。
⚠ 注意	如果不正确操作处理，本危险有时可能会导致轻伤、中等程度伤害，也可能会造成物质损失。

图标说明

⃠符号表示禁止。
具体内容参见⃠内和文字叙述。
左图表示 “ 禁止拆解 ”。

该三角形符号表示注意事项 (包括警告)。
具体内容参见三角形内和文字叙述。
左图表示 “ 小心触电 ”。

该三角形符号表示注意事项 (包括警告)。
具体内容参见三角形内和文字叙述。
左图表示 “ 一般注意事项 ”。

●符号表示强制。
具体内容参见●内和文字叙述。
左图表示 “ 一般强制事项 ”。

该三角形符号表示注意事项 (包括警告)。
具体内容参见三角形内和文字叙述。
左图表示 “ 注意高温 ”。

警告

⚠ 警 告

通电中

通电时请勿接触端子部。否则可能导致触电。

请勿对本产品进行分解。特别时通电时或刚断开电源后，电源单元内部可能因升压而处于高电压状态，可能导致触电。此外，还可能因内部的尖锐部件而受伤。

故障安全对策

因 CPU 单元从站或单元的故障或外部原因而导致异常时，请在外部采取安全措施，以确保系统整体的安全性。否则可能会因异常动作而导致严重事故。

请务必通过外部的控制回路构成紧急停止回路、联锁回路、限制回路等安全保护相关回路。

若输出继电器卡死、烧毁或输出晶体管毁损，输出可能会保持在 ON 或 OFF 状态。此时，必须在外部采取措施以确保系统安全。

以下情况时，CPU 单元将关闭所有基本输出单元和数字输出单元的所有输出，从站将按照从站侧的动作运行。
发生电源部异常时
发生电源连接不正确时
发生 CPU 异常 （WDT 异常）或 CPU 复位时
发生全部停止故障等级的控制器异常时
从接通电源到变为运行模式的启动过程中
发生系统初始化异常时
以上情况下，必须在外部采取措施以确保系统安全。

若从站等外部电源变为过载状态或短路状态，电压将下降，可能导致输出变为 OFF、输入无法导入。此时，必须在外部采取措施，以确保系统安全，如在必要时监视电源电压，并导入到控制中。

变量存储器、CJ 单元用存储器等异常时，可能发生意外输出。此时，必须在外部采取措施以确保系统安全。

请在通信系统或用户程序中采取措施，确保在数据链接通信或远程 I/O 通信中发生通信错误或误动作时的系统安全。

远程 I/O 中发生通信故障或发生全部停止故障等级的异常时，输出状态取决于所用产品的规格。
请在确认发生上述通信故障或全部停止故障等级的异常时，其动作规格如何，再采取安全对策。
请分别正确地设定从站及单元。

NJ/NX 系列在发生瞬时停电时，也可在一定时间内正常运行，因此可能收到受瞬时停电影响的外部设备传来的错误信号。
请在外部采取故障安全对策，必要时，对外部设备的电源电压进行监视，并作为联锁条件导入。

用户必须针对信号线路断线或瞬时断电导致的信号异常，采取相应的故障安全措施。
否则可能会因异常动作而导致严重事故。

电压 / 电流输入

请勿对从站或单元施加超过指定范围的电压或电流。
否则，可能会导致故障或火灾。

传送

通过 Sysmac Studio 传送用户程序、“ 构成 / 设定 ” 数据、设备变量、CJ 单元用存储器的值时，请在确认传送目标安全后再执行。
无论 CPU 单元处于什么动作模式，装置、机器都可能发生意外动作。

正式运行时

请对编写完成的用户程序及各种数据、设定值进行充分的动作确认后，再转移到正式运行。

注意

⚠ 注 意

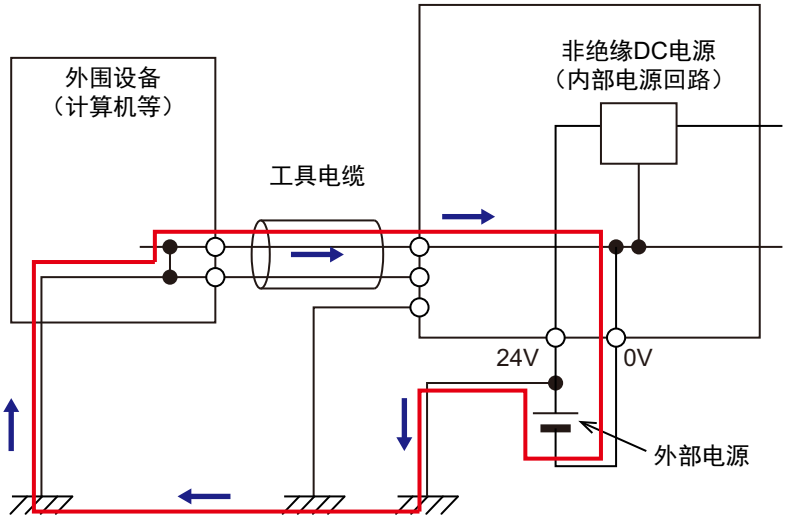
使用时

请不要在通电过程中及切断电源后立即触摸装置。否则可能导致烫伤。

配线时

请务必按照本手册或参考手册中指定的扭矩紧固端子台螺钉和电缆螺钉。螺钉松动可能导致起火及误动作。

将计算机或外围设备连接到使用非绝缘电源型 DC 单元的控制器上时，需将外部电源的 0V 侧接地，或者不对外部电源接地。
否则，会因为外围设备接地方式的差异导致外部电源短路。如下图所示，请勿将外部电源的 24V 侧接地。



在线编辑

请确认输入输出时间打乱后也不会受到影响，再执行在线编辑。进行在线编辑后，任务执行时间可能超过任务周期，与外部之间的输入输出无法更新，导致输入信号无法读取、输出时间紊乱。

显示错误信息“指令可能发生意外动作”时

请取消勾选 Sysmac Studio 的选项项目“程序检查”的“将输入输出变量传递到特定指令的特定参数时发生错误”时，指令可能发生意外动作，也可能对装置造成影响。
请务必确认与《NJ/NX 系列 指令基准手册 基本篇（SBCA-CN5-360）》中记载的可使用条件一致。

版本相关信息

上述错误信息的显示及选项设定仅在 Sysmac Studio Ver.1.02 以上的环境下可以实现。

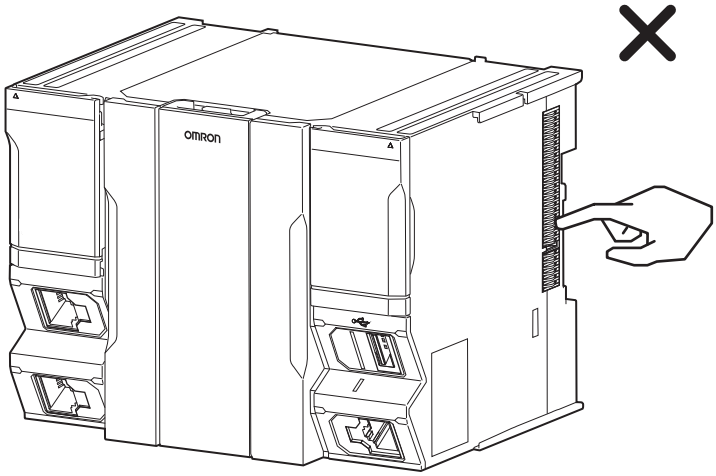
安全要点

分解和掉落

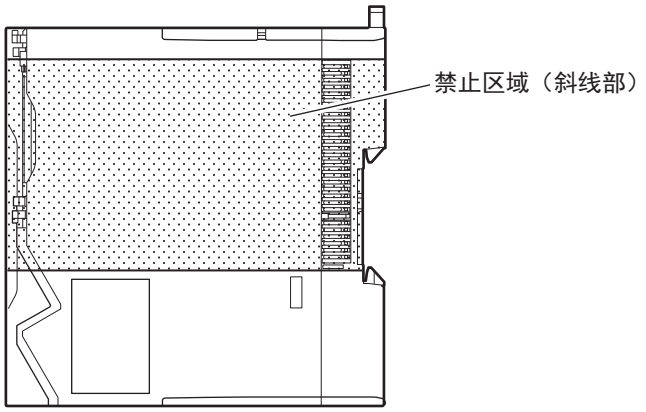
- 请勿对本产品进行分解、修理或改装。否则可能导致故障或起火。
- 请勿使产品掉落，或对其施加异常振动和冲击。否则可能导致产品故障、烧毁。

安装时

- 连接电源单元、CPU单元、I/O单元等时，单元间的连接器咬合后，请滑动至上下的滑片发出“咔哒”声，牢固锁定。
- 单元上不可粘贴贴纸或胶带。安装或拆卸单元时，粘着物或碎屑可能沾到 NX 总线连接器端子上，导致误动作。
- 请勿接触单元的 NX 总线连接器端子。否则可能弄脏 NX 总线连接器端子，导致单元误动作。



- 在下图所示的 CPU 单元禁止区域内，请勿用油性笔写字或乱涂乱画。安装或拆卸单元时，油墨或脏污可能沾到 NX 总线连接器端子上，导致单元误动作。



安装时

- 安装时请务必进行D种接地（第3种接地）。特别时将电源单元的LG端子和GR端子短接后，为了防止触电，请务必采用 D 种接地 （第 3 种接地）。

配线时

- 请按照本手册中的指定步骤正确配线。
  在接通电源前，应仔细检查所有的配线及开关等的设定。
- 请使用压接端子进行配线。
  请勿将仅实施了绞接的电线直接连接至端子台。
- 请勿强行扭曲或拉拽电缆。
  请勿在电缆的电线上放置重物。否则可能导致断线。
- 充分确认端子台、连接器的安装位置后，再进行安装。
- 如端子台、扩展电缆等带有锁定机构，则请务必确认锁紧后再使用。
- 出厂时若单元上部粘贴有防尘标签，通电前请务必撕下标签。否则可能导致无法散热，引起误动作。
- 连接计算机和CPU单元时，请先从AC插座上拔掉计算机的电源插头。此外，若计算机上有FG端子，连接时请确保其与电源单元的 GR 端子为同电位。
  否则计算机和控制器之间会出现电位差，可能导致故障或误动作。
- 请勿通过输出单元或从站向有极性的外部电源施加反向电压。无论控制器是否运行，若有反向电流通过，可能对连接的设备造成破坏。

电源设计时

- 系统中使用的电源单元请勿超出《NX 系列 CPU 单元 用户手册 硬件篇 （SBCA-CN5-418）》及《NJ系列 CPU单元 用户手册 硬件篇（SBCA-CN5-358）》中记载的电源单元的额定供给容量范围。
  否则可能发生停止运行或误动作，或者在断电时无法备份需要的数据。
  包括 NJ 系列 CPU 装置和扩展装置在内，请使用 NJ 系列电源单元。
  若使用 CJ 系列电源或将 NJ 系列电源与 CJ 系列单元组合，将不会动作。
- 请勿在输出端子或从站上施加超过最大额定值的电压或连接负载。
- 接通电源时可能产生浪涌电流。选择外部回路的保险丝、断路器时，请考虑溶断・检测特性和上述内容，设计时留出足够的余量。浪涌电流规格请参考《NX 系列 CPU 单元 用户手册 硬件篇（SBCA-CN5-418）》及《NJ 系列 CPU 单元 用户手册 硬件篇 （SBCA-CN5-358）》。
- 进行耐电压试验时，若通过开关施加或断开全部电压，因产生的脉冲电压，可能导致电源单元破损。
  应用电压请用试验机的旋钮慢慢变化。
- 绝缘电阻及耐电压试验时，请在电源单元的 L1 端子、 L2 端子和 GR 端子之间施加电压。
- 请勿对 AC 电源供给变频器产生的方波输出。否则可能因内部温度上升导致冒烟、烧毁。请务必输入《NX 系列 CPU 单元 用户手册 硬件篇（SBCA-CN5-418）》及《NJ 系列 CPU 单元 用户手册 硬件篇 （SBCA-CN5-358）》中所示的频率正弦波。
- 为防止外部配线短路，请采取安全措施，如设置断路器等。

电源接通时

- 从接通电源到变为运行模式约需 10 ～ 20 秒。在此期间的输出遵照从站或单元的规格。请设计故障安全回路，避免在使用电源单元的运行中输出等时，外部设备发生误动作。
- 请务必对外部回路进行配置，从而在接通控制器的电源后再接通控制系统的电源。若先接通控制系统电源后再接通控制器电源，则在接通控制器电源时，输出单元的输出会瞬间发生误动作，从而导致控制系统的输出临时出错。
- 若要在接通电源时将 SD 存储卡的备份文件数据传送到控制器中，请合理选择要传送的数据组。若将错误的数据组数据传送到控制器中，设备可能发生意外动作。
- 在单元版本 1.05 以下的 CPU 单元中，从接通电源到变为运行模式期间，请勿关闭控制器的电源。若关闭了电源，下次启动时可能发生 “ 电池备份存储器检查错误 ”。发生 “ 电池备份存储器检查错误 ” 时，保持变量变为初始值，CJ 单元用存储器的 HR ／ DM ／ EM 将清零。若要重新开始运行，需要对保持变量或 CJ 单元用存储器重新设定正确的数据。

电源 OFF 时

- BUSY LED 闪烁时，请勿关闭控制器的电源。 BUSY LED 闪烁时，表示正在向内置非易失性存储器备份 CPU 单元内的用户程序及设定值。若关闭电源，这些数据将无法正常备份。此外，下次启动时将发生全部停止故障等级的控制器异常，运行停止。
- 正在访问 SD 存储卡（SD BUSY LED 闪烁）时，请勿关闭电源或拔出 SD 存储卡。否则数据可能损坏，使用该数据时将无法动作。从通电中的 CPU 单元中取出 SD 存储卡时，请先按下 SD 存储卡停止供电按钮，待 SD BUSY LED 熄灯后，再拔出 SD 存储卡。
- 正在通过工具传送各种数据、用户程序时，请勿拔掉电缆或关闭控制器的电源。
- 进行下列操作时，请务必切断控制器的电源。
  安装或拆卸单元时
  组装装置时
  设定拨动开关或旋转开关时
  连接电缆或进行配线时
  连接或断开连接器时
  安装或拆卸风扇单元时
- 切断电源后，电源单元可能会对控制器继续供电数秒钟，此时， PWR LED 亮灯。上述操作请在确认 PWR LED 熄灭后再执行。

操作时

- 在进行以下任何一项操作前，请确认其不会对设备造成不良影响。
  变更 CPU 单元的动作模式 （包括接通电源时的动作模式设定）
  用户程序、设定的变更
  设定值 / 当前值的变更
  强制值刷新
- 变更从站或单元的设定并重启时，请充分确认连接目标的对象设备绝对安全。
- 联合CJ系列基本单元和EtherCAT从站等不同的功能模块对装置进行控制时，请在用户程序或外部采取对策，确保在一个功能模块停止时也不会造成影响。各功能模块发生部分停止故障等级的异常时，相应的输出遵照从站或单元的规格。
- 要对 EtherCAT 主机功能模块解除部分停止故障等级以上的控制器异常时，请充分确认连接目标的设备绝对安全。
  因部分停止故障等级以上的控制器异常而变为可操作状态以外（输出无效）的从站将变为可操作状态，输出变为有效。
  解除所有异常或重启从站时，请确认 EtherCAT 主机功能模块是否发生部分停止故障等级的控制器异常。
- 要对 CJ 系列高功能单元解除控制器异常时，请充分确认连接目标的设备绝对安全。对发生控制器异常且异常等级为监视信息以上的 CJ 系列高功能单元，解除其控制器异常后，该单元将重启。
  解除所有异常时，请确认 CJ 系列高功能单元是否发生监视信息等级以上的控制器异常。“ 控制器异常 ” 标签画面中不会显示监视信息等级的事件，因此可能在没有意识到的情况下重启 CJ 系列高功能单元。
  是否有监视信息等级的控制器异常，可通过监视窗口监视异常状态变量 _CJB_UnitErrSta[0,0] ～ _CJB_UnitErrSta[3,9] 来确认。

电池保持

- 用户程序及变量的初始值保存在CPU单元内部的非易失性存储器中，有保持属性的变量的当前值及CJ单元用存储器的 HR/DM/EM 值由电池保持其数据。
  若未安装电池或电池耗尽， CPU 单元将检测到电池备份存储器检查错误。
  此时，有保持属性的变量变为初始值， CJ 单元用存储器的 HR ／ DM ／ EM 将清零。
  为使装置能在这样的情况下仍安全动作，请充分验证有保持属性的变量的初始值设定及初始值时的动作。
- 绝对编码器原点位置偏置由电池保持。接通电源时，若 CPU 单元检测到电池电压低 / 电池未安装，绝对编码器原点位置偏置将清零，发生 “ 绝对编码器原点位置偏置读取异常 ”。请解除异常，执行原点复位，确定原点。若未确定原点，控制对象可能发生意外动作。

调试时

- 强制值刷新是指与用户程序执行结果无关，始终以设定的值执行 I/O 刷新的功能。因此，I/O 刷新对象以外的接点会因本功能的影响先变为设定的值，再根据用户程序替换。与 CJ 系列的强制设置 / 复位功能的动作不同，敬请注意。
- 强制值刷新信息不可通过 Sysmac Studio 上传 / 下载。
  因此，下载包含强制值刷新对象的数据后，请根据具体需要，变更为运行模式，通过 Sysmac Studio 执行强制值刷新操作。
  因强制状态不同，控制对象可能发生意外动作。
- 请勿将多个变量 AT （分配目标）指定到同一地址。
  因为要以多个变量名称访问一个实体，所以用户程序看起来会很乱，也可能导致程序出错。

一般通信

- 数据链接通信时，请参考 _ErrSta （控制器异常状态），仅在数据链接源装置未发生异常时，创建需要使用接收数据的用户程序。
  发生异常时，通过数据链接接收的数据可能不正确。
- 数据链接表设定不当时，可能导致设备动作异常。此外，即使设定了适当的数据链接表，也请在确认不会对设备产生不良影响后，再传送数据链接表。传送后将自动开始数据链接。
- 路由表从编程设备传送至 CPU 单元后，所有 CPU 总线单元将重启。请在重启后确认不会对设备造成影响，再执行传送。
- 运行过程中可传送标签数据链接参数，但在传送标签数据链接参数的过程中，相关节点间的标签数据链接将停止。请在确认不会对设备造成影响后，再执行传送。

EtherNet/IP 通信

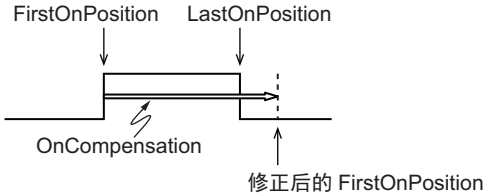
- EtherNet/IP请以规格范围内的通信距离及连接台数/连接方法使用。此外，请勿连接EtherCAT等其他网络。否则可能因过载导致网络崩溃或误动作。
- 若从编程设备传送内置 EtherNet/IP 的设定 （IP 地址或标签数据链接的设定），设定对象的所有 EtherNet/IP （节点）将重置。这是因为读入设定内容后生效。请在重置后确认不会对设备造成影响，再执行传送。
- 若使用中继集线器进行 EtherNet/IP 的标签数据链接通信 （周期链接通信），网络的通信负载会升高，容易发生冲撞 （冲突），将无法稳定地通信。在使用标签数据链接的网络中，请务必使用交换式集线器。
- 将两个 EtherNet/IP 作为不同的网络使用时，请分别通过不同的集线器与通信对象连接。
- 注意不要弄错两个 EtherNet/IP 的连接目标。

EtherCAT 通信

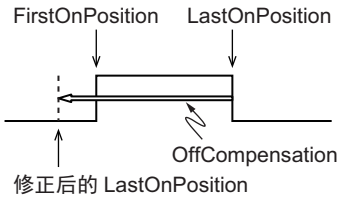
- EtherCAT 请以规格范围内的通信距离及连接台数 / 连接方法使用。
  此外，请勿连接 EtherNet/IP 或普通局域网等其他网络。否则可能因过载导致网络崩溃或误动作。
- 因 EtherCAT 主机和从站的修订版本组合不同，可能导致误动作或意外动作。将网络设定的修订版本检查功能设为无效后使用时，请通过 Sysmac Studio 确认主机上设定的从站修订版本和实际从站的修订版本，并查阅从站点的手册等，确认功能的兼容性。此外，实际从站的修订版本可通过 Sysmac Studio 或铭牌确认。
- 若传送用户程序，CPU 单元会重启，与 EtherCAT 从站的通信将中断。在此期间的从站输出遵照从站的规格。因 EtherCAT 网络配置不同，通信中断的时间也不同。
  传送用户程序时，请确认不会对装置造成影响后再执行。
- “低效运行设定”设定为“停止”时，若任一从站发生 EtherCAT 通信异常，所有从站和过程数据通信将停止。此时，伺服驱动器的动作遵照伺服驱动器的规格。请充分检验“低效运行设定”，确保装置发生异常时能安全运行。
- 刚接通电源时，EtherCAT 的通信可能尚未建立。请利用系统定义变量确认通信建立后，再创建用于控制的程序。
- 发生干扰时或从 EtherCAT 从站上拔下网络电缆时，正在通信的帧可能会丢失。帧丢失时，将无法传送从站的输入输出数据，可能发生意外动作。从站的输出遵照从站的规格。详情请参考各从站的手册。
  需要采取干扰对策或更换从站时，请执行以下处理。
  - 请将系统定义变量的输入数据无效作为联锁条件来创建用户程序。
  - 请务必将 EtherCAT 主机设定项目中的“PDO 通信连续超时检测此时”设定为 2 次以上。详情请参考
    📖《NJ/NX 系列 CPU 单元内置 EtherCAT 端口 用户手册 (SBCD-CN5-358)》。
- EtherCAT从站设定为“脱离”或“无效”时，该从站之后的从站都将停止通信，无法控制输出。请在确认不会对设备造成影响后，再发出“脱离”、“无效”的指示。
- 刚接通电源时，NX 总线的 I/O 数据通信可能尚未建立。请利用系统定义变量及 EtherCAT 耦合器单元的设备变量确认 I/O 数据通信建立后，再创建用于控制的程序。
- 请勿使用通用 Ethernet 集线器、中继集线器。若使用，可能发生全部停止故障等级的异常。

运动控制

- 进行 MC 调试时，请仔细确认轴编号。
- 若MC调试过程中Sysmac Studio-CPU单元之间的通信中断，将强制停止电机。请注意连接PC和CPU单元通信电缆未松脱，且不会对设备造成影响后，再执行 MC 调试。
- 通过CPU单元的用户程序或Sysmac Studio变更了凸轮数据的值时，请务必执行凸轮表保存指令。若不执行，下次接通电源时，凸轮数据的值将恢复为变更前的状态，装置可能发生意外动作。
- 将伺服驱动器的外部输入中分配的正转侧驱动禁止输入 （POT）、反转侧驱动禁止输入 （NOT）、原点附近输入 （DEC）分别作为正转侧极限输入、反转侧极限输入、原点附近输入使用。这些输入信号的信号宽度请务必确保大于控制周期。若输入信号的宽度小于控制周期，将无法检测到输入信号，导致无法正常动作。
- 变更了Sysmac Studio的[轴基本设定]画面中的“详细设定”时，请在正式运行前，确认装置或机器是否按计划动作。
  若运动控制功能模块的功能和轴中分配的EtherCAT从站的过程数据关系不正确，装置或机器可能发生意外动作。
- 请务必在很定速度的轴上使用 MC_DigitalCamSwitch （数字凸轮开关有效）指令。
  若将计数模式变更为旋转模式，使用 OnCompensation （ON 时间补偿）或 OffCompensation （OFF 时间补偿），轴速度发生急剧变化，将发生以下动作。
  - OnCompensation （ON 时间补偿）或 OffCompensation （OFF 时间补偿）的值超出轴旋转半圈的时间时， InOperation （操作中）将变为 FALSE。
  - OnCompensation（ON 时间补偿）的值超出 LastOnPosition（ON 结束位置）时，输出时间将变得不确定。



  - OffCompensation （OFF 时间补偿）的值超出 FirstOnPosition （ON 开始位置）时，输出时间将变得不确定。



- 请在确认 MC_DigitalCamSwitch（数字凸轮开关有效）指令的 InOperation（操作中）变为 TRUE 后，使用 NX_AryDOutTimeStamp （时间戳数字输出排列写入）指令。
- 变更了单位转换设定时，请重新执行原点复位。若未执行原点复位，控制对象可能发生意外动作。

恢复时

- 从站或单元中，所有或部分设定数据可能不在备份、恢复、核对的对象范围内。此外，设定为无效的从站或单元也不在备份、恢复、核对的对象范围内。使用恢复功能时，请进行充分的动作确认后再转移到正式运行。
- CPU单元利用电池来保持绝对编码器的信息，即“绝对编码器原点位置偏置”。满足以下任一条件时，请将绝对编码器原点位置偏置从恢复对象中排除，再执行恢复。然后，请重新设定绝对编码器原点。若未设定绝对编码器原点，控制对象可能发生意外动作。
  - 伺服电机或驱动器的个体变为创建备份时的个体。
  - 创建备份后，执行了绝对编码器的设置。
  - 绝对编码器的绝对值数据将丢失。

程序传送时

- 若通过SD存储卡程序传送功能传送接通电源时的动作模式为运行模式的程序，无论运行前的状态或设定如何，传送完成后都将切换为运行模式。确认装置开始运行且没有问题后，再执行传送。

更换电池时

- 切勿短接电池的正负极端子或对电池进行充电、拆解、加热或焚烧。请勿使电池受到强烈的冲击。否则会导致电池漏液、破裂、发热或起火。
- 若电池掉落到地板上或受到过度冲击，请丢弃该电池。受过冲击的电池在使用过程中会发生漏液。
- UL 规格规定，电池务必由熟练的技术人员进行更换。更换作业请由熟练的技术人员负责。
- 更换电池前，请通电 5 分钟以上，并在电源关闭后 5 分钟内（环境温度 25 ℃时）更换新电池。若无电池的状态持续 5 分钟以上，保持的数据可能会丢失。
- 更换电池时，建议在未通电状态下操作，避免 CPU 单元的精密部件因静电而损坏或误动作。不关闭电源、在通电状态下也可更换电池，但请在接触单元前先触摸接地金属，去除人体静电后再更换。
  更换电池后，请连接 Sysmac Studio，解除 “ 电池电压低 ”。

更换风扇单元时

- 刚关闭电源时，请勿用手触摸 CPU 单元中的散热器。
  否则可能导致受伤或烫伤。

更换单元时

- 更换CPU单元时，请先将开始运行所需的用户程序“构成/设定”的设定数据、变量、CJ单元用存储器的值传送到更换后的 CPU 单元及外部链接的设备后，再开始运行。标签数据链接的设定、路由表及 CPU 总线单元的系统设定也保存在 CPU 单元中，也请传送。
- CPU单元利用电池来保持绝对编码器的信息，即“绝对编码器原点位置偏置”。制作新装置或更换伺服电机等，需要变更 CPU 单元和伺服电机的个体组合时，请重新设定绝对编码器原点。
- 若硬件修订版本不同，性能也会不同。向不同硬件修订版本的 CPU 单元传送用户程序及各种数据、设定值时，请进行充分的动作确认后再转移到正式运行。

废弃时

- 对本体和电池的废弃处理有时受到当地法规限制。请遵照当地适用法规进行废置处理。

- 装有锂一次电池（高氯酸含量 6ppb 以上）的产品经过或出口至美国加利福尼亚州时，有标示以下信息的义务。
  Perchlorate Material - special handling may apply.
  See www.dtsc.ca.gov/hazardouswaste/perchlorate
- CPU 单元中配备有锂一次电池（高氯酸含量 6ppb 以上）。向美国加利福尼亚州出口装有CPU 单元的产品 （成品）时，请在个装箱、运输箱等包装上标示上述内容。

使用注意事项

保管和安装时

- 请按照本手册中的说明以正确安装。
- 请勿安装或保管在下列场所。否则可能导致运行停止、误动作。
  日光直射的场所；
  环境温度或相对湿度超出规格中规定范围的场所；
  温度变化剧烈容易引起结露的场所；
  有腐蚀性气体、可燃性气体的场所；
  尘土、粉尘、盐分、铁屑较多的场所；
  有水、油、化学品等飞沫喷溅的场所；
  直接致使本体产生振动或冲击的场所；
- 在下列场所使用时，请充分采取遮蔽措施。
  产生强高频干扰的设备附近；
  可能因静电等产生干扰的场所；
  产生强电场或磁场的场所；
  可能受到辐射的场所；
  附近有电源线或动力线通过的场所；
- 接触单元前请先采取触摸接地金属等除静电措施。
- 请勿安装在发热体附近，并确保通风。否则可能导致误动作、运行停止或烧毁。

安装时

- 若 CS/CJ 系列 I/O 连接电缆的连接器松脱，将发生 I/O 总线检查错误，并停止运行，敬请注意。
- 安装单元时，请勿触摸或碰撞 NX 总线连接器的端子。
- 安装单元时，请勿对 NX 总线连接器的端子施加压力。
  若在 NX 总线连接器端子变形的状态下安装单元并通电，可能因接触不良导致误动作。
- 为了保护CPU装置上的单元，请务必在CPU装置的末端安装端盖。若不安装，可能导致CPU单元故障或误动作。
- 安装单元后，请务必从两侧固定端板。若不固定，可能导致故障或误动作。
- 使用绝缘隔片安装CPU装置时，高度会高出约10mm。CPU装置的单元或连接线不得与其他设备接触。

配线时

- 请勿从单元开口处放入异物。否则可能引起烧毁、触电或故障。
- 请勿使配线废料或碎屑进入单元内部。否则会导致烧毁、故障、误动作。特别是施工时，请用物品覆盖。
- EtherCAT、EtherNet/IP的连接方法和使用的电缆请参考 📖《NJ/NX系列 CPU单元内置EtherCAT端口用户手册（SBCD-CN5-358）》、📖《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册（SBCD-CN5-359）》的记载内容。否则可能导致通信不良。
- 电源单元请在额定电源电压下使用。请特别注意供电不稳定的地方，确保能供给额定电压或频率的电源。
- 电源配线时请注意电线的电流容量。否则可能导致发热。若端子和端子采用交叉配线，各电线中流过的电流为各条线相加的值。采用交叉配线时，请注意所有电线的电流容量。
- 请不要在切断电源后立即触摸电源单元的端子。否则可能因残留电压导致触电。
- 将磁簧开关作为 AC 输入单元的输入接点使用时，请使用容许电流为 1A 以上的产品。
  若使用容许电流较小的磁簧开关，可能因浪涌电流导致接点熔敷。

异常处理

- 在使用异常状态获取指令检测结果的应用程序中，请在编程时考虑检测时对设备产生的影响。例如，在动作过程中更换了电池等，即使检测到轻微异常，因用户程序的关系，也可能对动作造成影响。
- 变更了控制器异常的重要程度时，发生异常时的输出状态可能因变更而发生变化。请在确认安全后使用。

恢复时、自动传送时

- 编辑“恢复指令文件”、“自动传送指令文件”时，请只记述可选择的数据组的“yes/no”。若编辑其他记述后执行恢复或自动传送，控制器可能发生意外动作。

程序传送时

- 使用 SD 存储卡程序传送功能时，为了防止传送计划外的程序，请设计为每次都需要输入密码。

更换从站或单元时

- 若更换了从站或单元，请参考各从站或各单元的手册，修改必要的设定。

任务设置

- 发生 “ 任务周期超限 ” 时，请在创建程序时确保在任务周期内，或设定任务周期。

运动控制

- 请利用系统定义变量确认已建立EtherCAT通信，再启动运动控制指令。在未建立EtherCAT通信的状态下，将无法正常启动运动控制指令。
- 请利用系统定义变量，对运动控制功能模块所控制的从站是否发生通信异常进行监视。在发生从站通信异常的状态下，将无法正常启动运动控制指令。
- 开始 MC 调试时，请在确认动作参数正确后，再开始 MC 调试动作。
- MC 调试过程中，请勿下载运动设定。

EtherCAT 通信

- 若要在运行过程中拔出 EtherCAT 从站电缆，请先脱离 EtherCAT 从站，或将后续 EtherCAT 从站在内的从站设为无效后再拔出。
- 请设定伺服驱动器，使其在控制器和伺服驱动器之间的 EtherCAT 通信发生异常时停止动作
- 要解除EtherCAT主机功能模块的“网络配置核对异常”、“过程数据通信异常”、“链接断开异常”时，请事先确认所有恢复对象的从站都处于加入状态。若有未加入的从站，EtherCAT 主机功能模块可能会访问指定节点地址以外的从站，或无法正确解除异常。
- 请务必使用指定的 EtherCAT 从站电缆。若使用非指定的电缆，EtherCAT 主机或 EtherCAT 从站可能检测出异常，出现以下现象。
  - 过程数据通信的刷新无法持续执行。
  - 过程数据通信的刷新无法在设定周期内完成。

更换电池时

- 请确认电池的标签，应使用制造后 2 年以内的电池进行更换。
- 长期闲置的 CPU 单元应在更换电池后再接通电源。更换电池后，若一次都不打开电源继续放置，可能导致电池寿命缩短。
- 更换电池时，请使用电池组 CJ1W-BAT01。

SD 存储卡

- 将 SD 存储卡插到底。
- 访问 SD 存储卡的过程中请勿断开控制器的电源。否则文件可能损坏。
  若 SD 存储卡中有损坏的文件，将通过 SD 存储卡修复功能，在接通电源时自动删除该文件。
- 使用欧姆龙生产的 SD 存储卡时，通过以下方法检测 SD 存储卡的寿命。
  - 系统定义变量 “_Card1Deteriorated （SD 存储卡使用寿命警告标志） ”
  - 事件 “SD 存储卡达到使用寿命 ”

  发生上述情况时，请更换 SD 存储卡。

在线编辑

- 以 CPU 单元版本 1.04 以上和 Sysmac Studio Ver.1.05 以上的组合进行在线编辑时，CPU 单元会将在线编辑时更新过的程序备份到内置非易失性存储器中。在此期间， Sysmac Studio 将显示正在备份的信息。显示该信息时，请勿断开控制器的电源。若断开控制器的电源，下次接通电源时，控制器将发生异常。

法规和标准

日本国外的使用

出口 （或向非居住者提供）本产品中符合外汇及外国贸易法规定的出口许可、批准对象货物 （或技术）要求的产品时，须依照该法获得出口许可、批准 （或劳务交易许可）。

符合 EU 指令

适用指令

- EMC 指令
- 低电压指令

概念

● EMC 指令

由于 OMRON 公司的产品为安装在各种机械、制造装置中使用的电气设备，为了使安装 PLC 的机械、装置更容易符合 EMC 标准，必须先使产品本身达到相关 EMC 标准 （*）。
但是，由于客户使用的机械、装置各不相同，EMC 性能因符合 EU 指令的产品安装的设备和控制柜的构成、配线及配置等条件而异，无法在客户使用状态下确认其适合性。因此，机械、装置整体最终的 EMC 适用性确认请客户自行实施。

* EMC (Electro-Magnetic Compatibility：电磁兼容性) 相关标准中，
  EMS (Electro-Magnetic Susceptibility：电磁敏感度) 应符合 EN 61131-2
  EMI (Electro-Magnetic Interference：电磁干扰) 应符合 EN 61131-2
  此外， Radiated emission 基于 10m 法。

● 低电压指令

对于以电源电压 50V AC ～ 1000V AC 以及 75V DC ～ 1500V DC 工作的设备，要求必须确保必要的安全性。适用标准为 EN 61010-2-201。

● 符合 EU 指令

NJ/NX 系列符合 EU 指令。为确保客户使用的机械、装置符合 EU 指令，必须注意下列几点。

- NJ/NX 系列必须安装在控制柜内。
- 连接到 DC 电源单元和 I/O 单元的 DC 电源，请使用 SELV 规格的电源。
- NJ/NX系列同时符合EU指令和EMI相关的通用发射标准。特别是Radiated emission(10m法)，因控制柜的配置、连至控制柜的其它设备、接线和其它条件而异。
  因此，即使使用符合 EU 指令的 NJ/NX 系列时，客户也必须确认机械、装置整体是否符合 EU 指令。

KC 标准的适用

在韩国使用本产品时，请遵守以下注意事项。

A 급 기기 (업무용방송통신기자재)
이 기기는 업무용(A 급) 전저파작합기기로서 판매자
또는 사용자는 이 점을 주의하시기 바라며, 가정외의
지역에서 사용하는 것을 목적으로 합니다.

ClassA 设备 （商用广播通信设备）

本设备为商用电磁波发生设备 （ClassA），设计用于家庭以外的场所。经销商和用户应注意以下内容。

船舶标准的适用

本产品符合各类船舶标准。适用各类船舶标准时，设定了使用条件，根据安装场所不同，可能无法使用。使用时，请务必与本公司联系。

各船舶标准的使用条件 （NK 、LR）

- 本产品必须安装在控制柜内。
- 控制柜开关口的缝隙应用垫片等完全密封。
- 电源线应与以下噪声滤波器连接。

噪声滤波器

厂家	型号
COSEL	TAH-06-683

软件许可和著作权

本产品中组装有第三方开发的软件。关于软件的许可和著作权，请参考
http://www.fa.omron.co.jp/nj_info_j/。

版本

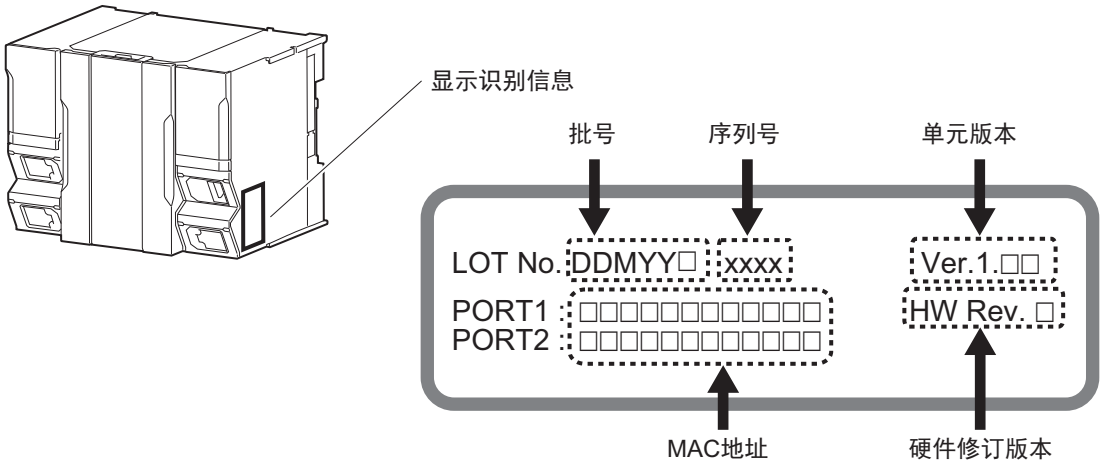
NJ/NX 系列的各单元及各 EtherCAT 从站的硬件或软件按照硬件修订版本或单元版本等编号管理。每次硬件或软件发生规格变更后，硬件修订版本或单元版本都会更新。因此，即使是同一型号的单元或 EtherCAT 从站，若硬件修订版本或单元版本不同，配备的功能或性能可能不同。

版本的确认方法

版本可通过标示的识别信息或 Sysmac Studio 确认。

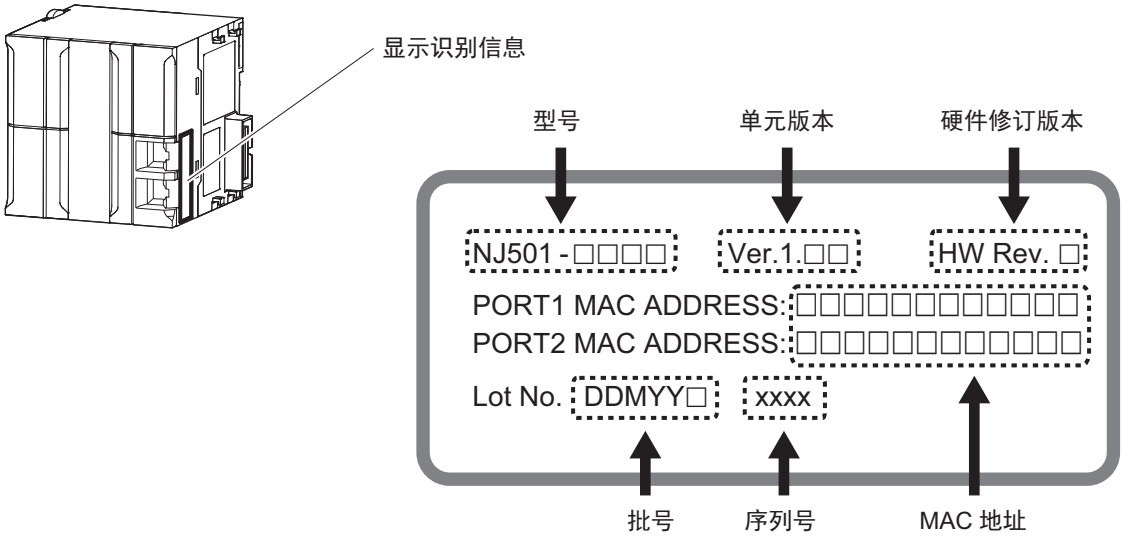
通过标示的识别信息确认

可通过产品侧面标示的识别信息确认版本。
NX 系列 CPU 单元 NX701- □□□□的识别信息标示如下图所示。



（注）硬件修订版本为 “ 无 ” 的单元中，不标示硬件修订版本。

NJ 系列 CPU 单元 NJ501- □□□□的识别信息标示如下图所示。



（注）硬件修订版本为 “ 无 ” 的单元中，不标示硬件修订版本。

通过 Sysmac Studio 确认的方法

可通过 Sysmac Studio 确认版本。单元和 EtherCAT 从站的确认方法不同。

● NX 系列 CPU 单元的版本确认方法

单元版本可通过在线状态的［生产信息］确认。但是，只能确认 CPU 单元的版本。

1 在多视图浏览器中右击［配置和设定］–［CPU・扩展装置］–［CPU 装置］，选择［显示生产信息］。
显示［生产信息］对话框。

● NJ 系列 CPU 单元的版本确认方法

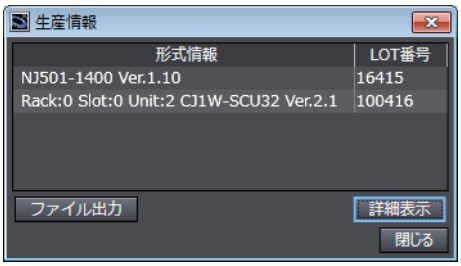
单元版本可通过在线状态的［生产信息］确认。但是，只能确认 CPU 单元、CJ 系列高功能 I/O 单元及 CJ 系列 CPU 总线单元的版本。 CJ 单元的基本 I/O 单元的版本无法通过 Sysmac Studio 确认。

1 在多视图浏览器中双击［配置和设定］–［CPU・扩展装置］。或者右击［配置和设定］–［CPU・扩展装置］，然后选择［编辑］。
显示单元编辑器。

2 右击单元编辑器的空白栏，选择［显示生产信息］。
显示［生产信息］对话框。

● 切换生产信息的显示内容

1 选择［生产信息］对话框右下方的［简易显示］或［详细显示］。
切换［生产信息］的简易显示和详细显示。



简易显示

详细显示

显示内容因简易显示还是详细显示而异。详细显示中，显示单元版本、硬件版本及软件版本。简易显示中，仅显示单元版本。

（注）硬件修订版本在硬件版本的右端以 “/” 分隔显示。硬件修订版本为 “ 无 ” 的单元中，不标示硬件修订版本。

● EtherCAT 从站的版本确认方法

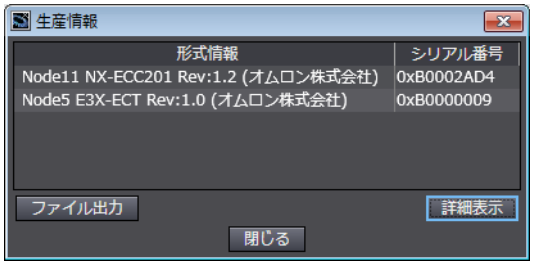
EtherCAT 从站的版本可通过在线状态的［生产信息］确认。确认方法如下所示。

1 在多视图浏览器中双击［配置和设定］–［EtherCAT］。或者右击［配置和设定］–［EtherCAT］，然后选择［编辑］。
显示 EtherCAT 配置编辑画面。

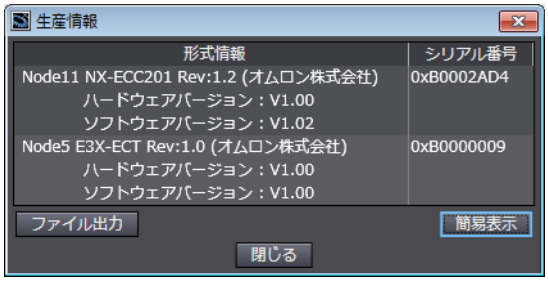
2 右击 EtherCAT 配置编辑画面中的主机，选择［显示生产信息］。
显示生产信息对话框。
单元版本以带 “Rev” 的形式显示。

● 切换生产信息的显示内容

1 选择［生产信息］对话框右下方的［简易显示］或［详细显示］。
切换［生产信息］的简易显示和详细显示。



简易显示



详细显示

CPU 单元的单元版本和 Sysmac Studio 的版本

NJ/NX 系列 CPU 单元的单元版本不同，所配备的功能也不同。使用升级后添加的功能时，需要使用对应版本的 Sysmac Studio。

CPU 单元的单元版本种类和 Sysmac Studio 版本的关系以及单元版本对应的支持功能一览请参考📖「A-13 NX 系列的版本相关信息」（P.A-177）及📖「A-14 NJ 系列的版本相关信息」（P.A-180）。

相关手册

本手册相关的手册如下表所示。请同时参考。

手册名称	Man.No.	型号	用途	内容
NX 系列 CPU 单元 用户手册 硬件篇	SBCA-CN5 -418	NX701- □□□□	希望了解 NX 系列 CPU 单元的概要 / 设计 / 安装 / 保养等基本规格时。 主要是硬件相关的信息。	对 NX 系列的系统整体概要及 CPU 单元相关的如下内容进行说明。 • 特长及系统构成 • 概要 • 各部分的名称和功能 • 一般规格 • 安装和配线 • 维护检查 请同时使用用户手册 软件篇（SBSA-CN5-359）。
NJ 系列 CPU 单元 用户手册 硬件篇	SBCA-CN5 -358	NJ501- □□□□ NJ301- □□□□ NJ101- □□□□	希望了解 NJ 系列 CPU 单元的概要 / 设计 / 安装 / 保养等基本规格时。 主要是硬件相关的信息。	对 NJ 系列的系统整体概要及 CPU 单元相关的如下内容进行说明。 • 特长及系统构成 • 概要 • 各部分的名称和功能 • 一般规格 • 安装和配线 • 维护检查 请同时使用用户手册 软件篇（SBSA-CN5-359）。
NJ/NX 系列 CPU 单元 用户手册 软件篇	SBCA-CN5 -359	NX701- □□□□ NJ501- □□□□ NJ301- □□□□ NJ101- □□□□	希望了解 NJ/NX 系列 CPU 单元的编程 / 系统启动时。 与软件相关的信息为主。	对 NJ/NX 系列 CPU 单元相关的如下内容进行说明。 • CPU 单元的动作 • CPU 单元的功能 • 初始设定 • 基于 IEC 61131-3 的语言规格和编程 请同时使用用户手册 硬件篇（SBCA-CN5-418 或 SBCA-CN5-358）。
NJ/NX 系列 指令基准 手册 基本篇	SBCA-CN5 -360	NX701- □□□□ NJ501- □□□□ NJ301- □□□□ NJ101- □□□□	希望了解 NJ/NX 系列的基本指令规格详情时。	对各指令 （IEC 61131-3 规格）的详情进行说明。 编程时请同时使用 “ 用户手册 硬件篇（SBCA-CN5-418 或 SBCA-CN5-358） / 软件篇 （SBSA-CN5-359） ”。
NJ/NX 系列 CPU 单元 用户手册 运动控制篇	SBCE-CN5 -363	NX701- □□□□ NJ501- □□□□ NJ301- □□□□ NJ101- □□□□	希望了解运动控制的设定及编程思路时。	对用于运动控制的 CPU 单元的设定、动作及编程思路进行说明。 编程时请同时使用 “ 用户手册 硬件篇（SBCA-CN5-418 或 SBCA-CN5-358） / 软件篇 （SBSA-CN5-359） ”。
NJ/NX 系列 指令基准 手册 运动篇	SBCE-CN5 -364	NX701- □□□□ NJ501- □□□□ NJ301- □□□□ NJ101- □□□□	希望了解运动指令规格的详情时。	对各运动指令的详情进行说明。 编程时请同时使用用户手册 硬件篇 (SBCA-CN5-418 或 SBCA-CN5-358)/ 软件篇 (SBSA-CN5-359)、用户手册 运动控制篇 (SBCE-CN5-363)”。
NJ/NX 系列 CPU 单元 内置 EtherCAT® 端口 用户手册	SBCD-CN5 -358	NX701- □□□□ NJ501- □□□□ NJ301- □□□□ NJ101- □□□□	使用 NJ/NX 系列 CPU 单元的内置 EtherCAT 端口时。	对内置 EtherCAT 端口进行说明。 对概要、构成、功能、安装进行说明。 请同时使用用户手册 硬件篇（SBCA-CN5-418 或 SBCA-CN5-358） / 软件篇 （SBSA-CN5-359）。

手册名称	Man.No.	型号	用途	内容
NJ/NX 系列 CPU 单元 内置 EtherNet/IP™ 端口 用户手册	SBCD-CN5 -359	NX701- □□□□ NJ501- □□□□ NJ301- □□□□ NJ101- □□□□	使用 NJ/NX 系列 CPU 单元的内置 EtherNet/IP 端口。	对内置 EtherNet/IP 端口进行说明。 对基本设定、标签数据链接、其它功能进行说明。 请同时使用用户手册 硬件篇 （SBCA-CN5-418 或 SBCA-CN5-358） / 软件篇 （SBSA-CN5-359）。
NJ 系列 数据库连接 CPU 单元 用户手册	SBCA-CN5 -411	NJ501-1 □ 20 NJ101- □□ 20	在 NJ 系列中使用数据库连接服务功能时。	对数据库连接服务功能进行说明。
NJ 系列 搭载 SECS/GEM 的 CPU 单元 用户手册	SBCA-CN5 -412	NJ501-1340	在 NJ 系列中使用 GEM 服务功能时。	对 GEM 服务的功能进行说明。
NJ 系列 NJ Robotics CPU 单元 用户手册	SBCA-CN5 -421	NJ501-4 □□□	用 NJ 系列进行机器人控制时。	对机器人控制功能进行说明。
NJ/NX 系列 故障排除手册	SBCA-CN5 -361	NX701- □□□□ NJ501- □□□□ NJ301- □□□□ NJ101- □□□□	希望了解 NJ/NX 系列检出的异常详情时。	对 NJ/NX 系列系统检出的异常的管理思路和各异常项目进行说明。 请同时使用用户手册 硬件篇 （SBCA-CN5-418 或 SBCA-CN5-358） / 软件篇 （SBSA-CN5-359）。
Sysmac Studio Version 1 操作手册	SBCA-CN5 -362	SYSMAC -SE2 □□□	了解 Sysmac Studio 的操作方法、功能。	对 Sysmac Studio 的操作方法进行说明。
NX 系列 EtherCAT® 耦合器单元 用户手册	SBCD-CN5 -361	NX-ECC □□□	希望了解 NX 系列 EtherCAT 耦合器单元及 EtherCAT 从站终端的使用方法时。	对 NX 系列 EtherCAT 耦合器单元和 NX 单元构成的 EtherCAT 从站终端的系统概要、配置方法，以及通过 EtherCAT 对 NX 单元进行设定、控制、监视所需的 EtherCAT 耦合器单元的硬件、设定方法和功能进行说明。
NX 系列 安全控制单元 用户手册	SGFM-CN5 -710	NX-SL □□□□ NX-SI □□□□ NX-SO □□□□	希望了解 NX 单元 安全控制单元的使用方法时。	对 NX 单元 安全控制单元的硬件、设定方法和功能进行说明。
图像处理系统 FH/FZ5 系列 用户手册	SDNB-CN5 -712	FH-1 □□□ FH-3 □□□ FZ5-L35 □ FZ5-6 □□ FZ5-11 □□	希望了解图像处理系统 FH/FZ5 系列的使用方法时。	对使用 FH/FZ5 系列所需的软件功能、设定和操作进行说明。
视觉传感器 FQ-M 系列 定位专用视觉传感器 用户手册	SDNB-CN5 -706	FQ-MS12 □	希望了解定位专用视觉传感器的使用方法时。	对定位专用视觉传感器的硬件、设定方法和功能进行说明。
视觉传感器 FZ3 系列 用户手册	SCHB-CN5 -762	FZ3- □□□□	希望了解视觉传感器 FZ3 系列的使用方法时。	对视觉传感器 FZ3 系列的软件功能、设定和操作进行说明。
位移传感器 ZW 系列 光纤同轴位移传感器 用户手册	SDNE-CN5 -703	ZW-CE1 □	希望了解位移传感器 ZW 系列的使用方法时。	对位移传感器 ZW 系列的硬件、设定方法和功能进行说明。
CJ 系列 高功能单元 用户手册 （NJ 系列连接篇）	SBCC-CN5 -846	CJ1W-AD □□□ CJ1W-DA □□□ CJ1W-MAD42	希望了解在 NJ 系列 CPU 单元上使用 CJ 系列单元的方法时。	对在 NJ 系列 CPU 单元中使用 CJ 系列单元的方法 （访问方法、用户程序 I/F 等）及注意点进行说明。 包括以下单元的手册。 模拟量 I/O 单元、绝缘型模拟量单元、温控单元、 ID 传感器单元、高速计数器单元、串行通讯单元、 DeviceNet 单元、 EtherNet/IP 单元、CompoNet 主站单元 请同时使用用户手册 硬件篇 （SBCA-CN5-358） / 软件篇 （SBSA-CN5-359）。
	SBCC-847	CJ1W-TC □□□		
	SBCC-848	CJ1W-CT021		
	SBCC-849	CJ1W-PDC15 CJ1W-PH41U CJ1W-AD04U		
	SBCD-CN5 -353	CJ1W-CRM21		
	SBCD-CN5 -354	CJ1W-SCU □□		
	SBCD-CN5 -355	CJ1W-EIP21		
	SBCD-CN5 -357	CJ1W-DRM21		
	SDGR-CN5 -703	CJ1W-V680 □□□		

手册名称	Man.No.	型号	用途	内容
可编程终端 NA 系列 用户手册 软件篇	SBSA-CN5-546	NA5- □ W □□□□	希望了解可编程终端 NA 系列的页面及各项目的功能时。	介绍可编程终端 NA 系列的页面及各项目的功能。
可编程终端 NS 系列 编程手册	SBSA-CN5-512	NS15- □□□□□ NS12- □□□□□ NS10- □□□□□ NS8- □□□□□ NS5- □□□□□	希望了解可编程终端 NS 系列的使用方法时。	介绍可编程终端 NS 系列的设定方法和功能。
CX-Designer 用户手册	SBSA-CN5-532	—	希望创建可编程终端 NS 系列的画面数据时	对 CX-Designer 的操作方法进行说明。

术语说明

术语	说明
AT （分配目标）指定	变量的属性之一。 该属性表示由用户指定变量的分配目标。可指定 I/O 端口或 CJ 单元用存储器的地址。
CJ 系列单元	在 CJ 系列的各种单元中，可使用 NJ 系列控制器者。
CJ 单元用存储器	NJ 系列 CPU 单元拥有的 I/O 存储器的一种，带地址。 只能通过 “ 有 AT （分配目标）指定 ” 属性的变量访问。访问 CJ 系列单元及 CJ 网络时使用。
CPU 单元	机器自动化控制器的控制中心。进行任务的执行、各单元和从站的 I/O 刷新等。 NJ/NX 系列中，为 NX701- □□□□、 NJ501- □□□□、 NJ301- □□□□。
EtherCAT 主机功能模块	功能模块之一。作为 EtherCAT 主机控制 EtherCAT 从站。
EtherNet/IP 功能模块	功能模块之一。控制 EtherNet/IP 端口。
FB	功能块的简称。
FUN	函数的简称。
I/O 端口	CPU 单元与外部设备 （从站 / 单元）进行数据交换所需的逻辑接口。
I/O 映射设置	对 I/O 分配变量的设定。 I/O 端口和变量之间的分配信息。
I/O 刷新	事先确定的存储器区域和外部进行周期性数据交换。
MC 调试	通过 Sysmac Studio 确认电机动作及配线的功能。
NX 单元	在 NX 系列的单元中，执行外部连接设备的 I/O 处理的单元。通信耦合器单元不包含在 NX 单元中。
PDO 通信	过程数据通信的简称。按过程数据通信周期 （＝主固定周期任务的任务周期），主机 — 从站之间周期性进行数据交换。
PLC 功能模块	功能模块之一。进行用户程序的执行、向运动控制功能模块发出指示，以及与 USB・SD 存储卡的接口。
POU	Program Organization Unit。 IEC 61131-3 中程序执行模式的单位。 作为构成整个用户程序的基本单位，对算法和本地变量表进行记述。 包括程序、功能块、函数三种。
SDO 通信	EtherCAT 通信的一种，使用可在任意时间传输信息的服务数据对象 （Service Data Objects: SDO）。
Sysmac Studio	计算机用软件，进行 NJ/NX 系列的设定、编程、调试及故障排除。包括运动控制用的功能及模拟功能。
上传	利用 Sysmac Studio 的 “ 同步 ” 功能，从控制器向 Sysmac Studio 传送数据。
一般信息	表示控制器信息及用户信息的事件重要程度的等级之一。并非异常，但仍记录到事件日志中，并向用户通知的信息。
事件日志功能	识别事件 （异常或现象），并记录下来。
事件设定	定义用户异常或用户信息。
事件任务	任务的执行条件成立时，执行一次用户程序。
联机 ST	将 ST 程序作为梯形图程序的一部分写入其中。或者是所写入的 ST 程序。
边缘	变量的属性之一。 BOOL 型变量的值从 FALSE 上升为 TRUE 时，或从 TRUE 下降为 FALSE 时，向功能块传递 TRUE 值。
凸轮数据变量	将凸轮数据转换为结构体排列后的变量。 结构体型排列指定的变量，由相位和变位数据组成。

术语	说明
监视信息	表示控制器信息及用户信息的事件重要程度的等级之一。 对控制没有影响的轻微异常，但仍记录到事件日志中，并向用户通知的信息
功能模块	按功能分割 CPU 单元软件构成的单位。
基本数据型	IEC 61131-3 定义的数据类型。 包括布尔型、位列型、整数型、实数型、持续时间型、日期型、时刻型、日期时刻型、字符串型。 与用户定义的数据类型 —“ 派生数据型 ” 相对的术语。
强制值刷新	用户进行调试等时，以强制值刷新来自外部的输入或向外部的输出。 即使是强制值刷新对象的区域，也可通过用户程序的指令覆盖。
联合体型	派生数据型的一种，使同一数据可以作为不同的数据类型处理。
全局变量	可从所有 POU （程序、功能块、函数）读写的变量。
轻度故障等级的控制器异常	NJ/NX 系列控制器的某些功能模块的部分功能无法控制的异常。 发生轻度故障后， NJ/NX 系列 CPU 单元将继续运行。
高功能单元设定	高功能单元用分配 DM 及单元固有设定的总称。
结构体型	派生数据型的一种，将多个不同的数据类型整合为一个。
常量	变量的属性之一。 变量指定为常量后，将无法通过指令、 ST 语言的运算符或 CIP 信息通信进行值的写入。
控制器	CPU 单元直接控制的范围。 NX 系列系统中，为 CPU 装置 +EtherCAT 从站 （包括通用 / 伺服驱动器）。 NJ 系列系统中，为 CPU 装置 + 扩展装置 +EtherCAT 从站 （包括通用 / 伺服驱动器）。
控制器异常	NJ/NX 系列系统中定义的异常。 “ 全部停止故障等级 ”、 “ 部分停止故障等级 ”、 “ 轻度故障等级 ”、 “ 监视信息 ” 的控制器事件的总称。
控制器事件	NJ/NX 系列系统的一种事件。系统定义为需要通知用户的异常、信息。系统检测到定义的发生原因后，发生控制器事件。
控制器信息	NJ/NX 系列系统中定义的信息，但并非异常。属于 “ 一般信息 ” 的控制器异常。
伺服 / 编码器输入从站	EtherCAT 从站中，可向 “ 轴 ” 分配的从站。 NJ/NX 系列中，指伺服驱动器和编码器输入从站。
轴	运动控制功能模块中的功能单位。分配外部伺服驱动的驱动机构、编码器输入从站的检测机构等。
轴组	运动控制功能模块中，汇集了多个 “ 轴 ” 的功能单位。
轴组变量	结构体中定义的系统定义变量，如各轴组的各种状态信息、部分轴组参数设定信息等。 用于运动控制指令的轴组指定及轴组的指令插补速度、异常信息等的监视。
轴变量	结构体中定义的系统定义变量，如各轴的各种状态信息、部分轴参数设定信息等。 用于运动控制指令的轴指定及轴的指令当前位置、异常信息等的监视。
系统共通处理	与 I/O 刷新或用户程序执行一起， CPU 单元在任务内部执行的系统处理。执行变量的任务间排他处理、数据追踪处理等。
系统服务	CPU 单元在任务处理的空闲时间执行的处理。执行通信处理、 SD 存储卡访问、自诊断处理等。

术语	说明
系统定义变量	所有属性由系统定义，用户绝对无法变更的变量。
初始值	变量的属性之一。以下情况下设定的变量值。 • 电源接通时 • 切换到运行模式时 • 指定为传送用户程序时初始化时 • 发生全部停止故障等级的控制器异常时
从站	执行远程 I/O 的主站 （主机）相对的子站。
从站终端	在通信耦合器单元上安装 NX 单元群的构件块型远程 I/O 单元的总称。从站终端是从站的一种。
从站 / 单元构成	EtherCAT 构成及单元构成的总称。
绝对编码器原点位置偏置	CPU 单元中保存的原点确定后的指令位置和从绝对编码器读取的绝对值数据的差，用于恢复带绝对编码器的伺服驱动器的当前位置。
全部停止故障等级的控制器异常	NJ/NX 系列控制器整体无法控制的异常。 CPU 单元将立即停止执行用户程序，断开包括远程 I/O 在内所有从站 / 单元的负载。
下载	利用 Sysmac Studio 的 “ 同步 ” 功能，从 Sysmac Studio 向控制器传送数据。
任务	“ 程序执行时的属性 ”。
任务周期	主固定周期任务及固定周期任务的执行间隔。
通信耦合器单元	NX 单元群和上位网络主机在网络上进行远程 I/O 通信所需的接口总称。例如，EtherCAT 耦合器单元为 EtherCAT 通信网络的通信耦合器单元。
固定周期任务	按一定周期执行 I/O 刷新和用户程序的任务。
设备变量	通过 I/O 端口访问特定设备所需的变量。
同步	自动比较 NJ/NX 系列控制器和 SysmacStudio 的信息，分层显示是否有不同以及不同的位置和内容，并将信息统一
名称空间	分割和分层管理函数、功能块定义、数据类型名称的机制。
网络公开	变量的属性之一。 利用 CIP 信息通信或标签数据链接功能，可在其他控制器或上位计算机上读写变量。
排列指定	变量指定的一种。整合相同数据类型的多个要素后得到的一个变量。每个要素从前往后以编号 （下标）指定。
派生数据型	由用户定义规格的数据类型。包括结构体型、联合体型、列举型。
范围指定	变量指定的一种。明确表示只能提取预先设定范围内的变量值。
通用从站	EtherCAT 从站中，不可向 “ 轴 ” 分配的从站。
功能	运算处理等，编写输出与输入一对一的元件时使用的 POU。
功能块	定时器、计数器等，编写输出与输入因状态不同而不同的元件时使用的 POU。
部分停止故障等级的控制器异常	NJ/NX 系列控制器的某些功能模块整体无法控制的异常。 发生部分停止故障后， NJ/NX 系列 CPU 单元将继续运行。
主固定周期任务	优先度最高的任务。
程序	与函数、功能块并列，三种 POU 之一。 分配到任务后动作。
过程数据通信	EtherCAT 通信的一种，使用以固定周期定进行实时信息交换的过程数据对象（Process Data Objects: PDO）。也称 “PDO 通信 ”。

术语	说明
变量	用户程序中使用的、表示任意数值或字符串的概念。 代入任意值，可变更变量的值。与之相对的是值始终不变的 “ 常数 ”。
变量存储器	无 AT （分配目标）指定的变量之当前值。只能通过 “ 无 AT （分配目标）指定 ” 属性的变量访问。
保持	变量的属性之一。在以下情况下设定了本属性时，将保持变量的值，若未设定，则为初始值。 • 断电后接通电源时 • 切换到运行模式时 • 指定为传送用户程序时不初始化时
指令	POU 的算法中记载的、欧姆龙指定的最小单位的处理要素。 指令中包含梯形图 （接点、线圈）、 FB 型、 FUN 型、 ST 语言的语句。
主存储器	CPU 单元在使用 OS、执行用户程序时使用的 CPU 单元内部的存储器。
运动控制功能模块	功能模块之一。按照用户程序中的运动指令所赋予的指令，执行运动控制、
运动控制指令	执行运动控制功能所需的运动控制定义指令。 运动控制指令中，包括以 PLCopen® 的运动控制用功能块为标准的指令和运动控制功能模块独立的指令。
用户事件	NJ/NX 系列系统的一种事件。由用户定义的事件。用户异常、用户信息的总称。
用户定义变量	所有变量属性由用户定义，可由用户变更的变量。
用户程序	多个程序的集合体。
单元	CPU 装置或扩展装置上可安装的设备。
单元构成	Sysmac Studio 中设定的单元构成信息，包括 CPU 单元上连接的单元种类和连接位置等信息。
文本	用户程序中使用的常数表现。
列举型	派生数据型的一种。将事先准备的名称列表 （列举值）中的一个作为值的数据类型。
列举值	将列举型变量可得到的多个值以文字的形式表示。 列举型变量的值取列举值中的任意一个。
本地变量	只能从定义的 POU 内部访问的变量。与之相对的是可从多个 POU 访问的 “ 全局变量 ”。 本地变量中包括内部变量、输入变量、输出变量、输入输出变量、外部变量。

手册修订记录

手册修订符号在封面和封底中记载的 Man.No. 末尾标注。

修订符号	修订年月	修订理由和修订页码
A	2011 年 7 月	第一版
B	2012 年 3 月	• 追加 NJ301- □□□□ • 追加 CPU 单元版本 1.01 以上的支持功能相关的记述 • 误记修正
C	2012 年 5 月	• 追加 CPU 单元版本 1.02 以上的支持功能相关的记述 • 误记修正
D	2012 年 8 月	• 追加 CPU 单元版本 1.03 以上的支持功能相关的记述 • 误记修正
E	2013 年 2 月	• 追加 CPU 单元版本 1.04 以上的支持功能相关的记述 • 误记修正
F	2013 年 4 月	• 追加 CPU 单元版本 1.05 以上的支持功能相关的记述 • 追加 NX 系列相关的记述 • 误记修正
G	2013 年 6 月	• 追加 CPU 单元版本 1.06 以上的支持功能相关的记述 • 误记修正
H	2013 年 9 月	• 追加 CPU 单元版本 1.07 以上的支持功能相关的记述 • 误记修正
J	2013 年 12 月	• 追加 CPU 单元版本 1.08 以上的支持功能相关的记述 • 误记修正
K	2014 年 7 月	• 误记修正
L	2015 年 1 月	• 追加 CPU 单元版本 1.10 以上的支持功能相关的记述 • 误记修正
M	2015 年 4 月	• 追加 NX701- □□□□、 NJ101- □□□□ • 误记修正
N	2015 年 10 月	• 追加硬件修订版本相关的记述 • 误记修正
P	2016 年 4 月	• 追加 CPU 单元版本 1.11 以上的支持功能相关的记述 • 误记修正

1

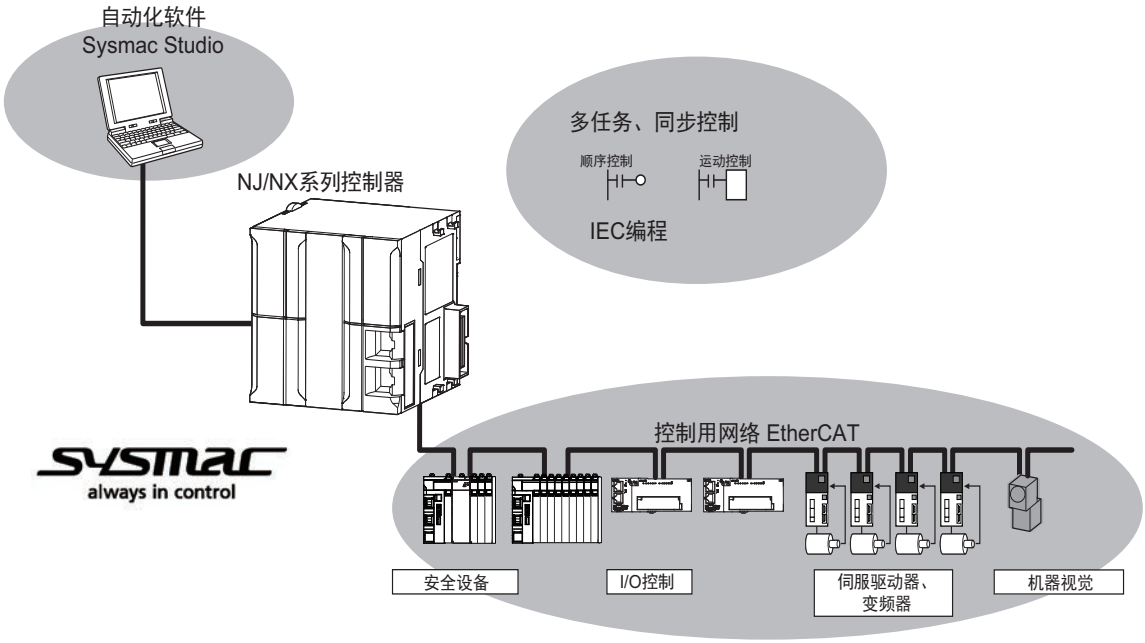
NJ/NX 系列控制器的概要

本章介 NJ/NX 系列控制器的特点、系统构成的概要、规格、使用步骤。

1-1 NJ/NX 系列控制器定义 1-2
1-1-1 特点 1-2
1-1-2 系统构成概要 1-5
1-2 规格概要 1-9
1-3 NJ/NX 系列的整体使用步骤 1-12
1-3-1 步骤概要 1-12
1-3-2 步骤详情 1-13

1-1 NJ/NX 系列控制器定义

机器自动化控制器 NJ 系列是一款兼顾机械控制必需的功能、高速性能和商用控制器所需的安全性、可靠性、维护性的新型控制器。
包含传统可编程控制器的功能，还附带运动控制所需的而各种功能，是一款综合型控制器。可在高速 EtherCAT 上同步控制安全设备、视觉装置、运动机器、 I/O 等输入输出设备。
欧姆龙提供以统一的通信规格和用户接口规格设计的欧姆龙控制设备 ——Sysmac 设备。机器自动化控制器 NJ/NX 系列可与 EtherCAT 从站等 Sysmac 设备以及自动化软件 Sysmac Studio 组合，实现最佳的性能和操作性。
在由 Sysmac 设备构成的系统中，机器与机器的连接非常方便，能够以统一的思路和实用工具操作。



1-1-1 特点

硬件的特点

- **标配控制用网络 EtherCAT**

  CPU 单元标配 EtherCAT 通信主机功能端口。
  EtherCAT 是一种以 Ethernet 系统为基础的高性能商用网络系统，可实现更高速、更高效的通信。各节点高速传输 Ethernet 帧，可实现恒定的短周期通信周期时间。
  采用控制网络 EtherCAT，可通过一个网络连接机器控制所需的 I/O 系统、伺服驱动器、变频器、机器视觉等设备。

- **支持 EtherCAT 从站终端**

  使用 EtherCAT 从站终端，节省安装空间。此外，利用种类丰富的 NX 单元，可灵活地构建系统。

● 在 EtherCAT 上实现安全子系统

通过使用 NX 系列 安全控制单元，可在顺序控制、运动控制系统上，将安全控制作为 EtherCAT 上的子系统整合起来。

版本相关信息

NX系列安全控制单元可在CPU单元版本1.06以上和Sysmac Studio Ver.1.07以上的组合中使用。

● 支持 CJ 系列单元

在 NJ 系列 CPU 单元中，除了 EtherCAT 网络的各种从站以外，还可在 I/O 总线系统上安装各种 CJ 系列单元 （基本 I/O 单元、高功能单元）使用。

● 标配 EtherNet/IP 通信功能端口

CPU 单元中标配 EtherNet/IP 通信功能端口。
EtherNet/IP 为使用 Ethernet 的多供应商网络。可作为控制器之间的网络或现场网络使用。使用标准 Ethernet 技术，可与各种通用 Ethernet 设备一起使用。

● 标配 USB 端口

计算机上安装的支持软件可通过 USB 直接与 CPU 单元连接。

● 标配 SD 存储卡槽

可通过用户程序访问 CPU 单元中安装的 SD 存储卡。

● 高硬件可靠性

备有与 PLC 同等的硬件可靠性和 RAS 功能。

● 利用多核处理器并列执行

CPU 单元 NX701- □□□□配备有多核处理器，可并列执行多个任务。因此，大规模设备也可实现高速控制。

软件的特点

● 整合顺序控制和运动控制

一台 CPU 单元覆盖实现顺序控制和运动控制。因此，可同时实现顺序控制和多轴同步控制。在同一控制周期中，执行顺序控制、运动控制和 I/O 刷新。
此外，控制周期与 EtherCAT 的过程数据通信周期一致。因此，可实现周期恒定、波动小、精度高的顺序控制及运动控制。

● 支持多任务

将 I/O 刷新或用户程序执行等一系列处理分配到多个任务，并分别指定执行条件和执行顺序，再将它们组合起来，可与应用程序一起灵活地构建控制系统。

- **国际标准规格 IEC 61131-3 标准的编程语言规格**

  配备符合 IEC 61131-3 的语言规格。部分经欧姆龙独家改良。备有 PLCopen® 标准的运动控制指令以及符合 IEC 规则的丰富的指令群 （POU）。

- **无需在意存储器映射，通过变量编程**

  与在电脑上使用高级语言一样，所有数据都通过变量访问。创建的变量可自动分配到 CPU 单元的存储器上，用于无需在意。

- **丰富的安全功能**

  备有丰富的安全功能，可通过操作权限的设定、 ID 来限制用户程序的执行。

- **控制器整体的监视功能**

  CPU 单元可对安装的单元、包括 EtherCAT 从站在内的控制器整体现象 （事件）进行监视。
  异常时的处理方法通过 Sysmac Studio 或 NS 系列显示器引导。发生的现象 （事件）将保存为记录。

- **自动化软件 Sysmac Studio**

  Sysmac Studio 是一个综合开发环境，能用一个软件覆盖控制器、外围设备或 EtherCAT 上的设备。对不同机器，可提供统一的操作性。从设计到调试、模拟、启动、开始使用后的变更等，可对所有工序提供支援。

- **丰富的模拟功能**

  备有丰富的模拟功能，包括虚拟控制器的执行功能、调试功能、任务执行时间的验证功能等。

1-1-2 系统构成概要

下面介绍 NX 系列和 NJ 系列的系统构成概要。

NX 系列的系统构成概要

NX 系列的系统构成如下所示。

● 基本构成

NX 系列的基本构成为 EtherCAT 网络配置和支持软件。

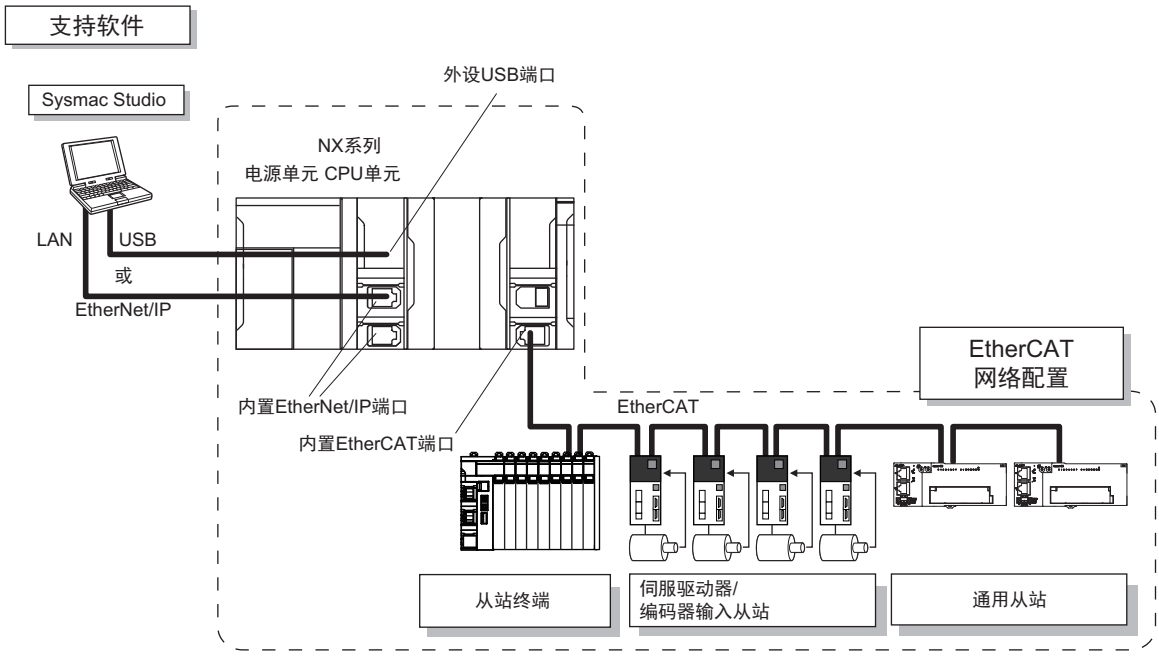
- EtherCAT 网络配置

  使用内置 EtherCAT 端口，可连接 EtherCAT 从站终端、数字 I/O、模拟 I/O 等通用从站以及伺服 / 编码器输入从站。使用这样的构成，可实现周期恒定、波动小、精度高的顺序控制及运动控制。

- 支持软件

  可将市售的 USB 电缆连接到 CPU 单元的外设 USB 端口上。或者将 Ethernet 电缆连接到内置 EtherNet/IP 端口上。

  支持软件的详细连接构成请参考📖「10-2 Sysmac Studio 的连接」(P.10-5)。



使用注意事项

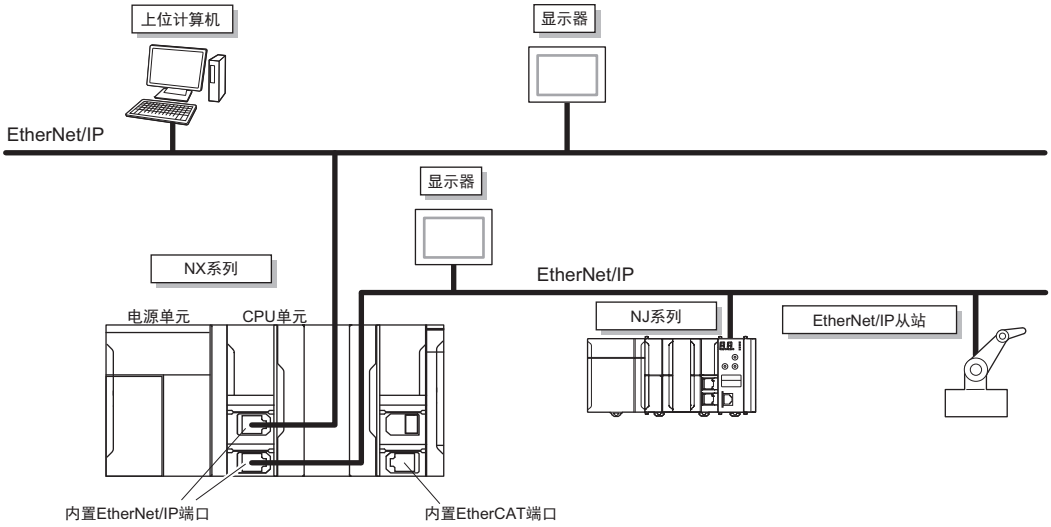
NX 单元与从站中断连接后使用。 CPU 单元的 NX 总线连接器用于将来扩展所需，因此不可连接 NX 单元使用。

参考

可在通信耦合器单元上直接连接 Sysmac Studio，进行从站终端的设定。详情请参考📖《NX 系列 EtherCAT 耦合器单元 用户手册 （SBCD-CN5-361）》。

● 网络配置

- 在内置 EtherNet/IP 端口上连接上位计算机、显示器、 NJ/NX 系列。
- NX 系列 CPU 单元中有两个 EtherNet/IP 端口。



网络配置的详情请参考📖「第 10 章 通信设定」(P.10-1)。

● 支持软件

进行 NX 系列的设定 / 监视 / 调试的编程设备如下所示。

- Sysmac Studio

  NX 系列中，基本使用 Sysmac Studio 作为控制器配置和设定、程序创建、调试、模拟的支持软件。

- 其他支持软件

  除了 Sysmac Studio 以外，软件包 Sysmac Studio 标准版中还包含以下软件。

构成软件	用途
Sysmac Studio	包括顺序控制及运动控制在内，使用除以下外的所有功能时使用。
Network Configurator	通过 EtherNet/IP 端口或单元进行标签数据链接时使用。*1
CX-Integrator	在 DeviceNet 单元和 CompoNet 主站单元之间进行远程 I/O 通信时使用。
CX-Protocol	在串行通讯单元上使用协议宏功能时使用。
CX-Designer	创建 NS 系列显示器画面时使用。

*1. 目标设备为 NJ/NX 系列控制器时， Sysmac Studio Ver.1.10 以下中也可设定。 CS/CJ 系列 PLC 为始发设备时，在 Network Configurator 上设定。

NJ 系列的系统构成概要

NJ 系列的系统构成如下所示。

● 基本构成

NJ 系列的基本构成为 EtherCAT 网络配置、 CJ 单元构成及支持软件。

- EtherCAT 网络配置

  使用内置 EtherCAT 端口，可连接 EtherCAT 从站终端、数字 I/O、模拟 I/O 等通用从站以及伺服 / 编码器输入从站。使用这样的构成，可实现周期恒定、波动小、精度高的顺序控制及运动控制。

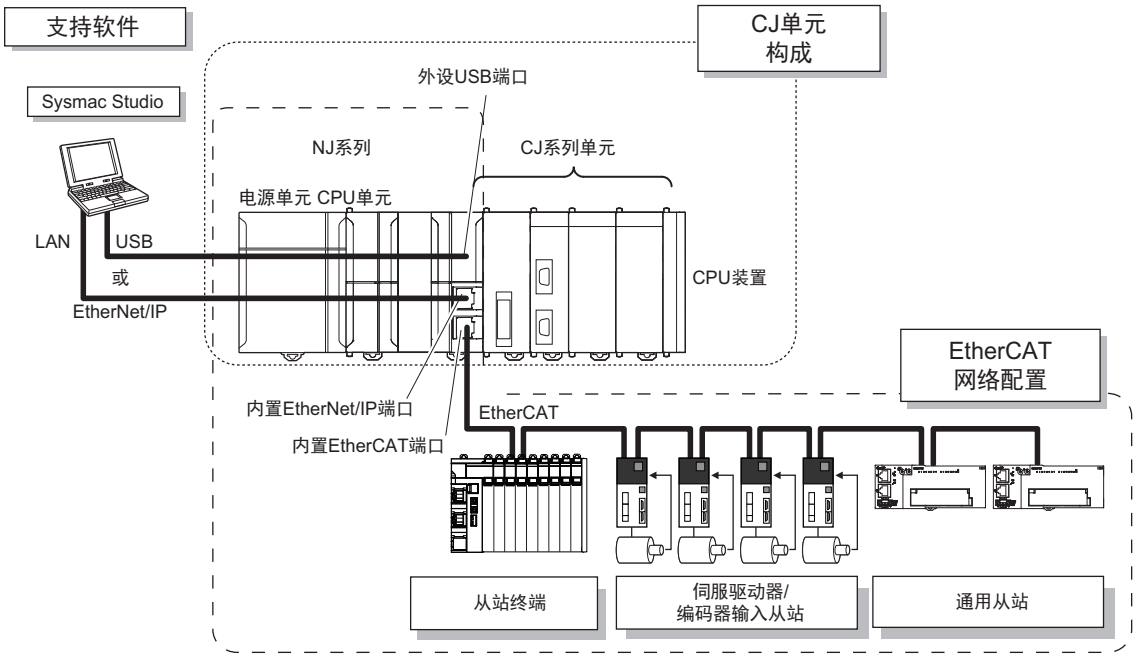
- CJ 单元构成

  除了 EtherCAT 网络以外，还可安装 CJ 系列单元 （基本 I/O 单元、高功能单元）。除了安装有 CPU 单元的 CPU 装置以外，还可使用扩展装置增设 CJ 系列单元。

- 支持软件

  可将市售的 USB 电缆连接到 CPU 单元的外设 USB 端口上。或者将 Ethernet 电缆连接到内置 EtherNet/IP 端口上。

  支持软件的详细连接构成请参考📖「10-2 Sysmac Studio 的连接」 (P.10-5)。

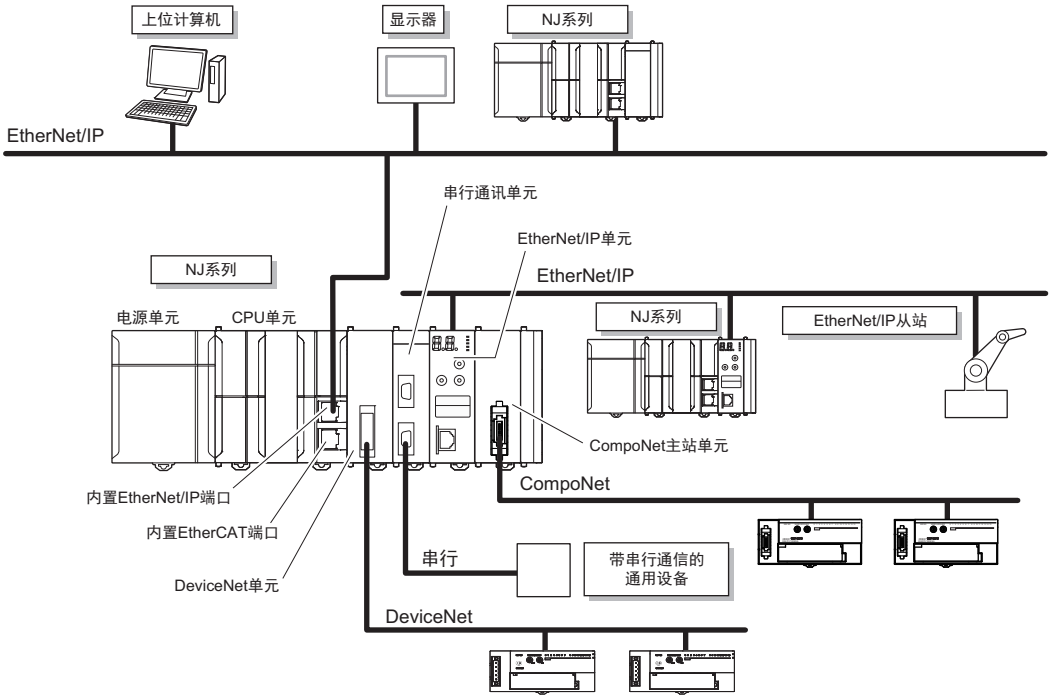


参考

可在通信耦合器单元上直接连接 Sysmac Studio，进行从站终端的设定。详情请参考📖《NX 系列 EtherCAT 耦合器单元 用户手册 （SBCD-CN5-361）》。

● 网络配置

- 在内置 EtherNet/IP 端口或 EtherNet/IP 单元（CJ1W-EIP21）上连接上位计算机、显示器、NJ 系列。
- DeviceNet 与 DeviceNet 单元连接，CompoNet 与 CompoNet 单元连接，使用串行通信时，应连接串行通讯单元。



网络配置的详情请参考📖「第 10 章 通信设定」(P.10-1)。

● 支持软件

进行 NJ 系列的设定 / 监视 / 调试的编程设备如下所示。

- Sysmac Studio

  NJ 系列中，基本使用 Sysmac Studio 作为控制器配置和设定、程序创建、调试、模拟的支持软件。
- 其他支持软件

  除了 Sysmac Studio 以外，软件包 Sysmac Studio 标准版中还包含以下软件。

构成软件	用途
Sysmac Studio	包括顺序控制及运动控制在内，使用除以下外的所有功能时使用。
Network Configurator	通过 EtherNet/IP 端口或单元进行标签数据链接时使用。*1
CX-Integrator	在 DeviceNet 单元和 CompoNet 主站单元之间进行远程 I/O 通信时使用。
CX-Protocol	在串行通讯单元上使用协议宏功能时使用。
CX-Designer	创建 NS 系列显示器画面时使用。

*1. 目标设备为 NJ/NX 系列控制器时， Sysmac Studio Ver.1.10 以下中也可设定。 CS/CJ 系列 PLC 为始发设备时，在 Network Configurator 上设定。

1-2 规格概要

下面介绍 NJ/NX 系列的主要性能规格。一般规格、性能规格、功能规格请参考📖「A-1 规格」(P.A-3)。

项目				NX701-		NJ501-			NJ301-		NJ101-	
				17 □□	16 □□	□5□□	□4□□	□3□□	12 □□	11 □□	10 □□	90 □□
编程	程序容量[*1]	大小		80MB		20MB			5MB		3MB	
		数量	POU 定义数	6,000		3,000			750		450	
			POU 实例数	48,000		9,000(*)			3,000(*)		1,800	
	变量容量	有保存属性[*2]	大小	4MB		2MB			0.5MB			
			变量数量	40,000		10,000			5,000(*)			
		无保存属性[*3]	大小	256MB		4MB			2MB			
			变量数量	360,000		90,000			22,500			
	数据类型	数据类型的数量		8,000		2,000			1,000			
	CJ 单元用存储器（可通过变量的 AT 指定进行指定）	通道 I/O（CIO）		—		6144 通道（0 ～ 6143）						
		工件继电器（WR）		—		512 通道（W0 ～ W511）						
		保持继电器（HR）		—		1536 通道（H0 ～ H1535）						
		数据存储器（DM）		—		32768 通道（D0 ～ D32767）						
		扩展数据存储器（EM）		—		32768 通道 ×25BANK（E0_00000 ～ E18_32767）			32768 通道 x 4BANK（E0_00000 ～ E3_32767）			
运动控制	控制轴数	最大控制轴数[*4]		256 轴	128 轴	64 轴	32 轴	16 轴	15 轴 (*)		6 轴	—
		最大使用实轴数[*5]		256 轴	128 轴	64 轴	32 轴	16 轴	8 轴	4 轴	2 轴	—
		最大单轴控制数		256 轴	128 轴	64 轴	32 轴	16 轴	15 轴 (*)		6 轴	—
		直线插补控制最大数		每 1 轴组 4 轴								—
		圆弧插补控制轴数		每 1 轴组 2 轴								—
	轴组最大数			64 组		32 组						—
	运动控制周期			与 EtherCAT 通信的过程数据通信周期相同								—
	凸轮	凸轮数据点数	每 1 凸轮表的最大点数	65,535 点								—
			所有凸轮表的最大点数	1,048,560 点					262,140 点			—
		凸轮表最大表数		640 表					160 表			—
外设（USB）端口	支持服务			Sysmac Studio 连接								
	物理层			USB2.0 标准 B 型连接器								
	传送距离			最大 5m								

项目			NX701-		NJ501-			NJ301-		NJ101-	
			17 □□	16 □□	□5□□	□4□□	□3□□	12 □□	11 □□	10 □□	90 □□
内置 EtherNet/IP 端口	端口数量		2		1						
	物理层		10BASE-T/100BASE-TX/1000BASE-T		10BASE-T/100BASE-TX						
	帧长度		最大 1,514 字节								
	介质存取方式		CSMA/CD								
	调制方式		基带								
	拓扑结构		星形								
	传送速度		1Gb/s（1000BASE-T）		100Mb/s （100BASE-TX）						
	传送介质		双绞线电缆 （带屏蔽：STP）: 类别 5、 5e 以上								
	传输距离 （集线器和节点之间的距离）最大值		100m								
	串联连接最大数量		使用交换式集线器时无限制								
	CIP 服务：标签数据链接 （周期链接通信）	最大连接数	256 / 端口 总计 512		32						
		Packet 间隔*6	各连接可设定 0.5 ～ 10000ms （以 0.5ms 为单位）		各连接可设定 1 ～ 10000ms （以 1ms 为单位） (*)						
		单元允许通信带宽	40,000 pps*7 （含 Heartbeat）		3,000 pps*7 （含 Heartbeat） (*)						
		最大标签集数	256 / 端口 总计 512		32						
		标签种类	网络变量		网络变量、 CIO / WR / HR / DM / EM						
		1 连接 （=1 标签集）的标签数	8 （标签集中包含控制器状态时为 7）								
		标签最大数量	256 / 端口 总计 512		256						
		每 1 节点的最大链接数据容量 （所有标签的总大小）	369,664 字节		19,200 字节						
		每 1 连接的最大数据容量	1,444 字节		600 字节						
		最大可登录的标签集数	256 / 端口 总计 512 （1 连接 =1 标签集）		32 （1 连接 =1 标签集）						
		1 标签集的最大容量	1,444 字节 （标签集中包含控制器状态时使用 2 字节）		600 字节 （标签集中包含控制器状态时使用 2 字节）						
		多点传送 Packet 过滤功能*8	可								
内置 EtherCAT 端口	通信标准		IEC 61158 Type12								
	EtherCAT 主站规格		符合 Class B （符合 Feature Pack Motion Control）								
	物理层		100BASE-TX								
	调制方式		基带								
	传送速度		100Mb/s （100BASE-TX）								
	Duplex 模式		Auto								
	拓扑结构		线、菊花链、分支布线								
	传送介质		类别 5 以上　双绞线电缆 （推荐铝带编织双重隔离屏蔽电缆、直线型）								
	节点间距离最大值		100m								
	最大从站数		512		192					64	

项目				NX701-		NJ501-			NJ301-		NJ101-	
				17 □□	16 □□	□5□□	□4□□	□3□□	12 □□	11 □□	10 □□	90 □□
单元构成	可安装的单元数	每 1 装置 （CPU 装置或扩展装置）中 CJ 单元的最大数		—		10						
		系统整体的 CJ 单元最大数		—		40						
		系统整体的 NX 单元最大数		4,000 （EtherCAT 从站终端上）							400 （EtherCAT 从站终端上）	
	最大扩展装置数			0		3						
	输入输出点数	CJ 单元最大输入输出点数		—		2,560						
	电源 （CPU 装置或扩展装置）	型号		NX-PA9001 NX-PD7001		NJ-P □ 3001						
		断电确定时间	AC 电源	30 ～ 45ms								
			DC 电源	5 ～ 20ms		22 ～ 25ms						

*1. 执行对象、变量表 （变量名等）的容量。
*2. 不含 CJ 单元用存储器的 HR、 DM、 EM。
*3. 不含 CJ 单元用存储器的 CIO、 WR。
*4. 所有轴种类的合计。
*5. 轴种类为 ［伺服轴］或 ［编码器轴］，且轴使用设定为 ［使用轴］的轴合计。
*6. 与节点数无关，以设定周期更新线路中的数据。
*7. pps 代表 Packet Per Second，表示 1 秒内可处理的收发信息包数。
*8. EtherNet/IP 端口安装了 IGMP 客户端，使用了支持 IGMP Snooping 的交换式集线器，所以会进行不必要的多点传送数据包的过滤。

（注） 表中带（*）的项目为版本升级后性能得到改善的功能。升级相关的信息请参考📖「A-13 NX 系列的版本相关信息」(P.A-177)及📖「A-14 NJ 系列的版本相关信息」 (P.A-180)。

1-3 NJ/NX 系列的整体使用步骤

下面介绍 NJ/NX 系列整体步骤的概要和详细信息。

1-3-1 步骤概要

NJ/NX 系列的使用流程如下。

STEP 1. 软件设计

设计系统的整体构成、任务构成 / 程序 / 变量。

- STEP1-1 设计 I/O 和处理
- STEP1-2 设计任务
- STEP1-3 设计程序

↓

STEP 2. 软件设定和编程

用支持软件创建 STEP1 中设计的系统构成，同时执行变量分配。然后，创建任务及程序，进行模拟等调试。

- STEP2-1 从站 / 单元构成
- STEP2-2 控制器设定
- STEP2-3 编程
- STEP2-4 离线调试

↓

STEP 3. 安装和硬件的设定

安装各种单元，并根据需要设定硬件。

↓

STEP 4. 配线

进行网络电缆或 I/O 的配线。

↓

STEP 5. 上机确认动作和运行

连接支持软件和实际系统，下载项目。上机确认动作，开始运行。

1-3-2 步骤详情

STEP1．软件设计

步骤	内容	参考
STEP1-1 **设计 I/O 和处理**	• 外部 I/O 和单元的构成 • 外部设备和刷新周期 • 程序的处理内容	《NX/NX 系列 CPU 单元 用户手册 硬件篇（SBCA-CN5-418）》 《NJ 系列 CPU 单元 用户手册 硬件篇（SBCA-CN5-358）》
STEP1-2 **设计任务**	• 任务构成 • 任务 — 程序的关系 • 各任务的周期 • 从站 / 单元的刷新周期 • 任务间变量的排他方法	「4-2-3 任务设置」(P.4-6)
STEP1-3 **设计程序**		
设计 POU （处理的分割单位）	• 程序 • 函数 / 功能块 • 选择各算法的语言	「第 6 章 编程」(P.6-1)
设计变量	• 区分 POU 共同使用的变量和各 POU 单独使用的变量 • 设计用于访问从站 / 单元的设备变量之变量名 • 设计变量的名称、保持等属性 • 设计变量的数据类型	「6-3 变量」(P.6-28)

STEP2．软件设定和编程

步骤	内容	Sysmac Studio 上的操作	参考
新建项目	1. 用 Sysmac Studio 新建项目。 2. 插入控制器。	[新建项目] 按钮 [插入] ｜ [控制器]	《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》

以下 “ 控制器配置和设定 ” 及 “ 编程 / 任务设置 ” 的顺序中，可任选。

STEP2-1 **从站 / 单元构成**			
1）从站 / 单元构成的创建	1. 离线或在线创建从站 / 单元构成。（在线时，先执行 STEP 5 的在线连接） 2. 使用从站终端时，应设定从站终端。	EtherCAT 从站设定编辑器 单元编辑器	「3-2-1 EtherCAT 从站构成的创建」(P.3-4) 「3-2-2 单元构成的创建」(P.3-6) 《NX 系列 EtherCAT 耦合器单元 用户手册（SBCD-CN5-361）》

2）向 I/O 端口分配设备变量	将设备变量登录到变量表中 （变量名由用户定义或自动生成）	I/O 映射	「3-3 I/O 端口和设备变量」(P.3-8)

（进行运动控制时）

3）“ 轴 ” 的创建和向伺服 / 编码器输入从站分配	创建 “ 轴 ”，分配到实轴或虚拟轴。 插补轴控制时，创建 “ 轴组 ”。	配置和设定丨在运动控制设定中设定	「3-5 “ 轴 ” 的创建和伺服驱动器 / 编码器输入从站的分配」(P.3-17)

STEP2-2 控制器设定	在 Sysmac Studio 中设定以下参数。		「第 4 章 控制器设定」(P.4-1)
	PLC 功能模块相关的初始设定	配置和设定丨在动作设定中设定	「4-2 PLC 功能模块的初始设定」(P.4-4)
	（NJ 系列时） 高功能单元的初始设定	配置和设定丨在 CPU・扩展装置中设定	「4-3 高功能单元的初始设定」(P.4-13)
	（进行运动控制时） 运动控制功能模块相关的初始设定	配置和设定丨在运动控制设定中设定	「4-4 运动控制功能模块的初始设定」(P.4-15)
	EtherCAT 主机功能模块 相关的初始设定	配置和设定丨在 EtherCAT 中设定	「4-5 EtherCAT 主机功能模块的初始设定」(P.4-17)
	EtherNet/IP 功能模块相关的初始设定	配置和设定丨在内置 EtherNet/IP 端口设定中设定	「4-6 EtherNet/IP 功能模块的初始设定」(P.4-18)

STEP2-3 编程			
1）变量的登录	• 在 Sysmac Studio 中，向全局变量表登录 POU 共通的变量 • 登录各程序的本地变量表 • 登录各功能块、函数的本地变量表	全局变量表编辑器 本地变量表编辑器	『Sysmac Studio Version 1 操作手册（SBCA-CN5-362）』 「6-3 变量」(P.6-28)
2）创建 POU 的算法	用各语言创建 POU （程序 / 功能块 / 函数）的算法	程序编辑器	「第 6 章 编程」(P.6-1) 《NJ/NX 系列 指令基准手册 基本篇（SBCA-CN5-360）》、 《NJ/NX 系列 指令基准手册 运动篇（SBCE-CN5-364）》
3）任务设置	进行任务相关的设置。	在任务设置中设置	「4-2-3 任务设置」(P.4-6)

STEP2-4 离线调试	在模拟器 （虚拟控制器）上 确认算法和任务执行时间		「第 7 章 动作确认和运行」(P.7-1)

STEP3． 安装和硬件的设定

步骤	内容	参考
1）安装	• 单元相互连接 • 安装到 DIN 导轨上	《NX/NX 系列 CPU 单元 用户手册 硬件篇（SBCA-CN5-418）》 《NJ 系列 CPU 单元 用户手册 硬件篇（SBCA-CN5-358）》
2）硬件的设定	• EtherCAT 从站的节点地址设定 • 利用高功能单元的正面旋转开关设定号机号機 No./单元编号	各 EtherCAT 从站的用户手册、各高功能单元的用户手册

STEP4． 配线

步骤	内容	参考
1） Ethernet 电缆的配线	• 内置 EtherCAT 端口的配线 • 内置 EtherNet/IP 端口的配线	《NJ 系列 CPU 单元 用户手册 硬件篇（SBCA-CN5-358）》
2） I/O 配线	• EtherCAT 从站的 I/O 配线 • 基本 I/O 单元、高功能单元的配线	各 EtherCAT 从站的用户手册、《NJ 系列 CPU 单元 用户手册 硬件篇（SBCA-CN5-358）》
	• 配线的检查	《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》
3）电脑 （Sysmac Studio）的配线	• USB 电缆的配线 • 内置 EtherNet/IP 端口的配线	《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》

STEP5． 上机确认动作和运行

步骤	内容	Sysmac Studio 上的操作	参考
1）与 Sysmac Studio 的在线连接、项目的下载	打开控制器的电源，与 Sysmac Studio 在线连接。然后，下载项目。*1 在安装单元上进行 STEP2-1 的 “ 从站 / 单元构成的创建 ” 时，请在 STEP2-1 之前执行。	[控制器] ｜ [通信设定] [控制器] ｜ [同步]	「第 7 章 动作确认和运行」 (P.7-1)

2）上机确认动作	1. 通过 I/O 映射或监视窗口对实际 I/O 进行强制值刷新，确认配线。 2. 进行运动控制时，在程序模式下通过 MC 调试画面确认配线。然后，通过点动进给确认电机的方向，通过相对定位确认移动量 （电子齿轮的设定），确认原点复位动作。 3. 设为运行模式，确认用户程序的动作。	—	「第 7 章 动作确认和运行」(P.7-1)
3）控制器的正式运行	开始正式运行。	—	—

*1. 下载时使用 Sysmac Studio 的同步功能。

2

CPU 单元的动作

下面介绍使用 CPU 单元需要的知识，包括 CPU 单元的原理及各种状态下的动作。

2-1 CPU 单元的动作概要 2-2
2-1-1 CPU 单元的原理概要 2-2
2-1-2 CPU 单元在各种状态下的动作概要 2-3

2-2 软件的原理 2-4
2-2-1 软件构成 2-4
2-2-2 软件的动作 2-5

2-3 与 I/O 的访问原理 2-10
2-3-1 变量的种类 2-10
2-3-2 变量和 I/O 的访问 2-13

2-4 顺序控制和运动控制的原理 2-19
2-4-1 控制的概要 2-19
2-4-2 顺序控制的原理 2-21
2-4-3 运动控制的原理 2-22
2-4-4 顺序控制和运动控制的同步 2-23

2-5 CPU 单元内数据的概要 2-25

2-6 CPU 单元在各种状态下的动作 2-26
2-6-1 CPU 单元的状态 2-26
2-6-2 各种状态下的动作 2-28
2-6-3 动作模式 2-29

2-1 CPU 单元的动作概要

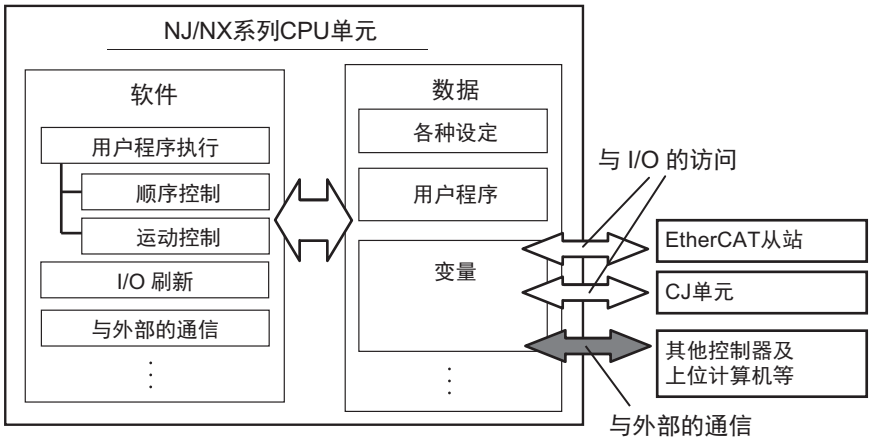
作为 CPU 单元的动作，下面介绍 CPU 单元的原理及 CPU 单元在各种状态下的动作概要。

2-1-1 CPU 单元的原理概要

为进行顺序控制和运动控制，NJ/NX 系列 CPU 单元需要执行用户程序、进行 I/O 刷新、与外部进行通信等。这些处理由 CPU 单元的软件执行。

此外， CPU 单元中还包含各种设定、用户程序、变量等数据。 CPU 单元使用这些数据进行处理。 CPU 单元和 I/O 的访问以及与外部的通信使用这些数据中的变量。

CPU 单元通过访问软件及使用变量的 I/O，可以通知执行顺序控制和运动控制。



（注）可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

本章介绍将 CPU 单元用于顺序控制、运动控制或两种控制时所需的基本知识，内容如下。

内容	参照页
软件的原理	P.2-4
与 I/O 的访问原理	P.2-10
顺序控制和运动控制的原理	P.2-19
CPU 单元的数据概要	P.2-25
CPU 单元在各种状态下的动作	P.2-26

参考

关于使用变量与外部进行通信的原理，请参考以下手册。

通信	参考手册
EtherNet/IP	• 内置EtherNet/IP端口时，《NJ/NX系列 CPU单元内置EtherNet/IP端口 用户手册 （SBCD-CN5-359）》 • EtherNet/IP 单元时，《CJ 系列 EtherNet/IP 单元 用户手册 NJ 系列 连接篇（SBCD-CN5-355）》
串行通信	《CJ 系列 串行通讯单元 用户手册 NJ 系列连接篇 (SBCD-CN5-354)》

2-1-2 CPU 单元在各种状态下的动作概要

CPU 单元会因发生异常、动作模式的切换而变为各种状态。因 CPU 单元的状态变化，用户程序的执行、I/O 刷新、与外部的通信处理动作也会变化。

「2-6 CPU 单元在各种状态下的动作」（P.2-26）中，介绍 CPU 单元在各种状态下的处理动作。

2-2 软件的原理

下面介绍 CPU 单元的软件构成及软件动作。

2-2-1 软件构成

CPU 单元的软件由 4 种功能构成。构成功能的单位为 “ 功能模块 ”。

各功能模块及其处理如下所示。

功能模块名称	处理
PLC 功能模块	进行任务的日程安排、向各功能模块的指示、事件日志、用户程序的执行、CJ 单元的 I/O 刷新[*1]、 USB 端口服务、 SD 存储卡服务、数据追踪处理等。
运动控制功能模块	进行运动控制处理。[*2]
EtherCAT 主机功能模块	EtherCAT 从站的 I/O 刷新[*3]、 EtherCAT 信息通信[*4] 等，作为 EtherCAT 主机与 EtherCAT 从站进行通信。
EtherNet/IP 功能模块	标签数据链接处理、内置 EtherNet/IP 端口服务等，进行 EtherNet/IP 通信的处理。

*1. CJ 单元是指在 CJ 系列 CPU 单元上可安装的单元中，可安装到 NJ 系列 CPU 单元的单元群。
*2. 根据运动控制指令发出的目标值 （位置、速度）等指示，进行运动计算，通过 EtherCAT 主机执行指令值输出、状态管理及信息获取。
*3. 通过过程数据通信 （以下简称 PDO 通信）进行。 PDO 通信是指在主机和从站之间以恒定的周期进行周期性数据交换的通信方式。
*4. 作为 EtherCAT 主机与 EtherCAT 从站进行通信。

2-2-2 软件的动作

软件执行的 CPU 单元处理，因执行处理的单元的状态、处理的执行条件不同，分为以下 4 种。

CPU 单元的处理种类	执行处理的 CPU 单元的状态	执行条件	处理示例
启动处理	启动中	仅接通电源时执行。	电源接通时的自诊断
任务中执行的处理	正常运行中及异常状态	在分配的任务中执行。任务分为按周期反复执行的任务和在指定的条件成立时仅执行 1 次的任务。	用户程序执行、 I/O 刷新
标签数据链接服务		仅在硬件或外部发出请求时执行。	标签数据链接
系统服务			USB 端口服务、 SD 存储卡服务

CPU 单元的状态请参考 「2-6-1 CPU 单元的状态」（P.2-26）。

下面介绍各处理的动作。

启动处理

启动处理仅接通电源时执行。
在启动处理中，执行以下处理。

处理	内容
电源接通时的自诊断	检测 “ 电源部异常 ”、 “CPU 复位 ”、 “CPU 异常 （WDT 异常） ”、 “ 电源连接不正确 ”。*1
数据检查	若带电源断开保持属性的变量值及 CJ 单元用存储器（DM/EM/HR）*2 的值在电源断开时无法保持，则系统定义变量 “_RetainFail” （电源断开保持失败标志）为 TRUE。
CJ 单元的识别*3	识别控制器上安装的所有 CJ 单元。
电源接通及电源断开的事件记录	记录电源接通及电源断开的事件。

*1. 关于 “ 电源部异常 ”、“CPU 复位 ”、“CPU 异常（WDT 异常）”、“ 电源连接不正确 ”，请参考 「12-1-2 CPU 单元无法动作的异常」（P.12-4）。
*2. 可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。
*3. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

任务中执行的处理

● 任务中执行的处理种类

任务中，将执行以下处理

处理	内容
I/O 刷新	与 EtherCAT 从站、 CJ 单元*1 的基本 I/O 单元或高功能单元进行数据输入输出。
用户程序执行	执行顺序控制的用户程序。或向运动控制发行指令。
运动控制	收到来自用户程序的指令，执行运动控制。
系统共通处理	执行数据追踪处理、标签数据链接处理等。

*1. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

任务中执行的处理详情请参考 📖「5-3-3　NX 系列的任务基本动作」（P.5-10）及 📖「5-4-3　NJ 系列的任务基本动作」（P.5-29）。

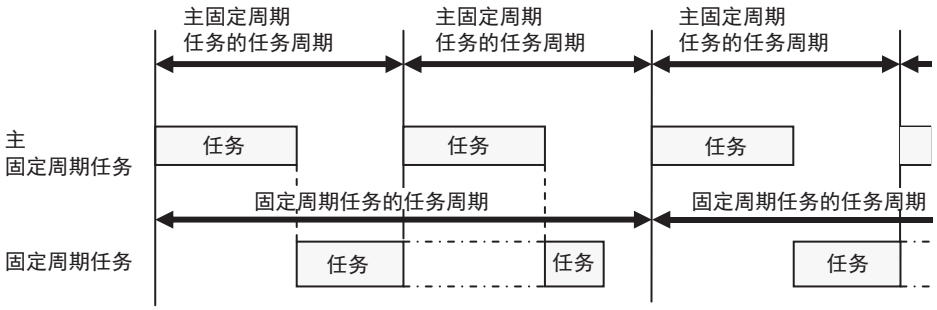
● 任务的动作

各处理将分配到任务中。因执行优先度和执行条件不同，任务分为以下 3 种。

固定周期任务的种类	执行优先度 （值越小，优先度越高）	执行条件	主要处理内容
主固定周期任务	4 执行优先度最高的任务。	以一定周期反复执行。	I/O 刷新、用户程序执行、运动控制
固定周期任务	5*1 优先度仅次于主固定周期任务的任务。	以一定周期反复执行。任务周期为主固定周期任务的任务周期整数倍。	I/O 刷新、用户程序执行、运动控制
	16,17,18	以一定周期反复执行。任务周期为主固定周期任务的任务周期整数倍。	处理内容因执行优先度不同而异。 执行优先度 16：I/O 刷新、用户程序执行 执行优先度 17、 18：用户程序执行
事件任务	8,48	在事先指定的条件成立时，仅执行 1 次。	用户程序执行

*1. 可使用执行优先度 5 的固定周期任务的 CPU 单元仅限 NX 系列 CPU 单元。

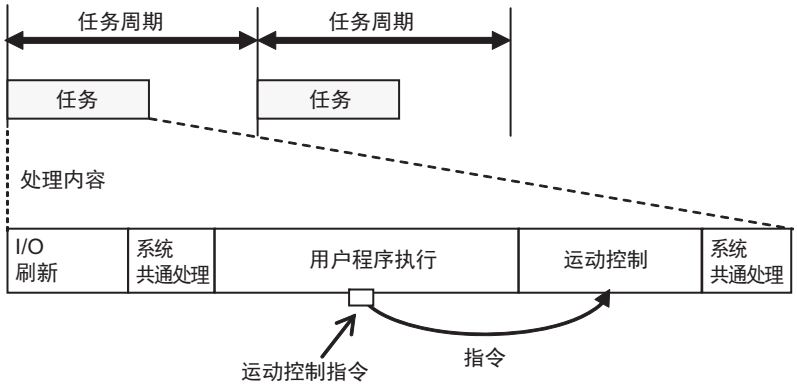
CPU 单元将优先执行优先度较高的任务。下图表示主固定周期任务和固定周期任务的动作示例。正在执行固定周期任务时，若执行了主固定周期任务，将优先执行主固定周期任务。



正在执行优先度更高的任务时，优先度低的任务动作在 NX 系列 CPU 单元和 NJ 系列 CPU 单元上不同。任务动作的详情请参考 「5-2　任务的概要」（P.5-5）。

● 任务中执行的处理动作

任务中分配的处理将在任务中按下图顺序执行。程序中记述了运动控制指令时，通过任务中的 " 用户程序执行 " 向 " 运动控制 " 发行指令。" 运动控制 " 将根据指令进行运动控制处理。



（注）" 运动控制 " 处理在主固定周期任务和执行优先度 5 的固定周期任务中执行。
详情请参考 「5-10-3　系统输入输出响应时间」（P.5-91）。

参考

在 NX 系列 CPU 单元上，可使用主固定周期任务和固定周期任务 （执行优先度 5）执行运动控制。需要区分两种运动控制时，主固定周期任务的运动控制称为 " 运动控制 1"，固定周期任务（执行优先度 5）的运动控制称为 " 运动控制 2"。

标签数据链接服务

● 标签数据链接服务的处理

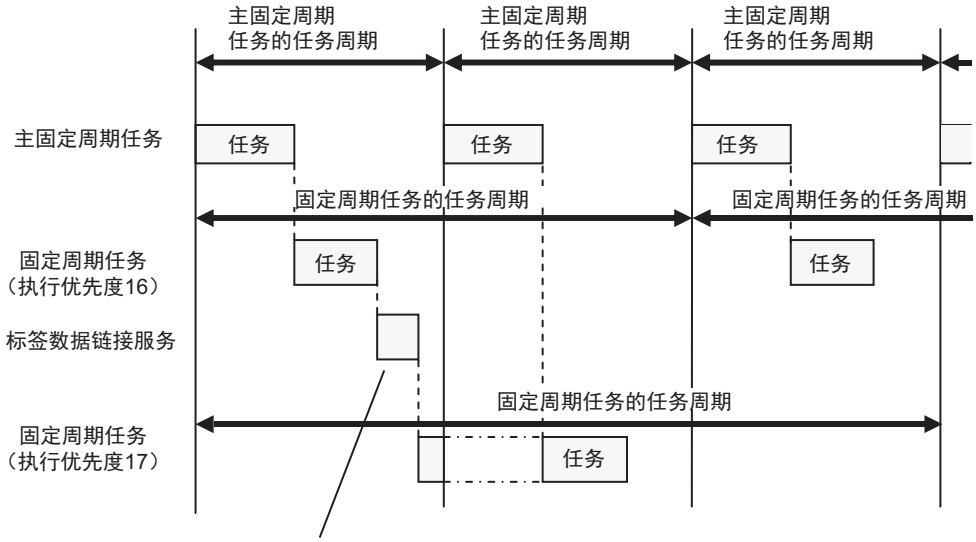
标签数据链接服务是指在 EtherNet/IP 网络上，使用其他控制器、设备和标签进行通信处理。使用 CPU 单元的内置 EtherNet/IP 端口或 CJ 系列 EtherNet/IP 单元 CJ1W-EIP21。

（注）可使用 CJ 系列 EtherNet/IP 单元 CJ1W-EIP21 的 CPU 单元仅限 NJ 系列 CPU 单元。

● 标签数据链接服务的动作

标签数据链接服务以恒定周期执行。周期或 1 次处理的执行时间因 CPU 单元的型号或标签数据链接设定不同而异。

此外，标签数据链接服务的执行优先度在固定周期任务 （执行优先度 16）和固定周期任务 （执行优先度 17）之间。因此，优先于固定周期任务 （执行优先度 17）执行，但一旦执行主固定周期任务或固定周期任务 （执行优先度 16），将优先执行主固定周期任务或固定周期任务 （执行优先度 16）。



将以固定周期任务（执行优先度16）和固定周期任务（执行优先度17）的优先度执行。

标签数据链接服务的详情请参考 📖「5-5　标签数据链接服务和系统服务」（P.5-47）。

系统服务

● 系统服务的种类

系统服务中包括以下处理。

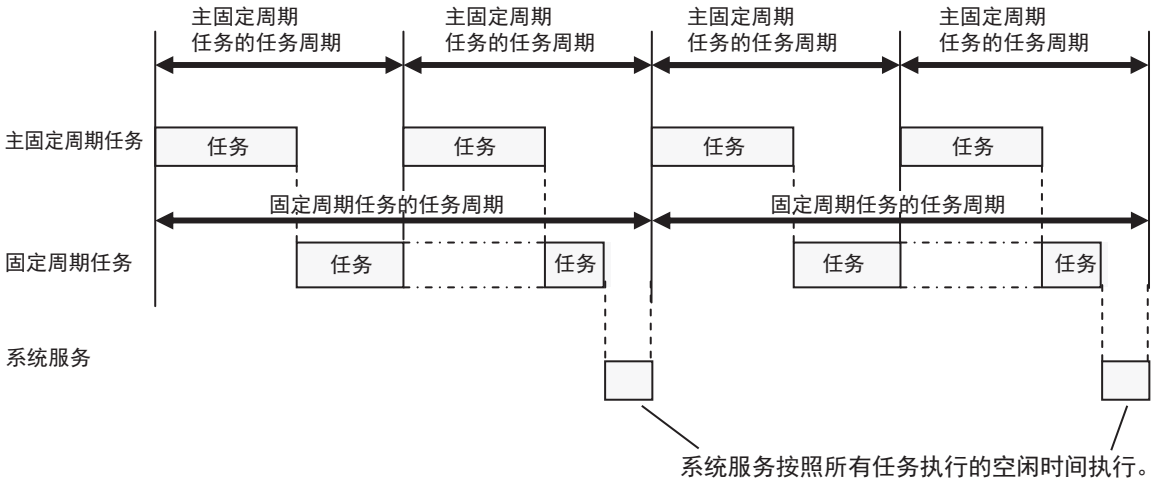
处理	内容
USB 端口服务	• 对来自编程设备、上位计算机的服务请求进行处理
内置 EtherNet/IP 端口服务	• 对来自编程设备、显示器、上位计算机或其他控制器的服务请求 （CIP 等）进行处理 • 通信指令 （CIP、套接字）的执行
内置 EtherCAT 端口服务	• EtherCAT 信息通信的执行
CJ 高功能单元服务*1	• 执行与 CJ 高功能单元的事件服务 • 通信指令 （CIP）的执行
SD 存储卡服务	• 从 FTP 客户端访问 • 通过 Sysmac Studio 对 SD 存储卡进行操作 • SD 存储卡指令的执行
自诊断处理	• 硬件的异常检测

*1. 与 CJ 高功能单元的分配区域数据交换在 “I/O 刷新 ” 时执行。
可使用 CJ 高功能单元的 CPU 单元仅限 NJ 系列 CPU 单元。

● 系统服务的动作

NX 系列 CPU 单元中，有来自硬件或 CPU 单元外部的请求时，系统服务将与其他处理并列执行。不受任务执行的影响，始终执行。

NJ 系列 CPU 单元中，有来自硬件或 CPU 单元外部的请求时，系统服务将在所有任务执行的空闲时间执行。



系统服务的详情请参考📖「5-5 标签数据链接服务和系统服务」（P.5-47）。

2-3 与 I/O 的访问原理

CPU 单元利用变量来访问 I/O。下面介绍变量与 I/O 的访问原理。
本手册中，将 EtherCAT 从站及 CJ 单元作为 I/O 处理。使用标签数据链接与其他控制器进行访问时，请参考 《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册 （SBCD-CN5-359）》。

2-3-1 变量的种类

NJ/NX 系列 CPU 单元中，通过用户程序，介由变量来访问 I/O 或 CPU 单元内部。
因是否可由用户设定属性或访问对象不同，变量分为以下几种。

<table>
<tr><th colspan="3">变量</th><th>可否设定属性</th><th>访问对象</th></tr>
<tr><td colspan="3">用户定义变量</td><td>可设定所有的属性</td><td>CPU 单元内部</td></tr>
<tr><td rowspan="3">准用户定义变量</td><td rowspan="2">设备变量</td><td>EtherCAT 从站设备变量</td><td rowspan="3">可设定部分属性*1</td><td>“ 轴 ” 中未分配的 EtherCAT 从站 *2*3</td></tr>
<tr><td>CJ 单元设备变量 *4</td><td>CJ 系列基本 I/O 单元、 CJ 系列高功能单元</td></tr>
<tr><td colspan="2">凸轮数据变量</td><td>伺服驱动器、编码器输入从站、CPU 单元内部</td></tr>
<tr><td rowspan="6">系统定义变量</td><td colspan="2">PLC 系统变量</td><td rowspan="6">所有的属性都不可设定</td><td>CPU 单元内部</td></tr>
<tr><td rowspan="3">运动控制系统变量</td><td>MC 共通变量</td><td rowspan="3">伺服驱动器、编码器输入从站、CPU 单元内部</td></tr>
<tr><td>轴变量</td></tr>
<tr><td>轴组变量</td></tr>
<tr><td colspan="2">EtherNet/IP 系统变量</td><td>内置 EtherNet/IP 端口</td></tr>
<tr><td colspan="2">EtherCAT 主机系统变量</td><td>内置 EtherCAT 主机端口</td></tr>
</table>

*1. 关于可设定的属性，请参考 「设备变量的属性」（P.3-12）。
*2. EtherCAT 从站包括 EtherCAT 从站终端上的 NX 单元。
*3. Sysmac Studio Ver.1.09 以上中，“ 轴 ” 中分配的 EtherCAT 从站也是 EtherCAT 从站设备变量的访问对象。
*4. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

用户定义变量

所有属性由用户定义的变量。用户定义变量的详情请参考 「6-3 变量」（P.6-28）。

准用户定义变量

可设定部分属性的变量。用于访问特定的对象数据。因访问对象不同，准用户定义变量分为设备变量和凸轮数据变量。

● 设备变量

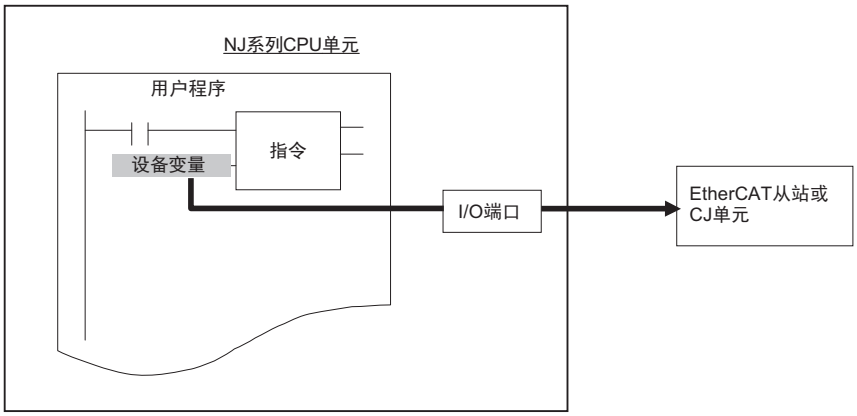
设备变量是指可访问设备数据的变量。设备是指通过 CPU 单元进行 I/O 刷新的对象的总称。表示 EtherCAT 从站或 CJ 单元。

因设备及设备访问对象的数据不同，分为以下几种。

设备变量的种类	设备	访问对象的数据
EtherCAT 从站设备变量	“轴”中未分配的 EtherCAT 从站 *1	EtherCAT 上的从站的过程数据*2
CJ 单元设备变量*3	CJ 系列基本 I/O 单元	基本 I/O 单元的实际 I/O 数据
	CJ 系列高功能单元	高功能单元的运行用数据*4、设定用数据*5

*1. Sysmac Studio Ver.1.09 以上中，“轴”中分配的 EtherCAT 从站也是 EtherCAT 从站设备变量的访问对象。
*2. 主机和从站之间以过程数据通信周期进行更换的 I/O 数据。
*3. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。
*4. 用于 CJ 单元运行的数据。使用 CJ 单元用存储器的 CIO 区域。
*5. 用于 CJ 单元设定的数据。使用 CJ 单元用存储器的 DM 区域。

设备变量通过 I/O 端口访问 EtherCAT 从站或 CJ 单元。 I/O 端口是指用于访问设备的逻辑端口。



I/O 端口及设备变量的详情请参考 📖「3-3-1 I/O 端口」（P.3-8）。

● 凸轮数据变量

在运动控制中，用于访问凸轮表数据的变量。详情请参考 📖《NJ/NX 系列 CPU 单元 用户手册 运动控制篇 （SBCE-CN5-363）》。

系统定义变量

系统定义变量是指名称等所有属性由 NJ 系统事先定义，且分配有特定功能的变量。用户无法变更名称等属性。
特定功能在各个功能模块中，因此各个功能模块中有系统定义变量。系统定义变量的种类如下所示。

功能模块	系统定义变量的种类
PLC 功能模块	PLC 系统变量
运动控制功能模块	运动控制系统变量
EtherNet/IP 功能模块	EtherNet/IP 系统变量
EtherCAT 主机功能模块	EtherCAT 主机系统变量

运动控制系统变量根据运动控制功能的内容分类如下。

运动控制系统变量	功能的内容
MC 共通变量	运动控制功能模块共通处理
轴变量	“ 轴 ” 的控制
轴组变量	轴组[*1] 的控制

*1. 轴组是指整合了多个 “ 轴 ” 的组。用于插补动作。

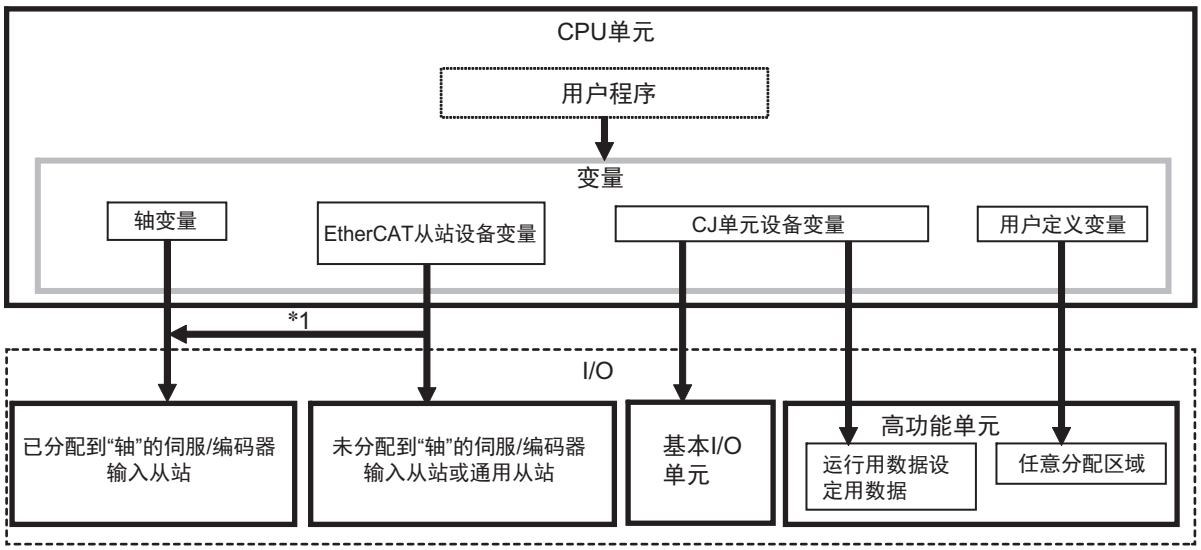
关于系统定义变量的详情，请参考 📖「A-4　系统定义变量一览」（P.A-38）。

2-3-2 变量和 I/O 的访问

CPU 单元在用户程序中使用变量。变量可访问所分配 I/O 的数据。 CPU 单元中，如下分配 I/O 和变量。CJ 单元的高功能单元中使用的变量因要访问的数据不同而异。

I/O		数据	变量
EtherCAT 从站	“ 轴 ” 中未分配的 EtherCAT 从站*1	—	EtherCAT 从站设备变量
	“ 轴 ” 中分配的 EtherCAT 从站	—	轴变量
CJ 单元*2	基本 I/O 单元	—	CJ 单元设备变量
	高功能单元	• 运行用数据 • 设定用数据	CJ 单元设备变量
		任意分配区域的数据*3	用户定义变量

*1. Sysmac Studio Ver.1.09 以上中，“ 轴 ” 中分配的 EtherCAT 从站也是 EtherCAT 从站设备变量的访问对象。
*2. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。
*3. 用户将扩展功能或从站的输入输出等分配到存储器区域后使用的数据。任意分配区域的数据不可通过设备变量访问。



与 EtherCAT 从站的访问

与 EtherCAT 从站的访问方法因 EtherCAT 从站的种类不同而异。

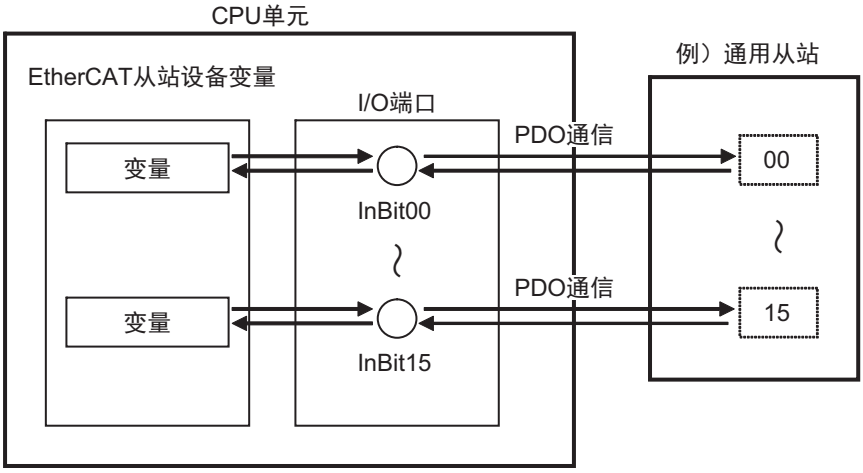
EtherCAT 从站的种类	访问方法
• 未分配到 “ 轴 ” 的伺服 / 编码器输入从站*1 • 通用从站	根据 EtherCAT 从站设备变量通过 I/O 端口访问
已分配到 “ 轴 ” 的伺服 / 编码器输入从站	通过轴变量直接访问*2

*1. Sysmac Studio Ver.1.09 以上中，“ 轴 ” 中分配的 EtherCAT 从站也可根据 EtherCAT 从站设备变量通过 I/O 端口进行访问。
*2. 伺服驱动器时，将 1 台伺服电机作为 1 个 “ 轴 ” 分配到轴变量中使用；编码器输入从站时，则为 1 个计数器。

（注） 不可分配到 “ 轴 ” 的 EtherCAT 从站称为 “ 通用从站 ”，可分配到 “ 轴 ” 的 EtherCAT 从站称为 “ 伺服 / 编码器输入从站 ”。
伺服驱动器 / 编码器输入从站请参考 《NJ/NX 系列 CPU 单元 用户手册 运动控制篇 （SBCE-CN5-363）》。

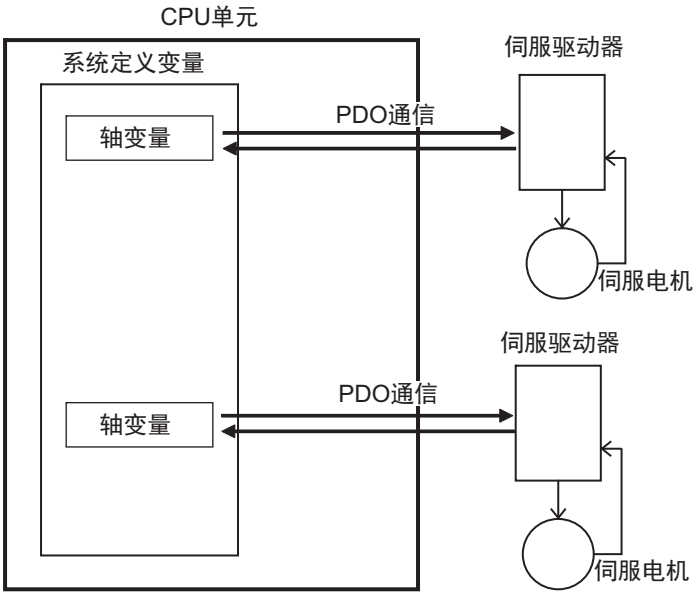
● 与未分配到 “ 轴 ” 的伺服 / 编码器输入从站或通用从站的访问

根据 EtherCAT 从站设备变量通过 I/O 端口访问。通过 I/O 端口访问时，利用 PDO 通信进行。



● 与已分配到 “ 轴 ” 的伺服 / 编码器输入从站的访问

轴变量可直接访问已分配到 “ 轴 ” 的伺服 / 编码器输入从站通过轴变量访问时，利用 PDO 通信进行。例如，使用伺服驱动器控制伺服电机时，针对轴变量中分配的伺服电机发出的控制指示将发送到伺服驱动器中。来自伺服电机的反馈将通过轴变量从伺服驱动器发送到 CPU 单元中。



轴变量的详情请参考 📖「3-5-2　轴变量 / 轴组变量」（P.3-17）。

版本相关信息

Sysmac Studio Ver.1.09 以上中，可将设备变量分配到 " 轴 " 中分配的伺服 / 编码器输入从站的 I/O 端口上。
可分配设备变量的 I/O 端口应满足以下任一条件。

- R/W 属性的值为 R （读取专用）。
- R/W 属性的值为 W（写入专用），Sysmac Studio 的 [轴基本设定] 画面的 [详细设定] 中，过程数据的设定为 < 不分配 >。

使用注意事项

将设备变量分配到 " 轴 " 中分配的伺服 / 编码器输入从站的 I/O 端口时，若进行以下操作，将解除设备变量的分配。敬请注意。

(1) Sysmac Studio Ver.1.09 以上中，将设备变量分配到 " 轴 " 中分配的伺服 / 编码器输入从站的 I/O 端口上。

(2) 保存项目数据。

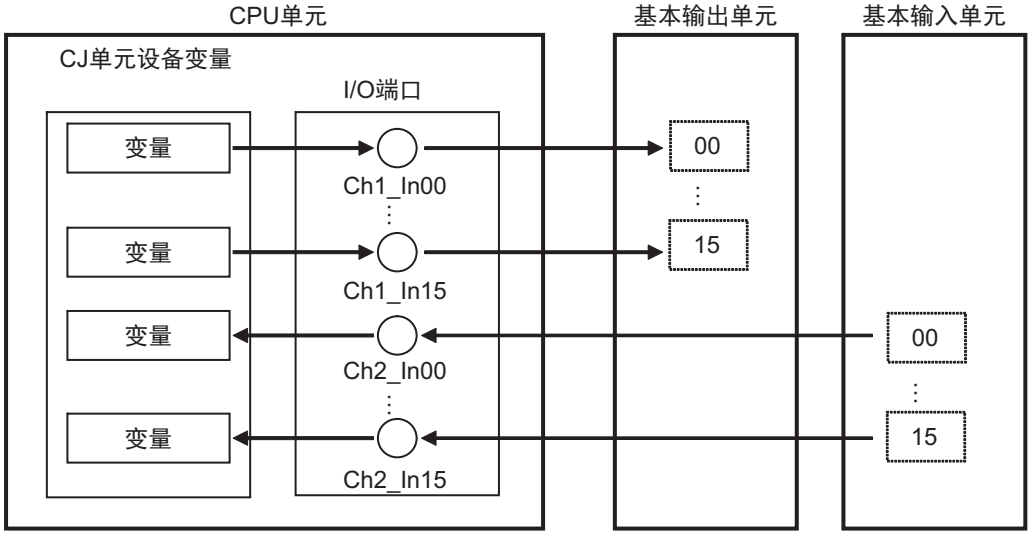
(3) 在 Sysmac Studio Ver.1.08 或以下版本中打开该项目数据。

参考

EtherCAT 通信分为 PDO 通信和 SDO 通信。对于伺服电机的位置控制等以固定控制周期更新输入输出数据的指令，使用 PDO 通信。对于参数传送等以指定时间读取或写入数据的指令，则使用 SDO 通信。详情请参考 《NJ/NX 系列 CPU 单元内置 EtherCAT 端口 用户手册 (SBCD-CN5-358)》。

与基本 I/O 单元的访问

在 NJ 系列 CPU 单元上访问基本 I/O 单元时，根据 CJ 单元设备变量通过 I/O 端口进行。



详情请参考 「3-3-1 I/O 端口」（P.3-8）。

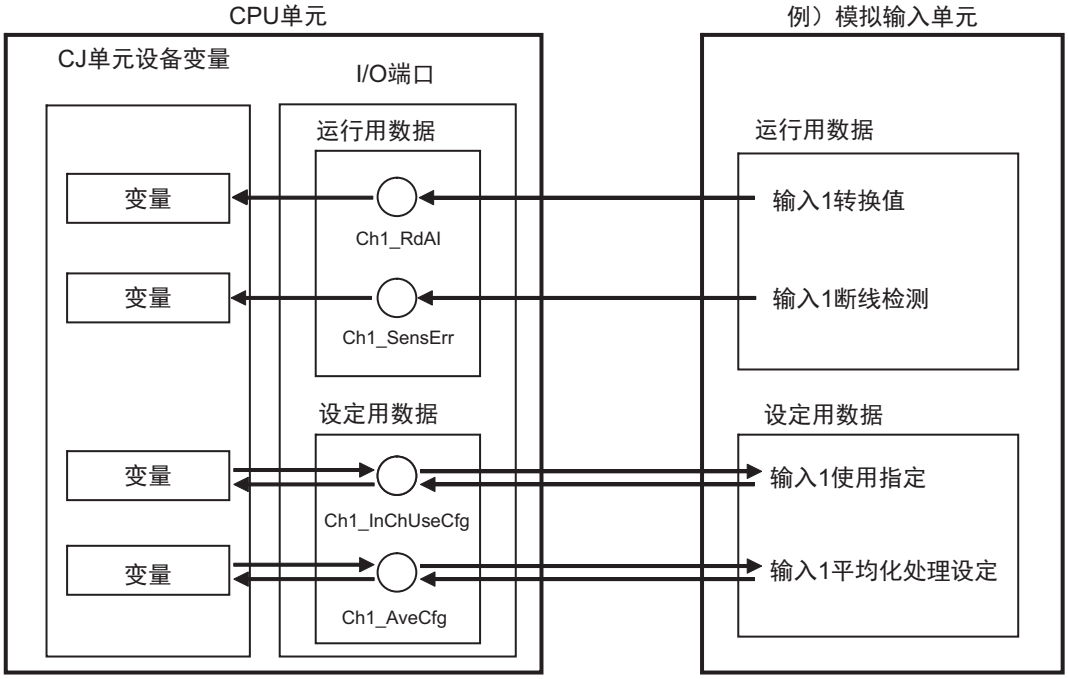
与高功能单元的访问

在 NJ 系列 CPU 单元上访问高功能 I/O 单元的方法因访问对象数据的不同分为两种。

访问方法	对象数据
根据 CJ 单元设备变量通过 I/O 端口访问	• 运行用数据 • 设定用数据
将用户定义变量 AT （分配目标）指定到 CJ 单元用存储器并访问	任意分配区域的数据

● 根据 CJ 单元设备变量通过 I/O 端口访问

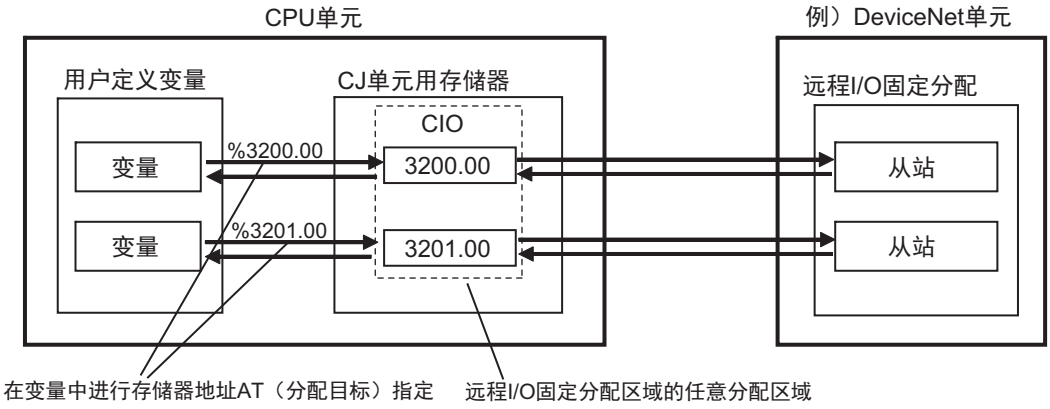
访问运行用数据及设定用数据时，根据 CJ 单元设备变量通过 I/O 端口进行。I/O 刷新时进行数据交换。



详情请参考 📖「3-3-1 I/O 端口」（P.3-8）。

● 将用户定义变量 AT （分配目标）指定到 CJ 单元用存储器并访问

访问任意分配区域时，将用户定义变量 AT （分配目标）指定到 CJ 单元用存储器后进行。 CJ 单元用存储器在 I/O 刷新时与 CJ 单元进行数据交换。



任意分配区域中包括以下区域。

- 使用 DeviceNet 单元时的固定分配区域
- 使用 DeviceNet 单元或 CompoNet 主机单元（通信模式 No.8）时，在 CX-Integrator 上指定的自由分配区域
- 使用高速计数器单元时的扩展存储器
- 使用过程输入输出单元时的扩展存储器

关于 CJ 单元用存储器，请参考 📖「A-7　CJ 单元用存储器的详情」（P.A-135）。关于 AT （分配目标）指定，请参考 📖「6-3-8　变量的各属性说明」（P.6-57）。

2-4 顺序控制和运动控制的原理

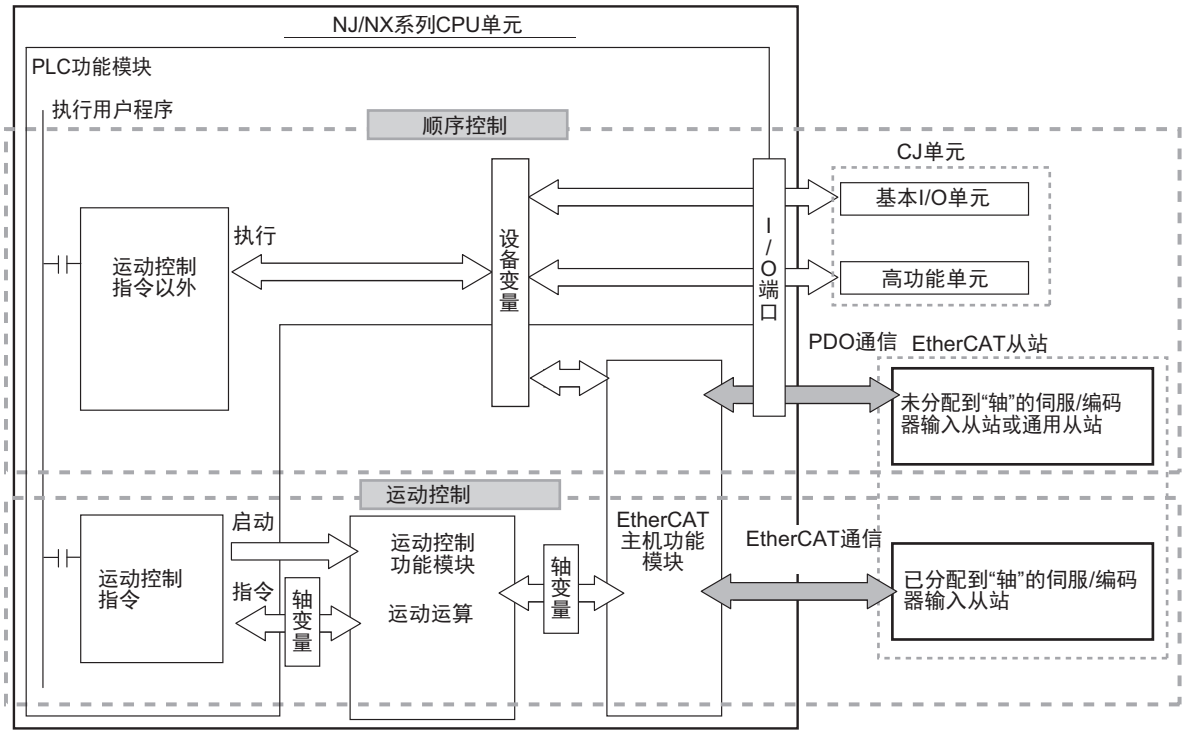
下面介绍 CPU 单元的顺序控制和运动控制的原理。

2-4-1 控制的概要

NJ/NX 系列 CPU 单元中可执行顺序控制和运动控制。

顺序控制是指执行用户程序中记述的运动控制指令以外的指令，对 EtherCAT 从站、 CJ 单元进行控制。控制时使用 PLC 功能模块、 EtherCAT 主机功能模块。

运动控制是指执行用户程序中记述的运动控制指令，对 " 轴 " 中分配的 EtherCAT 从站上的伺服驱动器 / 编码器输入从站进行控制。控制时使用 PLC 功能模块、运动控制功能模块、 EtherCAT 主机功能模块。



（注）可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

参考

控制原理对应的指令分类

根据控制原理，指令大致可分为以下两类。

指令的分类	定义
运动控制指令以外（顺序控制）	执行 PLC 功能模块中的用户程序，结束处理的指令。
运动控制指令	通过执行 PLC 功能模块中的用户程序来启动，对运动控制功能模块发出指令。 例)MC_Home （原点复位）、 MC_Move （定位）、 MC_CamIn （凸轮动作开始）等运动控制相关的各种操作

运动控制指令的详情请参考 《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。运动控制指令以外的指令详情请参考 《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

版本相关信息

Sysmac Studio Ver.1.09 以上中，可将设备变量分配到 “ 轴 ” 中分配的伺服 / 编码器输入从站的 I/O 端口上。
可分配设备变量的 I/O 端口应满足以下任一条件。

- R/W 属性的值为 R （读取专用）。
- R/W属性的值为W（写入专用），Sysmac Studio的[轴基本设定]画面的[详细设定]中，过程数据的设定为 < 不分配 >。

使用注意事项

将设备变量分配到 “ 轴 ” 中分配的伺服 / 编码器输入从站的 I/O 端口时，若进行以下操作，将解除设备变量的分配。敬请注意。

(1) Sysmac Studio Ver.1.09 以上中，将设备变量分配到 “ 轴 ” 中分配的伺服 / 编码器输入从站的 I/O 端口上。

(2) 保存项目数据。

(3) 在 Sysmac Studio Ver.1.08 或以下版本中打开该项目数据。

2-4-2 顺序控制的原理

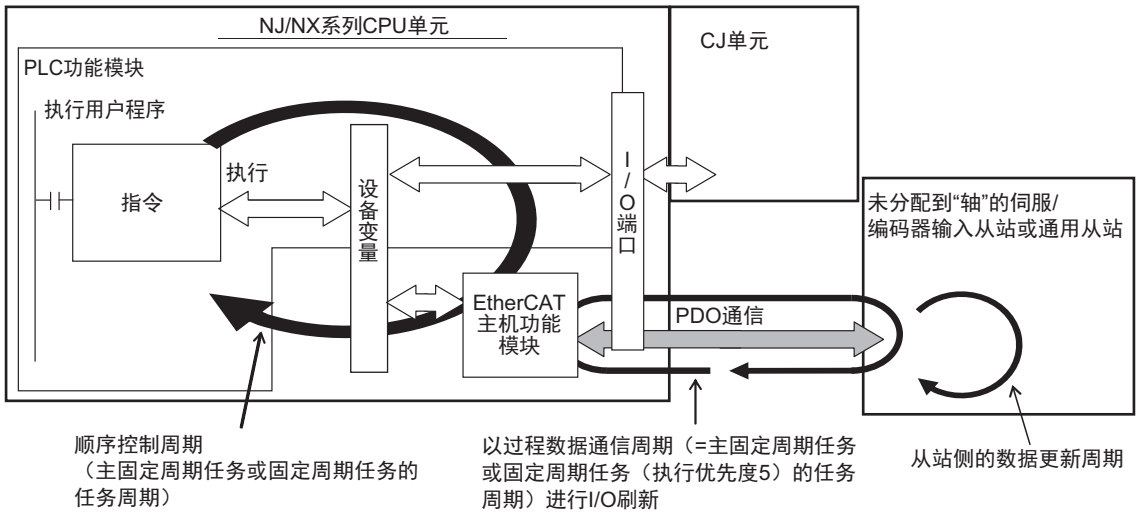
因顺序控制的对象设备不同而异。下面以功能模块的动作、控制周期为例对顺序控制的原理进行说明。

设备	顺序控制的原理	
	功能模块的动作	控制周期
未分配到 " 轴 " 的伺服 / 编码器输入从站*1 或通用从站	• PLC 功能模块将执行用户程序，更新设备变量。 • EtherCAT 主机功能模块使用设备变量，通过 I/O 端口与从站进行数据交换。	分配了程序的任务之任务周期 （主固定周期任务或固定周期任务的任务周期）*2
CJ 单元*3	利用 PLC 功能模块，进行用户程序的执行、设备变量的更新、与 CJ 单元的数据交换。	分配了程序的任务之任务周期 （主固定周期任务或固定周期任务的任务周期）*4

*1. Sysmac Studio Ver.1.09 以上中，通过将设备变量分配到 I/O 端口，" 轴 " 中分配的伺服 / 编码器输入从站也可进行顺序控制。
*2. 从站侧的数据更新周期取决于从站的设定。
*3. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。
*4. 与 CJ 单元进行数据交换的周期为分配了 CJ 单元 I/O 刷新的任务之周期。

未分配到 " 轴 " 的伺服 / 编码器输入从站或通用从站，以过程数据通信周期，即主固定周期任务或固定周期任务 （执行优先度 5）的任务周期进行 I/O 刷新。但是，用户程序的执行及设备变量的更新按分配了程序的任务之任务周期执行。因此，将从站的值反映到设备变量中并进行控制的周期为分配了程序的任务之任务周期。
希望以过程数据通信周期控制从站时，需要将控制从站的程序分配到主固定周期任务或固定周期任务（执行优先度 5）中。

详情请参考 📖「5-10-3　系统输入输出响应时间」（P.5-91）。



参考

- 可使用固定周期任务 （执行优先度 5）的单元仅限 NX 系列 CPU 单元。
- 在 NX 系列 CPU 单元上，可使用主固定周期任务和固定周期任务（执行优先度 5）执行过程数据通信。需要区分两个过程数据通信周期时，主固定周期任务的通信周期称为 " 过程数据通信周期 1"，固定周期任务 （执行优先度 5）的通信周期称为 " 过程数据通信周期 2"。
- 在 NJ 系列 CPU 单元上，只使用主固定周期任务执行过程数据通信。

2-4-3 运动控制的原理

下面以功能模块的动作、控制周期为例对运动控制的原理进行说明。

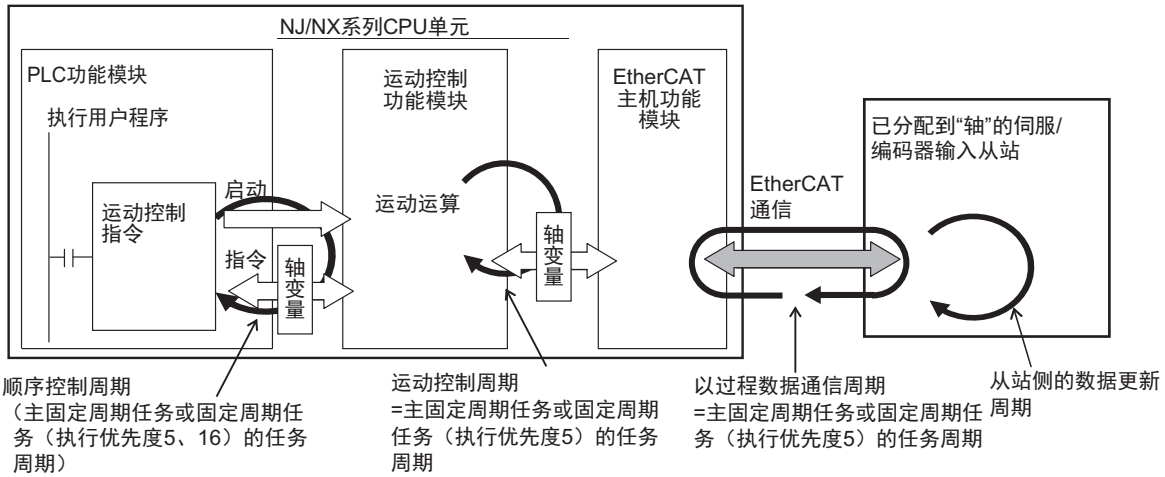
● 功能模块的动作

- PLC 功能模块通过启动用户程序中的运动控制指令，向运动控制功能模块发行运动控制指令。本指令中使用轴变量。
- 运动控制功能模块根据PLC功能模块的指令进行运动计算，然后按计算结果的指令值更新轴变量的值。
- EtherCAT 主机功能模块通过 EtherCAT 通信将轴变量的指令值发送到伺服驱动器等。

● 控制周期

运动控制周期为主固定周期任务或固定周期任务 （执行优先度 5）的任务周期。
运动控制功能模块的运动计算周期为主固定周期任务或固定周期任务 （执行优先度 5）的任务周期。
此外，与已分配到控制对象 “ 轴 ” 的伺服 / 编码器输入从站之间，以主固定周期任务或固定周期任务（执行优先度 5）的过程数据通信周期进行数据交换。过程数据通信周期与主固定周期任务或固定周期任务 （执行优先度 5）同步。
因此，运动控制周期为主固定周期任务或固定周期任务 （执行优先度 5）的任务周期，可完全同步地运行多个轴。
但存在以下限制。

- 运动控制指令的启动及运动控制指令的发行以顺序控制周期执行。
- EtherCAT 从站侧的数据更新周期取决于从站的设定。



使用注意事项

- 可使用固定周期任务 （执行优先度 5）的单元仅限 NX 系列 CPU 单元。
- 在NX系列CPU单元上，可使用主固定周期任务和固定周期任务（执行优先度5）执行运动控制。需要区分两种运动控制时，主固定周期任务的运动控制称为 “ 运动控制 1”，固定周期任务 （执行优先度 5）的运动控制称为 “ 运动控制 2”。
- 在 NJ 系列 CPU 单元上，仅使用主固定周期任务执行运动控制。

参考

- 用运动控制功能模块控制的EtherCAT从站需要在Sysmac Studio上进行“轴”的分配设定。因此，PLC 功能模块通过启动用户程序中的 “ 运动控制指令 ”，向运动控制功能模块发出指令。而且，可利用运动控制功能模块，通过轴变量获取信息。
- 运动控制系统的输入输出响应时间因分配（记述有运动控制指令的）程序的任务种类不同而异。详情请参考 「5-10-3　系统输入输出响应时间」（P.5-91）。

2-4-4 顺序控制和运动控制的同步

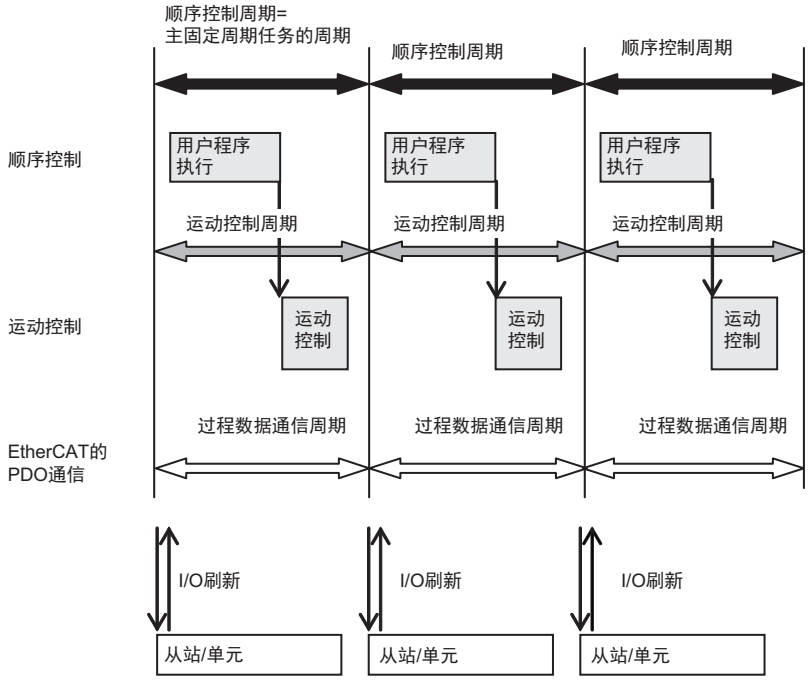
顺序控制的控制周期为分配了程序或 I/O 刷新的任务之任务周期。同时，运动控制始终以主固定周期任务或固定周期任务 （执行优先度 5）的任务周期执行。此外，运动控制中使用的 EtherCAT 从站的过程数据通信周期与主固定周期任务或固定周期任务 （执行优先度 5）同步。

NX 系列 CPU 单元中，可对按轴和轴组执行运动计算的运动控制进行设定。设定为运动控制 1 的轴及轴组按主固定周期任务的周期执行运动控制及过程数据通信。设定为运动控制 2 的轴及轴组按固定周期任务 （执行优先度 5）的周期执行运动控制及过程数据通信。

在 NJ 系列 CPU 单元上，按主固定周期任务的周期执行轴和轴组的运动控制和过程数据通信。

因此，若将顺序控制程序分配到执行运动控制的任务中，可使顺序控制和运动控制的控制周期及 EtherCAT 的过程数据通信周期同步。

将程序分配到主固定周期任务时的图如下所示。在下图中，顺序控制周期、运动控制的周期、 EtherCAT 的过程数据通信周期全部同步。



参考

- NX 系列 CPU 单元上运动控制、任务、过程数据通信周期的关系

运动控制	执行的任务	过程数据通信周期
运动控制 1	主固定周期任务	过程数据通信周期 1
运动控制 2	固定周期任务 （执行优先度 5）	过程数据通信周期 2

- NJ 系列 CPU 单元上运动控制、任务、过程数据通信周期的关系

运动控制	执行的任务	过程数据通信周期
运动控制	主固定周期任务	过程数据通信周期

2-5 CPU 单元内数据的概要

CPU 单元中包含各种设定、用户程序、当前值等数据。主要的数据及其内容如下所示。
关于 CPU 单元的所有数据，请参考 「A-6 CPU 单元内数据的属性」（P.A-132）。

数据类型			内容
各种设定	EtherCAT 构成	EtherCAT 从站构成	EtherCAT 从站的构成信息。
		EtherCAT 主机设定	EtherCAT 主机的通信周期等参数设定。
	单元构成和单元设定		识别 CPU 单元所需的单元构成信息及高功能单元的初始设定信息。
	I/O 映射		对根据单元构成自动生成的 I/O 端口变量进行的分配信息。
	控制器设定	动作设定	接通电源时的动作模式、安全等设定或系统服务设定。
		内置 EtherNet/IP 端口设定	EtherNet/IP 端口的以下设定。 TCP/IP 设定、 Ethernet 设定、 DHCP 设定、 DNS 设定、 FTP 设定、 NTP 设定、 SNMP 设定
	运动控制设定		“ 轴 ” / “ 轴组 ” 的轴轴变量 / 轴组变量及运动控制参数设定。
	凸轮数据设定		由运动控制指令的凸轮动作中使用的相位 / 变位数据构成的凸轮表。
	事件设定		用户定义的异常 / 信息的设定。
	任务设置		任务的种类或数量、任务的执行条件、任务名称、任务中执行的程序等设定。
	数据追踪设定		数据追踪触发条件等设定。
	标签数据链接表		EtherNet/IP 上的标签数据链接设定。
	控制器名称		CPU 单元的名称。
	操作权限认证		Sysmac Studio 对 CPU 单元进行操作所需的权限密码。
	内置时钟	设定时刻	CPU 单元内置时钟的时刻。
		时区设定	CPU 单元内置时钟的时区设定。
用户程序	POU （程序构成单位）		程序、函数、功能块的定义。包括本地变量表及其初始值。
	数据	数据类型	数据类型的定义。
		全局变量	全局变量的属性信息。初始值属性和保持属性等。
当前值	变量的值		变量的值。
	CJ 单元用存储器的值*1		CJ 单元用存储器中的 CIO、 WR、 HR、 DM、 EM 区域的值。
其它	事件日志		控制器的异常履历及电源 ON/OFF 或运行开始等异常以外的现象的履历。
	绝对编码器原点位置偏置		运动控制中，原点确定后的指令位置和从绝对编码器读取的绝对值数据的差，用于恢复带绝对编码器的伺服驱动器的当前位置。

*1. 可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

2-6 CPU 单元在各种状态下的动作

介绍 CPU 单元在各种状态下的用户程序执行、I/O 刷新、与外部的通信处理动作。此外，还对 CPU 单元处于 “ 正常运行中 ” 时，介绍用户程序执行状态的切换和动作模式。

2-6-1 CPU 单元的状态

CPU 单元有 “ 启动中 ”、“ 正常运行中 ”、“ 异常状态 ” 三种状态。各状态的定义如下所示。

状态	定义
启动中	软件正在执行启动处理的状态
正常运行中	由软件执行任务中执行的处理或系统服务，未发生控制器异常的状态
异常状态	由软件执行任务中执行的处理或系统服务，发生控制器异常的状态

此外，“ 正常运行中 ” 包括 “ 程序模式动作中 ”、“ 运行模式动作中 ”、“ 下载中 ” 三种状态。“ 正常运行中 ” 的 CPU 单元将根据用户操作变为各种状态。各状态的定义如下所示。

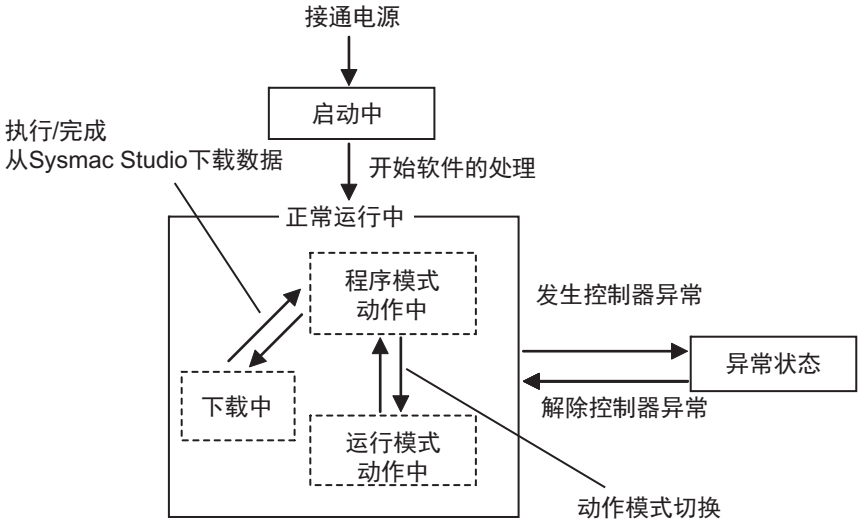
状态	定义
程序模式动作中	动作模式为程序模式时
运行模式动作中	动作模式为运行模式时
下载中	正在从 Sysmac Studio 下载数据的状态

（注）程序模式和运行模式请参考 「2-6-3 动作模式」（P.2-29）。

● CPU 单元的状态变化

接通电源后，CPU 单元变为 “ 启动中 ”。CPU 单元变为 “ 启动中 ” 并经过 10 ～ 20 秒后，开始软件处理，变为 “ 正常运行中 ”。“ 正常运行中 ” 若发生控制器异常，将变为 “ 异常状态 ”。解除控制器异常后，恢复为 “ 正常运行中 ”。

从 “ 启动中 ” 变为 “ 正常运行中 ” 后， CPU 单元将根据控制设定中设定的动作模式，变为 “ 程序模式动作中 ” 或 “ 运行模式动作中 ”。然后，通过动作模式的切换，“ 程序模式动作中 ” 时变为 “ 运行模式动作中 ”，“ 运行模式动作中 ” 时，变为 “ 程序模式动作中 ”。“ 程序模式动作中 ” 若从 Sysmac Studio 下载数据，将变为 “ 下载中 ”。下载完成后，恢复为 “ 程序模式动作中 ”。



参考

- CPU单元的状态可通过CPU单元正面的状态显示LED确认。关于状态显示LED的确认方法，请参考 📖「12-1　异常状态的动作」（P.12-2）。
- 关于 CPU 单元状态发生变化时的数据动作，请参考 📖「A-6　CPU 单元内数据的属性」（P.A-132）。
- 关于 CPU 单元状态发生变化时的变量值，请参考 📖「6-3-9　与状态变化对应的变量值的变化」（P.6-64）。

2-6-2 各种状态下的动作

用户程序的执行、I/O 刷新、与外部的通信动作因 CPU 单元的状态而变化。“ 启动中 ” 及 “ 正常运行中 ” 各处理的动作如下所示。

“ 异常状态 ” 请参考 📖「12-1-3 CPU 单元可动作的异常」（P.12-6）。

CPU 单元的处理	启动中的动作	正常运行中的动作		
		程序模式动作中	运行模式动作中	下载中
用户程序	停止。	停止。	执行。	停止。
EtherCAT 从站的 I/O 刷新	停止。	执行。		EtherCAT 通信变为可安全操作状态。*1
CJ 单元的 I/O 刷新*2	停止。	执行。		
与外部的通信	停止。	执行。		执行。*3

*1. 仅更新输入的值。
*2. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。
*3. 标签数据链接会继续，但标签数据链接的值不更新。输出标签的值保持下载前的值。输入标签的值不反映到变量中。

● I/O 输出接点的值

通过 I/O 刷新处理，各状态下 I/O 输出接点的值如下。

I/O 的输出接点	启动中的动作	正常运行中的动作		
		程序模式动作中	运行模式动作中	下载中
EtherCAT 从站的输出接点	遵照各从站的设定。*1	变为 EtherCAT 从站设备变量的值。*2		遵照各从站的设定。
CJ 系列基本输出单元的输出接点*3	变为 OFF。	变为 CJ 单元设备变量的值。*4		

*1. 接通电源后，开始 EtherCAT 通信所需的时间等从站设定请参考各从站的手册。
*2. 下载完成后， EtherCAT 从站将进行初始化处理。初始化处理中的输出遵照各从站的设定。
*3. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。
*4. 但是，下载完成时及动作模式切换时， CJ 单元设备变量的值将按照初始值属性初始化。

关于变量的初始值属性，请参考 📖「6-3-8 变量的各属性说明」（P.6-57）。

参考

动作模式切换后伺服驱动器的动作
运动控制中，若从运行模式切换为程序模式，将以各轴参数中设定的最大减速度减速停止。

EtherCAT 从站初始化处理中的动作模式切换
EtherCAT 从站正在进行初始化处理时，可将 CPU 单元的动作模式变更为运行模式。此时，请确认通信建立后，再创建用于控制从站数据的程序。通过用户程序，可使用系统定义变量 “_EC_PDSlavTbl” （过程数据通信中从站表），确认所有从站的过程数据输出和输入是否有效。

2-6-3 动作模式

动作模式是指可根据动作确认或正式运行等目的进行切换的模式。根据不同目的，动作模式分为运行模式和程序模式。各动作模式中，用户程序的执行状态不同。以下表示各动作模式的目的和用户程序的执行状态。

动作模式	目的	用户程序的执行状态
运行模式*1	执行用户程序的试运行或正式运行	执行。
程序模式	I/O 配线检查等，不执行用户程序即可进行的动作确认	不执行。

*1. 初始设定中，若 CPU 单元从 “ 启动中 ” 变为 “ 正常运行中 ”，将以运行模式进行动作。

参考

通过动作模式的切换，执行轴的停止、运动控制指令的中断等。动作模式切换时运动控制功能模块的动作请参考 《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。

各动作模式下是否可通过 Sysmac Studio 及显示器操作

各动作模式下，可否通过 Sysmac Studio 及显示器进行的主要操作如下。

操作		运行模式	程序模式
Sysmac Studio	同步	不可	可
	在线编辑	可	
	强制值刷新	可	
	变更变量或 CJ 单元用存储器*1 的值	可	
显示器	变更变量或 CJ 单元用存储器 *1 的值	可	

*1. 可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

动作模式切换时变量值的保持

若从运行模式切换为程序模式或从程序模式切换为运行模式，根据变量的保持属性不同，变量值如下所示。

变量的保持属性	变量的值
非保持	有初始值设定时 : 指定的初始值 无初始值设定时 : 系统定义的初始值*1
保持	保持动作模式变更前的值

*1. 因变量的数据类型不同而异。「初始值指定为空白时」（P.6-61）。

关于 CPU 单元状态发生变化时的变量值，请参考「6-3-9 与状态变化对应的变量值的变化」（P.6-64）。

动作模式的设定及切换

接通电源后开始运行时， CPU 单元将按照控制器设定的动作模式动作。此外，正常运行中时，用户可根据目的切换动作模式。设定及切换从可在 Sysmac Studio 上进行。

● 接通电源后开始运行时的动作模式设定

接通电源后开始运行时，按照 “ 电源接通时模式 ” 中设定的动作模式来运行 CPU 单元。在控制器设定的 “ 动作设定 ” 中，将 “ 电源接通时模式 ” 设定为运行模式或程序模式。关于 “ 电源接通时模式 ” 的设定，请参考 📖「4-2-2　控制器设定」（P.4-4）。

● 运行中的动作模式切换

动作模式通过 Sysmac Studio 切换。从菜单栏的 ［控制器］ ー ［动作模式］中选择运行模式或程序模式。

安全要点

进行接通电源时的动作模式设定及动作模式的切换前，请确认其不会对设备造成不良影响。

动作模式的确认

动作模式可通过 CPU 单元的 RUN LED 或 Sysmac Studio 确认。

● 通过 RUN LED 确认

根据 CPU 单元的 RUN LED 显示，动作模式如下。

RUN LED 的显示	动作模式
熄灭	程序模式
点亮	运行模式

● 通过 Sysmac Studio 确认

动作模式可通过 Sysmac Studio 的控制器状态窗口确认。以下控制器状态窗口中，显示 “ 运行模式 ”。

参考

要将运行模式中这一信息输出到外部时，请使用电源单元的运行中输出。电源单元的运行中输出请参考 📖《NX 系列 CPU 单元 用户手册 硬件篇 （SBCA-CN5-418）》及 📖《NJ 系列 CPU 单元 用户手册 硬件篇 （SBCA-CN5-358）》。

3

I/O 单元和从站 / 单元构成

本章介绍 I/O 端口的使用方法、从站 / 单元构成的创建和步骤、各功能的分配。

3-1 从站 / 单元构成的步骤 3-2

3-2 从站 / 单元构成的创建和核对 3-4

3-2-1 EtherCAT 从站构成的创建 3-4

3-2-2 单元构成的创建 3-6

3-2-3 单元构成的核对 3-6

3-3 I/O 端口和设备变量 3-8

3-3-1 I/O 端口 3-8

3-3-2 I/O 端口名称 3-9

3-3-3 设备变量 3-10

3-4 向单元分配变量 3-13

3-4-1 向单元分配变量的方法 3-13

3-4-2 单元中分配的变量的使用方法 3-15

3-5 “ 轴 ” 的创建和伺服驱动器 / 编码器输入从站的分配 3-17

3-5-1 概要 3-17

3-5-2 轴变量 / 轴组变量 3-17

3-5-3 “ 轴 ” 及轴变量的创建和使用方法 3-19

3-1 从站 / 单元构成的步骤

下面介绍从站 / 单元构成的步骤。

在整体步骤中，从站 / 单元构成与以下阴影标示的位置对应。

STEP 1. 软件设计

- STEP1-1 设计 I/O 和处理
- STEP1-2 设计任务
- STEP1-3 设计程序

↓

STEP 2. 软件设定和编程

- STEP2-1 从站 / 单元构成
- STEP2-2 控制器设定
- STEP2-3 编程
- STEP2-4 离线调试

↓

STEP 3. 安装和硬件的设定

↓

STEP 4. 配线

↓

STEP 5. 上机确认动作和运行

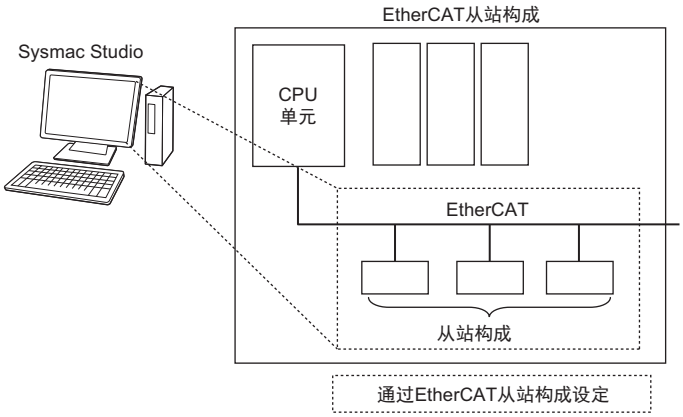
详情请参考 📖「1-3 NJ/NX 系列的整体使用步骤」（P.1-12）。

STEP1 EtherCAT 从站构成的创建 （使用 EtherCAT 时）、单元构成的创建 （使用 CJ 单元时）

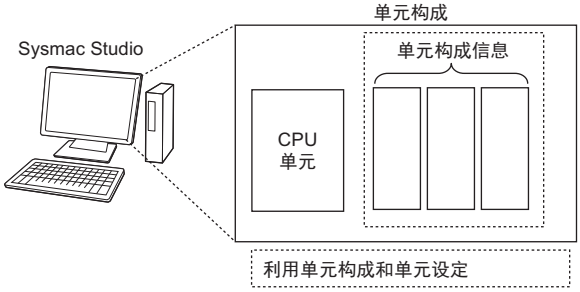
参考

「3-2-1 EtherCAT 从站构成的创建」（P.3-4）
「3-2-2 单元构成的创建」（P.3-6）

- EtherCAT 从站构成的创建



- 单元构成的创建

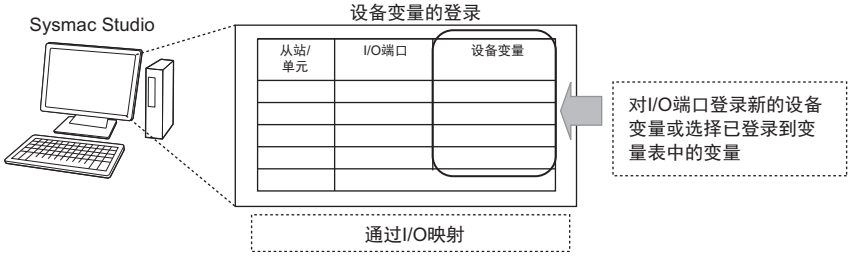


STEP2 向 I/O 端口分配设备变量

参考

「2-3-1 变量的种类」（P.2-10）
「3-3 I/O 端口和设备变量」（P.3-8）

- 设备变量的登录

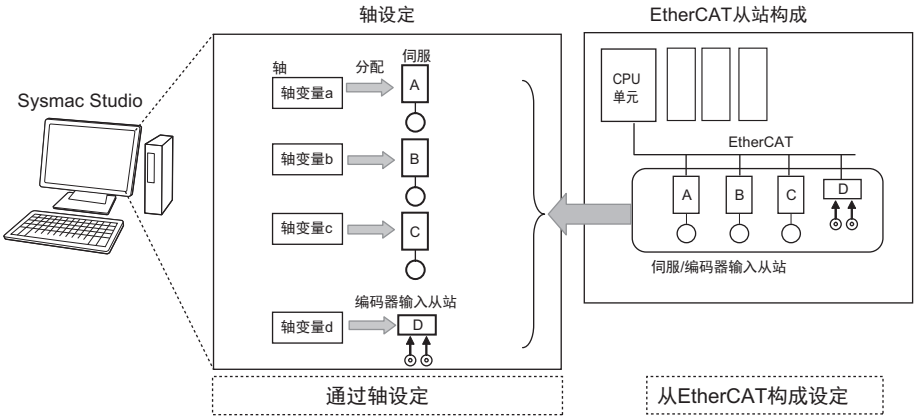


STEP3 “ 轴 ” 的创建和向伺服 / 编码器输入从站的分配 （运动控制时）

参考

「3-5 “ 轴 ” 的创建和伺服驱动器 / 编码器输入从站的分配」（P.3-17）

1. “ 轴 ” 的创建
2. 向 EtherCAT 构成上的伺服 / 编码器输入从站分配

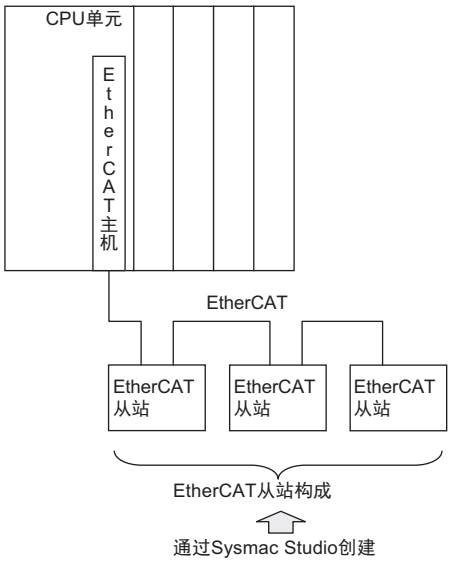


3-2 从站 / 单元构成的创建和核对

为了访问控制器中的各从站 / 单元，在 Sysmac Studio 上创建从站构成或单元构成。然后，对 Sysmac Studio 上创建的单元构成和实际的单元构成是否相同进行核对。

3-2-1 EtherCAT 从站构成的创建

在 Sysmac Studio 的 EtherCAT 构成编辑画面中创建将 CPU 单元识别为 “ 正 ” 的 EtherCAT 从站构成。



将自动登录构成从站的 “I/O 端口 ”。有后面的用户对 “I/O 端口 ” 分配 “ 设备变量 ”。
在用户程序中指定 “ 设备变量 ”，即可访问各从站。

EtherCAT 从站构成的创建方法请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》中的 “[EtherCAT] 的构成和设定 ”。

参考

连接 EtherCAT 从站终端时，除了创建 EtherCAT 从站构成外，还需要进行从站终端的构成和动作设定。从站终端的构成和动作设定的详情请参考📖《NX 系列 EtherCAT 耦合器单元 用户手册 (SBCD-CN5-361)》。

版本相关信息

EtherCAT 从站终端可在 CPU 单元版本 1.05 以上和 Sysmac Studio Ver.1.06 以上的组合中使用。

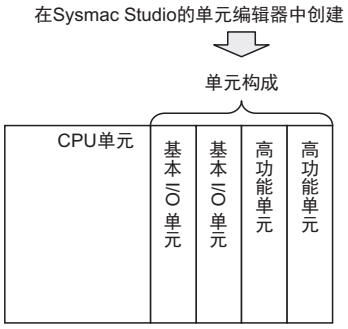
参考

EtherCAT 从站为伺服 / 编码器输入从站时，登录到 EtherCAT 从站构成后，再创建 “ 轴 ”，将自动登录轴变量。

「3-5 “ 轴 ” 的创建和伺服驱动器 / 编码器输入从站的分配」（P.3-17）。

3-2-2 单元构成的创建

在 Sysmac Studio 的 “ 单元构成和单元设定 ” 的单元编辑器中创建 CPU 单元识别为 “ 正 ” 的单元构成。



创建单元构成后，会在接通电源时自动检查实际的单元构成是否与视为 “ 正 ” 的 “ 单元构成 ” 一致。
同时，对 “ 单元构成 ” 中指定的单元自动登录 “I/O 端口 ”。
有后面的用户对 “I/O 端口 ” 分配 “ 设备变量 ”。
对于 “ 单元构成 ” 的单元，在用户程序上通过 “ 设备变量 ” 来访问。

单元构成的创建方法请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》中的 “[CPU·扩展装置] 的构成和设定 ”。

使用注意事项

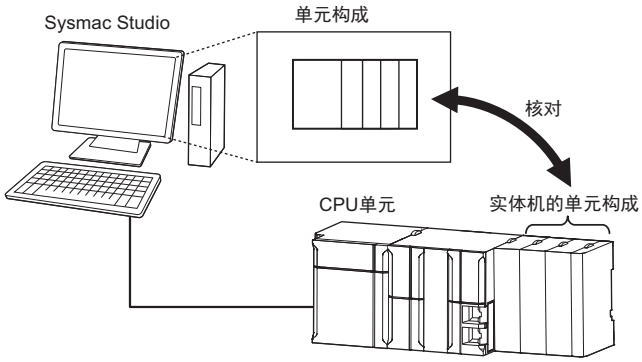
- 即使不创建 “ 单元构成 ”，也可启动控制器。但是，I/O 端口和设备变量不会自动登录，因此不可通过用户程序访问各单元。
- 在 NJ 系列 CPU 单元上使用 CJ 单元时，需要创建 “ 单元构成 ”。
  NX 系列 CPU 单元上不使用单元，因此无 “ 单元构成 ”。

3-2-3 单元构成的核对

在 Sysmac Studio 中，可对以下单元构成进行核对。

Sysmac Studio 上的单元构成和实际单元构成的核对

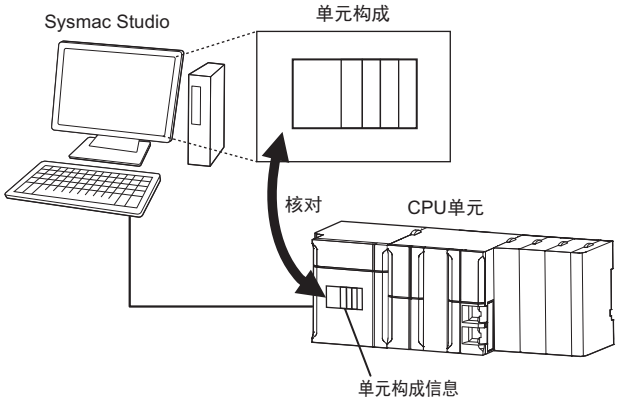
对 Sysmac Studio 上的单元构成和实际的单元构成是否一致进行核对。
第一次用 Sysmac Studio 向 CPU 单元传送单元构成前，需要确认该单元构成是否与实际单元构成一致时执行。



Sysmac Studio 上的单元构成和 CPU 单元内单元构成的核对

对 Sysmac Studio 上的单元构成和 CPU 单元内保持的单元构成是否一致进行核对。

用 Sysmac Studio 向已经传送了单元构成的 CPU 单元传送单元构成前，需要确认这些构成是否一致时执行。



3-3 I/O 端口和设备变量

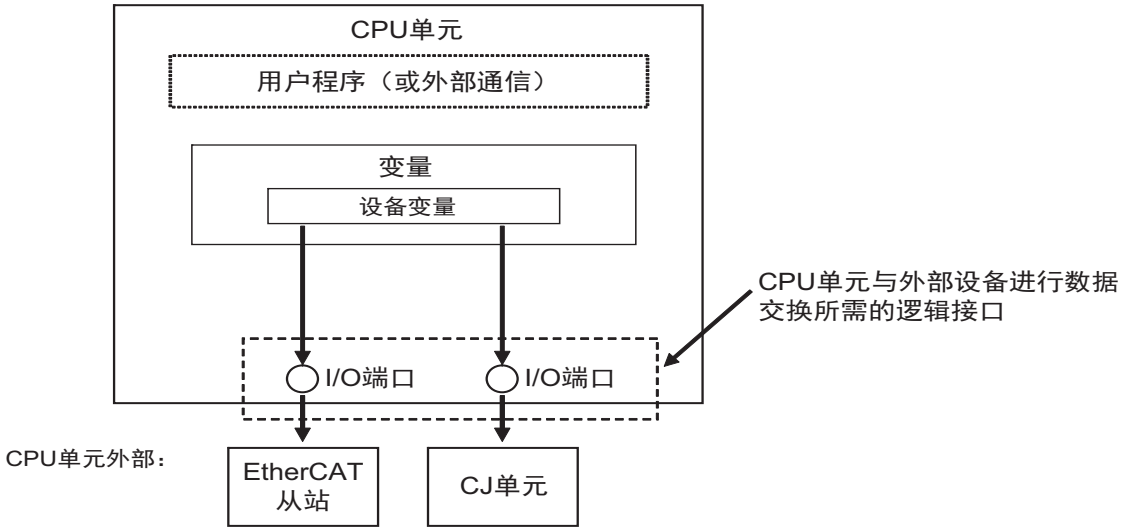
下面介绍访问 NJ/NX 系列控制器的 EtherCAT 从站及 CJ 系列单元所需的 I/O 端口及设备变量。

使用注意事项

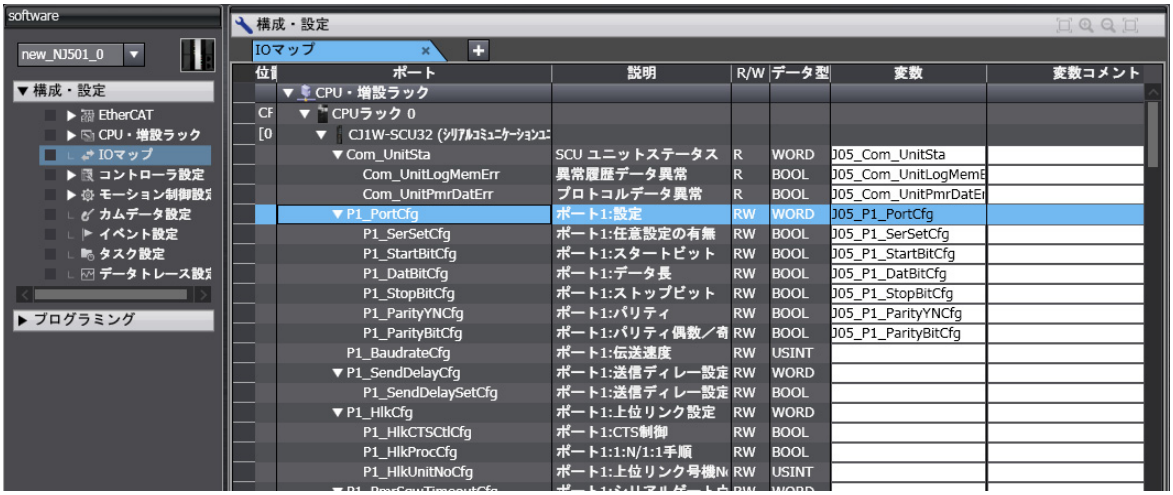
可使用 CJ 系列单元的 CPU 单元仅限 NJ 系列 CPU 单元。

3-3-1 I/O 端口

“I/O 端口 ” 是指 CPU 单元与外部设备 （从站 / 单元）进行数据交换所需的逻辑接口。
在 Sysmac Studio 上创建从站 / 单元构成后，可自动生成 I/O 端口。在 I/O 端口上分配设备变量后，通过用户程序访问。



在 Sysmac Studio 上创建 EtherCAT 从站构成及单元构成或从实际控制器中读取这些信息后，“I/O 端口 ” 将自动登录到 I/O 映射中。
登录的 “I/O 端口 ” 可在 Sysmac Studio 的 “I/O 映射 ” 中确认。
I/O 映射：



3-3-2 I/O 端口名称

I/O 端口名称会自动登录。设备（单元 / 从站）为 EtherCAT 从站时、CJ 系列基本 I/O 单元时、CJ 系列高功能单元时， I/O 端口名称有以下不同。

设备为 EtherCAT 从站时

设备为 EtherCAT 从站，且为远程 I/O 终端时， I/O 端口名称如下所示。

例） 16 点远程 I/O 终端时：

Bit00 ～ Bit15

在其他从站中，为 EtherCAT 对象字典中定义的对象名称或其一部分。

例）模拟输入单元时：

CH0_input16-bit

例） R88D-KN50H-ECT 时：

Position actual value、 Digital inputs 等

设备为 CJ 系列基本 I/O 单元时

设备为 CJ 系列基本 I/O 单元时， I/O 单元名称按以下规则生成。

● **基本 I/O 单元的 I/O 端口名称规格**

输入	输出
Ch□_In□□ └ 端子编号：00～15 └ 16点单位的通道：1～4	Ch□_Out□□ └ 端子编号：00～15 └ 16点单位的通道：1～4

● **不同点数的 I/O 端口名称示例**

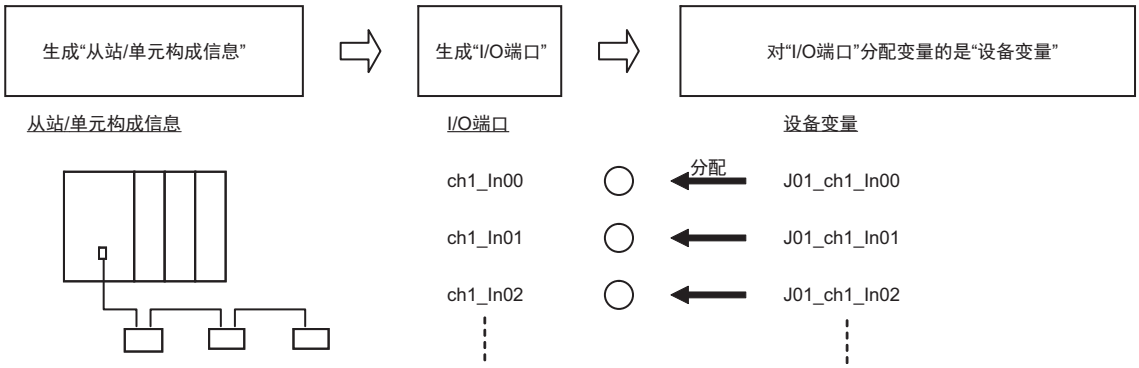
输入点数 输出点数	I/O 端口名称			
	输入	数据类型	输出	数据类型
32 点	Ch1_In	WORD	Ch1_Out	WORD
	Ch1_In00 ～ Ch1_In 15	BOOL	Ch1_Out00 ～ Ch1_Out 15	BOOL
	Ch2_In	WORD	Ch2_Out	WORD
	Ch2_In00 ～ Ch2_In 15	BOOL	Ch2_Out00 ～ Ch2_Out 15	BOOL

设备为 CJ 系列高功能单元时

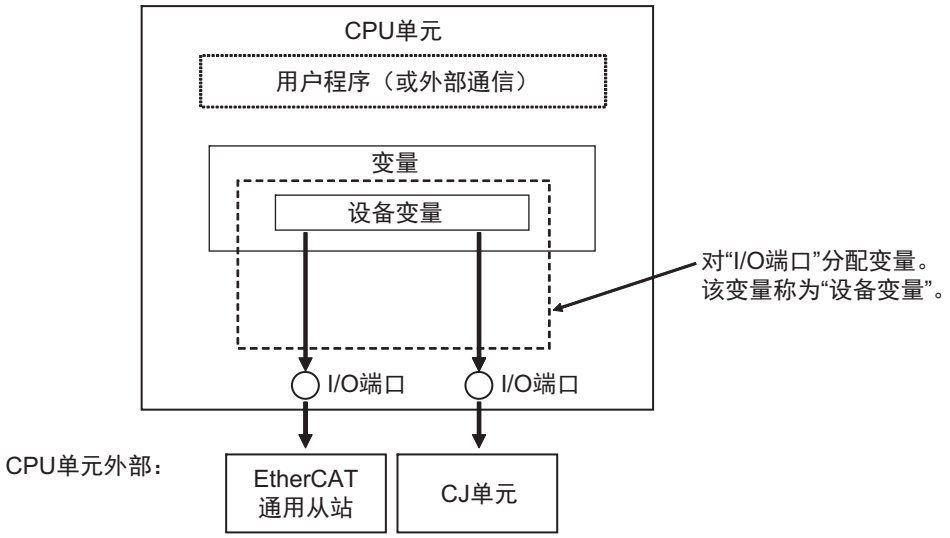
设备为 CJ 系列高功能单元时， I/O 端口名称为各单元型号及功能的固定名称。
例）模拟输入单元 CJ1W-AD041-V1 时
Ch1_PkHdCmd、 Ch1_AveCfg 等

3-3-3 设备变量

NJ/NX 系列中，不是分配 CPU 单元外部设备 （从站 / 单元）分配的存储器空间上的地址，而是对该 “I/O 端口 ” 分配 “ 变量 ”。
该变量称为 “ 设备变量 ”。



用户程序 （或外部通信）可利用该 “ 设备变量 ” 来访问设备 （从站 / 单元）。



关于整体变量中设备变量的定位，请参考📖「2-3-1 变量的种类」（P.2-10）。

Sysmac Studio 上的设备变量登录方法请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

设备变量的属性

设备变量的各属性如下所示。属性分为可变更设定的属性和不可变更的属性。

属性	设定内容	设定变更
变量名称	自动生成时：[设备名称 _] ＋ [I/O 端口名称] 默认 “ 设备名称 ”： • EtherCAT 从站时， “E” ＋ “001 开始的连号 ”。 • CJ 单元时， “J” ＋ “01 开始的连号 ”。 “I/O 端口名称 ” 请参考 📖「3-3-2 I/O 端口名称 」（P.3-9）。 手动输入时：任意变量名称	可以
数据类型	遵照 “I/O 端口 ” 的数据类型。	可以
AT （分配目标）指定	• NX 单元 （通过 EtherCAT 耦合器单元） ECAT://node#[EtherCAT 耦合器单元节点地址 ,NX 单元编号]/I/O 单元名称 • EtherCAT 从站 ECAT://node#[节点地址]/ [I/O 端口名称] • CJ 单元 IOBus://rack#[装置编号]/slot#[插槽编号]/ [I/O 端口名称]	不可
保持	• EtherCAT 从站设备变量：非保持 • CJ 单元设备变量中 分配到 “ 运行用数据 ” （CIO 区域）的变量：“ 非保持 ”、 分配到 “ 设定用数据 ” （DM 区域）的变量：“ 保持 ”	不可
初始值	无	可以
常量	无	可以
网络公开	非公开	可以
边缘	无	不可

（注） 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

各属性的含义请参考 📖「6-3-4 变量的属性 」（P.6-30）。

参考

- 在I/O映射中，可进行I/O端口的强制值刷新。因此，可以强制进行实际I/O的ON/OFF，对配线进行检查。
- 在I/O映射中，可以选择将设备变量登录到哪个变量表（全局变量表、各POU的本地变量表）中。

3-4 向单元分配变量

有些指令中以变量的形式指定 EtherCAT 从站终端的各单元。因此，用户需要事先将变量分配到各单元中。一旦将变量分配到单元后，即使变更单元的连接位置，该信息也会自动反映到变量的值中。因此，不需要在每次变更单元的连接位置时重新分配变量。

版本相关信息

向单元分配变量的功能可在CPU单元版本1.05以上和Sysmac Studio Ver.1.06以上的组合中使用。

参考

还可对从站终端以外的 EtherCAT 从站分配变量。其他公司生产的 EtherCAT 从站也一样。向 EtherCAT 从站分配变量的方法与向 EtherCAT 耦合器单元和 NX 单元分配变量的方法相同。

3-4-1 向单元分配变量的方法

即使在 Sysmac Studio 上进行 EtherCAT 从站终端的设定，也不会自动生成单元中分配的变量。请按以下步骤设定向单元分配的变量。

1 **在 Sysmac Studio 的 [构成和设定] ｜ [EtherCAT] 中，设定 EtherCAT 从站终端的构成。**

2 **在 [构成和设定] ｜ [I/O 映射] 中，显示 I/O 映射。**
显示设定的 EtherCAT 从站终端的各单元 I/O 映射。

3 **右击分配变量的单元的型号，选择 [显示节点位置端口]。**
将在 I/O 映射中追加 “Node location information” 端口。

4 **右击 “Node location information”，选择 [生成设备变量]。**
将在 “Node location information” 端口的 [变量] 栏中写入变量名称。

单元中分配的变量之数据类型为 _sNXUNIT_ID 型结构体。 _sNXUNIT_ID 型结构体的详情如下所示。

变量		名称	含义	数据类型
设定为任意值		指定单元	指定的单元	_sNXUNIT_ID
	NodeAdr	节点地址	EtherCAT 耦合器单元的节点地址	UINT
	IPAdr	IP 地址*1	EtherCAT 耦合器单元的 IP 地址	BYTE[5]
	UnitNo	单元编号	指定单元的单元编号	UDINT
	Path	路径 *1	指向指定单元的路径	BYTE[64]
	PathLength	“Path” 的有效数据长度	“Path” 的有效数据长度	USINT

*1. 只在控制器内部使用的信息。用户不会参照或编辑。

使用注意事项

单元中分配的变量值将在登录变量时自动设定。变量的值请勿变更。若用户变更了变量的值，控制器可能发生以外动作。

参考

向从站终端以外的 EtherCAT 从站分配变量时，变量的数据类型为 _sECAT_ID 型结构体。详情如下所示。

变量		名称	含义	数据类型
设定为任意值		指定从站	要指定的从站	_sECAT_ID
	NodeAdr	节点地址	指定从站的节点地址	UINT

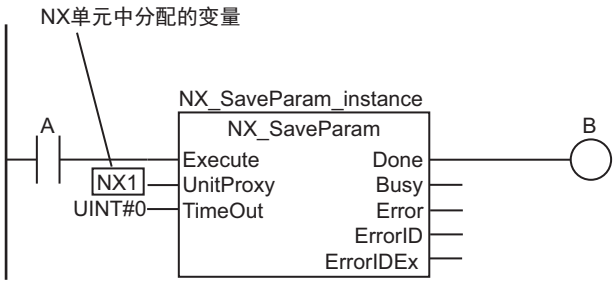
3-4-2 单元中分配的变量的使用方法

单元中分配的变量在用户程序上如下使用。无论哪种情况，都需要将与 I/O 映射中分配到单元的变量同名的变量，登录到变量表中。该变量的数据类型为 _sNXUNIT_ID 型结构体。

单元的指定

将单元中分配的变量直接作为参数传递到指定单元的指令中。

例） NX_SaveParam 指令的执行
指定分配了 “NX1” 这一变量的 NX 单元，执行 NX_SaveParam 指令。向变量 “UnitProxy” 传递 “NX1”。



单元属性的指定

指定单元中分配的变量之成员，可以只指定单元的部分参数。

例）根据 NX 单元的单元编号执行指令
若分配了 “NX1” 这一变量的 NX 单元的单元编号为 2，则从该单元的数据对象中读取数据。变量 “NX1.UnitNo” 表示单元编号。

```
IF (NX1.UnitNo = UINT#2) THEN
  NX_ReadObj_instance(Execute:=TRUE, UnitProxy:=NX1, Obj:=S_Obj, ReadDat:=Rdat);
END_IF;
```

多个单元的指定

将单元中分配的变量设为排列指定。使用循环处理，可对多个单元执行相同的处理。

例）执行将多个 NX 单元设为数据可写入模式的指令
“NX0”、“NX1”、“NX2”为分配到NX单元的变量。分别代入排列NXTable[0..2]中后，依次执行NX_ChangeWriteMode 指令。

- 变量表

名称	数据类型
NXTable	ARRAY[0..2] OF_sNXUNIT_ID

- ST 程序

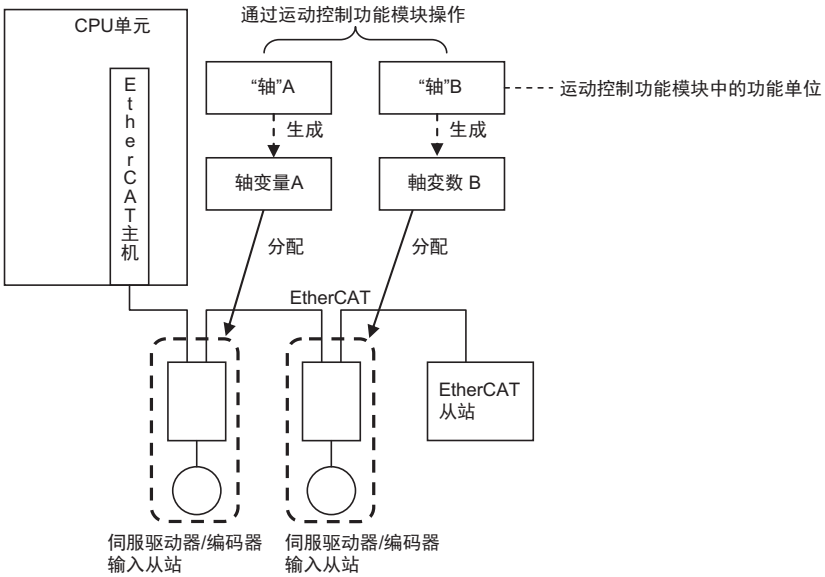
```
FOR i:= 0 TO 2 DO
  NX_ChangeWriteMode_instance[i](Execute:=FALSE);
END_FOR;

NXTable[0] := NX0;
NXTable[1] := NX1;
NXTable[2] := NX2;

FOR i:= 0 TO 2 DO
  NX_ChangeWriteMode_instance[i](Execute:=TRUE, UnitProxy:=NXTable[i]);
END_FOR;
```

3-5 “ 轴 ” 的创建和伺服驱动器 / 编码器输入从站的分配

下面介绍 NJ/NX 系列控制器上的 “ 轴 ” 创建和将伺服驱动器 / 编码器输入从站分配到 “ 轴 ”。

3-5-1 概要

用运动控制功能模块操作 EtherCAT 从站中的伺服驱动器或编码器输入从站时，需要在 Sysmac Studio 中创建 “ 轴 ”，然后分配为 EtherCAT 上的伺服轴、编码器轴。
将自动创建系统定义变量的轴变量。



利用用户程序中的运动控制指令指定轴变量，可方便地访问伺服驱动器或编码器输入从站并进行操作。

3-5-2 轴变量 / 轴组变量

轴变量 / 轴组变量分为以下种类。

变量种类		用途	访问对象的设备	生成方法
轴变量	系统定义的轴变量	用 1 个轴单独控制时	“ 轴 ” 中分配的 EtherCAT 从站 （伺服 / 编码器输入从站）	标配
	用 Sysmac Studio 创建了 “ 轴 ” 时，自动生成的轴变量			在 Sysmac Studio 中创建 “ 轴 ”，并分配轴
轴组变量	系统定义的轴组变量	对多个轴进行多轴协调控制时	“ 轴组 ” 中分配的 EtherCAT 从站 （伺服 / 编码器输入从站）	标配
	用 Sysmac Studio 创建了 “ 轴组 ” 时，自动生成的轴组变量			在 Sysmac Studio 中创建 “ 轴组 ”

轴变量 / 轴组变量的详情请参考 《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。

• 指定方法

可以用以下 Sysmac Studio 创建的变量和系统定义的变量指定。

种类	名称	
	轴变量	轴组变量
Sysmac Studio 创建的变量	MC_Axis*** （*** 按创建顺序从 000 开始升序排列） 用户可任意变更。	MC_Group*** （*** 按创建顺序从 000 开始升序排列） 用户可任意变更。
系统定义的变量	_MC_AX[***]*1 （排列要素编号按创建顺序从 0 开始升序排列）	_MC_GRP[***]*1 （排列要素编号按创建顺序从 0 开始升序排列）

*1. 在 NX 系列 CPU 单元中，_MC1_AX[***]、_MC2_AX[***] 及 _MC1_GRP[***]、_MC2_GRP[***] 也可使用。详情请参考 《NJ/NX 系列 用户手册 运动控制篇 （SBCE-CN5-363）》。

• 用途

有以下两种。

(1) 用运动控制指令进行轴指定或轴组指定：

通过指定为运动控制指令的输入输出变量，可对欧姆龙产的伺服 / 编码器输入从站进行操作。

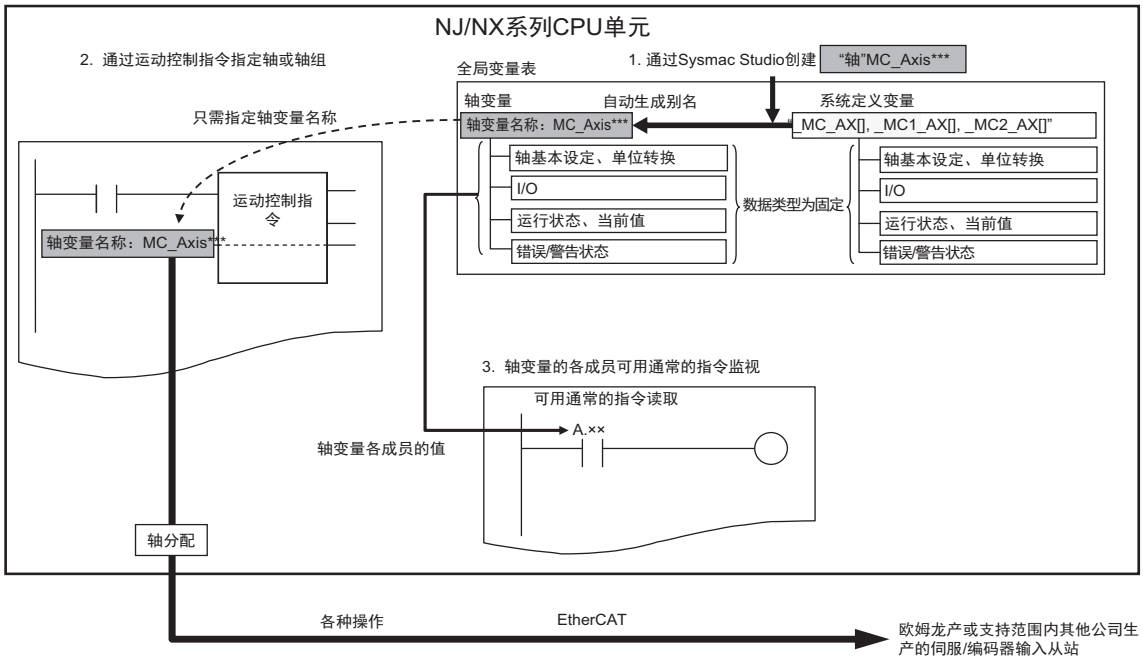
(2) 轴变量各成员的监视：

可用常用的指令对各伺服 / 编码器输入从站的当前位置或异常信息等进行监视。

参考

轴变量的详细介绍

1. 在 Sysmac Studio 中创建轴名称为 “A” 的轴。
   根据系统定义变量的轴变量，自动生成变量名称为 “A” 的 “ 轴变量 ”。轴变量由轴基本设定、单位转换、 I/O、运行状态、当前值、错误 / 警告状态构成。
2. 用运动控制指令的输入输出变量指定轴变量名称 “A”。可访问欧姆龙产或支持范围内其他公司生产的伺服 / 编码器输入从站，并进行各种操作。
3. 可根据需要指定轴变量，用常用的指令对各伺服 / 编码器输入从站的当前位置或异常信息等进行监视。



3-5-3 “ 轴 ” 及轴变量的创建和使用方法

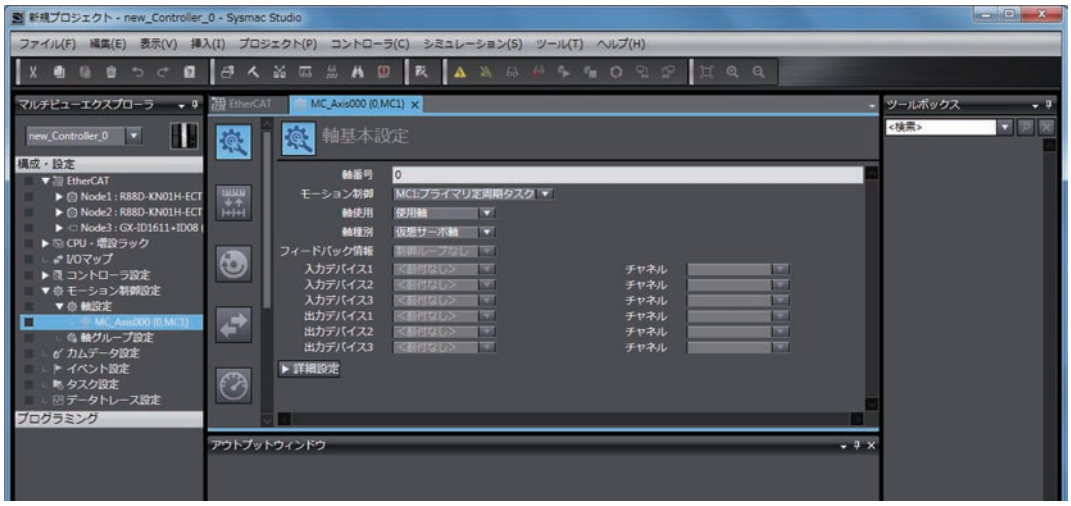
如下创建 “ 轴 ” 及轴变量并使用。

1 **右击多视图浏览器的[配置和设定]｜[运动控制设定]｜[轴设定]，从菜单中选择[添加]｜[轴设定]。**
根据需要，将默认的轴变量名称 “MC_Axis***” （*** 按创建顺序从 000 开始升序排列）改为其他名称。

2 **将Sysmac Studio的EtherCAT从站构成上的伺服驱动器/编码器输入从站分配到创建的“轴”中。**
用 Sysmac Studio 设定以下 “ 轴基本设定 ”。

区分	参数名称	设定内容
轴基本设定	轴编号	按轴的创建顺序自动设定。
	运动控制*1	选择 ［主固定周期任务］。
	轴使用	选择 ［使用轴］。
	轴种类	选择伺服轴或编码器轴。
	输入 / 输出设备	选择要分配到轴的 EtherCAT 从站。

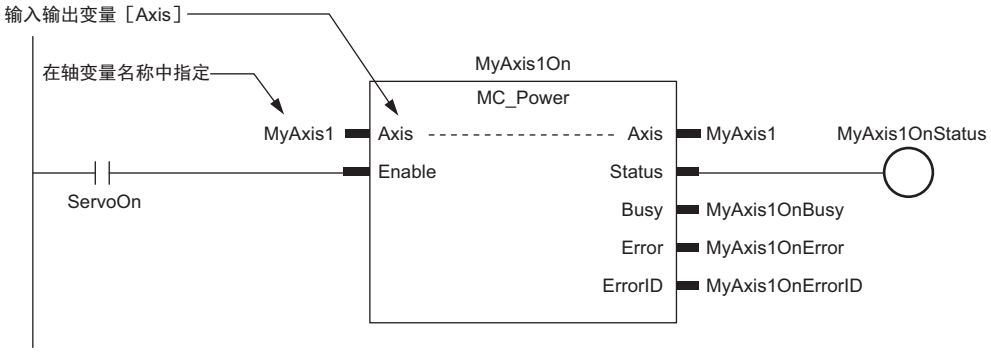
*1. NX 系列 CPU 单元时设定。



3 **用 Sysmac Studio 进行试运行所需的设定（单位转换、计数模式、极限等）及正式运行所需的设定，并作为项目传送到 CPU 单元中。**

4 **在用户程序中，对运动控制指令的输入输出变量 “Axis” 指定轴变量名称。**
轴变量名称指定为 “ 运动控制设定 ” 中插入的 “ 轴名称 ” （轴变量名称）或系统定义变量。可对相应的伺服驱动器 / 编码器输入从站执行运动控制。以下表示轴变量名称指定为 “MyAxis1” 时的示例。

例）



关于轴变量的详情，请参考 📖「3-5-2 轴变量 / 轴组变量」（P.3-17）。

使用注意事项

- NX 系列 CPU 单元时，请使用与［轴设定］及［轴组设定］的［运动控制］设定对应的系统定义变量。若使用不对应的系统定义变量，链接时将发生异常。

［运动控制］的设定	对应的系统定义变量
MC1：主固定周期任务	_MC_AX[***]、 _MC1_AX[***]、 _MC_GRP[***]、 _MC1_GRP[***]
MC2：固定周期任务 （优先度 5）	_MC2_AX[***]、 _MC2_GRP[***]

4

控制器设定

本章介绍各功能模块的初始设定。

4-1 控制器设定的概要 4-2

4-2 PLC 功能模块的初始设定 4-4

4-2-1 概要 4-4

4-2-2 控制器设定 4-4

4-2-3 任务设置 4-6

4-2-4 单元构成和单元设定 4-12

4-3 高功能单元的初始设定 4-13

4-4 运动控制功能模块的初始设定 4-15

4-4-1 概要 4-15

4-4-2 设定方法 4-16

4-5 EtherCAT 主机功能模块的初始设定 4-17

4-6 EtherNet/IP 功能模块的初始设定 4-18

4-1 控制器设定的概要

下面表示控制器设定的概要。

在整体步骤中，与以下阴影标示的位置对应。

STEP 1. 软件设计

- STEP1-1 设计 I/O 和处理
- STEP1-2 设计任务
- STEP1-3 设计程序

↓

STEP 2. 软件设定和编程

- STEP2-1 从站 / 单元构成
- **STEP2-2 控制器设定**
- STEP2-3 编程
- STEP2-4 离线调试

↓

STEP 3. 安装和硬件的设定

↓

STEP 4. 配线

↓

STEP 5. 上机确认动作和运行

详情请参考📖「1-3 NJ/NX 系列的整体使用步骤」（P.1-12）。

控制器设定	参考
● **PLC 基本模块相关的初始设定：** 控制器设定： 接通电源时的动作模式、写入保护的有无、系统服务监视设定等	📖「4-2 PLC 功能模块的初始设定」（P.4-4）
●**高功能单元的初始设定：** 单元构成和单元设定：高功能单元的初始设定	📖「4-3 高功能单元的初始设定」（P.4-13）
●**运动控制功能模块相关的初始设定：** • 轴参数：单轴动作时的运动控制用参数 • 轴组参数： 多轴协调控制动作时的运动控制用参数 • 凸轮表：凸轮动作中使用的、相位和变位间的设定表	📖「4-4 运动控制功能模块的初始设定」（P.4-15）
● **EtherCAT 主机功能模块相关的初始设定：** EtherCAT 构成上的 EtherCAT 主机参数： EtherCAT 主机的过程数据通信周期等参数设定	📖「4-5 EtherCAT 主机功能模块的初始设定」（P.4-17）
● **EtherNet/IP 功能模块相关的初始设定：** Ethernet 端口设定： EtherNet/IP 端口的 TCP/IP 设定、 Ethernet 设定等	📖「4-6 EtherNet/IP 功能模块的初始设定」（P.4-18）

4-2 PLC 功能模块的初始设定

下面介绍 PLC 功能模块的初始设定。

4-2-1 概要

PLC 功能模块初始设定包括以下内容。

- 控制器设定
- 任务设置

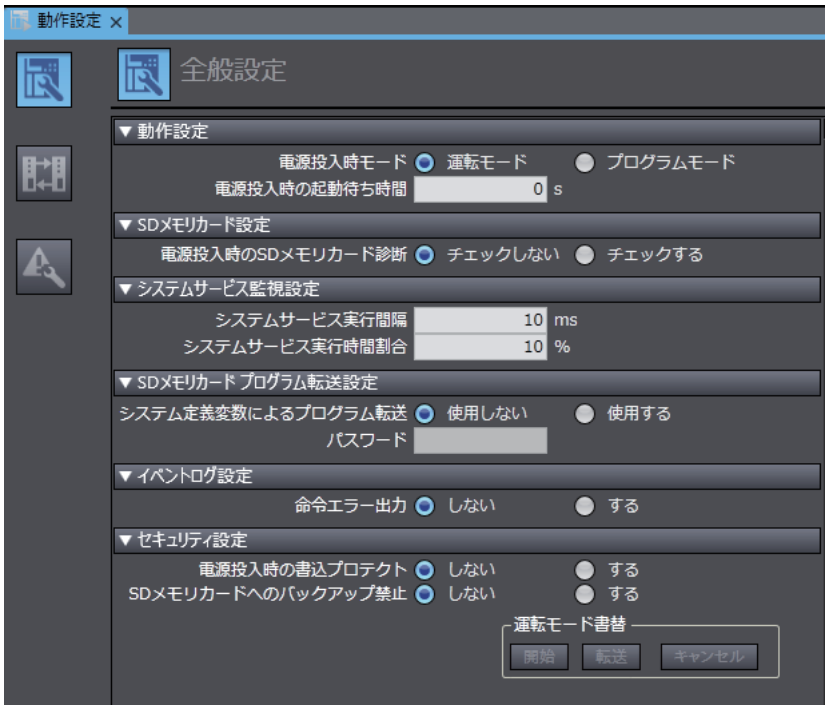
用 Sysmac Studio 分别设定 [配置和设定] ｜ [控制器设定]、[任务设定]。

4-2-2 控制器设定

动作设定标签

● 整体设定

电源接通时或动作模式切换时的动作定义等，与 CPU 单元支持的功能相关的设定。



设定项目	设定中项目	功能说明	设定值	默认	反映时序	运行模式中改写
动作设定	电源接通时模式	设定电源接通时的 CPU 单元动作模式。	运行模式 程序模式	运行模式	设定传送时	不可
	电源接通时的启动等待时间 *1	电源接通后正在启动时，设定等待 NS 系列显示器的启动，优先执行标签核对的时间。*2	0 ～ 10s	0s	设定传送时	不可
SD 存储卡设定	电源接通时的 SD 存储卡诊断	设定在电源接通时，是否对安装的 SD 存储卡执行自诊断 （文件系统的检查和修复）。	不检查 检查	不检查	设定传送时	不可
系统服务监视设定 *3	系统服务执行间隔 [ms]	设定执行系统服务的间隔。	10ms ～ 1s	10ms	设定传送时	不可
	系统服务执行时间比例 [%]	设定对 CPU 单元的整体处理执行系统服务的比例监视值。	5% ～ 50%	10%	设定传送时	不可
SD 存储卡程序传送设定*4	通过系统定义变量进行程序传送	设定是否使用 SD 存储卡程序传送。	使用 不使用	不使用	设定传送时	不可
	密码	设定执行 SD 存储卡程序传送时核对的密码。	半角英文和数字（区分大小写）， 8 个字符以上且 32 字符以内	无	设定传送时	不可
事件日志设定 *5	指令错误输出	发生指令错误时，设定是否将与之对应的事件输出到事件日志中。	否 是	否	设定传送时	不可
安全设定	电源接通时的写入保护	设定在接通控制器的电源时，是否自动启用写入保护。	否 是	否	电源接通时	可
	禁止向 SD 存储卡备份*6	设定是否禁止向 SD 存储卡备份。	否 是	否	设定传送时	可

*1. 将 NS 系列显示器连接到 CPU 单元的内置 EtherNet/IP 端口上，设定为在 NS 系列显示器和 CPU 单元同时接通电源时有效。可在 CPU 单元版本 1.10 以上和 Sysmac Studio Ver.1.12 以上的组合中使用。
*2. 利用本设定，可缩短 NS 系列显示器的标签核对处理时间。以标签核对为优先时，请设定为 “10”，不优先时，设定为 “0”。设定为 “10” 时，CPU 单元在电源接通后的 “ 启动中 ” 时，将优先执行 NS 显示器的标签核对处理约 10 秒钟，然后变为 “ 正常运行中 ”。在 “ 启动中 ” 优先执行标签核对处理的一部分，可以缩短标签核对所需的时间。
设定为 “1 ～ 10” 时，无论是否有 NS 系列显示器，都只按设定的时间优先执行标签核对处理，因此 CPU 单元变为 “ 正常运行中 ” 所需的时间会变长。未连接 NS 系列显示器时，或不同时接通电源时，请设定为 “0”。
*3. NX 系列 CPU 单元中无系统服务监视设定。
*4. 可在 CPU 单元版本 1.11 以上和 StudioVer.1.15 以上的组合中使用。
*5. 可在 CPU 单元版本 1.02 以上和 Sysmac StudioVer.1.03 以上的组合中使用。
*6. 可在 CPU 单元版本 1.03 以上和 Sysmac StudioVer.1.04 以上的组合中使用。

操作项目	功能说明
运行模式改写	更改可在运行模式中改写的设定项目之设定值。 关于操作，请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。 开始 ：变为可改写设定值的状态。 传送 ：向控制器传送设定值。将改写设定值。 Cancel(取消) ：变为禁止改写设定值的状态。

使用注意事项

将 [事件日志设定] ｜ [指令错误输出] 设定为 [是] 时，若反复执行有异常的指令，将每次输出指令错误。因此，事件日志的记录条数可能超出上限。此时，将从旧到新依次覆盖，敬请注意。

4-2-3 任务设置

● 任务设置

进行任务的追加和各任务的设定。

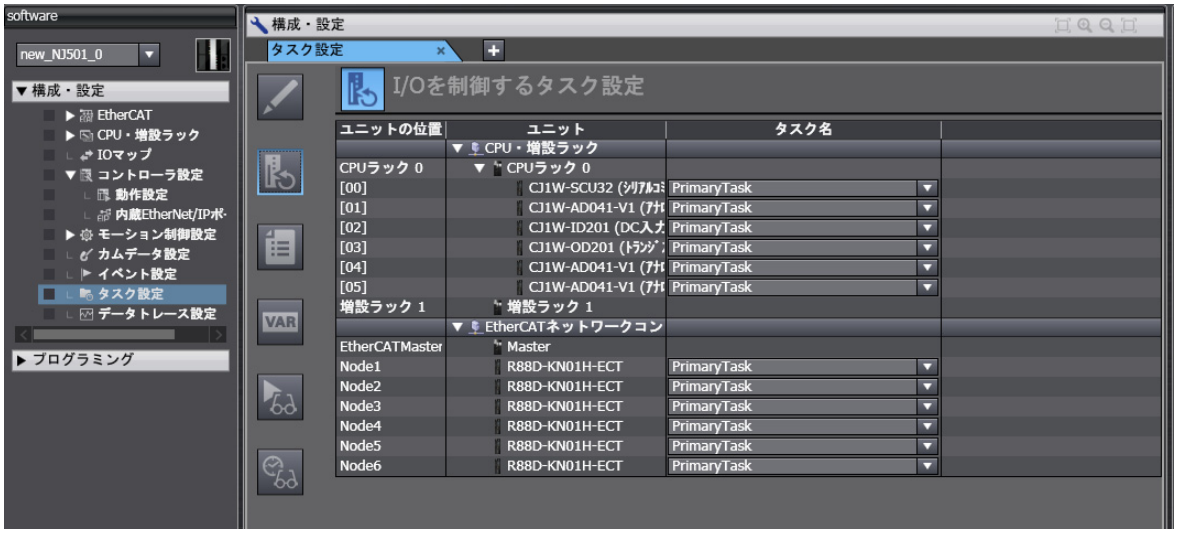
设定项目	设定中项目	功能说明	设定值	默认	反映时序	运行模式中改写
任务类型		选择任务类型。	主固定周期任务 固定周期任务〔优先度 5〕*1 固定周期任务〔优先度 16〕 固定周期任务〔优先度 17〕 固定周期任务〔优先度 18〕 事件任务 [优先度 8] 事件任务 [优先度 48]	主固定周期任务	设定传送时	不可
	任务的执行优先度	选择任务执行优先度。	根据 “ 任务类型 ” 自动选择	主固定周期任务：4	设定传送时	不可
任务名称		设定任务名称。	字符串	主固定周期任务：PrimaryTask 固定周期任务：PeriodicTask0 事件任务：EventTask0	设定传送时	不可

设定项目	设定中项目	功能说明	设定值	默认	反映时序	运行模式中改写
周期 / 执行条件		选择任务周期。	主固定周期任务： 📖参考「5-3-1 NX 系列的任务规格」（P.5-9） 固定周期任务： 📖参考「5-3-1 NX 系列的任务规格」（P.5-9） 事件任务： 通过指令执行，变量的条件式一致	主 固定周期任务：1ms*2 固定周期任务：2ms （优先度 5） 10ms （优先度 16、 17、 18） 事件任务： 通过指令执行	设定传送时	不可
任务周期超限检测		设定在超过任务周期时，是否检测错误。	• 检测 （发生轻度故障等级的控制器异常） • 不检测 （监视信息等级，记录到日志中。）	主 固定周期任务、固定周期任务：检测 事件任务： 固定为不检测	设定传送时	不可
任务执行超时时间		设定任务执行超时时间。若检测到超过超时时间，将发生任务执行超时。	主固定周期任务、固定周期任务： 周期时间 ×1 倍～周期时间 ×5 倍 事件任务： 执行优先度 8 时 1 ～ 500ms 执行优先度 48 时 1ms ～ 10s	主固定周期任务、固定周期任务： 周期时间 ×5 倍 事件任务： 执行优先度 8 时 200ms 执行优先度 48 时 1s	设定传送时	不可
变量访问时间 [%]		设定从控制器外部进行变量访问的时间与任务周期的比例。	主固定周期任务、固定周期任务： 1% ～ 50% 事件任务： 无	3%	设定传送时	不可

*1. 可使用执行优先度 5 的固定周期任务的 CPU 单元仅限 NX 系列 CPU 单元。
*2. NJ101- □□□□的主固定周期任务的默认值为 2ms。

● 设定控制 I/O 的任务

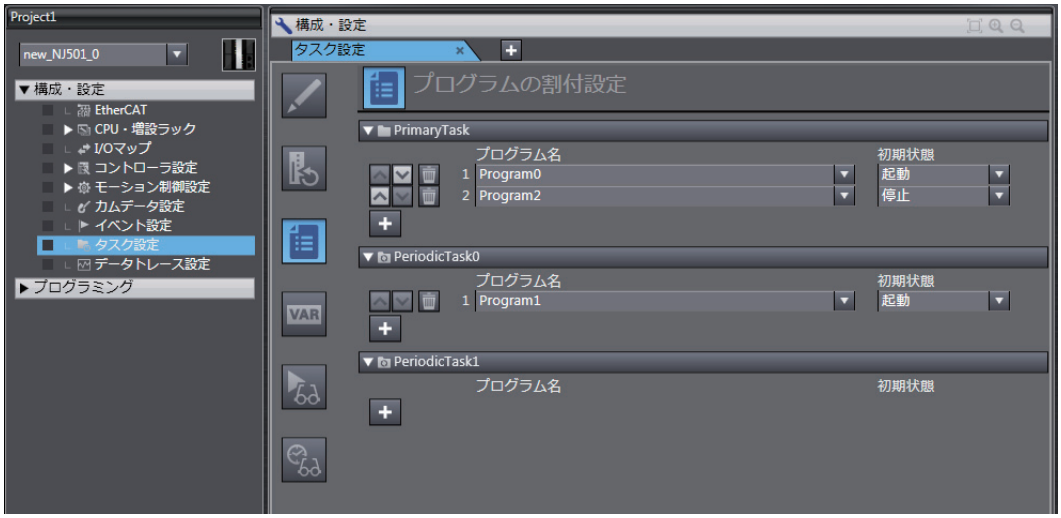
设定各输入输出的刷新执行时间。



设定项目	功能说明	设定值	默认	反映时序	运行模式中改写
任务名称	对在哪个任务中进行单元 / 从站的 I/O 刷新进行分配。	主固定周期任务 / 固定周期任务	主固定周期任务	设定传送时	不可

● 程序的分配设定

设定各任务的程序分配、程序执行顺序及各程序的运行开始时动作。



设定项目	功能说明	设定值	默认	反映时序	运行模式中改写
程序执行顺序	设定将哪个程序分配到任务中，以及该程序在任务中的执行顺序。	按程序执行顺序，从上到下分配到任务中。	Program0	设定传送时	不可
程序的初始状态*1	设定在运行开始时，是否启动该程序。	启动 停止	启动	设定传送时	不可

*1. 可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。

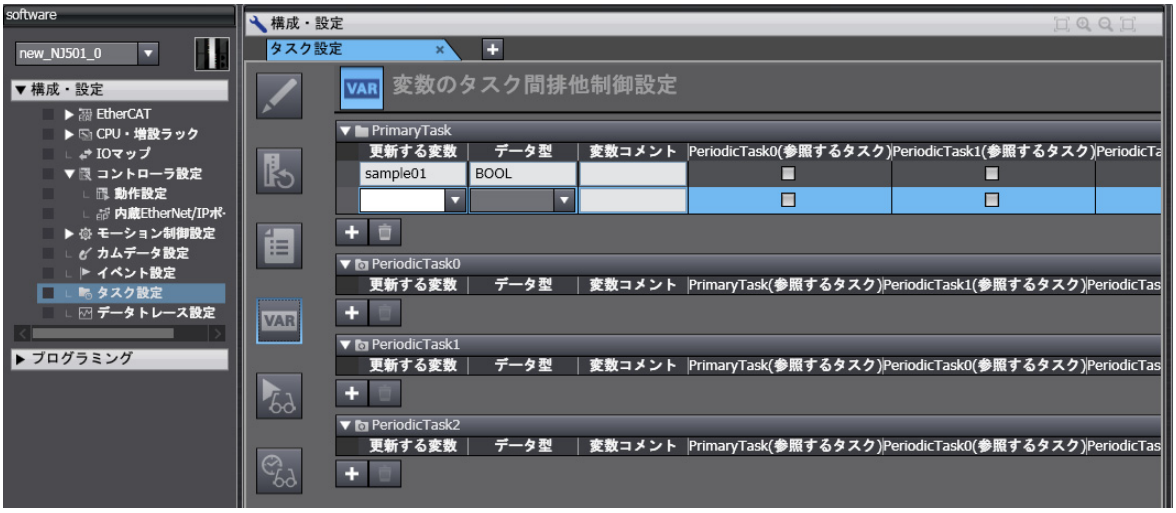
使用注意事项

- 记述有访问从站 / 单元所需的设备变量的程序，请分配到与［设定控制 I/O 的任务］中分配到对象从站 / 单元的任务相同的任务中。若分配到不同的任务中，链接时将发生异常。
- ［设定控制 I/O 的任务］的默认设定为主固定周期任务。将程序分配到主固定周期任务以外的任务时，可以事先对 ［设定控制 I/O 的任务］进行重新设定，防止链接时发生错误。
- NX 系列 CPU 单元时，对轴中分配的从站 / 单元进行控制的程序，请分配到与［轴设定］及［轴组设定］的 ［运动控制］设定对应的任务中。若分配到不对应的任务中，链接时将发生异常。此外，在程序中指定轴或轴组时，请使用与 ［运动控制］设定对应的系统定义变量。

［运动控制］的设定	与程序的分配对应的任务	对应的系统定义变量
MC1：主固定周期任务	主固定周期任务 固定周期任务 （执行优先度 16）	_MC_AX[]、 _MC1_AX[]、 _MC_GRP[]、 _MC1_GRP[]
MC2：固定周期任务 （优先度 5）	固定周期任务 （执行优先度 5）	_MC2_AX[]、 _MC2_GRP[]

● 变量的任务间排他控制设定

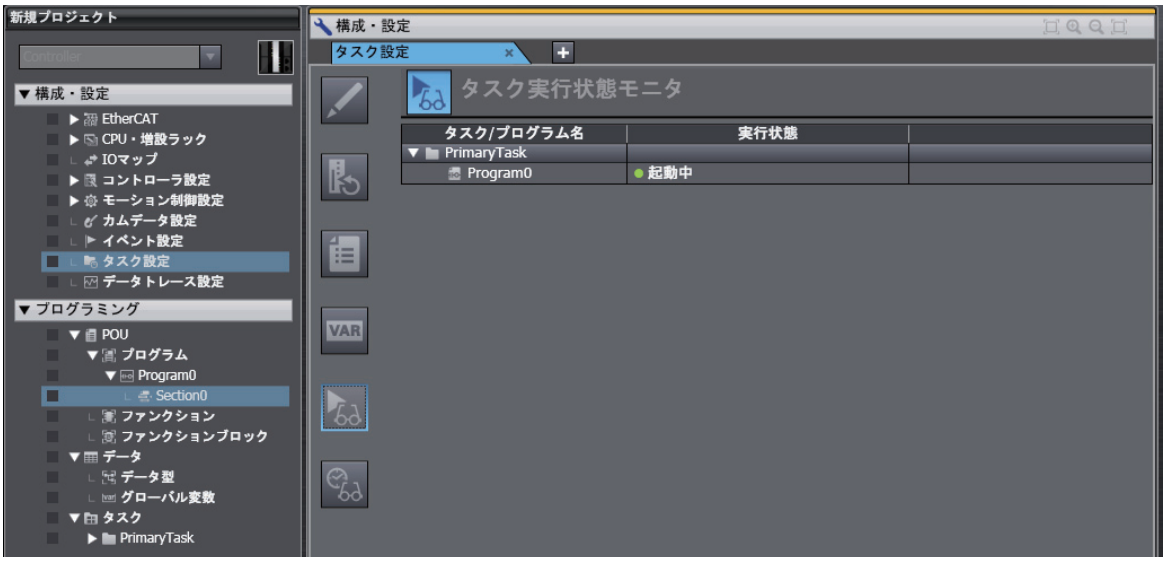
设定更新和参照全局变量的任务。



设定项目	设定中项目	功能说明	设定值	默认	反映时序	运行模式中改写
各任务	更新的变量	从主固定周期任务或固定周期任务中选择作为更新任务的变量。		无	设定传送时	不可
	数据类型	该变量的数据类型	无			
	变量注释	该变量的变量注释	无			
	参照任务	选择该变量的参照任务。				

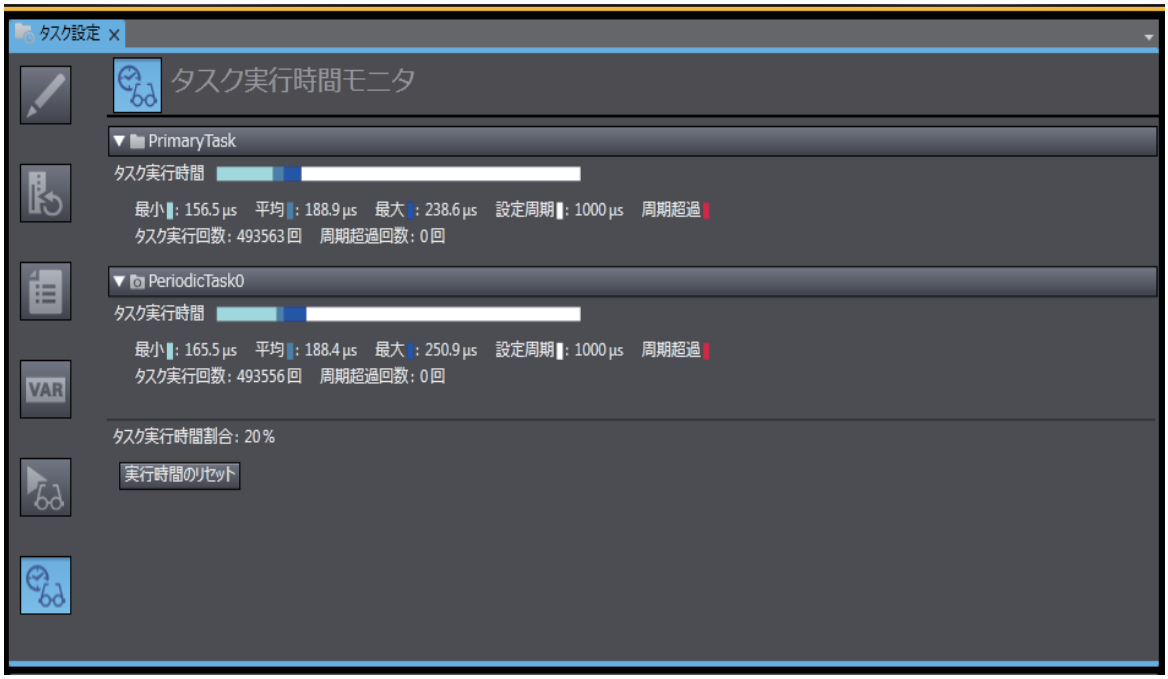
● 任务执行状态监视

显示各程序的执行状态。



● 任务执行时间监视

显示各任务的执行时间。



4-2-4 单元构成和单元设定

下面介绍 NJ 系列 CPU 单元上的设定。 NX 系列 CPU 单元中无本设定。

● 单元信息

● 所有单元共通

设定设备名称。
用单元编辑器登录单元后，将自动生成以下设备名称。
默认：“J” 后面按登录顺序从 “01” 开始连续编号
建议变更为任意名称。

参考

这里设定的 “ 设备名称 ” 将在自动生成 “ 设备变量 ” 时附加到 “I/O 端口名称 ” 前。

● 高功能单元时

请设定高功能单元的号机 No./ 单元编号。

使用注意事项

请务必将高功能单元的号机 No./ 单元编号设定为与正面旋转开关的值相同。如果不一致，实际机器将按照正面旋转开关的值动作。

● 基本 I/O 单元时

在基本 I/O 单元的单元信息中如下设定。

设定项目	设定中项目	功能说明	设定值	默认	反映时序	运行模式中改写
单元信息 ※ 以插槽为单位设定	基本输入单元的输入响应时间设定	设定基本输入单元的输入响应时间 （ON 响应时间＝ OFF 响应时间）。 可设定为无过滤器或在 0.5ms ～ 32ms 之间分段设定。 增大该值，可减少抖动或外部干扰的影响。 反之，减小该值，可以接收（比任务周期更长的条件下的）短脉冲输入。	无过滤器 0.5ms 1ms 2ms 4ms 8ms 16ms 32ms	8ms	电源接通时（包括 CPU 重置时）	不可

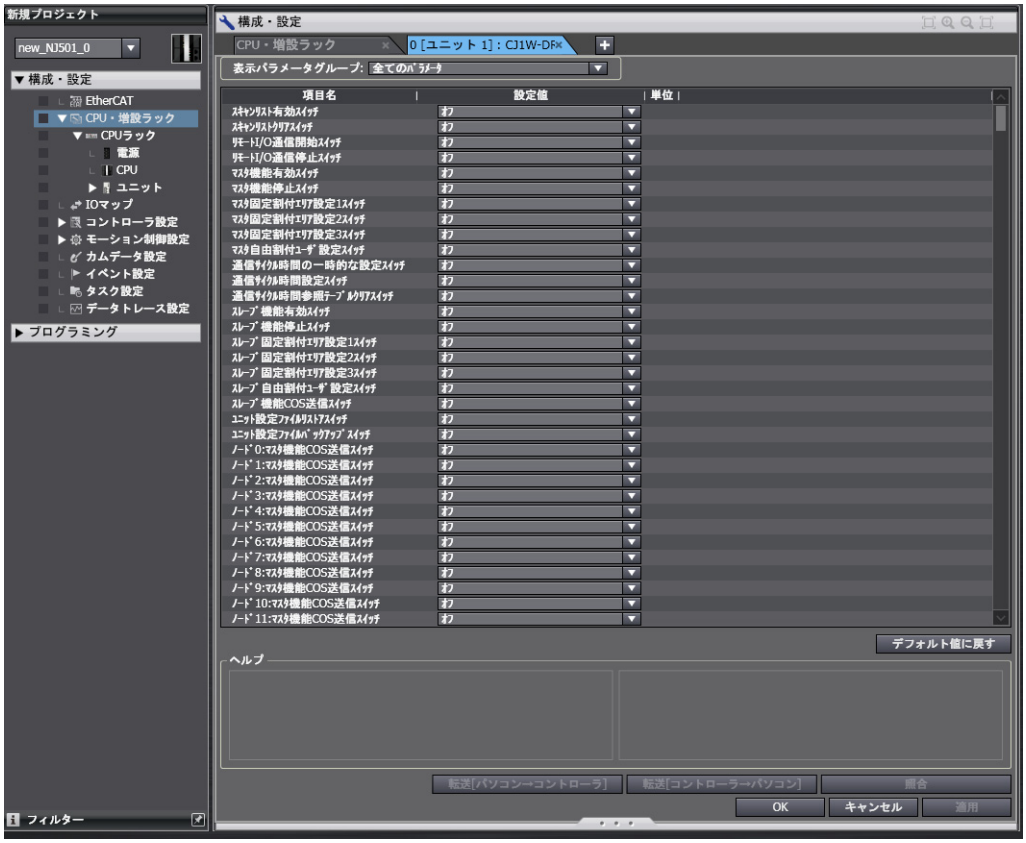
4-3 高功能单元的初始设定

下面介绍 NJ 系列 CPU 单元上的高功能单元初始设定。 NX 系列 CPU 单元中无本设定。
高功能单元的初始设定通过以下任意方法进行。

方法 1：从 Sysmac Studio 的单元设定画面进行设定

1 **在单元构成和单元设定中选择单元。**

2 **在以下单元设定画面中设定。**



3 **在线连接到 CPU 单元上，并传送设定。**

方法 2：用 Sysmac Studio 的 I/O 映射对 I/O 端口设定值

1 **用 Sysmac Studio 的 I/O 映射对 I/O 端口设定值。**

2 **重启该单元，或重置控制器 / 重新接通控制器的电源。**

方法 3：用 Sysmac Studio 设定 CJ 单元用设备变量的初始值

1 **用 Sysmac Studio 设定 CJ 单元用设备变量的初始值。**

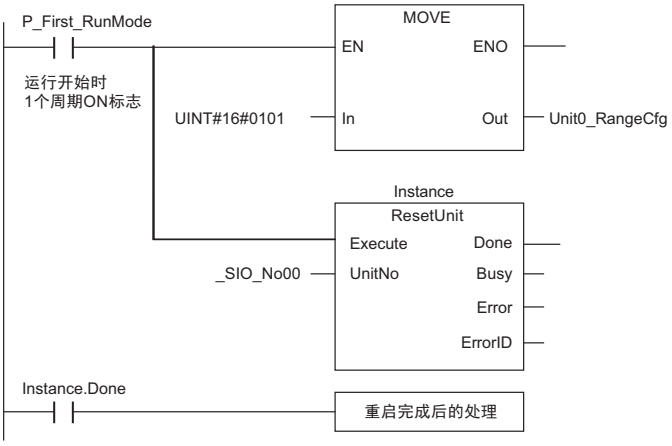
2 **用 Sysmac Studio 将变量表传送到 CPU 单元上。**
此时，勾选 “ 初始化保持变量的当前值 ”。

3 **重启该单元，或重置控制器 / 重新接通控制器的电源。**

方法 4：通过指令设定 CJ 单元用设备变量的值

1 **在用户程序中，开始运行时通过 MOVE（传送）指令等设定 CJ 单元用设备变量的值，然后重启该单元。**

例）



安全要点

变更高功能单元的设定并重启时，请充分确认连接目标的对象设备绝对安全。

4-4 运动控制功能模块的初始设定

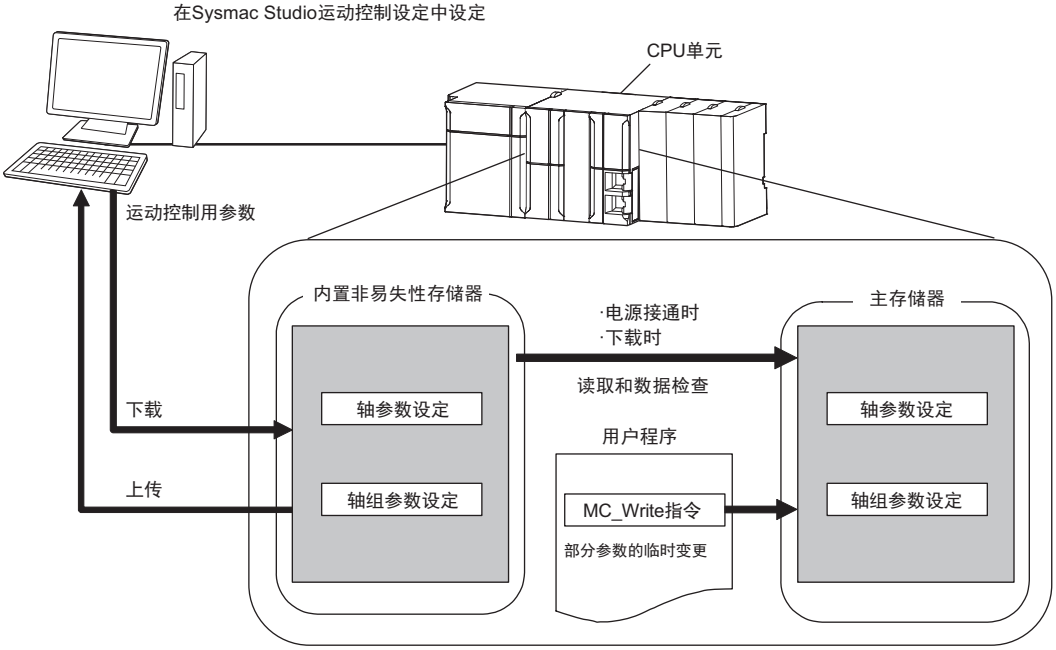
下面介绍运动控制功能模块的初始设定。

4-4-1 概要

运动控制功能模块的初始设定称为 “ 运动控制参数 ”。

“ 运动控制参数 ” 包括以下内容。

- 轴参数：单轴控制所需的设定
- 轴组参数：多轴协调控制所需的设定

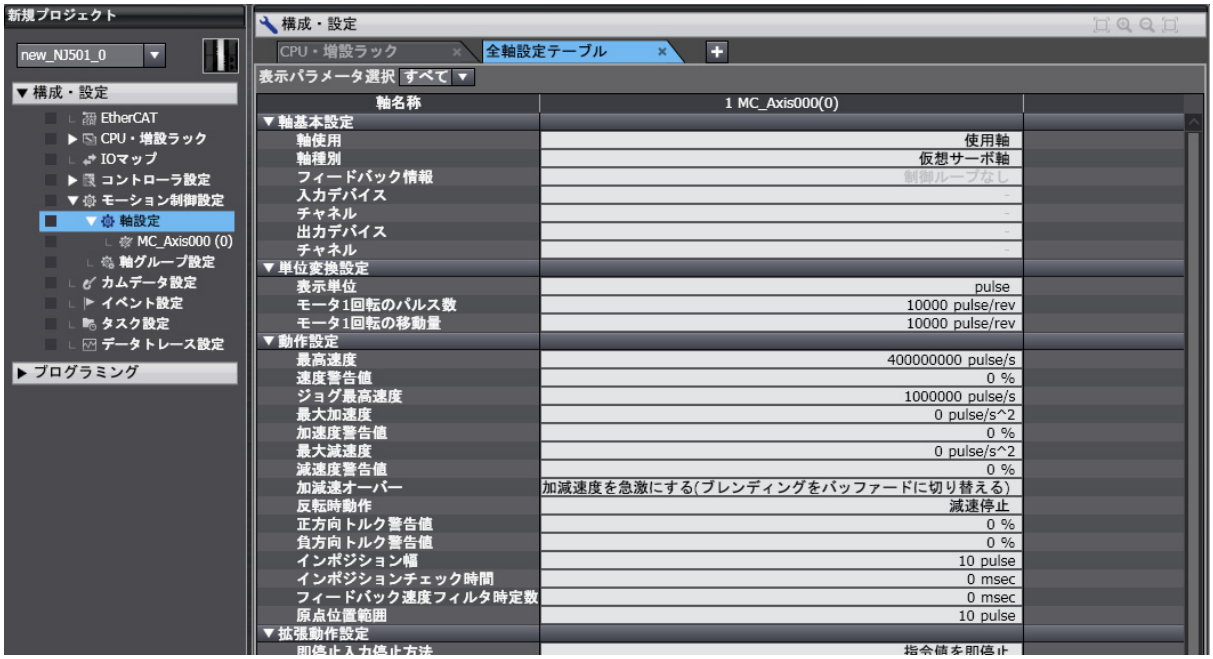


4-4-2 设定方法

运动控制用参数的设定通过以下任意方法进行。

方法 1：通过 Sysmac Studio 的运动控制设定进行设定

从 Sysmac Studio 中右击以下 [配置和设定] | [运动控制设定] | [轴设定]，在 [轴设置表] 中进行设定。



若将运动控制用参数下载到 CPU 单元，将保存到 CPU 单元内置的非易失性存储器中。电源接通时或下载时，下载的设定数据变为有效。

方法 2：用 MC_Write 指令设定

部分运动控制用参数可通过 MC_Write 指令临时替换。

详情请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。

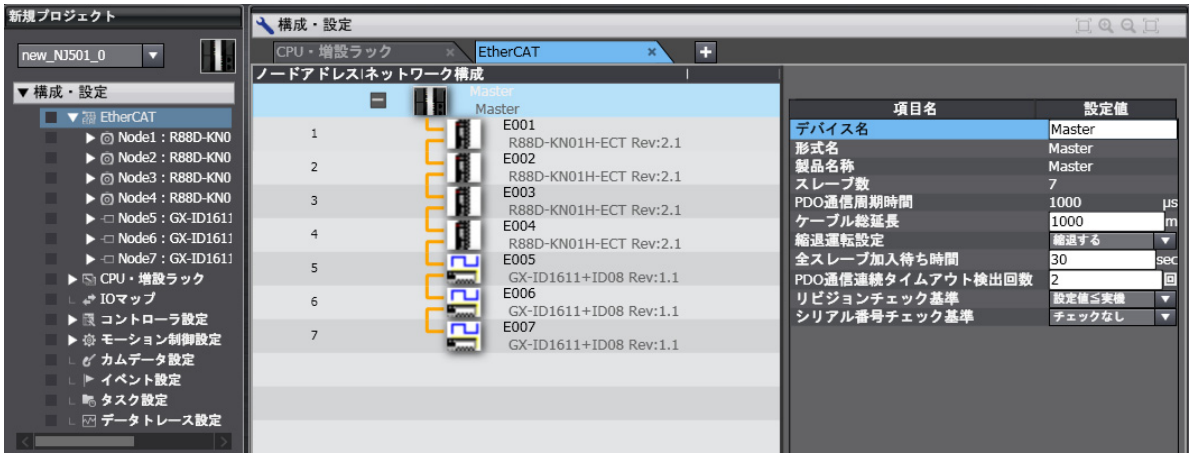
4-5 EtherCAT 主机功能模块的初始设定

下面介绍 EtherCAT 主机功能模块的初始设定。

EtherCAT 主机功能模块的初始设定包括以下内容。

- 设备名称
- 电缆总长
- 低效运行设定
- 所有从站加入等待时间
- PDO 通信连续超时检测次数
- 修订版本检查基准
- 序列号检查基准

在 Sysmac Studio 中选择 [配置和设定] | [EtherCAT]，然后选择 Master，将显示 EtherCAT 主机功能模块的初始设定画面。



详情请参考📖《NJ/NX 系列 CPU 单元内置 EtherCAT 端口 用户手册 (SBCD-CN5-358)》。

4-6 EtherNet/IP 功能模块的初始设定

下面介绍 EtherNet/IP 功能模块的初始设定。

EtherNet/IP 功能模块的初始设定包括以下内容。

- TCP/IP 设定
- LINK 设定
- FTP 设定
- NTP 设定
- SNMP 设定
- SNMP 捕捉设定
- FINS 设定 (*1)

*1 NX 系列 CPU 单元中无 FINS 设定。

用 Sysmac Studio 在 [配置和设定] ｜ [控制器设定] ｜ [内置 EtherNet/IP 端口设定] 中设定。

详情请参考📖《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册 (SBCD-CN5-359)》。

5

设计任务

本章介绍任务的原理和种类。

5-1 任务设计步骤的概要 5-3

5-2 任务的概要 5-5

5-2-1 任务定义 5-5
5-2-2 任务相关的指令 5-7
5-2-3 任务相关的系统定义变量 5-7

5-3 NX 系列的任务规格和基本动作 5-9

5-3-1 NX 系列的任务规格 5-9
5-3-2 NX 系列的任务分配说明 5-10
5-3-3 NX 系列的任务基本动作 5-10
5-3-4 NX 系列事件任务的执行条件 5-19
5-3-5 NX 系列事件任务的执行时序 5-23
5-3-6 事件任务完成前，执行条件再次成立时的动作 5-27

5-4 NJ 系列的任务规格和基本动作 5-28

5-4-1 NJ 系列的任务规格 5-28
5-4-2 NJ 系列的任务分配说明 5-29
5-4-3 NJ 系列的任务基本动作 5-29
5-4-4 NJ 系列事件任务的执行条件 5-37
5-4-5 NJ 系列事件任务的执行时序 5-41
5-4-6 事件任务完成前，执行条件再次成立时的动作 5-46

5-5 标签数据链接服务和系统服务 5-47

5-5-1 标签数据链接服务和系统服务的执行优先度和执行顺序 5-48
5-5-2 标签数据链接服务的处理内容和执行时序 5-50
5-5-3 系统服务的处理内容和执行时序 5-54

5-6 任务相关的分配和设定 5-57

5-6-1 分配到 I/O 刷新的任务中 5-57
5-6-2 将程序分配到任务中 5-65
5-6-3 任务的设定项目 5-66

5-7 确保变量的同时性 5-69

5-7-1 确保任务间变量值同时性的方法 5-69
5-7-2 从控制器外部进行的变量访问 5-75

5-8 任务相关的异常 5-79

5-9 任务执行状态、任务执行时间的监视 5-82

5-10 任务设计方法及输入输出响应时间 5-87
5-10-1 任务执行时间的确认 5-87
5-10-2 任务设计示例 5-90
5-10-3 系统输入输出响应时间 5-91

5-1 任务设计步骤的概要

下面介绍任务设计步骤的概要。

在整体步骤中，与以下阴影标示的位置对应。

STEP 1. 软件设计

- STEP1-1 设计 I/O 和处理
- **STEP1-2 设计任务**
- STEP1-3 设计程序

↓

STEP 2. 软件设定和编程

- STEP2-1 从站 / 单元构成
- STEP2-2 控制器设定
- STEP2-3 编程
- STEP2-4 离线调试

↓

STEP 3. 安装和硬件的设定

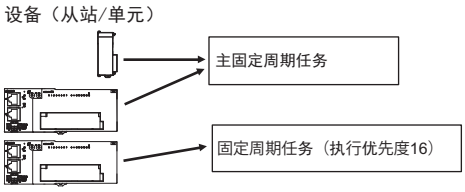
↓

STEP 4. 配线

↓

STEP 5. 上机确认动作和运行

详情请参考📖「1-3 NJ/NX 系列的整体使用步骤」（P.1-12）。

任务设计	参考
● **任务构成的设计** 根据控制对象装置所需的输入输出响应性能，设计任务构成。	「5-3-3 NX 系列的任务基本动作」（P.5-10） 「5-4-3 NJ 系列的任务基本动作」（P.5-29） 「5-10 任务设计方法及输入输出响应时间」（P.5-87）
● **设计哪个设备 （从站 / 单元）在主固定周期任务还是固定周期任务 （执行优先度 5、16）中执行 I/O 刷新**  指定在哪个任务中进行I/O刷新	「5-6-1 分配到 I/O 刷新的任务中」（P.5-57）
● **设计将哪个程序分配到主固定周期任务、固定周期任务（执行优先度 5、16 ～ 18）、事件任务 （执行优先度 8、 48）中的哪一个**	「5-6-2 将程序分配到任务中」（P.5-65）
● **设计任务间变量的排他方法** 在不同的任务中使用相同的全局变量时，设计任务间变量的排他方法。	「5-7-1 确保任务间变量值同时性的方法」（P.5-69）
● **从控制器外部访问变量时的任务设定** 从控制器外部访问 CPU 单元内的变量时，需要将变量访问和特定任务的用户程序执行同步时设计。变量访问中包括 EtherNet/IP 标签数据连接。	「5-7-2 从控制器外部进行的变量访问」（P.5-75）

Sysmac Studio 上的任务设定

任务的设定	参考
● **PLC 基本模块相关的初始设定** 任务设定：任务周期、输入输出设定、程序分配任务间接口设定等	「4-2 PLC 功能模块的初始设定」（P.4-4）

Sysmac Studio 上的离线调试

上机动作确认	参考
● **用模拟器 （虚拟 PLC）对顺序控制和运动控制进行上机调试。** ● **在任务执行时间监视画面中监视任务执行时间。**	「第 7 章 动作确认和运行」(P.7-1) 「5-9 任务执行状态、任务执行时间的监视」（P.5-82）

5-2 任务的概要

下面介绍任务的概要。

5-2-1 任务定义

任务是指在 I/O 刷新或用户程序执行等一系列处理过程中，指定执行条件和执行顺序的功能。

因执行条件及执行优先度不同，任务分为以下 3 种。

任务类型	任务数	任务执行优先度	定义	主要处理内容
主固定周期任务	1	4	按设定的任务周期执行 1 次。 优先于任何任务执行。 主固定周期任务的运动控制指令及 EtherCAT 的通信按主固定周期任务的周期执行。	I/O 刷新、用户程序执行、运动控制
固定周期任务	0 ～ 1	5*1	按设定的任务周期执行 1 次。 执行优先度仅次于主固定周期任务。 执行优先度 5 的固定周期任务的运动控制指令及 EtherCAT 的通信按执行优先度 5 的固定周期任务的周期执行。	I/O 刷新、用户程序执行、运动控制
	0 ～ 3	16、 17、 18	按设定的任务周期执行 1 次。 执行优先度 16 的固定周期任务的运动控制指令及 EtherCAT 的通信按主固定周期任务的周期执行。	处理内容因任务执行优先度不同而异。 执行优先度 16：I/O 刷新、用户程序执行 执行优先度 17、 18：用户程序执行
事件任务	0 ～ 32	8、 48	在指定的执行条件成立时，仅执行 1 次。	用户程序执行

*1. 可使用执行优先度 5 的固定周期任务的 CPU 单元仅限 NX 系列 CPU 单元。

版本相关信息

事件任务可在 CPU 单元版本 1.03 以上和 Sysmac Studio Ver.1.04 以上的组合中使用。

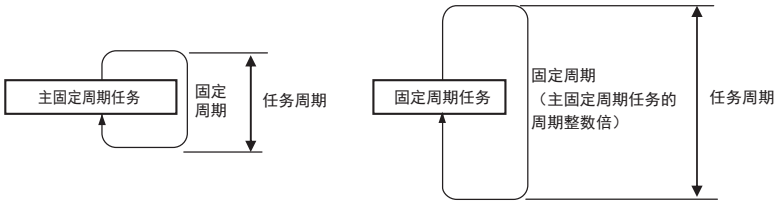
参考

在 NX 系列 CPU 单元上，可使用主固定周期任务和固定周期任务 （执行优先度 5）执行运动控制。需要区分两种运动控制时，分别称为 “ 运动控制 1” 和 “ 运动控制 2”。

● 主固定周期任务和固定周期任务

主固定周期任务及固定周期任务都按固定周期执行。

主固定周期任务及固定周期任务按固定周期执行时的间隔称为 “ 任务周期 ”。



各任务中可分配 1 ～ 128 个程序。分配的程序按分配顺序执行。这称为 “ 用户程序执行 ”。

与 CJ 单元、 EtherCAT 从站的数据交换称为 I/O 刷新。

I/O 刷新可按从站 / 单元分配到主固定周期任务或固定周期任务 （执行优先度 5、 16）中。默认为所有从站 / 单元的 I/O 刷新都分配到主固定周期任务中。

● 事件任务

事件任务在指定的执行条件成立时，执行 1 次。事件任务的执行条件有以下两种。

执行条件	规格
通过指令执行	若执行事件任务执行指令 ActEventTask，将执行事件任务。
变量的条件一致后执行	若指定的变量与事先设定的条件一致，则执行事件任务。

各任务中可分配 1 ～ 128 个程序。分配的程序按分配顺序执行。

使用注意事项

- 事件任务中不会执行I/O刷新或运动控制。因此，带I/O控制的程序或包括运动控制指令的程序不可分配到事件任务中。
- 事件任务不是每个任务周期重复执行的任务。因此，若程序中包含跨越多个任务周期的指令，将无法分配到事件任务中。跨越多个任务周期执行的指令是指部分基本指令，如 SD 存储卡指令、通信指令，以及所有的运动控制指令和所有的模拟指令。事件任务中无法使用的基本指令详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

参考

NX 系列 CPU 单元和 NJ 系列 CPU 单元上任务的动作不同。详情请参考📖「5-3 NX 系列的任务规格和基本动作」（P.5-9）及📖「5-4 NJ 系列的任务规格和基本动作」（P.5-28）。

5-2-2 任务相关的指令

任务相关的指令如下。

指令	指令名称	概要	
GetMyTaskStatus	读取当前任务状态	读取当前任务的以下任务状态。 上次任务执行时间、任务执行时间最大值、任务执行时间最小值、任务执行次数、任务周期超过标志、任务周期超过次数	
GetMyTaskInterval	读取当前任务设置周期	读取当前任务的任务周期。	
Task_IsActive	任务执行中判定	返回指定的任务是否正在执行。	
Lock	任务间独占锁定	开始锁定区间。	其他任务中相同编号的锁定区间将无法同时执行。
Unlock	解除任务间独占锁定	结束锁定区间。	
ActEventTask	事件任务启动	启动指定的事件任务。	

5-2-3 任务相关的系统定义变量

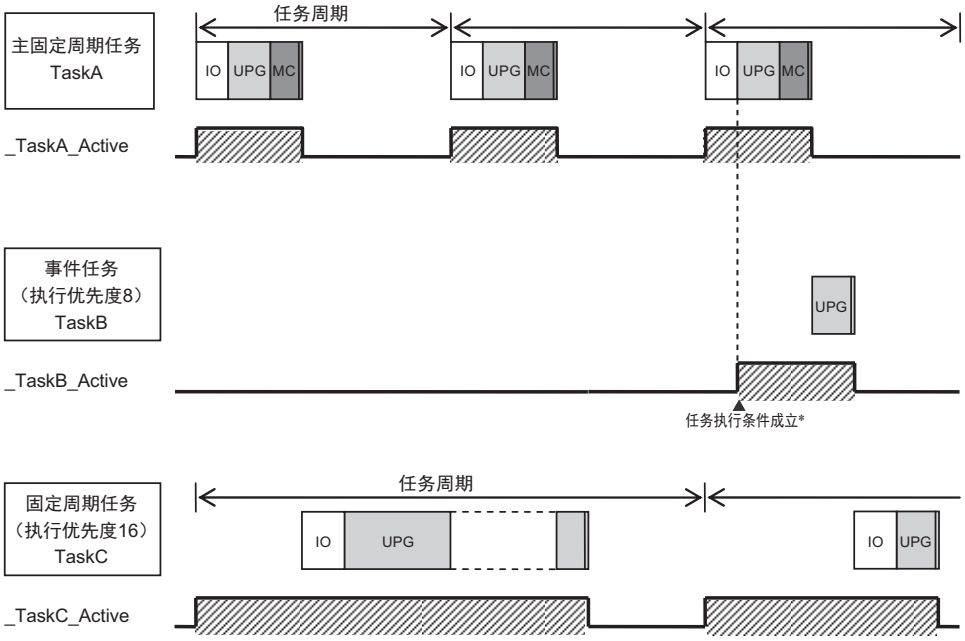
如下所示，表示任务状态的系统定义变量按不同任务分别准备。
但是，出于实时性及与各功能模块异常状态的同时性考虑，在用户程序中无法使用。仅在通过 Sysmac Studio 的数据追踪功能对任务状态进行采样时使用。
通过用户程序参照任务的状态时，请使用 GetMyTaskStatus 指令、 Task_IsActive 指令。

变量名称	名称	功能	数据类型	R/W
_< 任务名称 >_Active	任务执行中标志	正在执行任务时为 TRUE。 任务执行完成后到任务周期结束为止，为 FALSE。	BOOL	R
_< 任务名称 >_LastExecTime	上次任务执行时间	表示上次执行的任务执行时间。	TIME	R
_< 任务名称 >_MaxExecTime	任务执行时间最大值	表示任务执行时间的最大值。	TIME	R
_< 任务名称 >_MinExecTime	任务执行时间最小值	表示任务执行时间的最小值。	TIME	R
_< 任务名称 >_ExecCount	任务执行次数	表示任务的执行次数。 若当前值超出数据类型的最大值，当前值将恢复为 0，然后继续计数。	UDINT	R
_< 任务名称 >_Exceeded	任务周期超出标志	超出任务周期后，在任务执行完成时变为 TRUE。 若在任务周期内完成任务执行，则变为 FALSE。	BOOL	R
_< 任务名称 >_ExceedCount	任务周期超出次数	保存超出任务周期的次数。 若当前值超出数据类型的最大值，当前值将恢复为 0，然后继续计数。	UDINT	R

（注）例：任务名称 “MainTask” 的任务周期超过标志为 “_MainTask_Exceeded”。

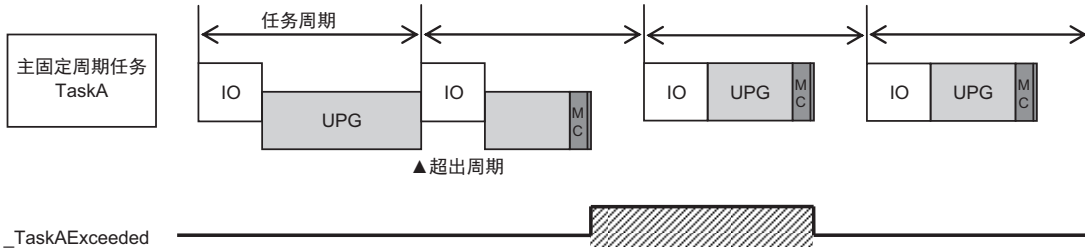
标志动作

● 任务执行中标志 （_< 任务名称 >_Active）



*用ActEventTask指令启动事件任务时，ActEventTask执行过程中，任务执行中标志将变为ON。

● 任务周期超出标志 （_< 任务名称 >_Exceeded）



5-3 NX 系列的任务规格和基本动作

下面介绍配备多核处理器的 NX 系列 CPU 单元的任务规格和基本动作。

5-3-1 NX 系列的任务规格

任务的规格如下所示。

项目	规格
任务类型	• 主固定周期任务 • 固定周期任务 （优先度 5、 16、 17、 18） • 事件任务 （优先度 8、 48）
任务个数	• 主固定周期任务：1 • 固定周期任务：0 ～ 4*1 • 事件任务：0 ～ 32*2
1 个任务的程序数	最大 128
主固定周期任务的任务周期	125μs 250μs ～ 8ms （可以 250μs 为单位设定）
固定周期任务的任务周期	• 优先度 5 125μs 250μs ～ 100ms （可以 250μs 为单位设定） • 优先度 16、 17、 18 1ms ～ 100ms （可以 250μs 为单位设定） 固定周期任务的任务周期设定为主固定周期任务的整数倍。 不可选择各任务的任务周期最小公倍数超出 600ms 的组合。

*1. 执行优先度 5、 16、 17、 18 分别为 0 或 1 个。
*2. 执行优先度 8、 48 分别为 0 ～ 32 个，两者加起来最多 32 个。

使用注意事项

使用固定周期任务 （执行优先度 5）时，请勿将主固定周期任务的任务周期设定为 4ms 以上。若设定为 4ms 以上，可能会发生从站应用异常。

5-3-2 NX 系列的任务分配说明

所有程序都需要分配到任意一个任务中。请根据程序的特性，按以下方法决定要分配的任务。

任务	适合分配的程序
主固定周期任务	• 需要严格遵守各周期的 I/O 刷新、用户程序执行、运动控制、系统共通处理的执行周期。 • 包括执行优先度最高、需要高速响应的控制。 • 包括执行优先度最高的运动控制指令。
固定周期任务 （优先度 5）	• 需要严格遵守各周期的 I/O 刷新、用户程序执行、运动控制、系统共通处理的执行周期。 • 包括执行优先度仅次于主固定周期任务、需要高速响应的控制。 • 包括执行优先度较高的运动控制指令。 • 将需要以最快速度控制的处理分配到主固定周期任务，将其他处理分配到固定周期任务 （执行优先度 5）中进行控制时使用。
固定周期任务 （优先度 16）	• 执行优先度相对较低，但需要按周期执行用户程序、系统共通处理。 • 包括将 I/O 刷新分配到主固定周期任务中的部分从站 / 单元的控制。 • 包括执行优先度较低的运动控制指令。 • 在针对主固定周期任务中控制的从站 / 单元的处理中，将执行优先度相对较低的处理分配到固定周期任务 （执行优先度 16）进行控制时使用。
固定周期任务 （优先度 17,18）	• 执行优先度相对较低，但需要按周期执行用户程序、系统共通处理。 • 包括不需要高速响应的运算处理、通信处理。
事件任务	• 仅在特定条件成立时执行。

5-3-3 NX 系列的任务基本动作

NX 系列 CPU 单元配备多核处理器，可并列执行主固定周期任务和固定周期任务 （执行优先度 5）。各任务按什么样的顺序执行，取决于各任务中设定的执行优先度。

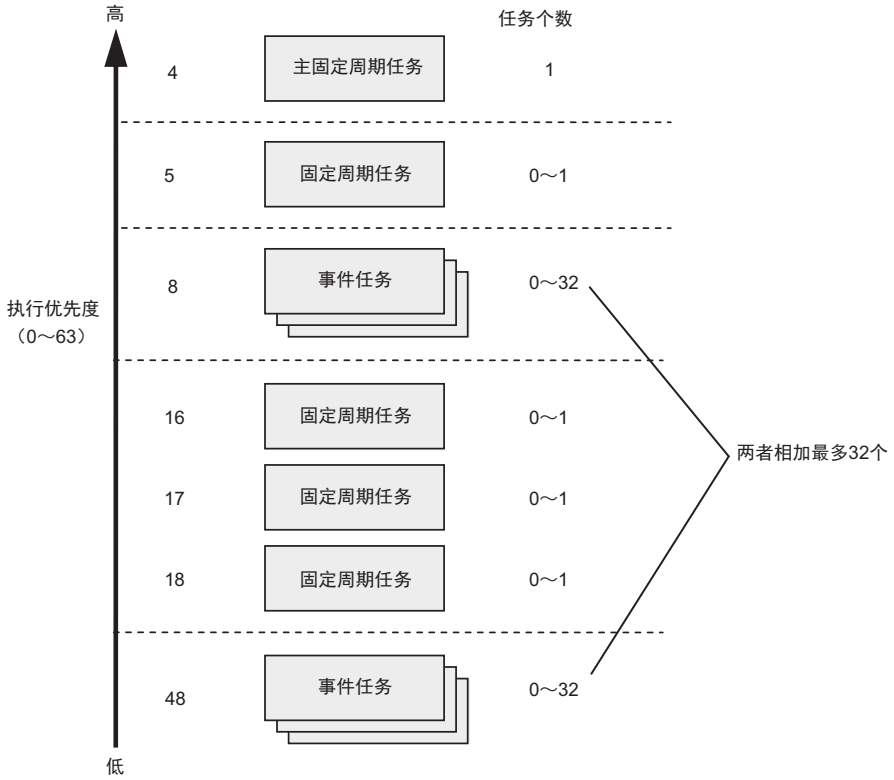
参考

NX 系列 CPU 单元中，可在执行任务时，并列执行标签数据链接服务和系统服务。

任务执行优先度

任务根据其类型决定执行优先度。 CPU 单元将优先执行优先度较高的任务。
正在执行某任务 Ta 时，若执行优先度比它高的 Tb 执行条件成立，将优先把 Tb 的处理分配到处理器。
不同任务类型的执行优先度如下所示。执行优先度的值越小，优先度越高。

任务	执行优先度	执行优先度的重复
主固定周期任务	4	—
固定周期任务	5、 16、 17、 18	不可将多个任务设定为相同的执行优先度。
事件任务	8、 48	可将多个任务设定为相同的执行优先度。 此时的执行顺序请参考「5-3-5 NX 系列事件任务的执行时序」（P.5-23）。



主固定周期任务和固定周期任务的任务周期

主固定周期任务和固定周期任务是指按周期反复执行的任务。
固定周期任务的任务周期必须设定为主固定周期任务的任务周期（称为“主周期”）的整数倍。因此，几次后两者的执行时序会重叠。
例如，设定为主周期 1ms、固定周期任务（执行优先度 16）的任务周期 4ms 时，则主固定周期任务每执行 4 次，主固定周期任务和固定周期任务的执行时序重叠。

参考

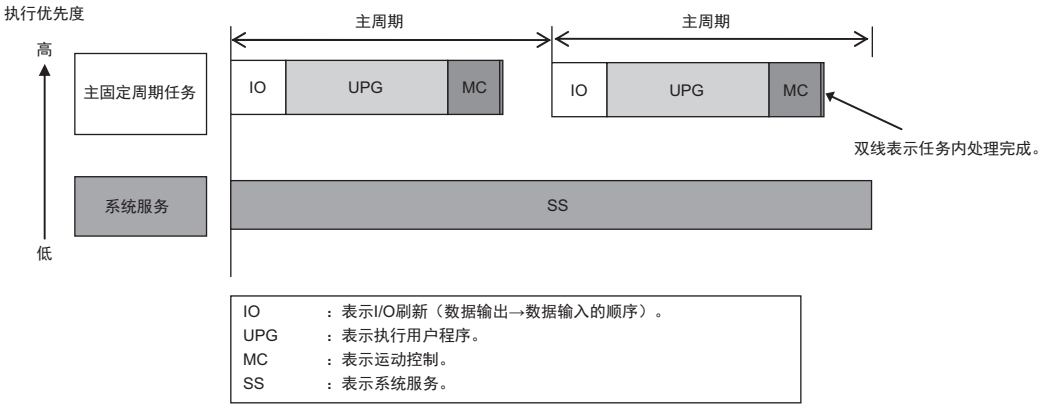
事件任务是指不按周期执行，而是在指定的执行条件成立时，仅执行 1 次的任务。因此，事件任务的执行时序取决于事件任务的执行条件成立的时间和事件任务的执行优先度。

任务的执行顺序示例

下面举例说明主固定周期任务和固定周期任务的执行顺序。
事件任务的执行顺序请参考「5-3-5 NX 系列事件任务的执行时序」（P.5-23）。

● 仅主固定周期任务时

按主固定周期执行主固定周期任务。
图中的系统服务是指通信处理等 CPU 单元在任务之外执行的处理。系统服务的详情请参考「系统服务的处理内容」（P.5-54）。

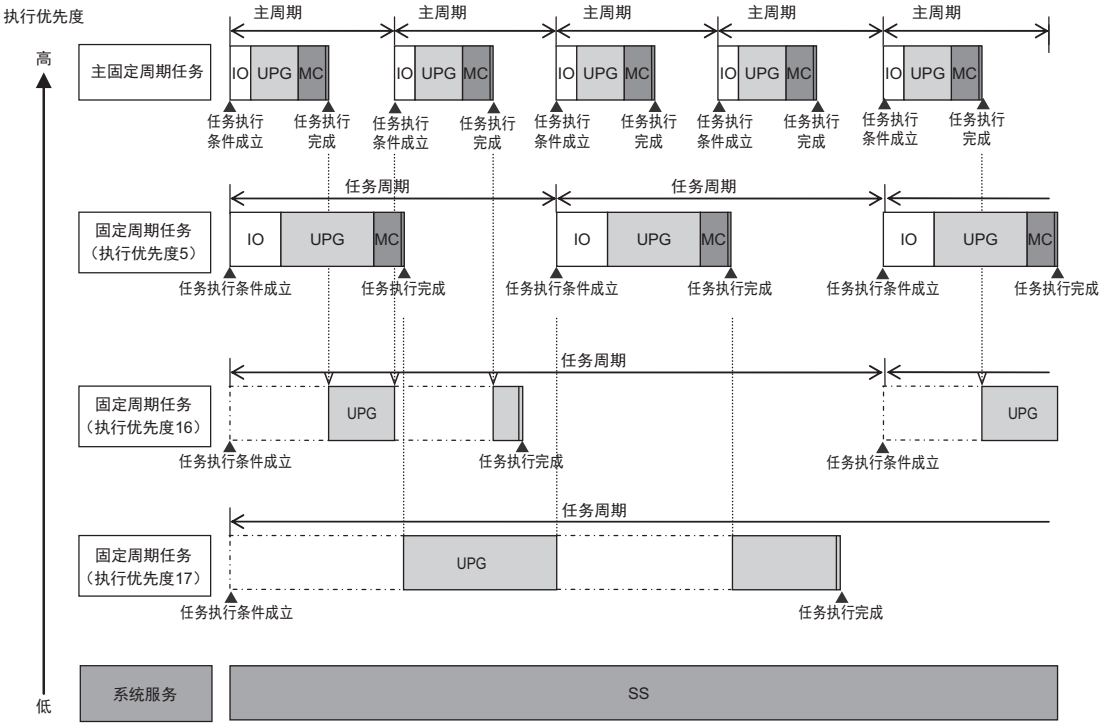


● **主固定周期任务、固定周期任务（执行优先度 5）、固定周期任务（执行优先度 16）、固定周期任务 （执行优先度 17）同时存在时**

- 各任务分为以下两组执行。

组	任务类型
组 1	主固定周期任务 固定周期任务 （执行优先度 16）
组 2	固定周期任务 （执行优先度 5、 17、 18） 事件任务 （执行优先度 8、 48）

- 同一组内的各任务按执行优先度从高到低的顺序优先执行。
- 主固定周期任务的执行优先度最高，因此将立即按主周期执行。
- 固定周期任务 （执行优先度 5）与主固定周期任务并列执行。
- 固定周期任务 （执行优先度 16）待主固定周期任务完成后执行。
- 固定周期任务 （执行优先度 17）的执行优先度低于固定周期任务 （执行优先度 5），因此在未执行固定周期任务 （执行优先度 5）的时间内执行。
- 本示例中，固定周期任务 （执行优先度 5）的任务周期为主周期的 2 倍。此外，固定周期任务 （执行优先度 16）的任务周期为主周期的 4 倍。因此，固定周期任务 （执行优先度 5）每经过 2 个主周期与主固定周期任务的开始时间同步，固定周期任务 （执行优先度 16）则为每 4 个主周期。
- 系统服务不受任务的影响，始终执行。



使用注意事项

可能在多个任务间读写同一个变量时，请务必使用 “ 变量的任务间排他控制功能 ”。否则，在执行某个任务的过程中，变量的值可能被其他任务变更。
详情请参考📖「5-7-1 确保任务间变量值同时性的方法」（P.5-69）。

任务和动作模式

CPU 单元的动作模式和任务有以下关系。

任务	规格
主固定周期任务	• 无论是运行模式还是程序模式，都执行。 • 但是，执行用户程序的仅限运行模式。
固定周期任务	
事件任务	仅在运行模式下执行。

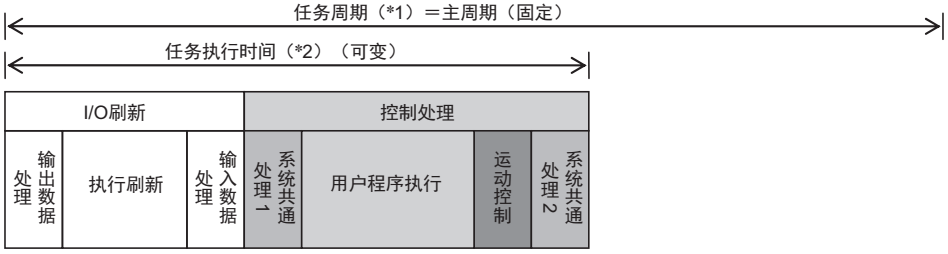
使用注意事项

- 动作模式从程序模式切换为运行模式时，若事件任务的执行条件已成立，将不执行事件任务。当动作模式为运行模式，且事件任务的执行条件从不成立变为成立时，执行事件任务。
- 即使动作模式为运行模式，如果发生全部停止故障等级的异常，也不会执行事件任务。

各任务类型的处理内容

● 主固定周期任务

任务的执行优先度最高，可高速、高精度执行的任务。
按指定的周期执行系统共通处理、 I/O 刷新、用户程序、运动控制。

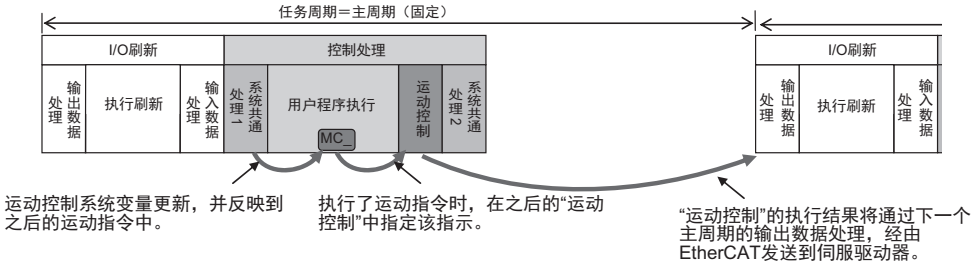


*1 ：任务周期： 任务按固定周期执行，为设定的固定时间。
*2 ：任务执行时间： 从任务的执行条件成立到执行完成为止，实际花费的时间。

处理		处理内容
I/O 刷新	输出数据处理	• 生成向执行 I/O 刷新的 I/O 输出的 OUT 刷新数据。 • 设定为强制值刷新时，将在 OUT 刷新数据中反映强制值 （输出）。
	执行刷新	• 与 I/O 执行数据交换。
	输入数据处理	• 事件任务的条件式一致判定。 • 导入执行了 I/O 刷新的 I/O （输入）单元之 IN 刷新数据。 • 设定为强制值刷新时，将在导入的 IN 刷新数据中反映强制值 （输入）。
系统共通处理 1		• 变量的任务间排他控制处理 （参照任务时） • 运动输入处理*1 • 数据追踪处理 （采样、触发判定）
用户程序执行		• 按分配顺序执行任务中分配的程序。
运动控制*2		• 通过主固定周期任务和固定周期任务 （执行优先度 16）中分配的用户程序中的运动控制指令，发出运动控制指示并执行。 • 对下一个主固定周期任务的 “I/O 刷新 ” 进行运动输出处理
系统共通处理 2		• 变量的任务间排他控制处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理 （［任务设定］中设定的 “ 变量访问时间 ” 分） • EtherNet/IP 标签数据链接处理时，若对标签 （网络公开属性的变量）设定更新任务，将作为上述变量访问处理执行。

*1. 运动控制系统变量的轴当前值 （位置、速度、转矩）及运动控制系统变量的伺服驱动器状态将更新。
*2. 若为主固定周期任务的用户程序中的运动控制指令，如下图所示，将在随后的 “ 运动控制 ” 中执行该指令发出的指示，并在下一主固定周期任务的 “I/O 刷新 ” 时输出到伺服驱动器。

（注）上表中各单元格的处理将按记载顺序执行。

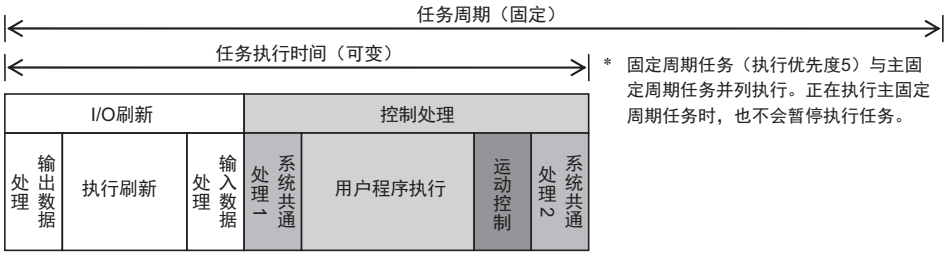


此外，若为固定周期任务 （执行优先度 16）的用户程序中的运动控制指令，将在该固定周期 （执行优先度 16）的下一个主固定周期任务的 “ 运动控制 （MC） ” 时，执行该指令发出的指示。

详情请参考📖「5-10-3 系统输入输出响应时间」（P.5-91）。

● 固定周期任务 （执行优先度 5）

任务的执行优先度仅次于主固定周期任务，可高速、高精度执行的任务。
按指定的周期执行系统共通处理、 I/O 刷新、用户程序、运动控制。
固定周期任务 （执行优先度 5）仅在 NX 系列 CPU 单元中可使用。

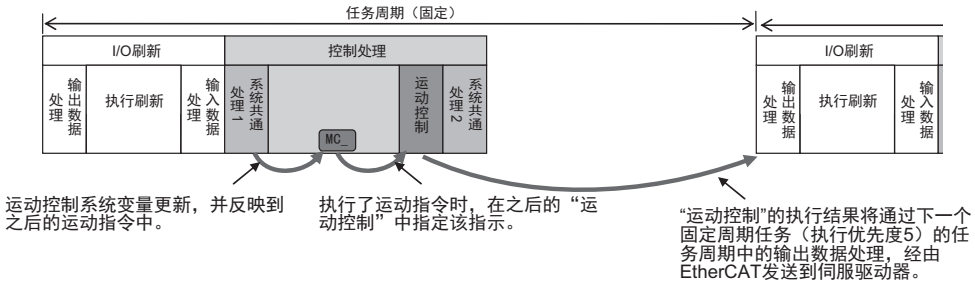


* 固定周期任务（执行优先度5）与主固定周期任务并列执行。正在执行主固定周期任务时，也不会暂停执行任务。

处理		处理内容
I/O 刷新	输出数据处理	• 生成向执行 I/O 刷新的 I/O 输出的 OUT 刷新数据。 • 设定为强制值刷新时，将在 OUT 刷新数据中反映强制值 （输出）。
	执行刷新	• 与 I/O 执行数据交换。
	输入数据处理	• 导入执行了 I/O 刷新的 I/O （输入）单元之 IN 刷新数据。 • 设定为强制值刷新时，将在导入的 IN 刷新数据中反映强制值 （输入）。
系统共通处理 1		• 变量的任务间排他控制处理 （参照任务时） • 运动输入处理*1 • 数据追踪处理 （采样、触发判定）
用户程序执行		• 按分配顺序执行任务中分配的程序。
运动控制*2		• 通过固定周期任务 （执行优先度 5）中分配的用户程序中的运动控制指令，发出运动控制指示并执行。 • 对下一个固定周期任务 （执行优先度 5）的 “I/O 刷新 ” 进行运动输出处理
系统共通处理 2		• 变量的任务间排他控制处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理 （［任务设定］中设定的 “ 变量访问时间 ” 分） • EtherNet/IP 标签数据链接处理时，若对标签 （网络公开属性的变量）设定更新任务，将作为上述变量访问处理执行。

*1. 运动控制系统变量的轴当前值 （位置、速度、转矩）及运动控制系统变量的伺服驱动器状态将更新。
*2. 若为固定周期任务 （执行优先度 5）的用户程序中的运动控制指令，如下图所示，将在随后的 “ 运动控制 ” 中执行该指令发出的指示，并在下一固定周期任务 （执行优先度 5）的 “I/O 刷新 ” 时输出到伺服驱动器。

（注）上表中各单元格的处理将按记载顺序执行。



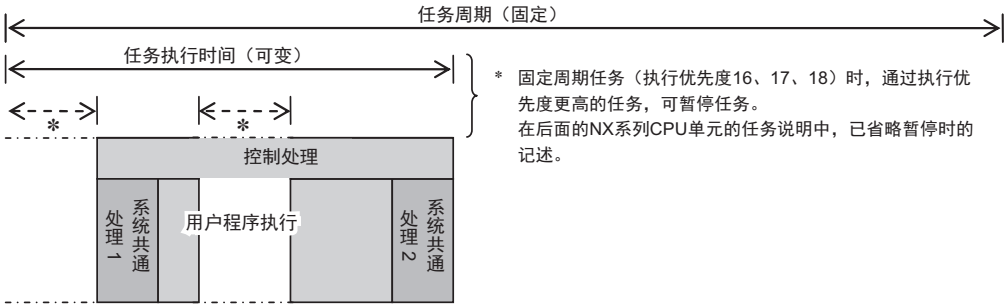
● 固定周期任务 （执行优先度 16、 17、 18）

按指定为主周期整数倍的任务周期，执行用户程序的任务。整个任务中可分配 0 ～ 3 个。

固定周期任务 （执行优先度 16）中，可记述控制程序，对 ［设定控制 I/O 的任务］设定为固定周期任务 （执行优先度 16）的从站 / 单元进行控制。

根据在固定周期任务中进行 I/O 控制或不控制，处理内容有以下不同。

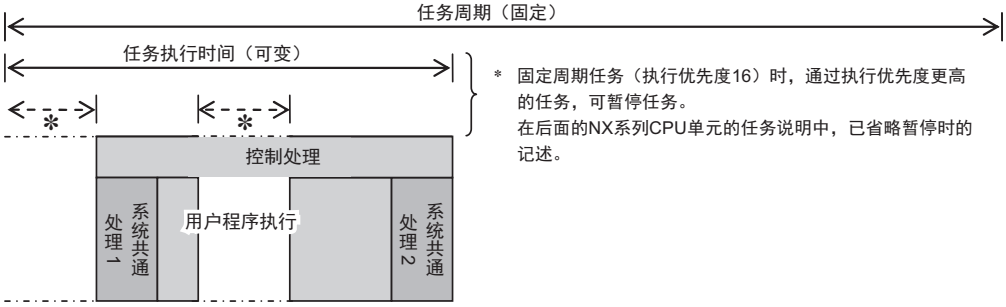
- 在固定周期任务中不进行 I/O 控制时



处理	处理内容
系统共通处理 1	• 变量的任务间排他控制处理 （参照任务时） • 数据追踪处理 （采样、触发判定）
用户程序执行	• 按分配顺序执行任务中分配的程序。
系统共通处理 2	• 变量的任务间排他控制处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理 （［任务设定］中设定的 “ 变量访问时间 ” 分） • EtherNet/IP 标签数据链接处理时，若对标签 （网络公开属性的变量）设定更新任务，将作为上述变量访问处理执行。

（注）上表中各单元格的处理将按记载顺序执行。

- 在固定周期任务 （执行优先度 16）中进行 I/O 控制时



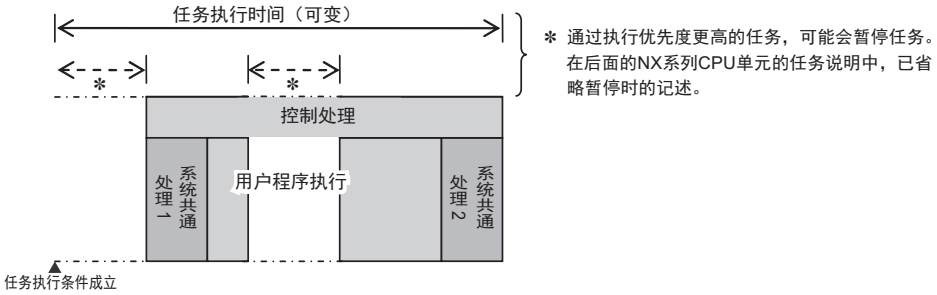
处理	处理内容
系统共通处理 1	• IN 刷新数据的反映*1 • 变量的任务间排他控制处理 （参照任务时） • 数据追踪处理 （采样、触发判定）
用户程序执行	• 按分配顺序执行任务中分配的程序。
系统共通处理 2	• 变量的任务间排他控制处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理 （［任务设定］中设定的 “ 变量访问时间 ” 分） • EtherNet/IP 标签数据链接处理时，若对标签 （网络公开属性的变量）设定更新任务，将作为上述变量访问处理执行。 • OUT 刷新数据的反映*2

*1. 导入在［设定控制 I/O 的任务］中设定为固定周期任务（执行优先度 16）的 EtherCAT 从站的 IN 刷新数据。主固定周期任务的 I/O 刷新时，程序数据通信中输入的数据即为 IN 刷新数据。
*2. 将［设定控制 I/O 的任务］中设定为固定周期任务 （执行优先度 16）的用户程序执行结果反映到 OUT 刷新数据中。主固定周期任务的 I/O 刷新时，通过程序数据通信输出 OUT 刷新数据。

（注）上表中各单元格的处理将按记载顺序执行。

● 事件任务

在指定的执行条件成立时，仅执行 1 次的任务。整个任务中可分配 0 ～ 32 个。
事件任务的处理内容如下所示。



处理	处理内容
系统共通处理 1	• 变量的任务间排他控制处理 （参照任务时）*1
用户程序执行	• 按分配顺序执行任务中分配的程序。
系统共通处理 2	• 变量的任务间排他控制处理 （更新任务时）*1

*1. 变量的任务间排他控制处理的详情请参考📖「5-7-1 确保任务间变量值同时性的方法」（P.5-69）。

5-3-4 NX 系列事件任务的执行条件

事件任务是指在指定的执行条件成立时，仅执行 1 次的任务。事件任务的执行条件有以下两种。

执行条件	执行事件任务的时序	使用时
通过 ActEventTask 指令执行	执行 ActEventTask 指令时	• 希望在用户程序中写明执行哪个事件任务时 • 在变量的条件式一致判定前，事件任务的执行条件可能发生变化时
变量的条件式一致后执行	指定的变量值与指定的条件式一致时*1	希望省略执行事件任务所需的用户程序，将用户程序简化时

*1. 关于判定指定的变量值与指定的条件式是否一致的时间，请参考📖「执行条件为变量的条件式一致时的执行时序」（P.5-24）。

通过 ActEventTask 指令执行事件任务

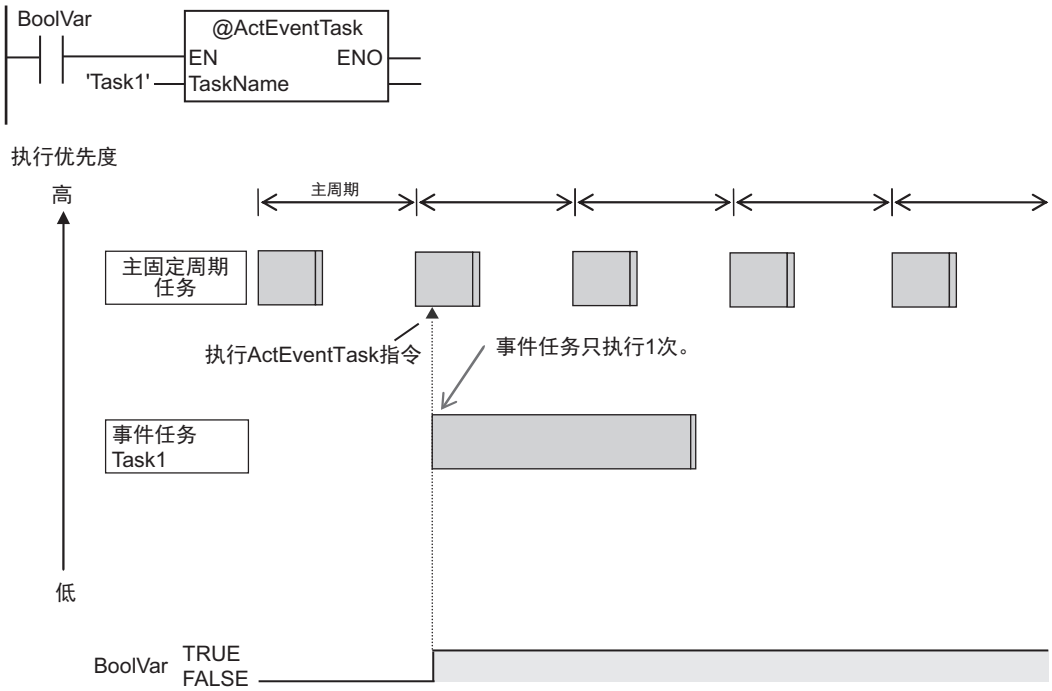
若在程序中执行 ActEventTask 指令，将执行 1 次指定的事件任务。 ActEventTask 指令的详细规格请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

若使用 ActEventTask 指令，看到用户程序后，能立即知道是启动哪个事件任务。此外，在变量的条件式一致判定前，即使事件任务的执行条件可能发生变化，仍为有效。

● 使用 ActEventTask 指令的用户程序示例

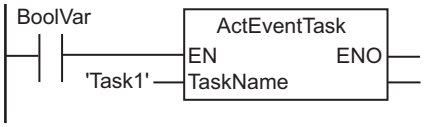
例 1）变量值发生变化时，执行 1 次事件任务

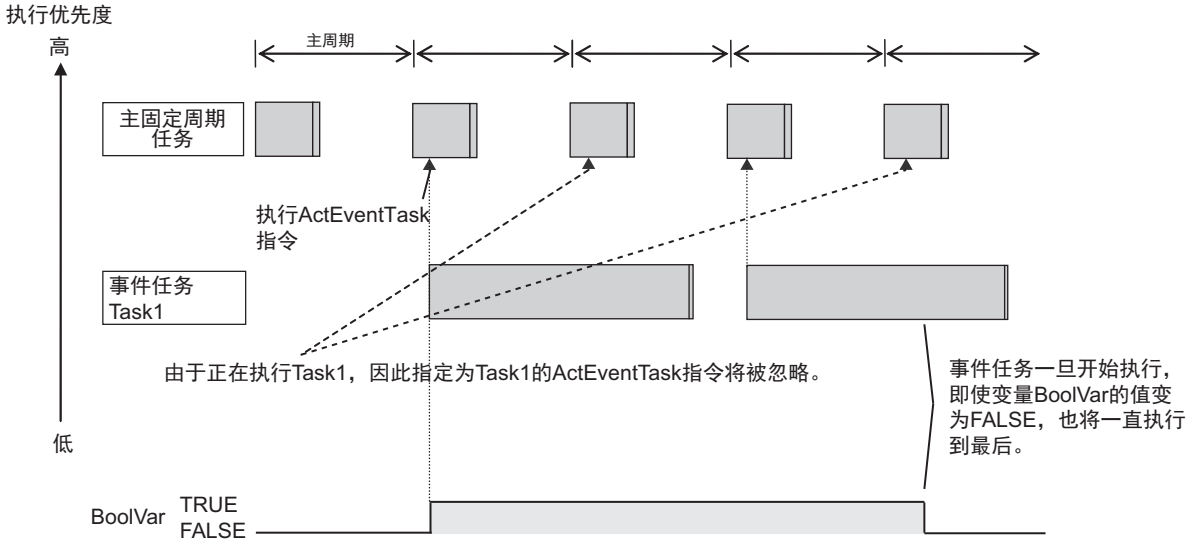
如下图所示，若在 ActEventTask 指令中加入输入上升沿微分的动作选项，当 BOOL 型变量 BoolVar 的值从 FALSE 变为 TRUE 时，将执行 1 次事件任务 Task1。



例 2）变量为指定值时，反复执行事件任务

如下图所示，若 ActEventTask 指令中不加入输入上升沿微分的动作选项，当 BOOL 型变量 BoolVar 的值为 TRUE 时，将反复执行事件任务 Task1。但是，在执行 Task1 时，即使执行指定了 Task1 的 ActEventTask 指令，也将忽略。此外，事件任务一旦开始执行，即使在执行过程中 BoolVar 的值变为 FALSE，也将一直执行到最后。





变量的条件式一致后执行事件任务

变量的条件式一致后执行的方法是指 Sysmac Studio 中指定的变量值与指定的条件式一致时，执行 1 次事件任务。该变量的值与条件式一致期间，不会持续执行事件任务，而是在变量的值从与条件式不一致变为一致时，执行 1 次事件任务。

变量的条件式一致后执行时，不需要执行事件任务所需的用户程序。

●可指定条件式的变量种类

可指定条件式的变量种类如下所示。

变量的种类		可否指定
系统定义变量		可*1
准用户定义变量		可
用户定义变量	全局变量	可
	程序中使用的变量	可
	功能块中使用的变量	可*2
	函数中使用的变量	不可

*1. 以下变量除外。
EN,ENO,P_Off,P_CY,P_First_RunMode,P_First_Run,P_PRGER

*2. 输入输出变量除外。

● **可指定条件式的变量的数据类型**

可指定条件式的变量的数据类型如下所示。

数据类型的分类	数据类型		可否指定
基本数据型	布尔型、位列型、整数型、实数型		可
	持续时间型、日期型、时刻型、日期时刻型、字符串型		不可
数据类型的指定	排列指定	排列整体	不可
		要素	可*1
派生数据型	结构体型	结构体整体	不可
		成员	可*2
	联合体型	联合体整体	不可
		成员	可*2
	列举型		可

*1. 排列要素的数据类型仅限布尔型、位列型、整数型、实数型中的一个。
*2. 成员的数据类型仅限布尔型、位列型、整数型、实数型中的一个。

● 可指定的条件式

因指定条件式的变量数据类型不同，可指定的条件式如下。
指定条件式的变量数据类型为位列型、整数型、实数型时，需要设定变量的值和要比较的比较常数。

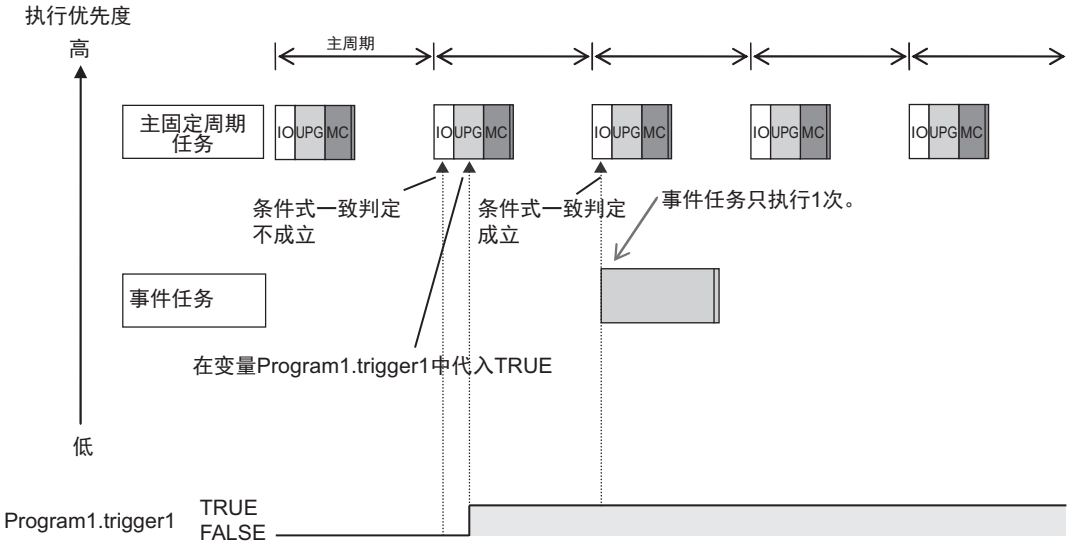
数据类型	可指定的条件式
布尔型、布尔型排列要素、布尔型结构体成员、布尔型联合体成员	上升沿 FALSE → TRUE
	下降沿 TRUE → FALSE
位列型、整数型、实数型、 或以上数据类型的排列要素、结构体成员、联合体成员	变量 = { 比较常数 }
	变量 ≠ { 比较常数 }
	变量 > { 比较常数 }
	变量 ≧ { 比较常数 }
	变量 < { 比较常数 }
	变量 ≦ { 比较常数 }

● 比较常数的值的有效范围

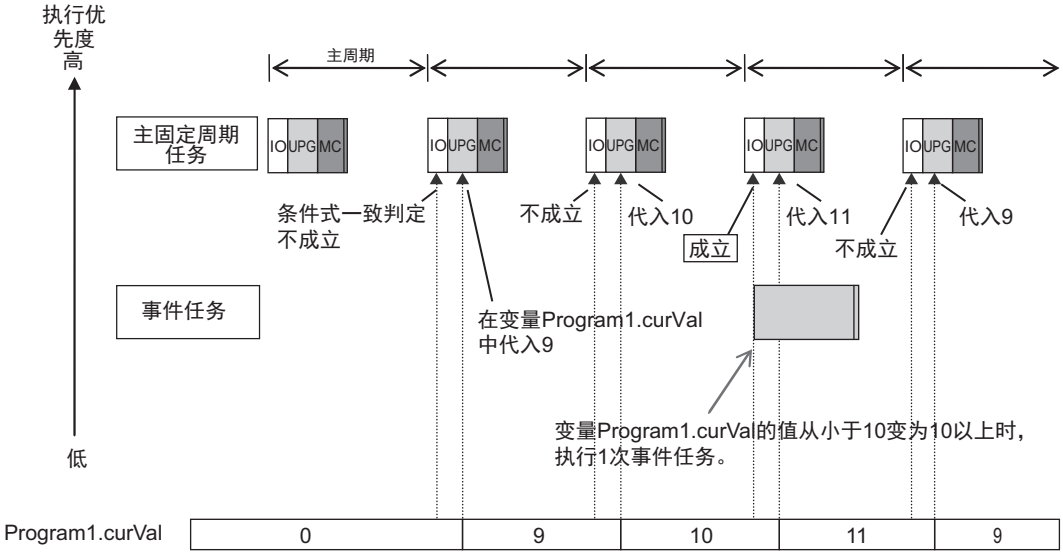
指定条件式的变量数据类型为位列型、整数型、实数型时，需要设定变量的值和要比较的比较常数。
比较常数的值的有效范围与指定条件式的变量之数据类型值的有效范围相同。
各数据类型的值的有效范围请参考📖 「基本数据型和派生数据型」（P.6-31）。
例如，指定条件式的变量之数据类型为 BYTE 时，比较常数的值的有效范围为 BYTE#16#00 ～ BYTE#16#FF。

● 变量的条件式一致后执行事件任务的示例

例 1) 事件任务的执行条件设定为 BOOL 型变量 Program1.trigger1 的上升沿时 Program1.trigger1 的值从 FALSE 变为 TRUE 时，执行 1 次事件任务。



例 2) 事件任务的执行条件设定为 INT 型变量 Program1.curVal ≧ 10 时 Program1.curVal 的值从小于 10 变为 10 以上时，执行 1 次事件任务。



使用注意事项

主固定周期任务中，指定变量的值发生变化时，将在下一主固定周期任务中判定是否与条件式一致。因此，执行事件任务的时间为：变量的值满足条件式的周期的下一个主固定周期任务中，进行条件式一致判定后。

5-3-5 NX 系列事件任务的执行时序

事件任务的执行优先度为 8 或 48。在配备多核处理器的 NX 系列 CPU 单元中，执行条件成立的事件任务将按照任务的执行优先度开始执行。在有些用户程序中，可能与主固定周期任务或固定周期任务，或与执行优先度不同的事件任务并列执行。

可能在多个任务间读写同一个变量时，请务必使用以下功能，控制主固定周期任务、固定周期任务或执行优先度不同的事件任务的执行。

目的	使用的功能
在事件任务和其他任务中访问相同的全局变量时	变量的任务间排他控制功能 • 变量的任务间排他控制设定功能 • Lock （任务间独占锁定）指令 • Unlock （解除任务间独占锁定）指令
确认其他任务的执行状态时	Task_IsActive （任务执行中判定）指令

详情请参考📖「5-7-1 确保任务间变量值同时性的方法」（P.5-69）。

事件任务的执行时序因事件任务的执行条件不同而异。
此外，事件任务可将多个任务分配为相同的执行优先度。执行优先度相同的多个事件任务的执行条件成立时，需要注意。

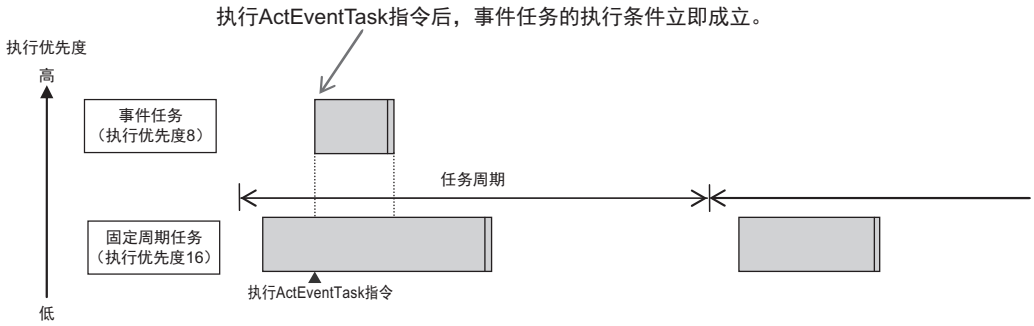
执行时序随事件任务执行条件的不同

因事件任务的执行条件为利用 ActEventTask 指令还是利用变量的条件式一致不同，执行条件的成立时间不同。

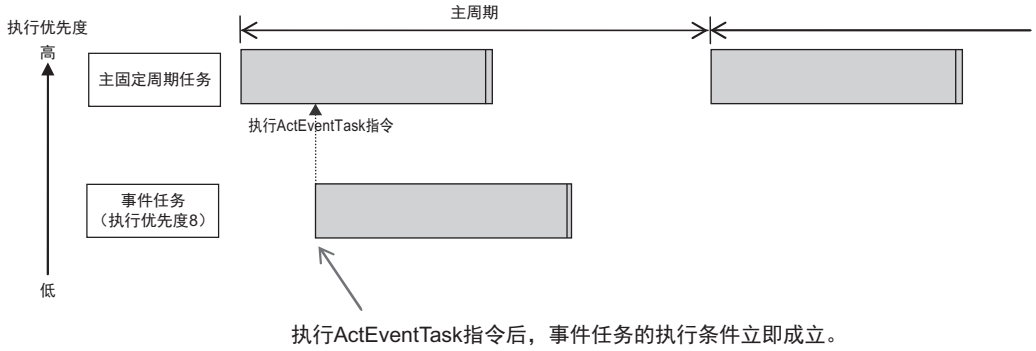
● 执行条件为利用 ActEventTask 指令时的执行时序

事件任务的执行条件为利用 ActEventTask 指令时，执行 ActEventTask 指令后，事件任务的执行条件立即成立。执行条件成立的事件任务将按照任务的执行优先度开始执行。

例 1) 要启动相比执行指令的任务，执行优先度更高的事件任务时



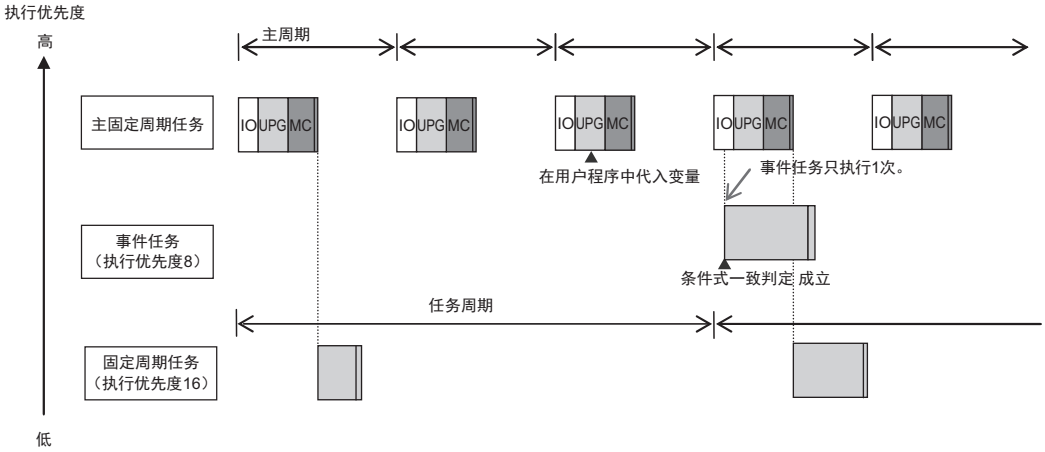
例 2) 要启动相比执行指令的任务，执行优先度更低的事件任务时



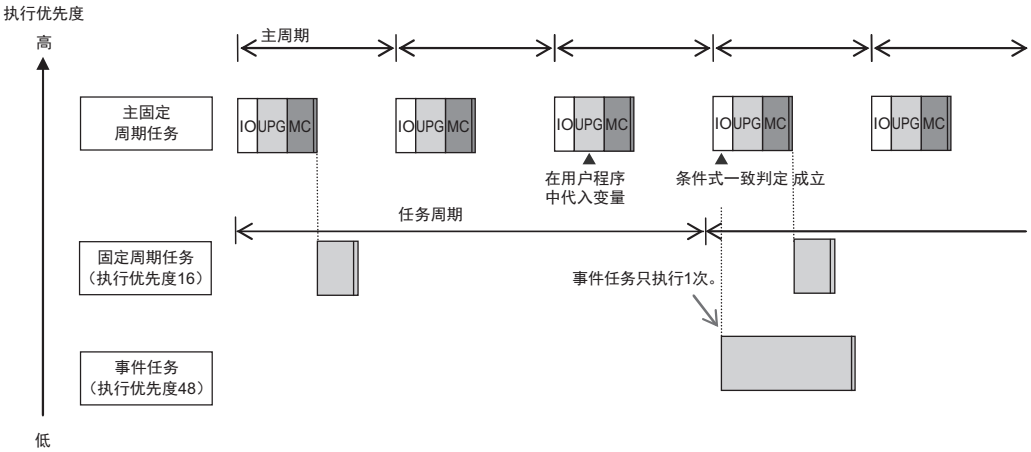
● 执行条件为变量的条件式一致时的执行时序

条件式一致判定在主固定周期任务中执行。在条件式的判定处理中，判定为条件一致时，事件任务的执行条件成立。执行条件成立的事件任务将按照任务的执行优先度开始执行。

例 1) 存在固定周期任务 （执行优先度 16）和事件任务 （执行优先度 8）时



例 2) 存在固定周期任务 （执行优先度 16）和事件任务 （执行优先度 48）时



使用注意事项

- NX 系列 CPU 单元中，由主固定周期任务的 I/O 刷新发生条件式一致时以及由主固定周期任务中分配的用户程序执行发生时，事件任务执行条件的成立时间相同，具体如下。

条件式一致发生原因	事件任务的执行条件成立的时间
主固定周期任务的 I/O 刷新	下一个主固定周期任务的判定处理
主固定周期任务的用户程序执行	下一个主固定周期任务的判定处理

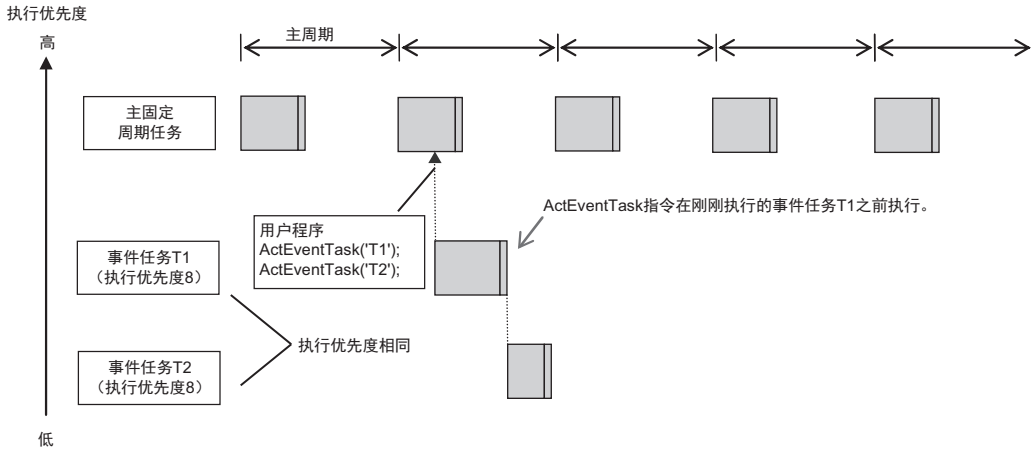
- 执行事件任务的时间：前一次判定时为条件式不一致，下一次判定时为条件式一致时。因此，即使发生从不一致到一致的变化，若在下次判定前恢复为不一致，将不会执行事件任务。

执行优先度相同的事件任务的执行时序

事件任务可将多个任务分配为相同的执行优先度。利用 ActEventTask 指令，执行优先度相同的多个事件任务的执行条件成立时，将按执行指令的顺序执行事件任务。

例 1）执行了 2 个 ActEventTask 指令时

如下图所示，先执行的 ActEventTask 指令所指定的事件任务 T1 将在事件任务 T2 之前执行。



例 2）变量的条件式一致和 ActEventTask 指令同时存在时

假设事件任务 T1、 T2、 T3 的执行条件设定如下。

- T1：变量 VarA 的条件式一致
- T2：变量 VarB 的条件式一致
- T3：ActEventTask 指令

如下图所示，在同一主周期中，I/O 刷新后 VarA 的条件式一致、在用户程序中执行 ActEventTask 指令、用户程序中发生 VarB 的条件式一致时，按以下顺序动作。

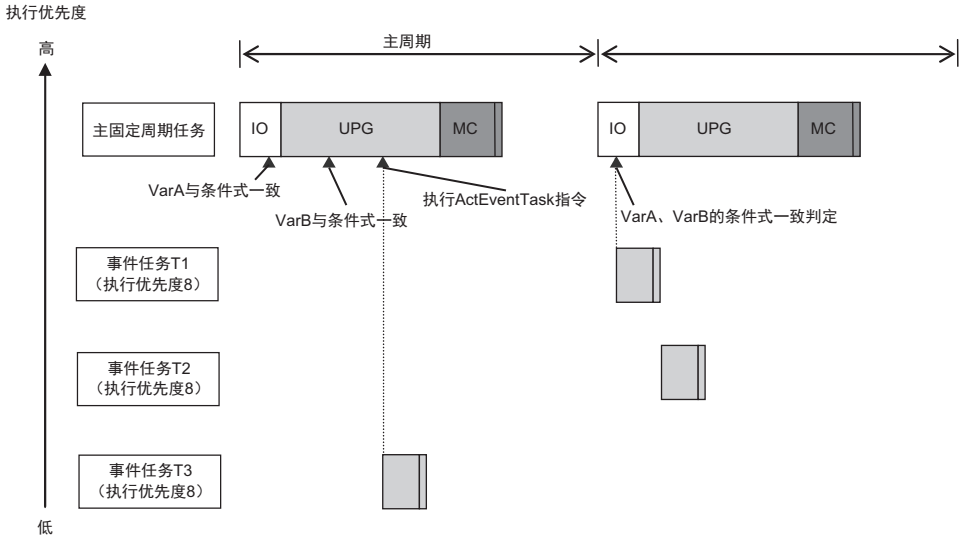
(1) I/O 刷新后 VarA 与条件式一致。

(2) 用户程序中 VarB 与条件式一致。

(3) 此时，由于尚未执行条件式的一致判定，因此 T1 和 T2 均不执行。

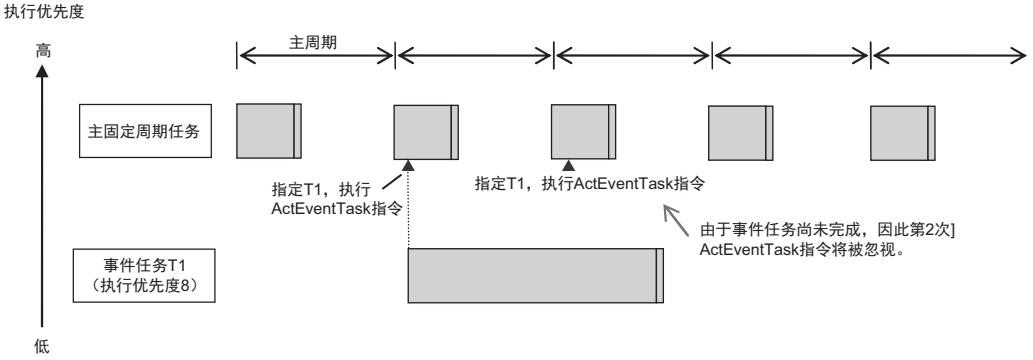
(4) 在程序中执行 ActEventTask 指令，执行 T3。

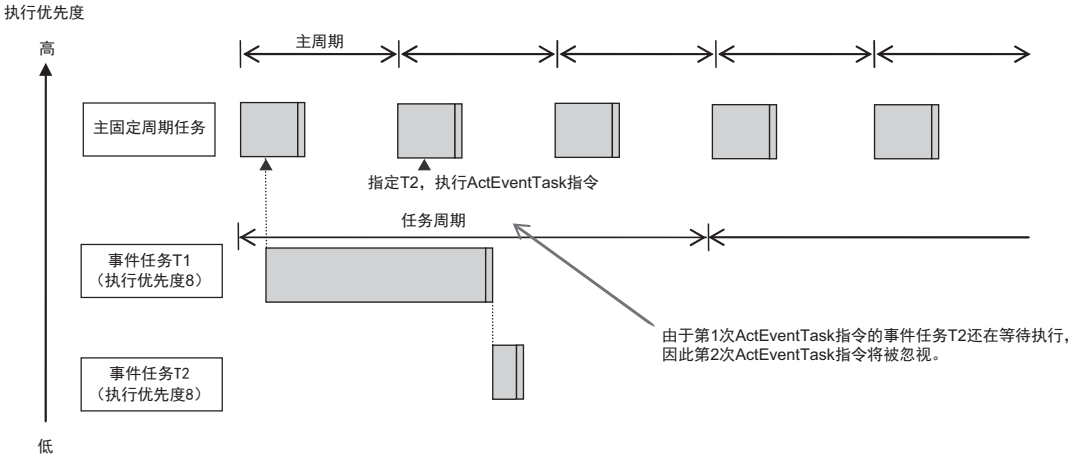
(5) 在下一主固定周期任务中执行刷新后，VarA 和 VarB 判定为与条件式一致，执行 T1 和 T2。若在同一周期内判定有多个条件式一致，则事件任务的执行顺序不确定。下图为先执行 T1 时的情况。



5-3-6 事件任务完成前，执行条件再次成立时的动作

事件任务完成前，该事件任务的执行条件再次成立时，第 2 次执行条件的成立将被忽略。事件任务完成前是指事件任务执行中及事件任务执行等待中两种情况。此外，事件任务一旦开始执行，即使执行条件变为不成立，也将一直执行到最后。





5-4 NJ 系列的任务规格和基本动作

下面介绍 NJ 系列 CPU 单元的任务规格和基本动作。

5-4-1 NJ 系列的任务规格

任务的规格如下所示。

项目	规格
任务类型	• 主固定周期任务 • 固定周期任务 （优先度 16、 17、 18） • 事件任务 （优先度 8、 48）
任务个数	• 主固定周期任务：1 • 固定周期任务：0 ～ 3*1 • 事件任务：0 ～ 32*2
1 个任务的程序数	最大 128
主固定周期任务的 任务周期	500μs*3、 1ms、 2ms、 4ms
固定周期任务的任务周期	固定周期任务的任务周期设定为主固定周期任务的整数倍。 可设定的周期请参考后面介绍的 “ 固定周期任务中可设定的任务周期 ” 的表。

*1. 执行优先度 16、 17、 18 分别为 0 或 1 个。
*2. 执行优先度 8、 48 分别为 0 ～ 32 个，两者加起来最多 32 个。
*3. NJ301- □□□□中，单元版本为 1.03 以上的环境下可以使用。
NJ101- □□□□不适用。

● 固定周期任务中可设定的任务周期

主固定周期任务的任务周期	可设定的固定周期任务的任务周期
500μs*1	1ms、 2ms、 3ms、 4ms、 5ms、 8ms、 10ms、 15ms、 20ms、 25ms、 30ms、 40ms、 50ms、 60ms、 75ms、 100ms
1ms	1ms、 2ms、 3ms、 4ms、 5ms、 8ms、 10ms、 15ms、 20ms、 25ms、 30ms、 40ms、 50ms、 60ms、 75ms、 100ms
2ms	2ms、 4ms、 8ms、 10ms、 20ms、 30ms、 40ms、 50ms、 60ms、 100ms
4ms	4ms、 8ms、 20ms、 40ms、 60ms、 100ms

*1. NJ301- □□□□中，单元版本为 1.03 以上的环境下可以使用。
NJ101- □□□□不适用。

5-4-2 NJ 系列的任务分配说明

所有程序都需要分配到任意一个任务中。请根据程序的特性，按以下方法决定要分配的任务。

任务	适合分配的程序
主固定周期任务	• 需要严格遵守 I/O 刷新的执行周期。 • 执行优先度最高。 • 包括执行优先度较高的运动控制指令。
固定周期任务 （优先度 16）	• 伴随 I/O 刷新的执行。 • 执行优先度相对较低，但需要按周期执行。 • 包括执行优先度较低的运动控制指令。
固定周期任务 （优先度 17,18）	• 执行优先度相对较低，但需要按周期执行。
事件任务	• 仅在特定条件成立时执行。

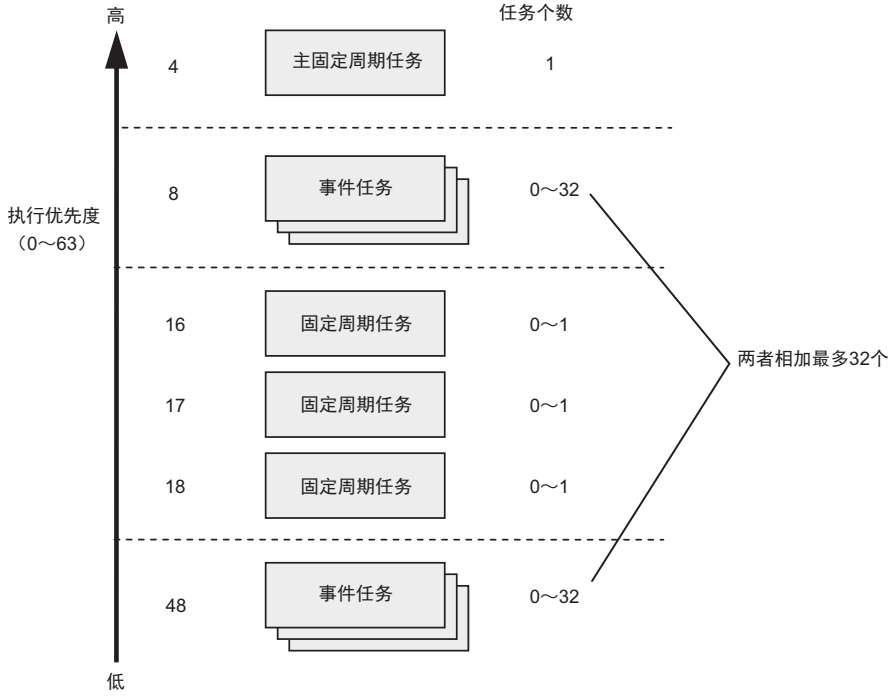
5-4-3 NJ 系列的任务基本动作

CPU 单元不可同时执行多个任务。各任务按什么样的顺序执行，取决于各任务中设定的执行优先度。

任务执行优先度

任务根据其类型决定执行优先度。
正在执行某任务 Ta 时，若执行优先度比它高的 Tb 执行条件成立，将中断 Ta 的执行，先执行 Tb。在 Tb 的处理结束后，再开始 Ta 的处理。
不同任务类型的执行优先度如下所示。执行优先度的值越小，优先度越高。

任务	执行优先度	执行优先度的重复
主固定周期任务	4	—
固定周期任务	16、 17、 18	不可将多个任务设定为相同的执行优先度。
事件任务	8、 48	可将多个任务设定为相同的执行优先度。 此时的执行顺序请参考📖「5-4-5 NJ 系列事件任务的执行时序」（P.5-41）。



主固定周期任务和固定周期任务的任务周期

主固定周期任务和固定周期任务是指按周期反复执行的任务。

固定周期任务的任务周期必须设定为主固定周期任务的任务周期（称为“主周期”）的整数倍。因此，几次后两者的执行时序会重叠。

例如，设定为主周期 1ms、固定周期任务（执行优先度 16）的任务周期 4ms 时，则主固定周期任务每执行 4 次，主固定周期任务和固定周期任务的执行时序重叠。

参考

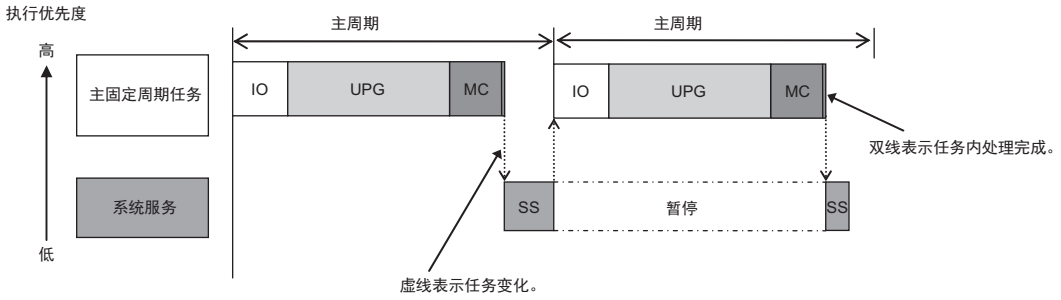
事件任务是指不按周期执行，而是在指定的执行条件成立时，仅执行 1 次的任务。因此，事件任务的执行时序取决于事件任务的执行条件成立的时间和事件任务的执行优先度。

任务的执行顺序示例

下面举例说明主固定周期任务和固定周期任务的执行顺序。
事件任务的执行顺序请参考📖「5-4-5 NJ 系列事件任务的执行时序」（P.5-41）。

● 仅主固定周期任务时

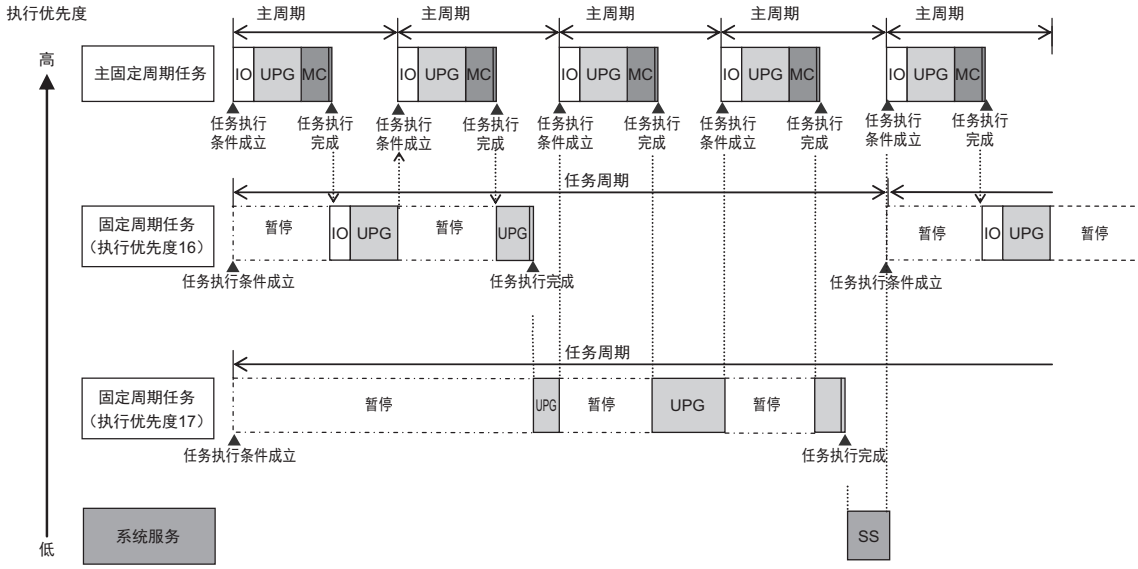
按主固定周期执行主固定周期任务。
图中的系统服务是指通信处理等 CPU 单元在任务之外执行的处理。系统服务将在任务执行的空闲时间执行。系统服务的详情请参考📖「 系统服务的处理内容」（P.5-54）。



IO	：表示I/O刷新（数据输出→数据输入的顺序）。
UPG	：表示执行用户程序。
MC	：表示运动控制。
SS	：表示系统服务。

● **主固定周期任务、固定周期任务（执行优先度 16）、固定周期任务（执行优先度 17）同时存在时**

- 主固定周期任务的执行优先度最高，因此将立即按主周期执行。
- 固定周期任务（执行优先度 16）的执行优先度低于主固定周期任务，因此在未执行主固定周期任务的时间内执行。
- 固定周期任务 （执行优先度 17）的执行优先度更低，因此在未执行上述两个任务的时间内执行。
- 本示例中，固定周期任务（执行优先度 16）的任务周期为主周期的 4 倍。因此，每经过 4 个主周期，主固定周期任务和固定周期任务 （执行优先度 16）的执行开始时间同步。
- 系统服务将在所有任务的空闲时间执行。

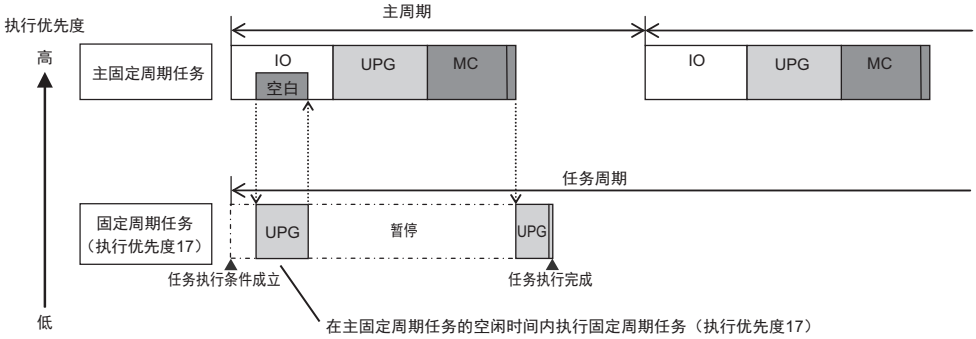


使用注意事项

可能在多个任务间读写同一个变量时，请务必使用“变量的任务间排他控制功能”。否则，在执行某个任务的过程中，变量的值可能被其他任务变更。
详情请参考📖「5-7-1 确保任务间变量值同时性的方法」（P.5-69）。

使用任务空闲时间的其他任务的执行

不可中断执行优先度较高的任务，而执行优先度更低的任务。但是，在 NJ 系列 CPU 单元中，执行优先度较高的任务中发生空闲时间时，将利用该时间，开始执行优先度较低的任务。例如，I/O 刷新中等待数据传送完成的时间。但是，执行优先度较高的任务重新开始处理后，将立即中断执行优先度较低的任务。具体情况如下所示。



使用注意事项

即使发生主固定周期任务的空闲时间，固定周期任务（执行优先度 16）的用户程序必须在主固定周期任务完成后执行。
但是， NJ 系列 CPU 单元中，上图的固定周期任务（执行优先度 17 或 18）无该限制。
此外， NJ 系列 CPU 单元中，固定周期任务（执行优先度 16）的 I/O 刷新也没有该限制，可在主固定周期任务的空闲时间内执行。

任务和动作模式

CPU 单元的动作模式和任务有以下关系。

任务	规格
主固定周期任务	• 无论是运行模式还是程序模式，都执行。
固定周期任务	• 但是，执行用户程序的仅限运行模式。
事件任务	仅在运行模式下执行。

使用注意事项

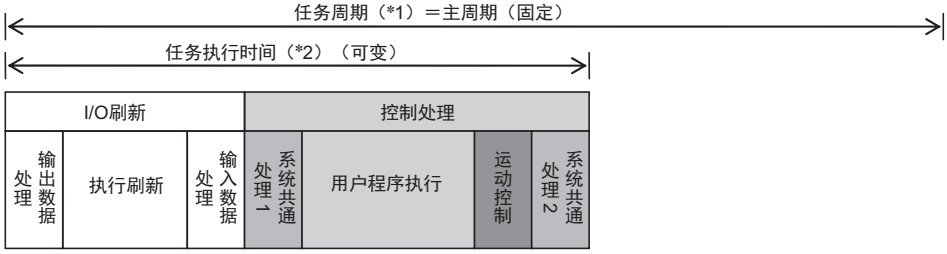
- 动作模式从程序模式切换为运行模式时，若事件任务的执行条件已成立，将不执行事件任务。当动作模式为运行模式，且事件任务的执行条件从不成立变为成立时，执行事件任务。
- 即使动作模式为运行模式，如果发生全部停止故障等级的异常，也不会执行事件任务。

各任务类型的处理内容

● 主固定周期任务

任务的执行优先度最高，可高速、高精度执行的任务。

按指定的周期执行系统共通处理、I/O 刷新、用户程序、运动控制。主固定周期任务中包含 “ 运动控制（MC） ” 的处理。



*1 ：任务周期： 任务按固定周期执行，为设定的固定时间。

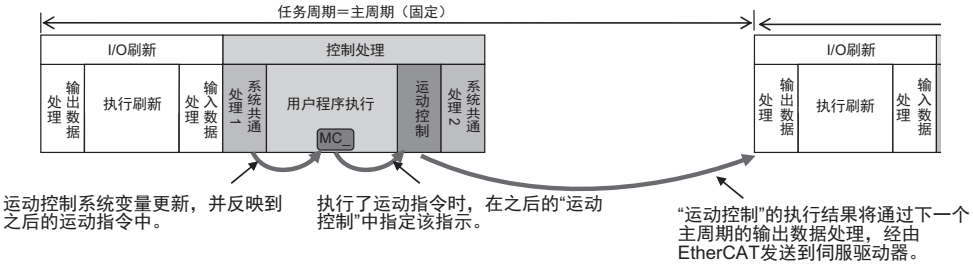
*2 ：任务执行时间： 从任务的执行条件成立到执行完成为止，实际花费的时间。

处理		处理内容
I/O 刷新	输出数据处理	• 生成向执行 I/O 刷新的 I/O 输出的 OUT 刷新数据。 • 设定为强制值刷新时，将在 OUT 刷新数据中反映强制值 （输出）。
	执行刷新	• 与 I/O 执行数据交换。
	输入数据处理	• 导入执行了 I/O 刷新的 I/O （输入）单元之 IN 刷新数据。 • 设定为强制值刷新时，将在导入的 IN 刷新数据中反映强制值 （输入）。
系统共通处理 1		• 变量的任务间排他控制处理 （参照任务时） • 运动输入处理*1 • 数据追踪处理 （采样、触发判定） • 事件任务的条件式一致判定
用户程序执行		• 按分配顺序执行任务中分配的程序。
运动控制*2		• 通过用户程序中的运动控制指令，发出运动控制指示并执行。 • 对下一个主固定周期任务的 “I/O 刷新 ” 进行运动输出处理
系统共通处理 2		• 变量的任务间排他控制处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理 （［任务设定］中设定的 “ 变量访问时间 ” 分） • EtherNet/IP 标签数据链接处理时，若对标签 （网络公开属性的变量）设定更新任务，将作为上述变量访问处理执行。

*1. 运动控制系统变量的轴当前值 （位置、速度、转矩）及运动控制系统变量的伺服驱动器状态将更新。

*2. 若为主固定周期任务的用户程序中的运动控制指令，如下图所示，将在随后的 “ 运动控制 ” 中执行该指令发出的指示，并在下一主固定周期任务的 “I/O 刷新 ” 时输出到伺服驱动器。

（注）上表中各单元格的处理将按记载顺序执行。



此外，若为固定周期任务的用户程序中的运动控制指令，将在该固定周期的主固定周期任务的 “ 运动控制 （MC） ” 时，执行该指令发出的指示。

详情请参考📖「5-10-3 系统输入输出响应时间」（P.5-91）。

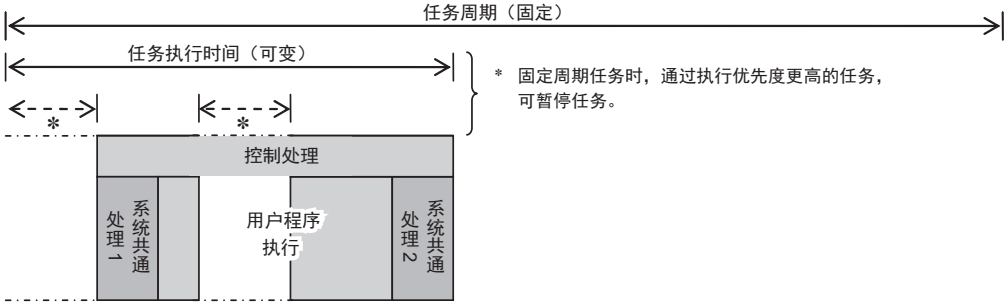
● 固定周期任务

按指定为主周期整数倍的任务周期，执行用户程序的任务。整个任务中可分配 0 ～ 3 个。

固定周期任务 （执行优先度 16）可执行 I/O 刷新。

根据在固定周期任务中进行 I/O 控制或不控制，处理内容有以下不同。

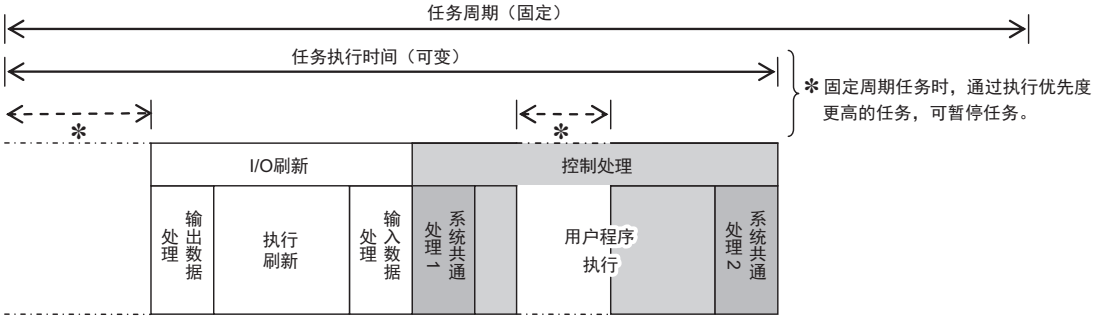
- 在固定周期任务中不进行 I/O 控制时



处理	处理内容
系统共通处理 1	• 变量的任务间排他控制处理 （参照任务时） • 数据追踪处理 （采样、触发判定）
用户程序执行	• 按分配顺序执行任务中分配的程序。
系统共通处理 2	• 变量的任务间排他控制处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理 （［任务设定］中设定的 “ 变量访问时间 ” 分） • EtherNet/IP 标签数据链接处理时，若对标签 （网络公开属性的变量）设定更新任务，将作为上述变量访问处理执行。

（注）上表中各单元格的处理将按记载顺序执行。

- 在固定周期任务中进行 I/O 控制时



处理		处理内容
I/O 刷新 *1	输出数据处理	• 生成向执行 I/O 刷新的 I/O 输出的 OUT 刷新数据。 • 设定为强制值刷新时，将在 OUT 刷新数据中反映强制值 （输出）。
	执行刷新	• 与 I/O 执行数据交换。
	输入数据处理	• 导入执行了 I/O 刷新的 I/O （输入）单元之 IN 刷新数据。 • 设定为强制值刷新时，将在导入的 IN 刷新数据中反映强制值 （输入）。
系统共通处理 1		• 变量的任务间排他控制处理 （参照任务时） • 数据追踪处理 （采样、触发判定）
用户程序执行		• 按分配顺序执行任务中分配的程序。
系统共通处理 2		• 变量的任务间排他控制处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理 （［任务设定］中设定的 “ 变量访问时间 ” 分） • EtherNet/IP 标签数据链接处理时，若对标签 （网络公开属性的变量）设定更新任务，将作为上述变量访问处理执行。

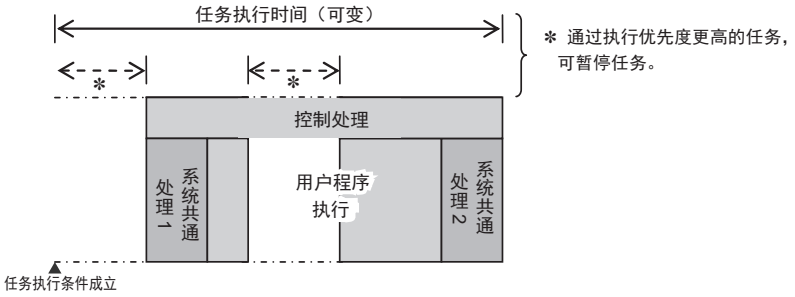
*1. 执行 CJ 单元的 I/O 刷新。

（注）上表中各单元格的处理将按记载顺序执行。

● 事件任务

在指定的执行条件成立时，仅执行 1 次的任务。整个任务中可分配 0 ～ 32 个。

事件任务的处理内容如下所示。



处理	处理内容
系统共通处理 1	• 变量的任务间排他控制处理 （参照任务时）*1
用户程序执行	• 按分配顺序执行任务中分配的程序。
系统共通处理 2	• 变量的任务间排他控制处理 （更新任务时）*1

*1. 变量的任务间排他控制处理的详情请参考「5-7-1 确保任务间变量值同时性的方法」（P.5-69）。

5-4-4 NJ 系列事件任务的执行条件

事件任务是指在指定的执行条件成立时，仅执行 1 次的任务。事件任务的执行条件有以下两种。

执行条件	执行事件任务的时序	使用时
通过 ActEventTask 指令执行	执行 ActEventTask 指令时	• 希望在用户程序中写明执行哪个事件任务时 • 在变量的条件式一致判定前，事件任务的执行条件可能发生变化时
变量的条件式一致后执行	指定的变量值与指定的条件式一致时*1	希望省略执行事件任务所需的用户程序，将用户程序简化时

*1. 关于判定指定的变量值与指定的条件式是否一致的时间，请参考「执行条件为变量的条件式一致时的执行时序」（P.5-43）。

通过 ActEventTask 指令执行事件任务

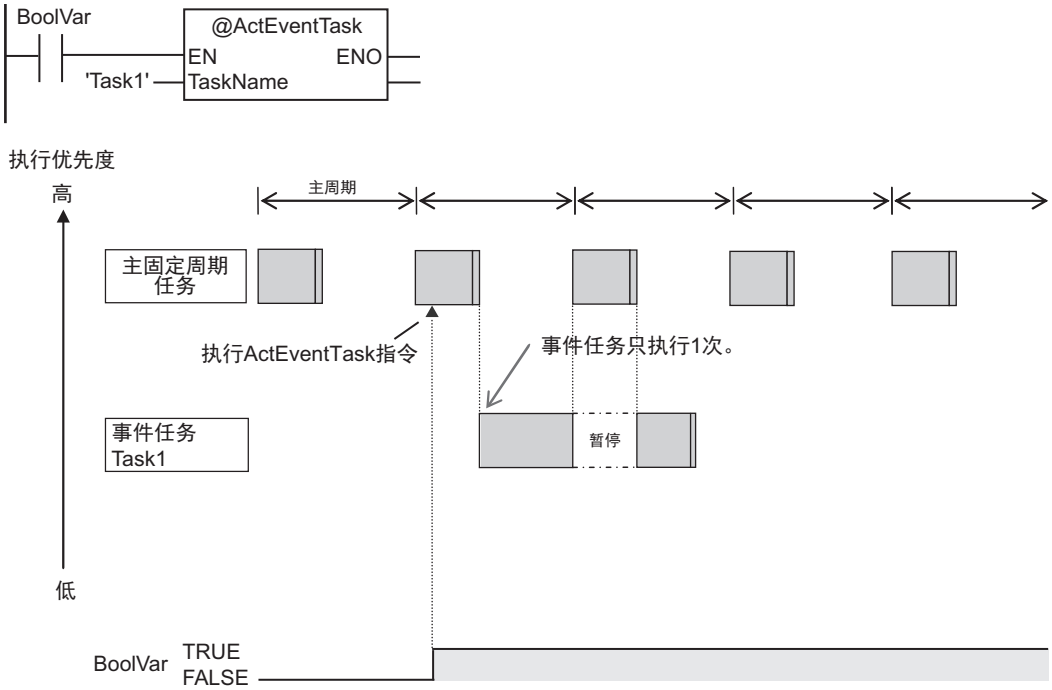
若在程序中执行 ActEventTask 指令，将执行 1 次指定的事件任务。 ActEventTask 指令的详细规格请参考《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

若使用 ActEventTask 指令，看到用户程序后，能立即知道是启动哪个事件任务。此外，在变量的条件式一致判定前，即使事件任务的执行条件可能发生变化，仍为有效。

● 使用 ActEventTask 指令的用户程序示例

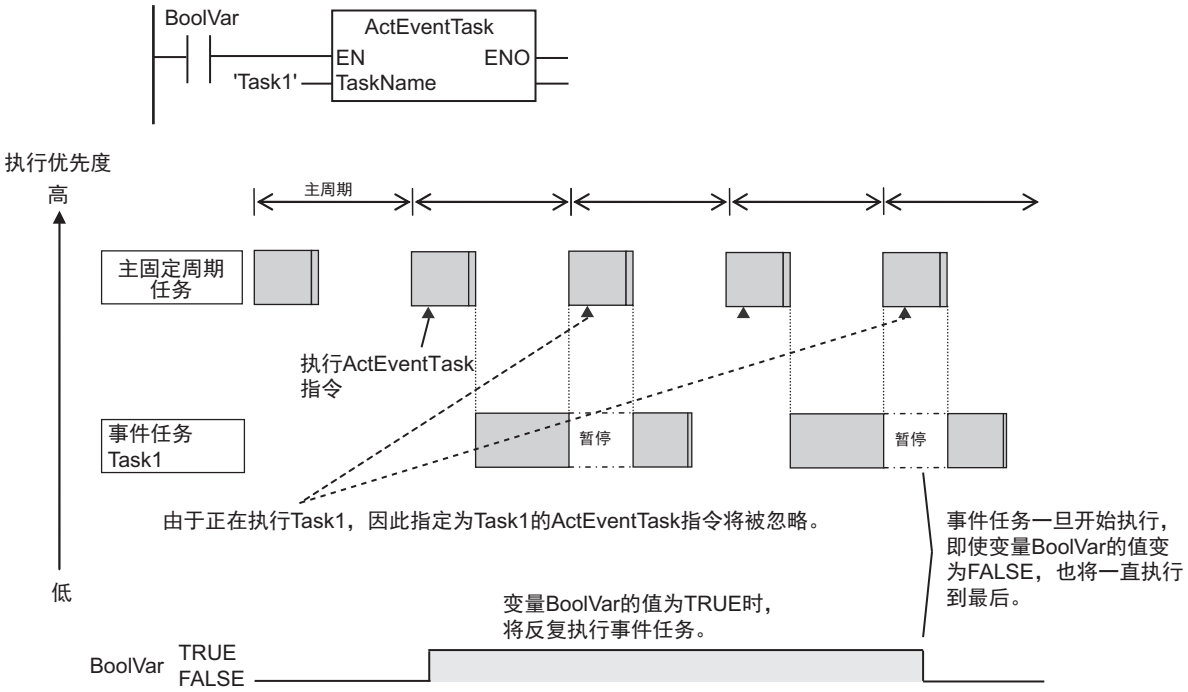
例 1）变量值发生变化时，执行 1 次事件任务

如下图所示，若在 ActEventTask 指令中加入输入上升沿微分的动作选项，当 BOOL 型变量 BoolVar 的值从 FALSE 变为 TRUE 时，将执行 1 次事件任务 Task1。



例 2）变量为指定值时，反复执行事件任务

如下图所示，若 ActEventTask 指令中不加入输入上升沿微分的动作选项，当 BOOL 型变量 BoolVar 的值为 TRUE 时，将反复执行事件任务 Task1。但是，在执行 Task1 时，即使执行指定了 Task1 的 ActEventTask 指令，也将忽略。此外，事件任务一旦开始执行，即使在执行过程中 BoolVar 的值变为 FALSE，也将一直执行到最后。



变量的条件式一致后执行事件任务

变量的条件式一致后执行的方法是指 Sysmac Studio 中指定的变量值与指定的条件式一致时，执行 1 次事件任务。该变量的值与条件式一致期间，不会持续执行事件任务，而是在变量的值从与条件式不一致变为一致时，执行 1 次事件任务。
变量的条件式一致后执行时，不需要执行事件任务所需的用户程序。

● 可指定条件式的变量种类

可指定条件式的变量种类如下所示。

变量的种类		可否指定
系统定义变量		可*1
准用户定义变量		可
用户定义变量	全局变量	可
	程序中使用的变量	可
	功能块中使用的变量	可*2
	函数中使用的变量	不可

*1. 以下变量除外。
EN,ENO,P_Off,P_CY,P_First_RunMode,P_First_Run,P_PRGER
*2. 输入输出变量除外。

● 可指定条件式的变量的数据类型

可指定条件式的变量的数据类型如下所示。

数据类型的分类	数据类型		可否指定
基本数据型	布尔型、位列型、整数型、实数型		可
	持续时间型、日期型、时刻型、日期时刻型、字符串型		不可
数据类型的指定	排列指定	排列整体	不可
		要素	可*1
派生数据型	结构体型	结构体整体	不可
		成员	可*2
	联合体型	联合体整体	不可
		成员	可 *2
	列举型		可

*1. 排列要素的数据类型仅限布尔型、位列型、整数型、实数型中的一个。
*2. 成员的数据类型仅限布尔型、位列型、整数型、实数型中的一个。

● 可指定的条件式

因指定条件式的变量数据类型不同，可指定的条件式如下。
指定条件式的变量数据类型为位列型、整数型、实数型时，需要设定变量的值和要比较的比较常数。

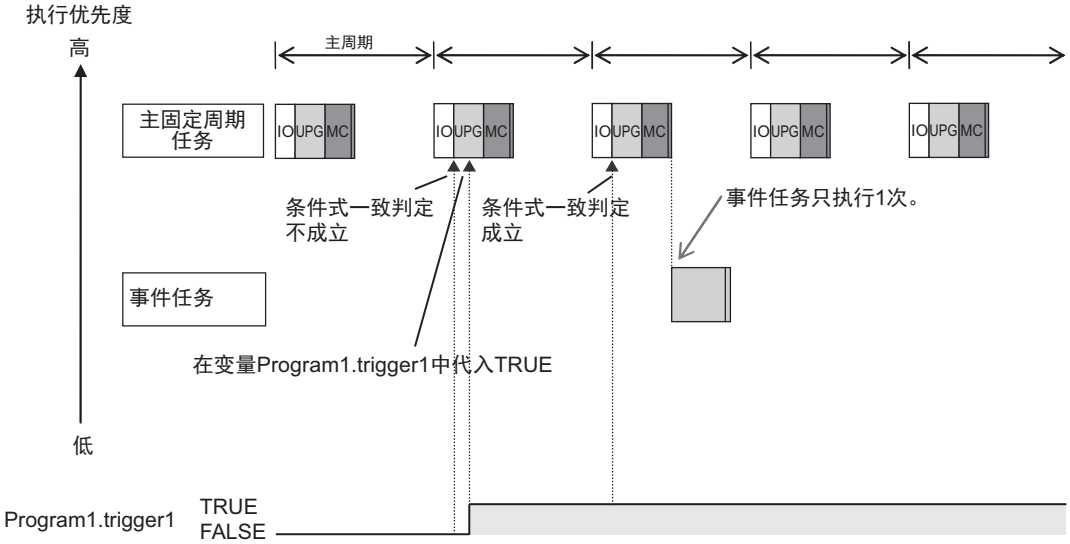
数据类型	可指定的条件式
布尔型、布尔型排列要素、布尔型结构体成员、布尔型联合体成员	上升沿 FALSE → TRUE
	下降沿 TRUE → FALSE
位列型、整数型、实数型、 或以上数据类型的排列要素、结构体成员、联合体成员	变量 = { 比较常数 }
	变量 ≠ { 比较常数 }
	变量 > { 比较常数 }
	变量 ≧ { 比较常数 }
	变量 < { 比较常数 }
	变量 ≦ { 比较常数 }

● 比较常数的值的有效范围

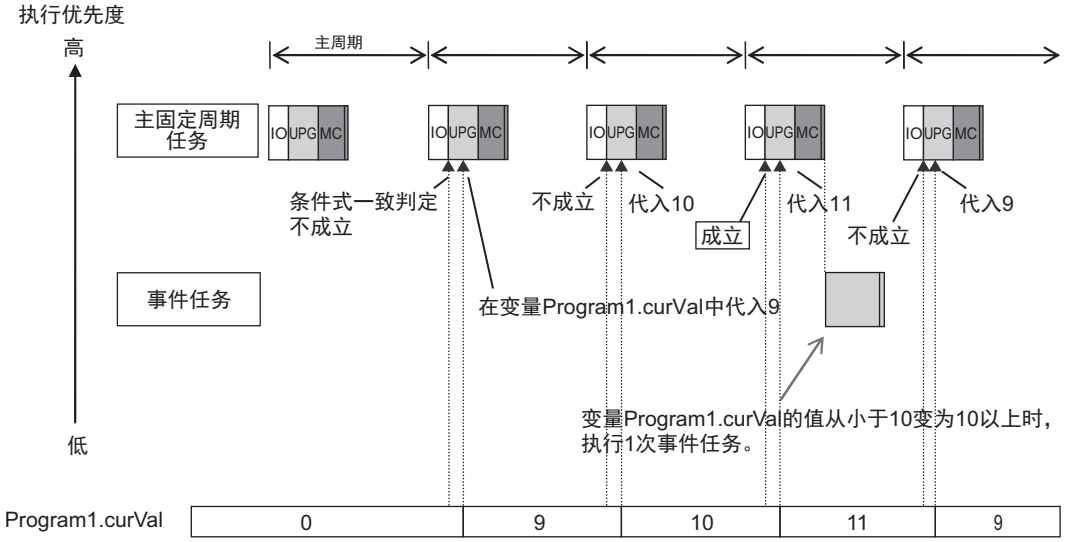
指定条件式的变量数据类型为位列型、整数型、实数型时，需要设定变量的值和要比较的比较常数。
比较常数的值的有效范围与指定条件式的变量之数据类型值的有效范围相同。
各数据类型的值的有效范围请参考📖 「基本数据型和派生数据型」（P.6-31）。
例如，指定条件式的变量之数据类型为 BYTE 时，比较常数的值的有效范围为 BYTE#16#00 ～ BYTE#16#FF。

● 变量的条件式一致后执行事件任务的示例

例 1)事件任务的执行条件设定为 BOOL 型变量 Program1.trigger1 的上升沿时
Program1.trigger1 的值从 FALSE 变为 TRUE 时，执行 1 次事件任务。



例 2)事件任务的执行条件设定为 INT 型变量 Program1.curVal ≧ 10 时
Program1.curVal 的值从小于 10 变为 10 以上时，执行 1 次事件任务。



使用注意事项

执行条件式一致判定的时间为执行主固定周期任务的用户程序前。因此，在主固定周期任务的用户程序中，即使指定变量的值与条件式一致，仍要在下一个主固定周期任务中执行判定。所以，执行事件任务的时间为：变量的值与条件式一致的主固定周期任务之下一个主固定周期任务完成后。

5-4-5 NJ 系列事件任务的执行时序

事件任务的执行优先度为 8 或 48。若正在执行其他任务时事件任务的执行条件成立，将先执行优先度高的任务，优先度低的任务将中断。这与主固定周期任务或固定周期任务相同。但是，事件任务的执行时序因事件任务的执行条件不同而异。
此外，事件任务可将多个任务分配为相同的执行优先度。执行优先度相同的多个事件任务的执行条件成立时，需要注意。

执行时序随事件任务执行条件的不同

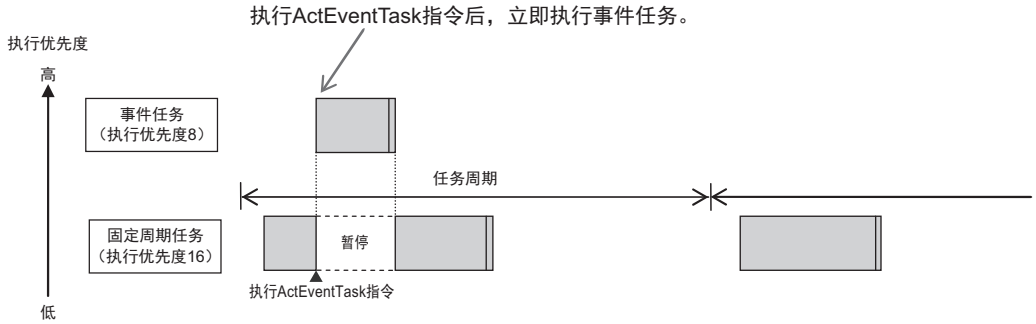
因事件任务的执行条件为利用 ActEventTask 指令还是利用变量的条件式一致不同，执行时序不同。

● 执行条件为利用 ActEventTask 指令时的执行时序

事件任务的执行条件为利用 ActEventTask 指令时，执行 ActEventTask 指令后，立即执行事件任务。

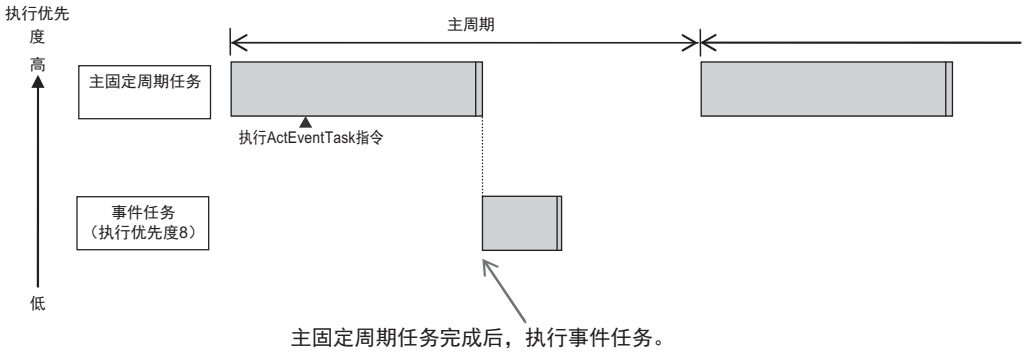
例 1）执行 ActEventTask 指令的固定周期任务的执行优先度为 16、事件任务的执行优先度为 8 时

事件任务 （执行优先度 8）的执行优先度高于固定周期任务 （执行优先度 16），因此将中断固定周期任务 （执行优先度 16），执行事件任务 （执行优先度 8）。



例 2）在主固定周期任务中执行 ActEventTask 指令时

事件任务的执行优先度低于主固定周期任务，因此将在主固定周期任务完成后执行事件任务。

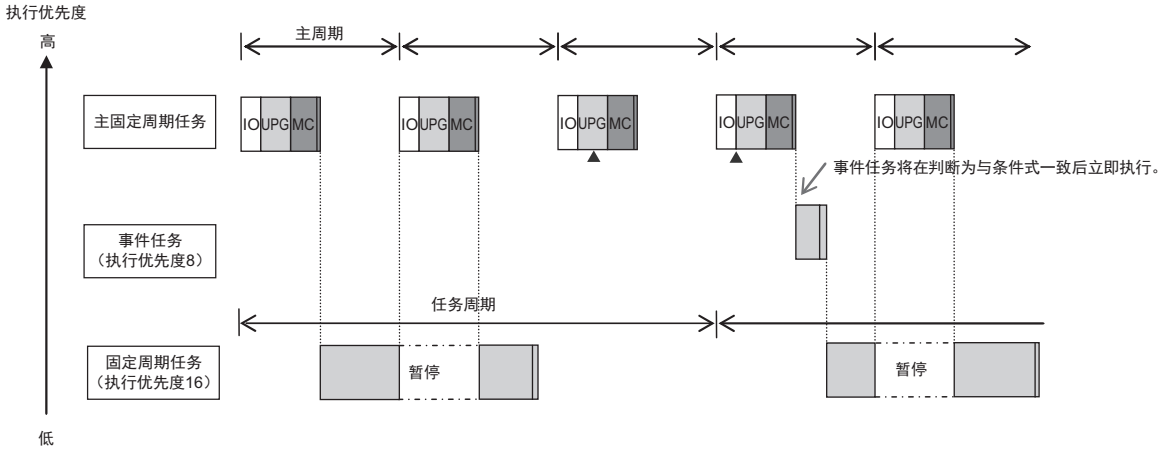


● 执行条件为变量的条件式一致时的执行时序

条件式一致判定在主固定周期任务中执行。所以，执行事件任务的时间为：指定变量的值与条件式一致后，最先执行的主固定周期任务之后。但是，存在执行优先度高于事件任务的任务时，将先执行该任务。

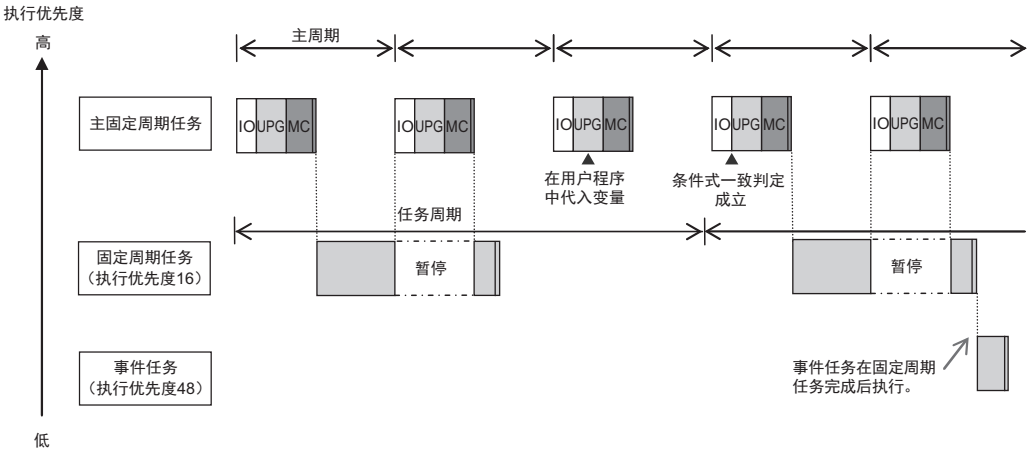
例 1）存在固定周期任务 （执行优先度 16）和事件任务 （执行优先度 8）时

发生条件式一致后，在最先执行的主固定周期任务之后，执行事件任务。事件任务 （执行优先度 8）的执行优先度高于固定周期任务 （执行优先度 16），因此将在事件任务之后执行固定周期任务 （执行优先度 16）。



例 2）存在固定周期任务 （执行优先度 16）和事件任务 （执行优先度 48）时

事件任务的执行优先度低于固定周期任务 （执行优先度 16），因此将在固定周期任务 （执行优先度 16）完成后执行事件任务。



使用注意事项

- 由主固定周期任务的I/O刷新发生条件式一致时以及由主固定周期任务中分配的用户程序执行发生时，事件任务执行时序不同，具体如下。这是因为执行条件式一致判定的是主固定周期任务中的系统共通处理 1。主固定周期任务的处理将按 I/O 刷新➞系统共通处理 1 ➞用户程序执行的顺序执行。

条件式一致发生原因	事件任务的执行时序
主固定周期任务的 I/O 刷新	该主固定周期任务完成后
主固定周期任务的用户程序执行	下一主固定周期任务完成后

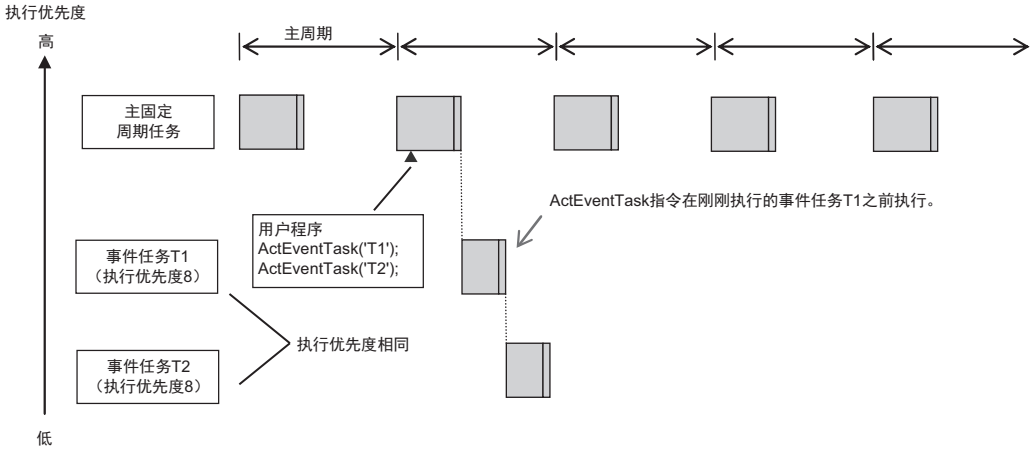
- 执行事件任务的时间：前一次判定时为条件式不一致，下一次判定时为条件式一致时。因此，即使发生从不一致到一致的变化，若在下次判定前恢复为不一致，将不会执行事件任务。

执行优先度相同的事件任务的执行时序

事件任务可将多个任务分配为相同的执行优先度。执行优先度相同的多个事件任务的执行条件成立时，按执行条件成立的顺序执行。

例 1）执行了 2 个 ActEventTask 指令时

如下图所示，先执行的 ActEventTask 指令所指定的事件任务 T1 将在事件任务 T2 之前执行。



例 2）变量的条件式一致和 ActEventTask 指令同时存在时

假设事件任务 T1、 T2、 T3 的执行条件设定如下。

- T1：变量 VarA 的条件式一致
- T2：变量 VarB 的条件式一致
- T3：ActEventTask 指令

如下图所示，在同一主周期中，I/O 刷新后 VarA 的条件式一致、在用户程序中执行 ActEventTask 指令、用户程序中发生 VarB 的条件式一致时，按以下顺序动作。

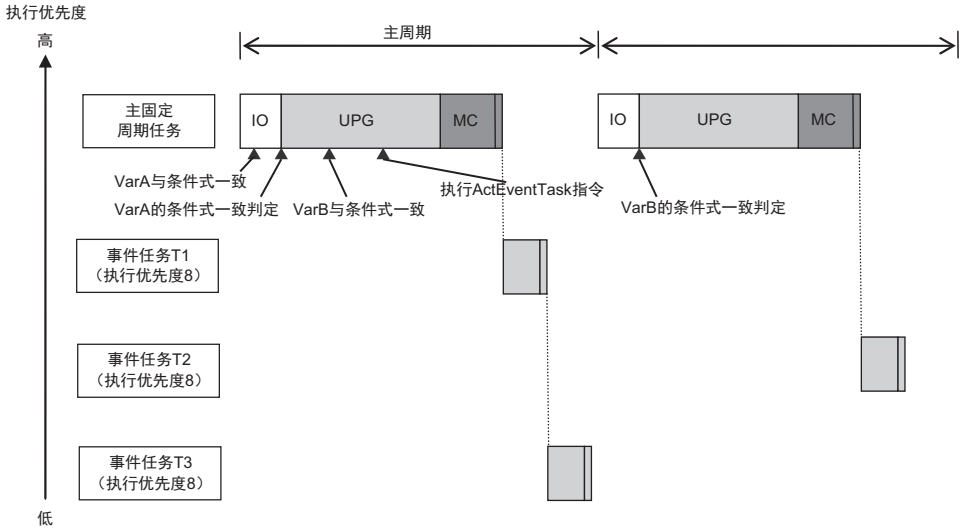
(1) I/O 刷新后 VarA 与条件式一致。

(2) 在系统共通处理 1 中， VarA 判定为与条件式一致，执行 T1。由于 VarB 与条件式不一致，因此不执行 T2。

(3) 用户程序中 VarB 与条件式一致。

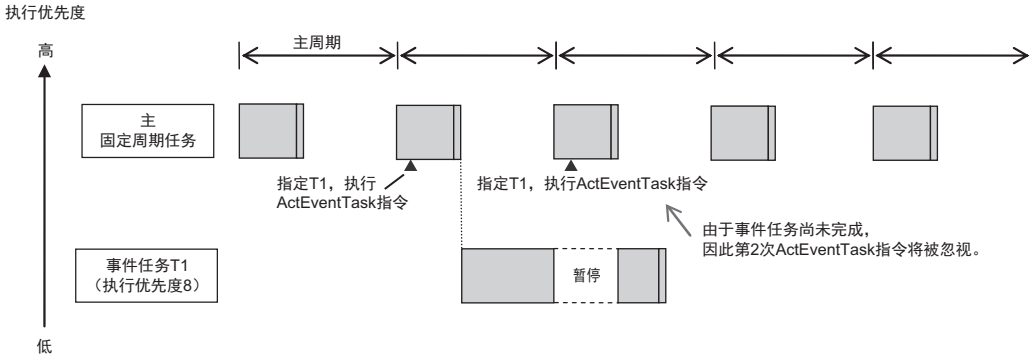
(4) 在程序中执行 ActEventTask 指令，执行 T3。

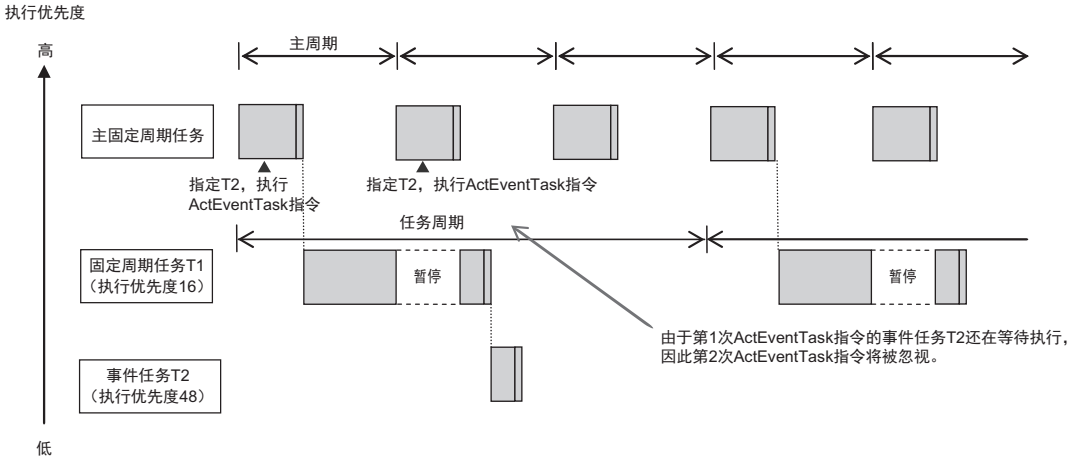
(5) 在下一主周期的系统共通处理 1 中， VarB 判定为与条件式一致，执行 T2。



5-4-6 事件任务完成前，执行条件再次成立时的动作

事件任务完成前，该事件任务的执行条件再次成立时，第 2 次执行条件的成立将被忽略。事件任务完成前是指事件任务执行中及事件任务执行等待中两种情况。此外，事件任务一旦开始执行，即使执行条件变为不成立，也将一直执行到最后。





5-5 标签数据链接服务和系统服务

CPU 单元除了主固定周期任务、固定周期任务、事件任务之外，还会执行其他处理。这些处理包括标签数据链接服务和系统服务两种。两者的处理内容、执行优先度、执行时序如下所示。

项目	标签数据链接服务	系统服务
执行优先度	固定周期任务 （执行优先度 16）和固定周期任务 （执行优先度 17）之间。	执行优先度比所有任务低。
处理内容	在 EtherNet/IP 网络上，使用其他控制器、设备和标签进行数据交换。使用 CPU 单元的内置 EtherNet/IP 端口或 CJ 系列 EtherNet/IP 单元 CJ1W-EIP21。*1	USB 端口服务或 SD 存储卡服务等执行优先度较低的处理的总称。
执行时序	执行间隔或 1 次处理的执行时间因 CPU 单元的型号或标签数据链接设定不同而异。*2	NX 系列 CPU 单元中，不受任务或标签数据链接服务的影响，始终执行。 NJ 系列 CPU 单元中，在任务或标签数据链接服务的空闲时间内执行。*3

*1. 标签数据链接的详细处理内容请参考《NJ/NX 系列 CPU 单元 内置 EtherNet/IP 端口 用户手册（SBCD-CN5-359）》或《CJ 系列 EtherNet/IP 单元 用户手册 NJ 连接篇 （SBCD-CN5-355）》。
可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

*2. 标签数据链接的执行间隔和执行时间的详情请参考「5-5-2 标签数据链接服务的处理内容和执行时序」（P.5-50）。

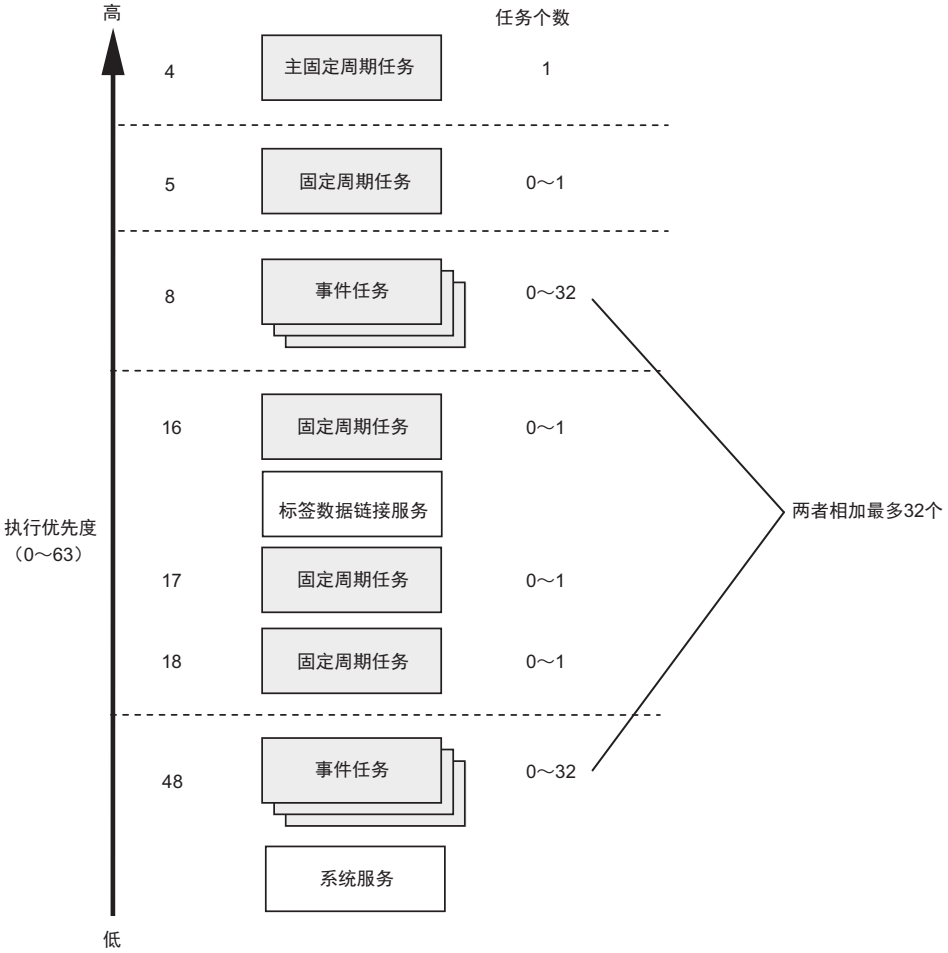
*3. 但是， NJ 系列 CPU 单元中，无法确保足够的系统服务执行时间时，将暂停执行优先度 17 以上的任务，而执行系统服务。执行系统服务所需的时间在 Sysmac Studio 的 “ 系统服务监视设定 ” 中设定。

5-5-1 标签数据链接服务和系统服务的执行优先度和执行顺序

下面介绍标签数据链接服务和系统服务的执行优先度和执行顺序示例。

标签数据链接服务和系统服务的执行优先度

标签数据链接服务和系统服务的执行优先度如下图所示。标签数据链接服务的执行优先度在固定周期任务 （执行优先度 16）和固定周期任务 （执行优先度 17）之间。系统服务的执行优先度比所有任务低。



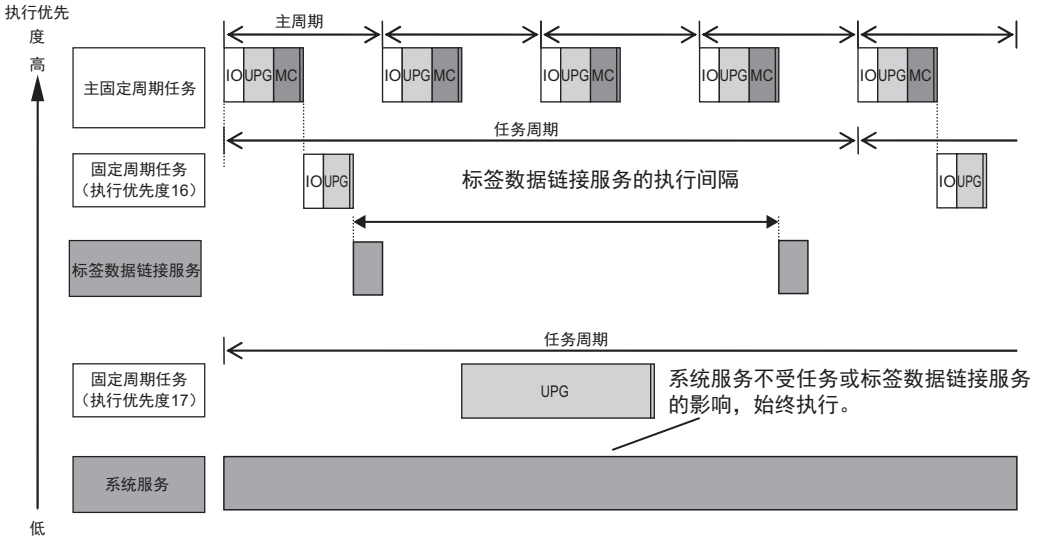
（注）可使用固定周期任务 （执行优先度 5）的单元仅限 NX 系列 CPU 单元。

标签数据链接服务和系统服务的执行顺序示例

主固定周期任务、固定周期任务（执行优先度 16）、固定周期任务（执行优先度 17）、标签数据链接服务、系统服务同时存在时，各任务的执行顺序示例。

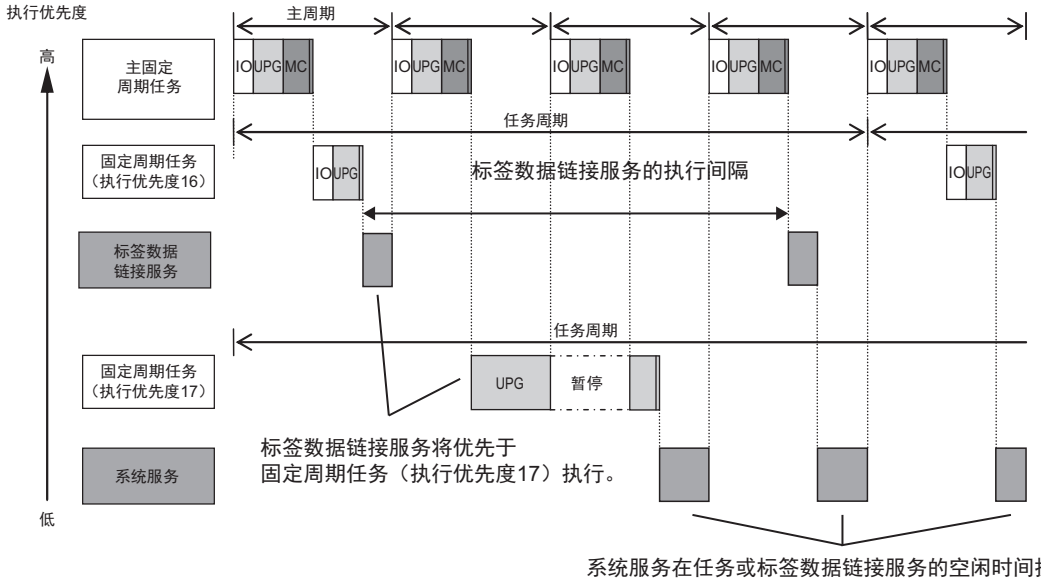
● NX 系列 CPU 单元时

- 标签数据链接服务优先于固定周期任务（执行优先度 17）执行。但是，主固定周期任务或固定周期任务（执行优先度 16）的优先度更高，因此优先执行。
- 系统服务不受任务或标签数据链接服务的影响，始终执行。



● NJ 系列 CPU 单元时

- 标签数据链接服务优先于固定周期任务（执行优先度 17）执行。但是，主固定周期任务或固定周期任务（执行优先度 16）的优先度更高，因此优先执行。
- 系统服务在所有任务或标签数据链接服务的空闲时间执行。



标签数据链接服务的执行间隔或 1 次的执行时间因 CPU 单元的型号或标签数据链接设定不同而异。详情请参考📖「5-5-2 标签数据链接服务的处理内容和执行时序」（P.5-50）。

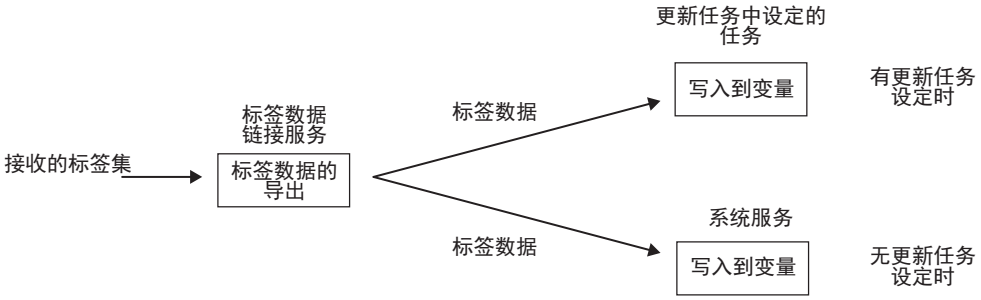
5-5-2 标签数据链接服务的处理内容和执行时序

标签数据链接服务的处理内容

标签数据链接处理细分为多个处理。这些处理通过标签数据链接服务、系统服务、任务来执行。下面表示使用 CPU 单元内置 EtherNet/IP 端口进行标签数据链接处理时的示例。

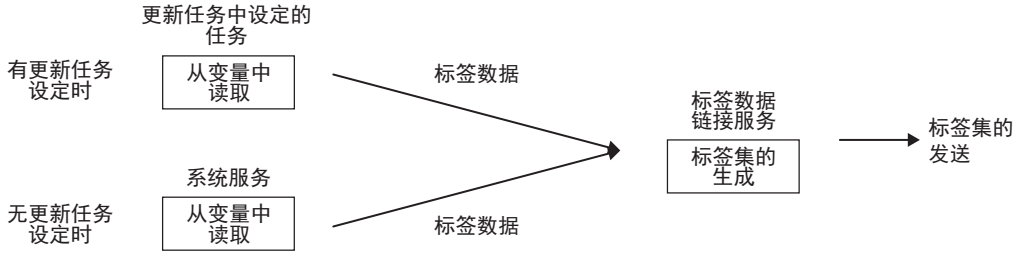
● 标签数据接收处理的流程

- 通过标签数据链接服务，从接收的标签集中导出标签数据。
- 设定有更新任务时，利用该任务，将标签数据的值写入到标签上分配的变量中。
- 未设定更新任务时，利用系统服务，将标签数据的值写入到标签上分配的变量中。



● 标签数据发送处理的流程

- 设定有更新任务时，利用该任务，从标签上分配的变量中读取标签数据的值。
- 未设定更新任务时，利用系统服务，从标签上分配的变量中读取标签数据的值。
- 然后，通过标签数据链接服务，根据标签数据生成标签集，然后发送。



参考

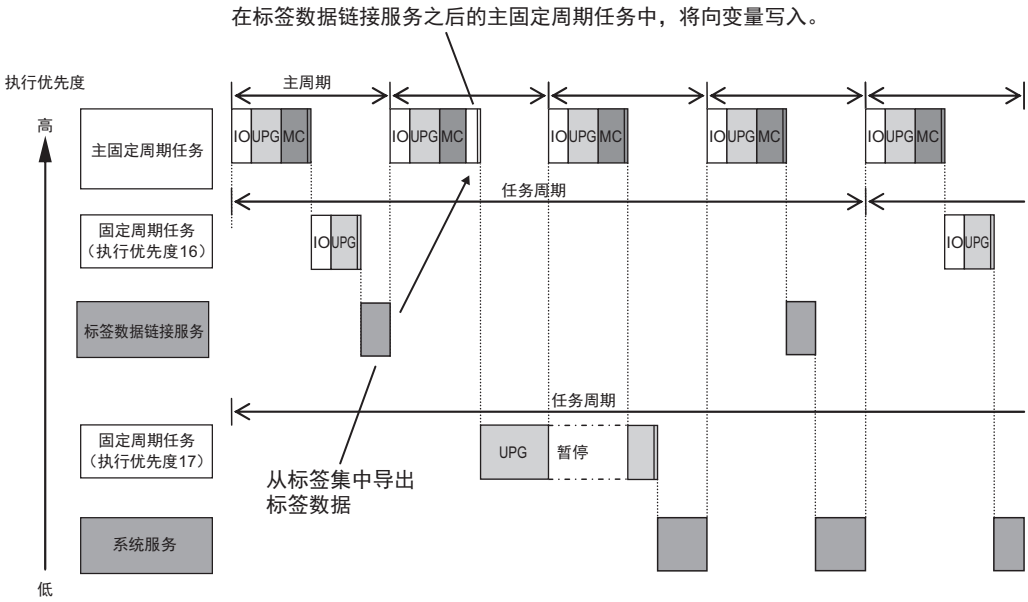
有或无更新任务设定时标签数据链接处理时序的不同

有或无更新任务设定时，向变量写入值的处理不同。有更新任务设定时，利用该任务写入值，无时，利用系统服务。因此，标签数据链接处理的执行时序也会不同。

以下表示在 NJ 系列 CPU 单元中接收标签数据链接时，执行时序的不同。

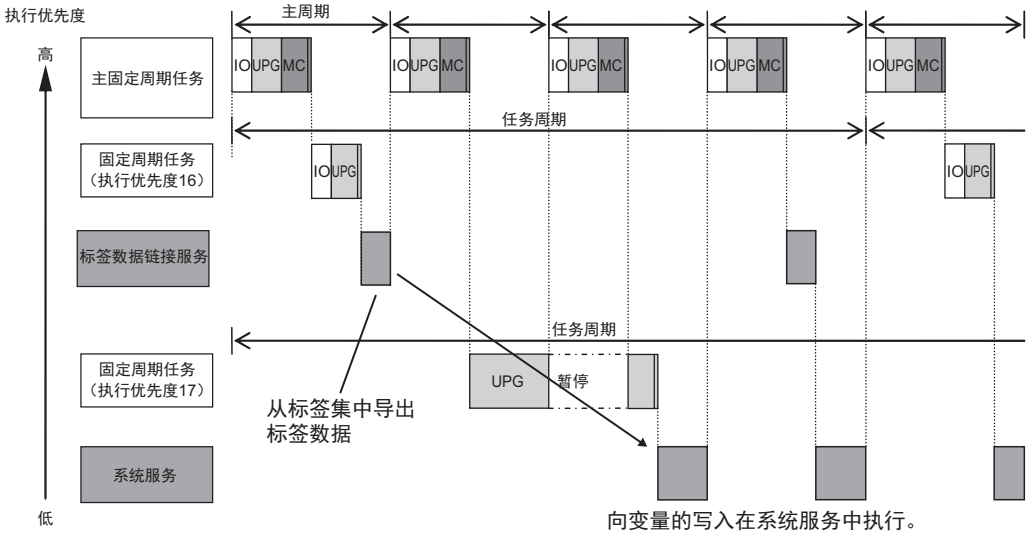
- 有更新任务设定时

  将主固定周期任务设定为更新任务时的示例。在标签数据链接服务之后的主固定周期任务中，写入到变量。



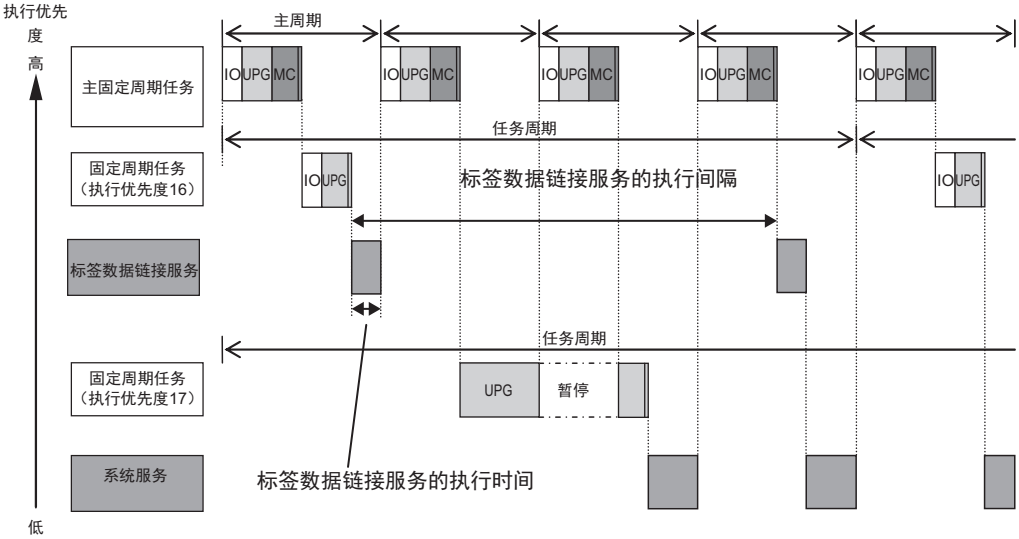
- 无更新任务设定时

  向变量的写入在系统服务中执行。



标签数据链接服务的执行时序

标签数据链接服务的执行间隔和执行时间因 CPU 单元的型号或标签数据链接设定不同而异。下表为参考值。但是，NJ 系列 CPU 单元时，若执行了优先度高于标签数据链接服务的任务，执行间隔和执行时间都可能大于以下的值。



● 使用 CPU 单元内置 EtherNet/IP 端口时

CPU 单元	标签数据链接设定			执行间隔 [ms]	执行时间 [ms]
	标签数	连接数	RPI[ms]		
NJ501- □□□□	8	1	1	1	0.02 ～ 0.04
		4	2	1	0.02 ～ 0.10
		32	50	1	0.02 ～ 0.10
NJ301- □□□□	8	1	1	1	0.03 ～ 0.06
		4	2	1	0.03 ～ 0.15
		32	50	1	0.03 ～ 0.15
NJ101- □□□□	8	1	1	1	0.04 ～ 0.09
		4	2	1	0.04 ～ 0.28
		32	50	1	0.04 ～ 0.28

参考

NX 系列 CPU 单元时，标签数据链接服务的执行不会影响到固定周期任务（执行优先度 17、18）及系统服务。

● **使用 CJ 系列 EtherNet/IP 单元 CJ1W-EIP21 时**

CPU 单元	标签数据链接设定				执行间隔 [ms]	执行时间 [ms]
	标签数	连接数	RPI[ms]	发送 / 接收		
NJ501- □□□□	8	1	1	发送	3	0.10
				接收	3	0.10
		4	2	发送	4	0.22
				接收	4	0.13
		32	50	发送	12	0.68
				接收	6 ～ 35	0.23 ～ 0.66
NJ301- □□□□	8	1	1	发送	3	0.16
				接收	3	0.16
		4	2	发送	3	0.21
				接收	4	0.33
		32	50	发送	15	1.05
				接收	8 ～ 35	0.29 ～ 1.85
NJ101- □□□□	8	1	1	发送	3	0.21
				接收	3	0.21
		4	2	发送	6	0.46
				接收	13	0.80
		32	50	发送	12 ～ 50	2.97 ～ 4.09
				接收	25	2.36

版本相关信息

标签数据链接处理所执行的处理，根据 CPU 单元的单元版本与 Sysmac Studio 版本的组合，有如下不同。

CPU 单元的单元版本	Sysmac Studio 的版本	
	Ver.1.03 以下	Ver.1.04 以上
Ver.1.03 以上	系统服务中的内置 EtherNet/IP 端口服务	标签数据链接服务*1
Ver.1.02 以下	系统服务中的内置 EtherNet/IP 端口服务	

*1. 在 Sysmac Studio 中将单元版本设定为 Ver.1.02 以下时，标签数据链接处理在系统服务中的内置 EtherNet/IP 端口服务中执行。

5-5-3 系统服务的处理内容和执行时序

系统服务的处理内容

系统服务的处理中包括以下内容。

系统服务的处理名称	内容
USB 端口服务	• 对来自编程设备、上位计算机的服务请求进行处理
内置 EtherNet/IP 端口服务	• 对来自编程设备、显示器、上位计算机或其他控制器的服务请求 （CIP 等）进行处理 • 通信指令 （CIP、套接字）的执行
内置 EtherCAT 端口服务	• EtherCAT 信息通信的执行
CJ 高功能单元服务*1	• 执行与 CJ 高功能单元的事件服务 • 通信指令 （CIP）的执行
SD 存储卡服务	• 从 FTP 客户端访问 • 通过 Sysmac Studio 对 SD 存储卡进行操作 • SD 存储卡指令的执行
自诊断处理	• 硬件的异常检测

*1. 与 CJ 高功能单元的分配区域数据交换在 “I/O 刷新 ” 时执行。
可使用 CJ 高功能单元的 CPU 单元仅限 NJ 系列 CPU 单元。

系统服务的执行时序

NX 系列 CPU 单元的系统服务不受任务或标签数据链接服务的影响，始终执行。系统服务的执行时间不会发生不足。

NJ 系列 CPU 单元的系统服务在任务或标签数据链接服务的空闲时间内执行。
但是，无法确保足够的系统服务执行时间时，将暂停执行优先度 17 以上的任务，而执行系统服务。
执行系统服务所需的时间在 Sysmac Studio 的 “ 系统服务监视设定 ” 中设定。

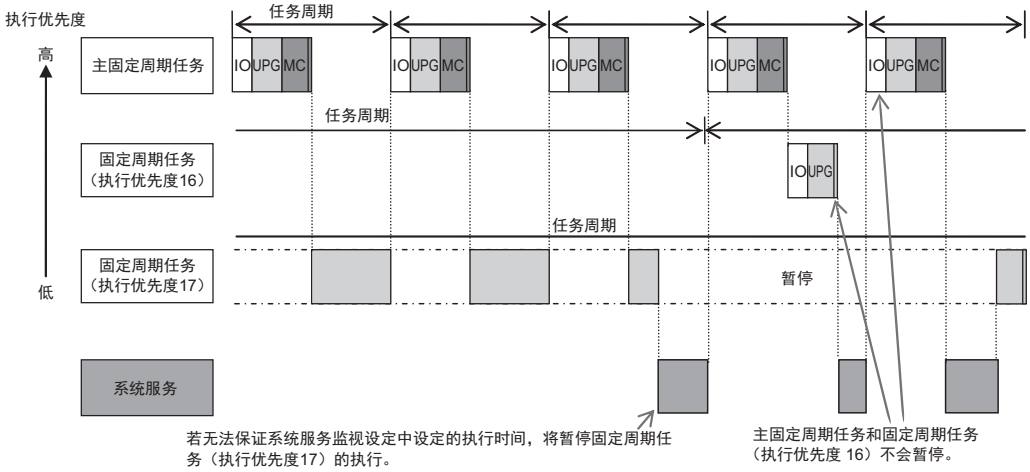
系统服务的处理没有优先顺序。各处理在进行时间切片后并列执行。

● 无法确保足够的系统服务执行时间时的时序

NJ 系列 CPU 单元中，任务或标签数据链接服务的空闲时间较少，无法确保足够的系统服务执行时间时，将暂停执行优先度 17 以上的任务，而执行系统服务。

暂停执行优先度 17 以上的任务后，仍无法确保系统服务的执行时间时，将发生 “ 系统服务时间不足 ”，停止执行用户程序。

执行系统服务所需的时间在 Sysmac Studio 的系统服务监视设定中设定。



● 系统服务监视设定

系统服务监视设定的作用是监视 NJ 系列 CPU 单元上是否能确保足够的系统服务执行时间。“ 系统服务执行间隔 ” 用于监视所有任务的未执行时间是否超出 “ 系统服务执行间隔 ” 的 “ 系统服务执行时间比例 ”。

系统服务监视设定可在 Sysmac Studio 的 “ 动作设定 ” ─ “ 一般设定 ” 中设定。

设定大项目	设定中项目	设定小项目 [单位]	功能说明	设定值	默认	反映时序	运行模式中改写
动作设定 ─ 动作设定标签 ─ 一般设定	系统服务监视设定	系统服务执行间隔 [ms]	设定执行系统服务的间隔。	10ms ～ 1s	10ms	设定传送时	不可
		系统服务执行时间比例 [%]	设定监视值，用于监视执行系统服务的比例。	5% ～ 50%	10%	设定传送时	不可

参考

NX 系列 CPU 单元中，系统服务的执行时间不会发生不足。因此，无系统服务监视设定。也不会发生系统服务时间不足的异常。

使用注意事项

- NJ 系列 CPU 单元中，为了确保足够的系统服务和任务执行时间，请在满足系统服务处理响应性能的范围内，将系统服务监视设定设为最小的值。系统服务监视设定的作用是监视是否能确保设定的系统服务执行时间。并非用于确保设定的系统服务执行时间。
- NJ系列CPU单元中，进行任务设计时，应确保足够的系统服务执行时间。关于任务的设定方法，请参考「5-10 任务设计方法及输入输出响应时间」（P.5-87）。
- NJ 系列 CPU 单元中，想要增加系统服务的执行时间时，请增大任务周期等，以增加所有任务的空闲时间。
- NJ系列CPU单元中，暂停执行优先度17以上的任务后，仍无法确保系统服务监视设定中设定的系统服务执行时间时，将发生 “ 系统服务时间不足 ”，停止执行用户程序。
- NJ 系列CPU 单元中，发生系统服务执行时间不足，执行优先度 17 以上的任务暂停时，暂停的任务中可能发生 “ 任务周期超限 ”。设计任务时，应保证系统服务按系统服务监视设定中设定的时间动作后，仍能在周期内完成执行优先度 17 以上的任务。

5-6 任务相关的分配和设定

下面介绍任务相关的分配和设定。

5-6-1 分配到 I/O 刷新的任务中

将 EtherCAT 从站及 CJ 系列单元的 I/O 刷新分配到任务中。可分配的任务和分配单位因 I/O 刷新对象的不同而异。分配单位是指可分配到 1 个 I/O 刷新上的 I/O 刷新对象的总和。例如，分配单位为从站终端时，即使 1 台通信耦合器单元上连接了多台 NX 单元，可分配 I/O 刷新的任务仅为 1 个。

想要在未分配 I/O 刷新的任务中进行输入输出时，请参考「未分配 I/O 刷新的任务的输入输出」（P.5-59）。

I/O 刷新对象、可分配的任务和分配单位的关系如下。

I/O 刷新对象	可分配的任务	分配单位
通信耦合器单元 但是，从站终端上有 NX 系列安全控制单元	主固定周期任务或固定周期任务（执行优先度 5、16）*1*2	从站终端
通信耦合器单元 但是，在从站终端上，有分配到轴的 NX 系列	主固定周期任务或固定周期任务（执行优先度 5）	
通信耦合器单元 但是，在从站终端上，无分配到轴的 NX 系列	主固定周期任务或固定周期任务（执行优先度 5、16）*2	
[轴] 中分配的 EtherCAT 从站	主固定周期任务或固定周期任务（执行优先度 5）*2	从站
上述以外的 EtherCAT 从站	主固定周期任务或固定周期任务（执行优先度 5、16）*2	
CJ 系列的基本 I/O 单元*3	主固定周期任务或固定周期任务（执行优先度 16）	单元
CJ 系列的高功能 I/O 单元 *3		
CJ 系列的 CPU 总线单元 *3		

*1. 有多个安装了 NX 系列安全控制单元的从站终端时，不可将固定周期任务（执行优先度 5）和固定周期任务（执行优先度 5）以外的固定周期任务组合后分配。
*2. 可使用固定周期任务 （执行优先度 5）的单元仅限 NX 系列 CPU 单元。
*3. 可使用各种 CJ 系列单元的 CPU 单元仅限 NJ 系列 CPU 单元。

安全要点

联合 CJ 系列基本单元和 EtherCAT 从站等不同的功能模块对装置进行控制时，请在用户程序或外部采取对策，确保在一个功能模块停止时也不会造成影响。各功能模块发生部分停止故障等级的异常时，相应的输出遵照从站或单元的规格。
部分停止故障等级的控制器异常请参考📖「12-1-3 CPU 单元可动作的异常」（P.12-6）。

参考

使用 EtherCAT 从站和 CJ 单元时，若将 I/O 刷新分配到不同的任务中，可缩短主固定周期任务的任务执行时间。主固定周期任务中请分配 EtherCAT 从站，固定周期任务 （执行优先度 16）中请分配 CJ 单元。

Sysmac Studio 上的设定操作

未分配到轴的各从站 / 单元应在 Sysmac Studio 的 [任务设定]–[设定控制 I/O 的任务] 中设定要进行 I/O 刷新的任务。
详情请参考📖「设定控制 I/O 的任务」（P.4-8）。

已分配到轴的各从站 / 单元若在 Sysmac Studio 的 [运动控制设定] 中选择要使用的运动控制，将设定要进行 I/O 刷新的任务。
详情请参考📖《NJ/NX 系列 CPU 单元 用户手册 运动控制篇 （SBCE-CN5-363）》。

I/O 刷新的时序

EtherCAT 从站及 CJ 系列单元的 I/O 刷新时序如下所示。

I/O 刷新对象	控制 I/O 的任务	I/O 刷新的周期
EtherCAT 从站	主固定周期任务	主固定周期任务的任务周期
	固定周期任务 （执行优先度 5）*1	固定周期任务 （执行优先度 5）的任务周期
	固定周期任务 （执行优先度 16）	主固定周期任务的任务周期*2
CJ 单元*3	主固定周期任务	主固定周期任务的任务周期
	固定周期任务 （执行优先度 16）	固定周期任务 （执行优先度 16）的任务周期

*1. 可使用固定周期任务 （执行优先度 5）的单元仅限 NX 系列 CPU 单元。
*2. EtherCAT 通信通过主固定周期任务的 I/O 刷新执行。用固定周期任务（执行优先度 16）控制 EtherCAT 从站时，I/O 刷新的数据更新周期变为固定周期任务 （执行优先度 16）的周期。
*3. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

● 过程数据通信的优先顺序

EtherCAT 的过程数据通信通过 I/O 刷新执行。在 NX 系列 CPU 单元上，可使用主固定周期任务和固定周期任务 （执行优先度 5）执行过程数据通信。

在主固定周期任务和固定周期任务 （执行优先度 5）中执行过程数据通信时，将优先主固定周期任务的过程数据通信。因此，即使两个任务的过程数据通信周期或过程数据大小相同，也是固定周期任务（执行优先度 5）的 I/O 刷新时间更长。

在用户程序上访问 I/O 的方法

通过用户程序访问 I/O 端口时，需要使用设备变量。

请通过 ［设定控制 I/O 的任务］中所设定任务的程序访问设备变量。

未分配 I/O 刷新的任务的输入输出

若试图用未分配 I/O 刷新的任务输出数据，Sysmac Studio 的程序检查时将发生错误。例如，在从站终端中，将各 NX 单元的处理分配到不同的任务时，只能通过分配有 I/O 刷新的任务来输出数据。此时，需要用未分配 I/O 刷新的任务向分配有 I/O 刷新的任务传递数据，然后从分配有 I/O 刷新的任务输出数据。

下面为用未分配 I/O 刷新的任务向分配有 I/O 刷新的任务传递数据并进行输出的示例程序。

此外，本示例程序中，利用未分配 I/O 刷新的任务来输入外部数据。若执行上述处理，Sysmac Studio 的程序检查时会出现警告，但仍可执行。

● 单元构成

使用从站终端。从站终端的单元构成如下所示。

型号	产品名称
NX-ECC201	EtherCAT 耦合器单元
NX-ID3317	DC 输入单元
NX-OD3256	晶体管输出单元

● I/O 映射

I/O 映射如下。表中仅标示示例程序中使用的接点。

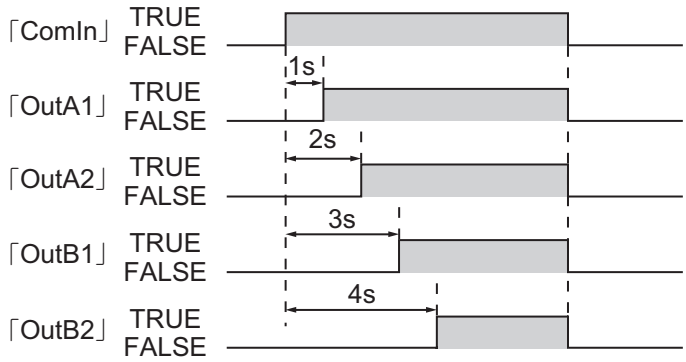
位置	端口	说明	R/W	数据类型	变量	变量注释	变量种类
Unit1	NX-ID3317						
	Input Bit 00	输入接点 00	R	BOOL	ComIn	共通输入值	全局变量
Unit2	NX-OD3256						
	Output Bit 00	输出接点 00	W	BOOL	OutA1	输出值 A1	全局变量
	Output Bit 01	输出接点 01	W	BOOL	OutA2	输出值 A2	全局变量
	Output Bit 02	输出接点 02	W	BOOL	OutB1	输出值 B1	全局变量
	Output Bit 03	输出接点 03	W	BOOL	OutB2	输出值 B2	全局变量

● 输入输出规格

输入输出规格如下所示。

- “ComIn”的值从FALSE变为TRUE后，1秒后“OutA1”、2秒后“OutA2”、3秒后“OutB1”、4秒后“OutB2”的值从 FALSE 变为 TRUE。
- “ComIn”的值从TRUE变为FALSE后，“OutA1”、“OutA2”、“OutB1”、“OutB2”的所有值从TRUE变为FALSE。

时序图如下所示。



● 任务的处理内容

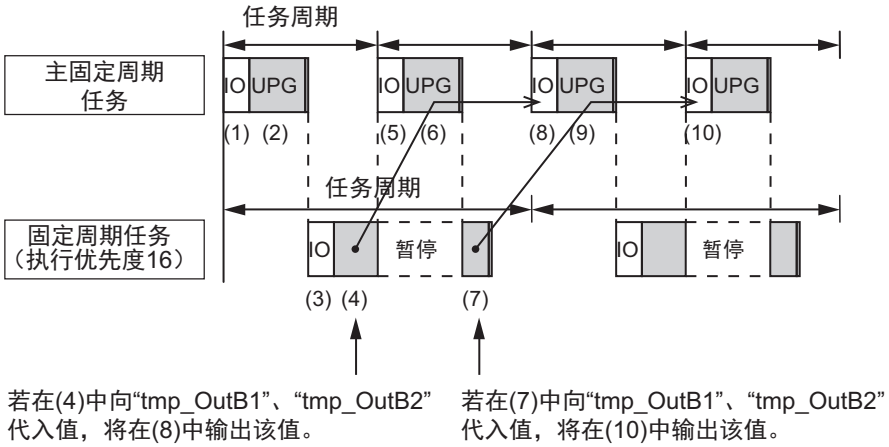
任务中使用主固定周期任务和固定周期任务（执行优先度 16）。将 I/O 刷新分配到主固定周期任务中。

“OutA1”、“OutA2” 的控制在主固定周期任务中执行。从 “ComIn” 的输入，到运算、“OutA1”、“OutA2” 的输出等一系列处理全部在主固定周期任务中执行。

“OutB1”、“OutB2” 的控制中，从 “ComIn” 的输入到运算为止在固定周期任务中执行。固定周期任务中的运算结果将代入到临时变量的全局变量 “tmp_OutB1”、 “tmp_OutB2” 中。主固定周期任务中，将 “tmp_OutB1”、 “tmp_OutB2” 的值代入 “OutB1”、 “OutB2” 后，将输出 “OutB1”、 “OutB2”。

固定周期任务可能因主固定周期任务的执行而暂停。将固定周期任务中的运算结果代入 “tmp_OutB1”、 “tmp_OutB2” 的处理，是在固定周期任务暂停前执行还是在暂停后执行，会影响到主固定周期任务输出固定周期任务运算结果的时序。

处理流程如下图所示。假设固定周期任务的任务周期为主固定周期任务的任务周期的 2 倍。



(1) 通过主固定周期任务的 I/O 刷新，进行 “ComIn” 的输入。

(2) 通过执行主固定周期任务的用户程序进行运算，将值代入 “OutA1”、 “OutA2” 中。
此外，将 “tmp_OutB1”、 “tmp_OutB2” 的值代入 “OutB1”、 “OutB2” 中。但是，由于未通过固定周期任务将值代入 “tmp_OutB1”、 “tmp_OutB2”，所以 “tmp_OutB1”、 “tmp_OutB2” 的值为初始值。因此， “OutB1”、 “OutB2” 的值也是初始值。

(3) 通过固定周期任务的 I/O 刷新，进行 “ComIn” 的输入。

(4) 通过执行固定周期任务的用户程序进行运算。
正在执行固定周期任务时，若开始执行主固定周期任务，将暂停固定周期任务。将值代入 “tmp_OutB1”、 “tmp_OutB2” 的处理，是在固定周期任务暂停前还是暂停后，受固定周期任务中的处理时序影响。

(5) 通过主固定周期任务的 I/O 刷新，进行 “OutA1”、“OutA2”、“OutB1”、“OutB2” 的输出。
同时，将进行第 2 次 “ComIn” 输入。

(6) 通过执行主固定周期任务的用户程序进行运算，将值代入 “OutA1”、 “OutA2” 中。
此外，将 “tmp_OutB1”、“tmp_OutB2” 的值代入 “OutB1”、“OutB2” 中。在 (4) 中将值代入“tmp_OutB1”、“tmp_OutB2”后，固定周期任务的运算结果将反映到“OutB1”、“OutB2” 中。在 (4) 中未将值代入 “tmp_OutB1”、“tmp_OutB2” 时，“tmp_OutB1”、“tmp_OutB2” 的值为初始值。因此， “OutB1”、 “OutB2” 的值也是初始值。

(7) 通过执行固定周期任务的用户程序，进行 (4) 后面的运算。
在 (4) 中未将值代入 “tmp_OutB1”、 “tmp_OutB2” 时，在这里将值代入 “tmp_OutB1”、 “tmp_OutB2” 中。

(8) 通过主固定周期任务的 I/O 刷新，进行 “OutA1”、“OutA2”、“OutB1”、“OutB2” 的输出。
在 (4) 中将值代入 “tmp_OutB1”、 “tmp_OutB2” 后，固定周期任务的运算结果将作为 “OutB1”、 “OutB2” 输出。
在 (7) 中将值代入 “tmp_OutB1”、“tmp_OutB2” 后，将输出 “OutB1”、“OutB2” 的初始值。

(9) 通过执行主固定周期任务的用户程序进行运算，将值代入 “OutA1”、 “OutA2” 中。
此外，将 “tmp_OutB1”、“tmp_OutB2” 的值代入 “OutB1”、“OutB2” 中。在 (7) 中将值代入 “tmp_OutB1”、 “tmp_OutB2” 后，固定周期任务的运算结果将在这里反映到 “OutB1”、 “OutB2” 中。

(10) 通过主固定周期任务的 I/O 刷新，进行 “OutA1”、“OutA2”、“OutB1”、“OutB2” 的输出。
在 (7) 中将值代入 “tmp_OutB1”、 “tmp_OutB2” 后，固定周期任务的运算结果将在这里作为 “OutB1”、 “OutB2” 输出。

此外，为了防止 “tmp_OutB1”、 “tmp_OutB2” 的值在主固定周期任务内参照前被固定周期任务覆盖，应使用 Lock、Unlock 指令，进行任务间的排他控制。任务间的排他控制详情请参考📖「5-7-1 确保任务间变量值同时性的方法」（P.5-69）。

● 全局变量表

全局变量如下所示。

全局变量的变量表

名称	数据类型	初始值	分配目标	注释
ComIn	BOOL	FALSE	ECAT://node#[1,1]/Input Bit 00	共通输入值
OutA1	BOOL	FALSE	ECAT://node#[1,2]/Output Bit 00	输出值 A1
OutA2	BOOL	FALSE	ECAT://node#[1,2]/Output Bit 01	输出值 A2
OutB1	BOOL	FALSE	ECAT://node#[1,2]/Output Bit 02	输出值 B1
OutB2	BOOL	FALSE	ECAT://node#[1,2]/Output Bit 03	输出值 B2
tmp_OutB1	BOOL	FALSE		B1 的临时变量
tmp_OutB2	BOOL	FALSE		B2 的临时变量

● 主固定周期任务的梯形图

主固定周期任务的梯形图如下所示。

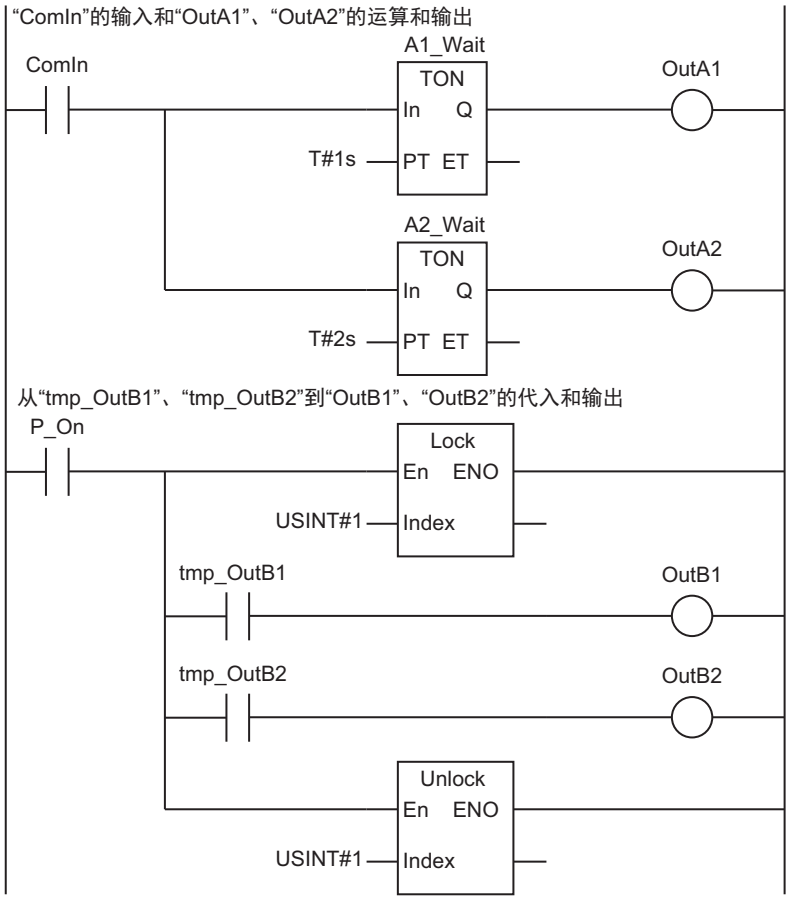
内部变量表

名称	数据类型
A1_Wait	TON
A2_Wait	TON

外部变量表

名称	数据类型	注释
ComIn	BOOL	共通输入值
OutA1	BOOL	输出值 A1
OutA2	BOOL	输出值 A2
OutB1	BOOL	输出值 B1
OutB2	BOOL	输出值 B2
tmp_OutB1	BOOL	B1 的临时变量
tmp_OutB2	BOOL	B2 的临时变量

算法部分



● 固定周期任务的梯形图

固定周期任务的梯形图如下所示。

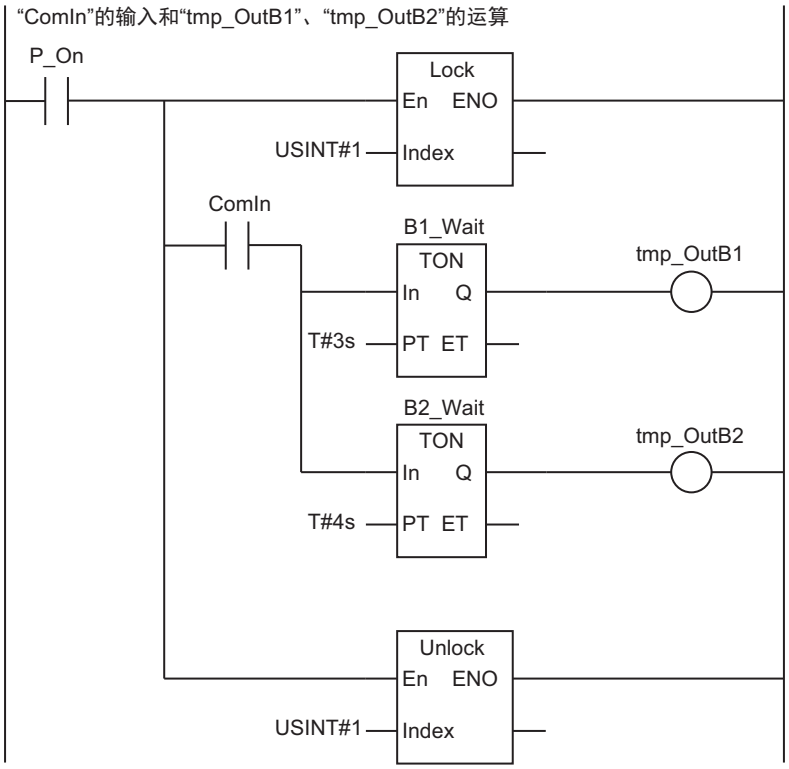
内部变量表

名称	数据类型
B1_Wait	TON
B2_Wait	TON

外部变量表

名称	数据类型	注释
ComIn	BOOL	共通输入值
tmp_OutB1	BOOL	B1 的临时变量
tmp_OutB2	BOOL	B2 的临时变量

固定周期任务的算法部分



5-6-2 将程序分配到任务中

将要执行的程序分配到任务中 （每个任务最多 128 个）。同时，设定各程序的运行开始时动作。

程序的执行顺序

在 Sysmac Studio 中设定任务内程序的执行顺序。

程序的运行开始时初始状态

设定各程序的运行开始时动作。运行开始时初始状态是指控制器的动作模式从程序模式变更为运行模式后，首先执行分配有该程序的任务时，是否执行该程序。设定范围为 “ 启动 ”、 “ 停止 ” 之一。
初始状态为 “ 停止 ” 时，若通过 PrgStart 指令发出执行指示，将在下次执行该程序的时序起执行。初始状态为 “ 启动 ” 时，若通过 PrgStop 指令发出停止指示，将在下次执行该程序的时序起停止。

● Sysmac Studio 上的设定操作

在 Sysmac Studio 的 [配置和设定] ｜ [任务设定] ｜ [程序的分配设定] 中，对向任务分配程序、任务中程序的执行顺序及各程序的初始状态进行设定。详情请参考「程序的分配设定」（P.4-9）。

版本相关信息

Sysmac Studio 的程序运行开始时初始状态设定及 PrgStart 指令、PrgStop 指令可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。

任务中可分配的 POU

任务中可分配 0 ～ 128 个程序。
可分配的 POU 仅限程序。FB 实例及 FUN 不可直接分配到程序中。同一个程序不可分配到多个任务中。

5-6-3 任务的设定项目

各任务的设定项目如下所示。

● 主固定周期任务的设定项目

设定项目		设定范围	默认	反映时序	运行模式中改写
任务类型		选择 “ 主固定周期任务 ”。	—	设定传送时	不可
	执行优先度	固定为 “4”	—		
任务名称		字符串	PrimaryTask		
周期 / 执行条件	任务周期*1	• NX 系列 CPU 单元时 125μs、 250μs ～ 8ms （以 250μs 为单位） • NJ 系列 CPU 单元时 500μs*2、 1ms、 2ms、 4ms	1ms*3		
任务周期超限检测		设定在超过有任务执行时间设定的任务周期时，是否检测错误。 • 检测 （发生轻度故障等级的控制器异常） • 不检测 （作为监视信息记录到事件日志中） 详情请参考📖「 任务周期超限 」（P.5-79）。	检测		
任务执行超时时间		设定在无限循环等任务不会完成处理时的超时检测时间。 设定为任务周期的倍数。 1 倍～ 5 倍 详情请参考📖「 任务执行超时 」（P.5-80）。	5 倍		
变量访问时间 ［%］		设定变量访问时间与任务周期的比例。 1% ～ 50% 📖「 变量访问时间设定 」（P.5-77）。	3%		

*1. EtherCAT 设定的过程数据通信周期 （过程数据通信周期 1）与本周期的值相同。
*2. NJ301- □□□□中，单元版本为 1.03 以上的环境下可以使用。
NJ101- □□□□不适用。
*3. NJ101- □□□□的主固定周期任务的默认值为 2ms。

● 固定周期任务 （执行优先度 5）的设定项目

设定项目		设定范围	默认	反映时序	运行模式中改写
任务类型		选择 “ 固定周期任务 （优先度 5） ”。*1	—	设定传送时	不可
	执行优先度	将自动选择为 “5”。	—		
任务名称		字符串	Periodic Task0		
周期 / 执行条件	任务周期*2	• NX 系列 CPU 单元时 125μs、 250μs ～ 100ms （以 250μs 为单位）	2ms		
任务周期超限检测		与主固定周期任务相同。	与主固定周期任务相同。		
任务执行超时时间					
变量访问时间 ［%］					

*1. 可使用固定周期任务 （执行优先度 5）的单元仅限 NX 系列 CPU 单元。
*2. EtherCAT 设定的过程数据通信周期 2 与本周期的值相同。

● 固定周期任务 （执行优先度 16、 17、 18）的设定项目

设定项目		设定范围	默认	反映时序	运行模式中改写
任务类型		选择以下任意一个。 “ 固定周期任务 （优先度 16） ” “ 固定周期任务 （优先度 17） ” “ 固定周期任务 （优先度 18） ”	—	设定传送时	不可
	执行优先度	将自动选择为 “16”“17”“18” 中的一个。	—		
任务名称		字符串	Periodic Task0		
周期 / 执行条件	任务周期	• NX 系列 CPU 单元时 参考 「5-3-1 NX 系列的任务规格」（P.5-9） • NJ 系列 CPU 单元时 参考 「5-4-1 NJ 系列的任务规格」（P.5-28）	10ms		
任务周期超限检测		与主固定周期任务相同。	与主固定周期任务相同。		
任务执行超时时间					
变量访问时间 ［%］					

● 事件任务的设定项目

设定项目	设定范围	默认	反映时序	运行模式中改写
任务类型	选择以下任意一个。 “ 事件任务 （优先度 8） ” “ 事件任务 （优先度 48） ”	—	设定传送时	不可
任务名称	字符串	Event Task0		
执行条件	选择 “ 通过指令执行 ”、“ 变量的条件式一致 ” 中的任意一个。	通过指令执行		
任务执行超时时间	设定在无限循环等任务不会完成处理时的超时检测时间。设定单位为 1ms。 • 执行优先度 “8” 时：1 ～ 500ms • 执行优先度 “48” 时：1ms ～ 10s	• 执行优先度 “8” 时：200ms • 执行优先度 “48” 时：1s		

● Sysmac Studio 上的设定操作

在 Sysmac Studio 的 [配置和设定] ｜ [任务设定] 中，进行任务的追加和各任务的设定。

详情请参考📖「任务设置」（P.4-6）。

5-7 确保变量的同时性

下面介绍在任务之间确保变量值同时性的方法和从控制器外部进行变量访问。

5-7-1 确保任务间变量值同时性的方法

通过多个不同的任务对同一个全局变量进行读写时，确保任务间全局变量值同时性的方法有以下两种。它们统称为 “ 变量的任务间排他控制功能 ”。

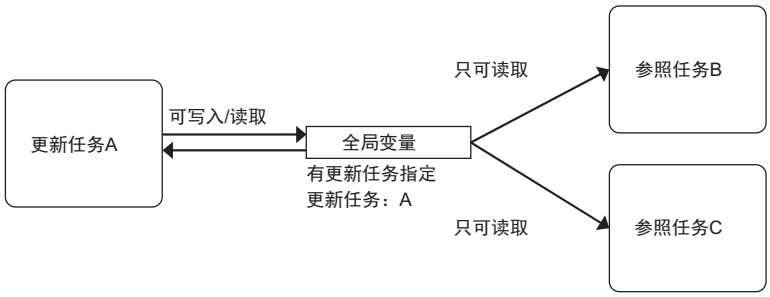
方法 1：通过 1 个任务写入到全局变量中，通过其他的多个任务读取时
使用变量的任务间排他控制设定功能。
方法 2：想要通过多个任务写入到全局变量时
使用任务间排他控制指令。

方法 1）变量的任务间排他控制设定功能

● **概要**

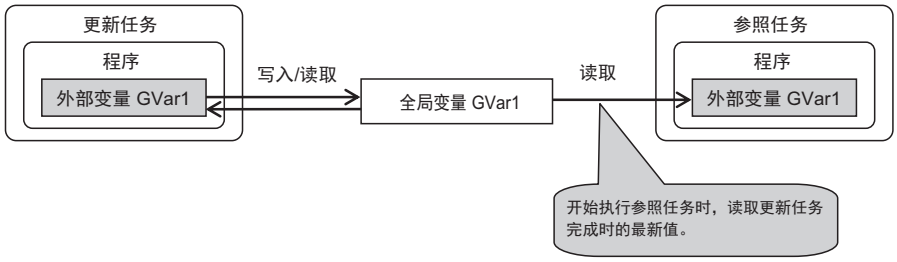
指定更新和参照全局变量的任务，确保对于参照侧任务而言全局变量值的同时性。

指定 1 个对全局变量进行读写的任务（更新任务）。针对该全局变量，指定只进行读取的任务（参照任务）。由此，可确保全局变量值的同时性。



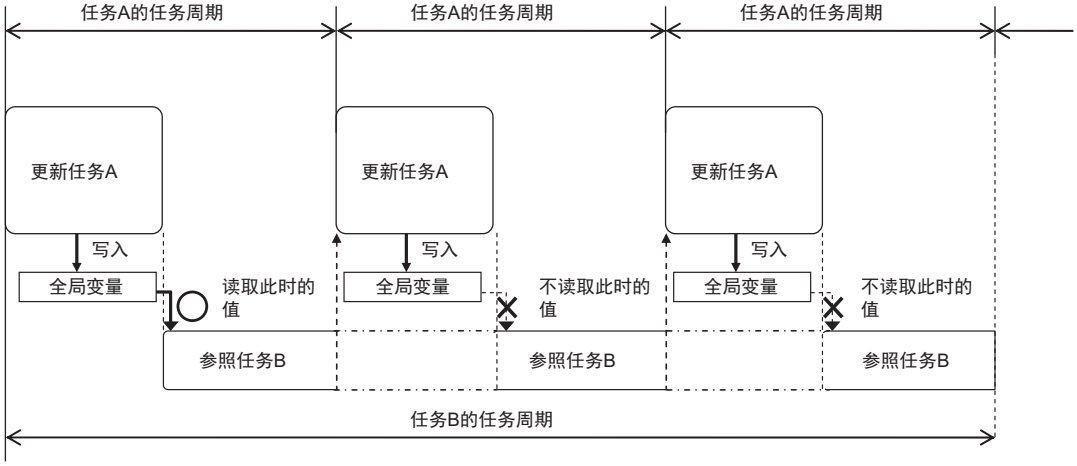
● **用途示例**

存在通过主固定周期任务写入的全局变量，在固定周期任务上读取该变量的值时，要在该固定周期任务内确保变量值的同时性时使用。



● **原理**

对于设定了更新任务的全局变量，待更新任务完成后，参照任务在自己的任务开始执行时，将其写入的值读取出来。

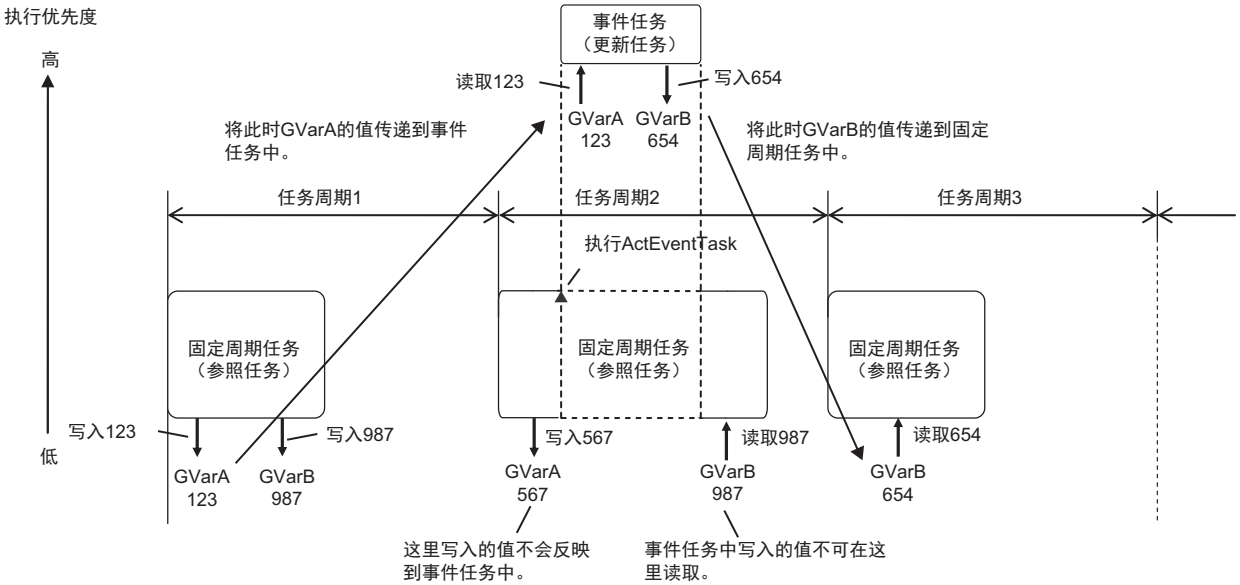


因此，无需特别的用户程序，即可确保任务内全局变量值的同时性。

若使用通过参照任务将值写入全局变量的指令， Sysmac Studio 的程序检查时将发生错误。

使用注意事项

使用 ActEventTask 指令时，请注意在两个任务之间，读取全局变量何时的值、何时更新值。
例如，下图情况下，事件任务读取的全局变量 GVarA 的值为任务周期 1 结束时的值。因此，即使在任务周期 2 中通过固定周期任务写入了 GVarA 的值，也不会反映到事件任务中。
此外，事件任务写入的全局变量GVarB的值传递到固定周期任务的时间为任务周期3的开始时间。因此，即使在任务周期 2 中通过固定周期任务读取 GVarB 的值，也无法读取事件任务写入的值。



使用 ActEventTask 指令执行事件任务时，若要接收全局变量的值，请勿使用 “ 变量的任务间排他控制功能 ”。
使用 ActEventTask 指令时，为了确保全局变量的同时性，任务执行中判定指令 Task_IsActive 为有效。Task_IsActive 指令是指判定指定的任务是否正在执行或处于待机状态的指令。使用本指令执行事件任务的过程中，请勿从其他任务参照事件任务进行写入所需的参数。Task_IsActive 指令的详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

● 限制

- 对1个全局变量只设定1个更新任务。若要通过多个任务写入同一全局变量，请使用后述的任务间排他控制指令。
- 在结构体 / 联合体变量中设定更新任务时，应对结构体 / 联合体变量整体设定 1 个更新任务。不可对每个结构体 / 联合体成员设定更新任务。
- 在排列变量中设定更新任务时，应对排列变量整体设定 1 个更新任务。不可对每个排列要素设定更新任务。

使用注意事项

对于需要确保值同时性的全局变量，请勿用参照任务、更新任务以外的任务写入值。对于设定了更新任务的全局变量，若用非更新任务、非参照任务的用户程序写入，将无法保证该全局变量值的同时性。若创建这样的程序，程序检查时会出现警告。

参考

- 使用变量的任务间排他控制功能的全局变量中，可将表示该全局变量的外部变量设定为数据追踪的采样对象。由此，可通过数据追踪，分别对更新任务和参照任务的全局变量值进行采样。数据追踪请参考「8-5-4 数据追踪」（P.8-41）。

● Sysmac Studio 上的设定操作

在 Sysmac Studio 的 [任务设定] ｜ [变量的任务间排他控制设定] 中，设定用于指定更新任务的全局变量 （“ 要更新的变量 ”）以及与之对应的参照任务 （“ 要参照的任务 ”）。

详情请参考「变量的任务间排他控制设定」（P.4-10）。

方法 2）任务间排他控制指令

通过多个任务写入同一全局变量时，应使用任务间排他控制指令（Lock/Unlock 指令），确保全局变量的同时性。

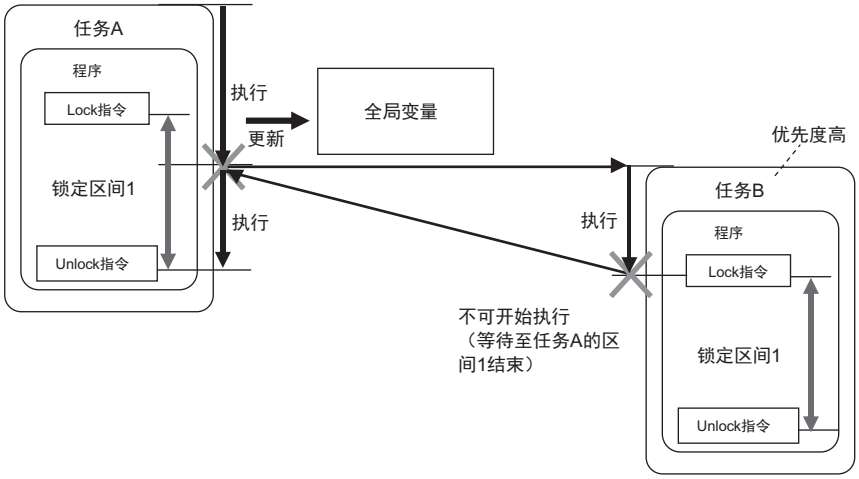
任务间排他控制指令中，正在执行从 Lock 指令到 Unlock 指令之间的区间 (锁定区间) 时，其他任务中相同锁定编号的锁定区间将无法执行。将写入到全局变量的操作设定在锁定区间内，通过多个任务写入全局变量时，也可确保值的同时性。

详情请参考《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》的 Lock （任务间排他锁定）指令、 Unlock （解除任务间排他锁定）指令。

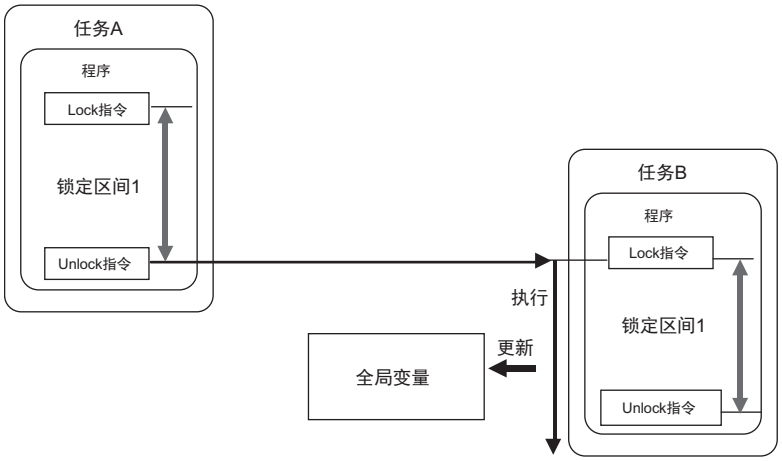
例）

假设任务 A、任务 B 都是拥有锁定区间 1 的任务，且任务 B 的优先度高于任务 A。

正在执行任务 A 的锁定区间 1 时，若任务 B 的执行条件成立，则任务 A 将在锁定区间 1 的中途暂停，执行任务 B。但是，有任务 A 中的锁定区间 1 尚未结束，因此任务 B 会在锁定区间 1 的处理前暂停。任务 B 暂停后，将重新开始任务 A 的锁定区间 1。



任务 A 的锁定区间结束后，任务 A 将再次暂停，执行任务 B 的锁定区间 1 之后的内容。在任务间独占全局变量的写入处理，确保值的同时性。



使用注意事项

- 锁定区间请设为最小需要长度。若锁定区间过长，可能超出任务的执行周期。
- Lock 指令和 Unlock 指令请务必在同一 POU 的同一段落中成对使用。

在任务间访问相同的全局变量时的示例

下面介绍使用多个全局变量向其他任务发出处理请求的情况。
在以下示例程序中，将一个全局变量 “gReq” 作为处理的排他标志，通过用户程序进行排他控制。此时，对于访问任务间使用的多个全局变量 “gPar1”、“gPar2”、“gReq” 的逻辑，需要用 “ 变量的任务间排他控制功能 ” 进行排他控制。

下面表示使用任务间排他控制指令 （Lock/Unlock 指令）进行排他控制的示例程序。

● 全局变量

名称	数据类型	注释
gReq	BOOL	请求标志
gPar1	ULINT	参数 1
gPar2	DATA_AND_TIME	参数 2

● 请求处理的任务 (MainTask)

- 内部变量

名称	数据类型	注释
ReqTrg	BOOL	请求触发
cCnt	ULINT	100ms 计数值
cTime	DATA_AND_TIME	当前时刻

- ST 程序

```
cCnt:=Get100msCnt();                    (*100ms 获取计数器值 *)
cTime:=GetTime();                       (* 获取当前时刻 *)

Lock(1);                                (* 任务间独占锁定开始 *)
IF ReqTrg=TRUE AND gReq=FALSE THEN      (* 独占标志的参照 *)
  gPar1:=cCnt;                          (*SubTask 中处理的参数组 *)
  gPar2:=cTime;                         (*SubTask 中处理的参数组 *)
  gReq:=TRUE;
  ReqTrg:=FALSE;
END_IF;
Unlock(1);                              (* 解除任务间独占锁定 *)
```

● 请求处理的任务 (SubTask)

- 内部变量

名称	数据类型	注释
ReqBusy	BOOL	—
UserDefFB_ins	UserDefFB	执行处理的用户定义 FB 的实例

- ST 程序

```
Lock(1);                                (* 任务间独占锁定开始 *)
IF gReq=TRUE AND ReqBusy=FALSE THEN     (* 独占标志的参照 *)
  ReqBusy:=TRUE;
  UserDefFB_ins.PutData:=gPar1;         (* 通过 MainTask 读取参数 *)
  UserDefFB_ins.PutDate:=gPar2;         (* 通过 MainTask 读取参数 *)
  gReq:=FALSE;                          (* 独占标志的重置 *)

  UserDefFB_ins.Execute:=TRUE;
END_IF;
Unlock(1);                              (* 解除任务间独占锁定 *)

UserDefFB_ins();

IF UserDefFB_ins.Done:=TRUE THEN
  UserDefFB_ins.Execute:=FALSE;
  ReqBusy:=FALSE;
END_IF;
```

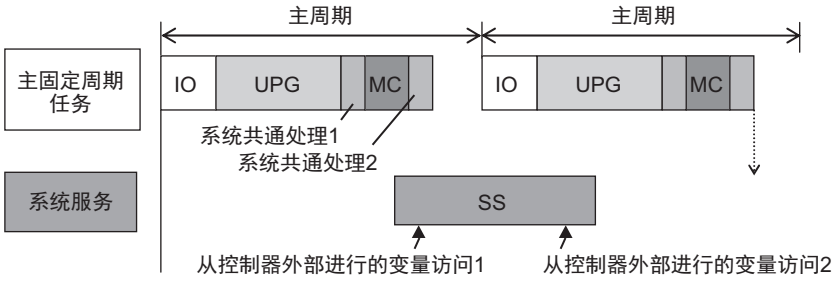
5-7-2 从控制器外部进行的变量访问

从控制器外部进行的变量访问通过系统服务执行。系统服务的执行优先度比任务低。因此，从控制器外部访问多个变量时，各变量值的更新可能不在一个任务周期内执行。用户程序中，若同时存在值发生更新的变量和未更新的变量，控制器可能发生意外动作。
为防止这样的情况，应在任务的 “ 系统共通处理 2” 中执行控制器外部的变量访问，这样可以确保在同一任务周期内执行多个变量值的更新。

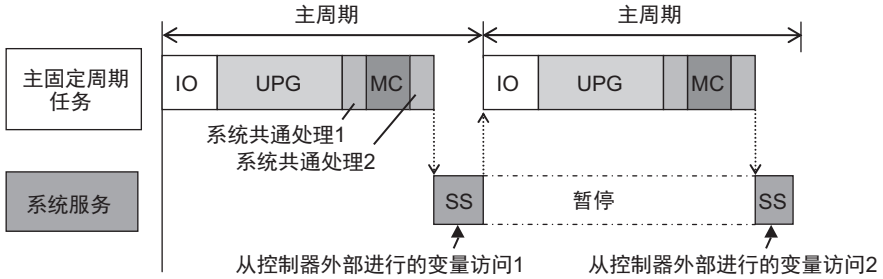
● 从控制器外部进行的变量访问通过系统服务执行时

无论是 NX 系列 CPU 单元还是 NJ 系列 CPU 单元，多个变量的访问可能不在同一任务周期内执行。

●NN系列CPU单元时

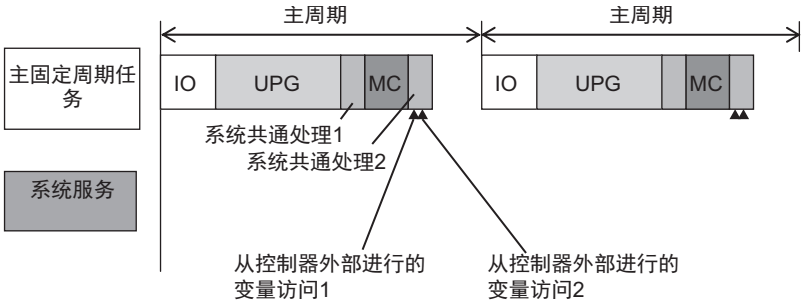


●NJ系列CPU单元时



● 从控制器外部进行的变量访问通过系统共通处理 2 执行时

无论是 NX 系列 CPU 单元还是 NJ 系列 CPU 单元，都在同一任务周期内执行多个变量的访问。



下面介绍通过 “ 系统共通处理 2” 执行从控制器外部进行的变量访问的方法。

从控制器外部访问变量的方法

从控制器外部访问变量的方法有以下 4 种。

- Sysmac Studio
- NS/NA 系列显示器
- EtherNet/IP 的标签数据链接
- 上位计算机等发出的 CIP 通信指令

但是，变量访问方法为 Sysmac Studio 时，任务的 “ 系统共通处理 2” 中可执行的处理仅为值的更新处理。值的参照处理在系统服务中执行。

通过系统共通处理 2 执行变量访问的任务

从控制器外部进行的变量访问通过 “ 系统共通处理 2” 执行时，变量的种类如下。

变量	更新值的任务
变量的任务间排他控制设定中指定的全局变量	Sysmac Studio 的 [配置和设定] ｜ [任务设定] ｜ [变量的任务间排他控制设定] 中指定的更新任务
EtherCAT 从站用设备变量	Sysmac Studio 的 [配置和设定] ｜ [任务设定] ｜ “ 设定控制 I/O 的任务 ” 中指定的任务
CJ 基本 I/O 单元用设备变量	
CJ 高功能单元用设备变量	主固定周期任务
在 CJ 单元用存储器中 AT （分配目标）指定的变量	

（注）可使用 CJ 单元及 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

通过系统共通处理 2 执行变量访问所需的设定

要通过任务的 “ 系统共通处理 2” 执行从控制器外部进行的变量访问，需要在 Sysmac Studio 上进行以下两个设定。

- 变量的任务间排他控制设定 （对象变量为全局变量时）
- 变量访问时间设定

变量的任务间排他控制设定

从控制器外部进行的变量访问通过系统共通处理 2 执行时，若对象变量为全局变量，需要进行变量的任务间排他控制设定。变量的任务间排他控制功能是指：指定更新对象全局变量的任务，其他任务或来自控制器外部的访问将无法更新对象全局变量的值。

变量的任务间排他控制功能详情请参考📖「变量的任务间排他控制设定」（P.4-10）。

使用注意事项

使用 EtherNet/IP 的标签数据链接时，对于同一标签集中的标签 （网络公开属性的变量），请务必将同一任务作为更新任务。否则，标签集内多个标签的更新时序可能跨多个不同的任务周期。

变量访问时间设定

若通过任务的 “ 系统共通处理 2” 执行从控制器外部进行的变量访问，任务执行时间可能会延长。用户需要在 Sysmac Studio 上设定变量访问时间的上限值。变量访问处理时间的上限值称为 “ 变量访问时间 ”。

● 变量访问时间的计算方法

变量访问时间按以下公式计算。

变量访问时间 [μs]= 变量的总大小 [字节]*a + 变量的数量 *b + 访问数量 *c + d

以上公式中常数 a ～ d 的值因 CPU 单元的型号不同而异，变化如下。

CPU 单元的型号	常数的值 [μs]			
	a	b	c	d
NX701- □□□□	0.0005	0.033	2.67	7.22
NJ501- □□□□	0.001	0.49*1	1.41	6.68
NJ301- □□□□	0.0015*2	0.56*3	2.15	7.52
NJ101- □□□□	0.0015	1.07	3.83	10.29

*1. CPU 单元版本低于 1.02 时为 “0.58”。
*2. CPU 单元版本低于 1.02 时为 “0.0009”。
*3. CPU 单元版本低于 1.02 时为 “1.03”。

● 变量访问时间的设定方法

变量访问时间在 Sysmac Studio 的 [配置和设定] ｜ [任务设定] ｜ [任务设定] 画面中设定。设定分任务进行，输入相对于任务周期的比例。默认值为 3%。设定方法的详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

● 变量访问时间的设定示例

变量访问时间的设定示例如下所示。
假设针对 CPU 单元 NJ501- □□□□中动作的任务，有以下 3 种来自外部控制器的变量访问。

访问 No.	变量访问源	访问变量的总大小 [字节]	访问变量的数量	访问数量
1	EtherNet/IP 的标签数据链接	600	8	1
2	EtherNet/IP 的标签数据链接	200	4	1
3	CIP 通信指令	1,000	1	1

访问 No. 为 1 的变量访问时间根据以下公式计算。
访问 No. 为 1 的变量访问时间 =600*0.001+8*0.49+1*1.41+6.68
=12.61[μs]

同样，各访问 No. 的变量访问时间为以下值。

访问 No.	变量访问时间 [μs]
1	12.61
2	10.25
3	9.58

1 个任务周期中，若只发生这些访问中的一个，将以时间最长的访问 No.1 为准，设定变量访问时间。变量 No.1 的变量访问时间为 12.61μs，所以假设任务周期为 500μs，变量访问时间应设定为 12.61/500 ≒ 3%。

1 个任务周期中，若这些访问逐个发生，则变量访问时间根据以下公式计算。
变量访问时间 =(600+200+1000)*0.001+(8+4+1)*0.49+(1+1+1)*1.41+6.68
=19.08[μs]

假设任务周期为 500μs，则变量访问时间设定为 19.08/500 ≒ 4%。

● 实际的访问时间大于变量访问时间的设定值时的处理

实际的访问时间大于变量访问时间的设定值时，根据 1 个周期中的变量访问次数，执行以下处理。请设定为有足够余量的变量访问时间，确保多个变量的访问在同一任务周期内执行。

1 个任务周期中的变量访问次数	处理
多次	执行变量访问时间的设定值以内的变量访问。未能执行的变量访问将在下一个任务周期内执行。 因此，多个变量的访问无法在一个任务周期内执行。
1 次	超过变量访问时间的设定值执行变量访问。因此，任务执行时间会变长。

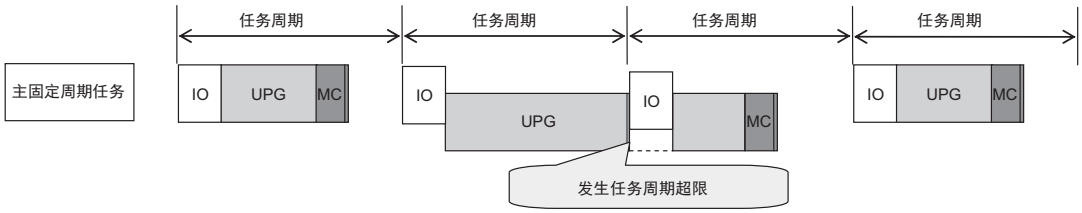
5-8 任务相关的异常

下面介绍以下异常。
- 任务周期超限
- 运动控制周期超限
- 任务执行超时
- I/O 刷新周期超限
- 系统服务时间不足

任务周期超限

任务的执行时间超过设定的任务周期时，发生 “ 任务周期超限 ”。
轻度故障等级的控制器异常。发生错误后仍继续运行。
对象为主固定周期任务、固定周期任务。

此外，可通过设定而不发生任务周期超限。在 Sysmac Studio 的 “ 任务设定 ” — “ 任务设定 ” 的 “ 任务周期超限检测 ” 中设定。默认为 “ 检测 ”。



异常名称	异常等级	处理
任务周期超限	轻度故障	修改任务设定及程序，重新下载。请通过 Sysmac Studio 解除异常。

无论 “ 任务周期超限检测 ” 设定如何，任务的处理未能在周期内完成时，将向系统定义变量 “_< 任务名称 >_Exceeded”（任务周期超过标志）、“_< 任务名称 >_ExceedCount”（任务周期超过次数）、“_ErrSta”（控制器异常状态）及事件日志输出信息。
任务处理未在周期内完成时的 I/O 刷新动作因输入输出对象的不同有以下差异。

输入输出的对象	任务处理未在周期内完成时的 I/O 刷新动作
EtherCAT 从站*1	输出：输出上一周期的值。 输入：执行 I/O 刷新，但输入数据不反映到正在执行的用户程序中。
CJ 单元	完成任务前，不会执行 I/O 刷新。

*1. 包括 EtherCAT 从站终端上的 NX 单元。

使用注意事项

发生 “ 任务周期超限 ” 时，请在创建程序时确保在任务周期内，或变更任务周期。

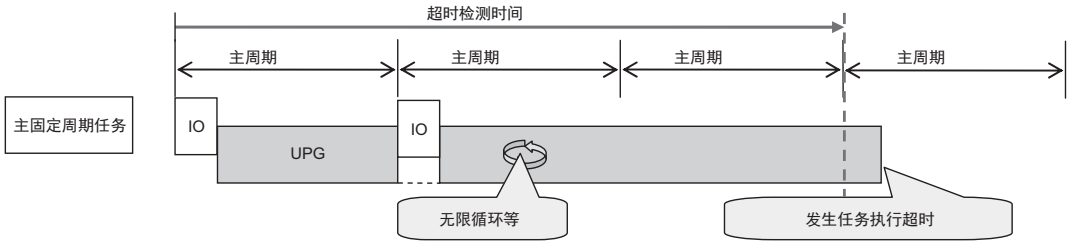
运动控制周期超限

“ 运动控制 （MC） ” 处理连续未能以主周期 （＝运动控制周期）完成时发生。运动控制功能模块发生部分停止故障等级的控制器异常。此时，将同时发生 “ 任务周期超限 ”。

异常名称	异常等级	处理
运动控制周期超限	部分停止故障	请减少程序的处理内容，或在动作不会发生问题的范围内，将控制周期设定为更大的值。

任务执行超时

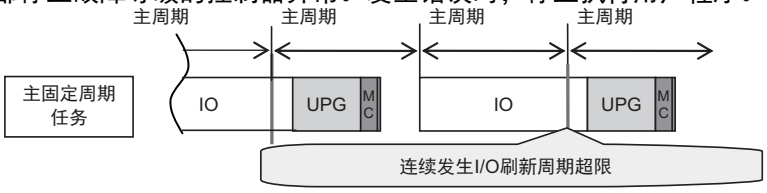
经过指定时间 （“ 超时检测时间 ”）后任务仍未完成时，将发生 “ 任务执行超时 ”。
全部停止故障等级的控制器异常。发生错误时，停止执行用户程序。
用户程序中的无限循环等，因逻辑错误而导致任务无法正常动作时也会发生。



异常名称	异常等级	处理
任务执行超时	全部停止故障等级	修改任务设定，重新下载用户程序。 需要重新接通电源或进行复位操作。

I/O 刷新周期超限

主固定周期任务及固定周期任务 （执行优先度 5、 16）中，若连续出现任务周期内无法完成 I/O 刷新的状态，将发生 “I/O 刷新周期超限 ”。
全部停止故障等级的控制器异常。发生错误时，停止执行用户程序。



异常名称	异常等级	处理
I/O 刷新周期超限	全部停止故障等级	修改任务设定，重新下载。 需要重新接通电源或进行复位操作。

系统服务时间不足

NJ 系列 CPU 单元中，无法确保系统服务监视设定中设定的系统服务执行时间时，将发生“ 系统服务时间不足 ”。

全部停止故障等级的控制器异常。发生错误时，停止执行用户程序。

异常名称	异常等级	处理
系统服务时间不足	全部停止故障等级	修改任务设定及系统服务监视设定，重新下载。 需要重新接通电源或进行复位操作。

参考

NX 系列 CPU 单元中，系统服务的执行时间不会发生不足。因此，无系统服务监视设定。也不会发生系统服务时间不足的异常。

5-9 任务执行状态、任务执行时间的监视

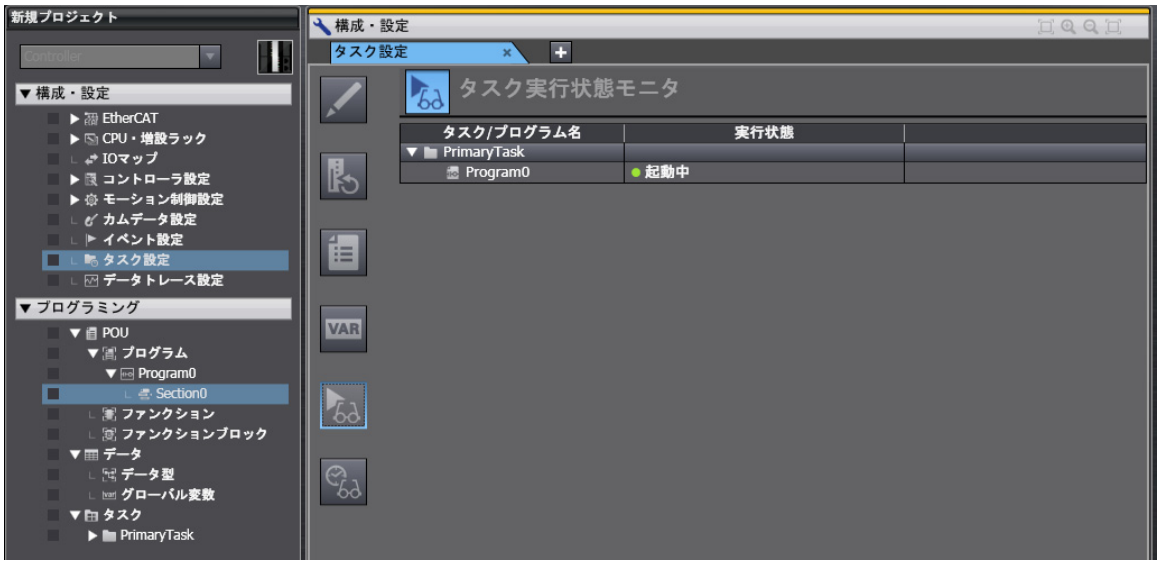
通过 Sysmac Studio 进行在线操作，可监视任务的执行状态、任务的执行时间。

任务执行状态的监视

可通过 Sysmac Studio 监视各任务中程序的执行状态 （启动中 / 停止中）。

● Sysmac Studio 上的操作

通过 Sysmac Studio 与 CPU 单元在线连接，选择 [配置和设定] ｜ [任务设定]。单击任务执行状态监视按钮，将显示以下画面。



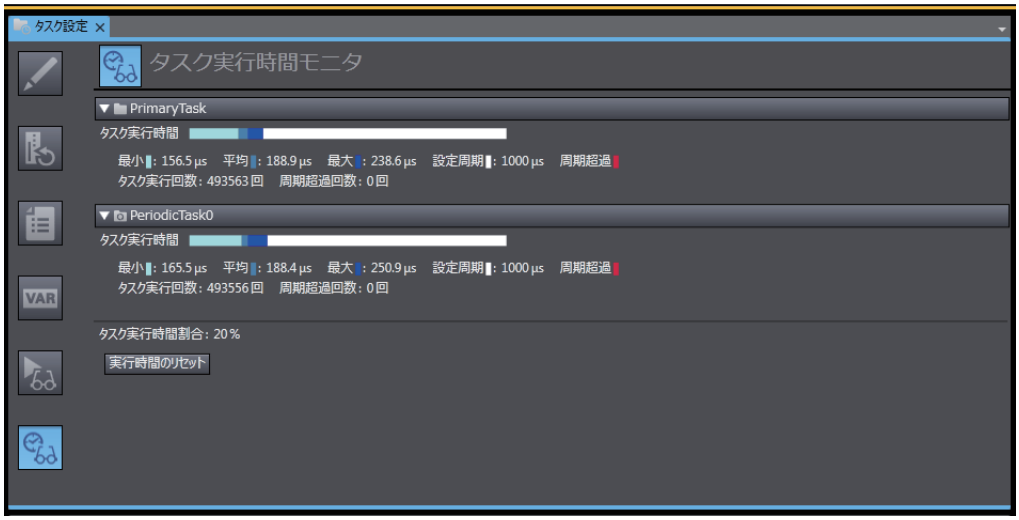
任务执行时间的监视

可通过 Sysmac Studio 监视各任务的执行时间。

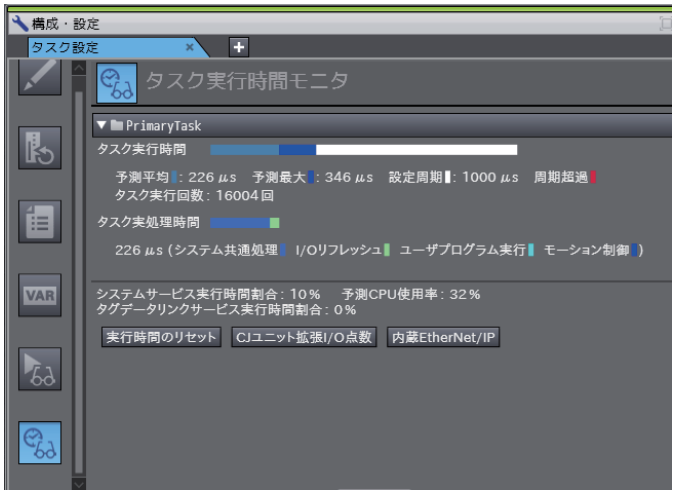
● Sysmac Studio 中可监视的值

连接实际控制器和连接模拟器时，画面有以下不同。

连接控制器时



连接模拟器时

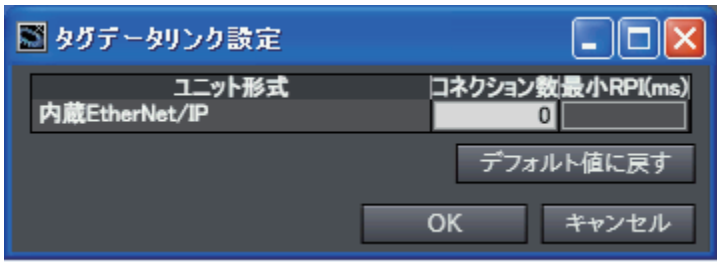


参考

NJ 系列 CPU 单元中，为了确认或预测系统服务执行时间，显示任务执行时间比例、系统服务执行时间比例、预测 CPU 使用率、标签数据链接范围执行时间比例、 CJ 单元扩展 I/O 点数、内置 EtherNet/IP。

NX 系列 CPU 单元的系统服务不受任务或标签数据链接服务的影响，始终执行，因此不显示上述内容。

连接模拟器时的内置 EtherNet/IP 画面



设定项目如下所示。

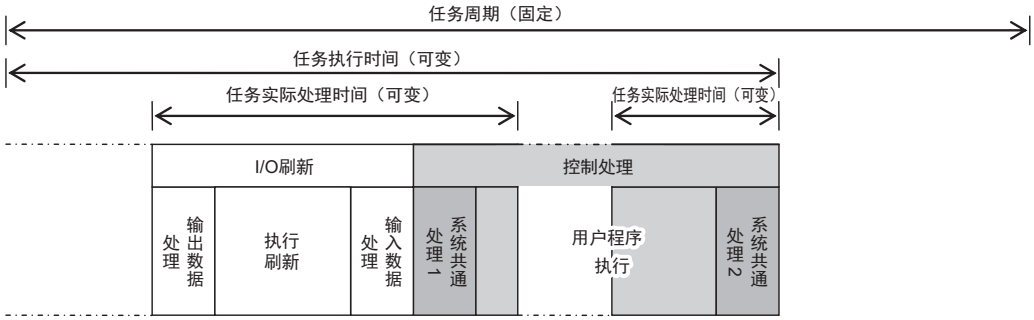
使用的端口 / 单元	设定项目	功能说明	设定值	默认
CPU 单元内置 EtherNet/IP 端口	连接数	标签数据链接的连接数。	0 ～ 32	0
	最小 RPI	标签数据链接的连接中，最小的数据包间隔 (RPI) 设定值。	1ms ～ 10s（最小单位 1ms）	—
CJ 系列 EtherNet/IP 单元 CJ1W-EIP21*1	标签数	标签数据链接的标签数。	0 ～ 256	0

*1. 在 “CJ 单元扩展 I/O 点数 ” 画面中输入。

可监视的项目如下所示。

监视项目		说明	连接控制器时	连接模拟器时
任务执行时间*1	最小	任务执行时间的最小值。	有	无
	平均	任务执行时间的平均值。		有
	最大	任务执行时间的最大值。		
设定周期		设定的任务周期。*2		
超出周期		任务执行时间超过任务周期时 （系统定义变量 " 任务周期超过标志 " 为 TRUE 时），在上面的栏中显示其程度。 *2		
任务执行次数		表示任务的执行次数。 系统定义变量 " 任务执行次数 " 的值。		
任务实际处理时间*3		以柱形图表示系统共通处理、 I/O 刷新、用户程序执行、运动控制处理的时间比例。（不输出时间 （数值）。）	无	有
	预测平均	任务实际处理时间的预测平均值。		
	预测最大	任务实际处理时间的预测最大值。		
任务执行时间比例*4		以小时为单位所有任务执行时间的比例。	有	无

*1. 从任务开始执行到完成为止，实际花费的时间。包括开始执行到完成期间执行的其他任务和系统服务的时间。
在 NX 系列 CPU 单元上连接模拟器时，仅显示主固定周期任务。
*2. 事件任务时不显示。
*3. 从任务的执行条件成立到执行完成为止，该任务本身花费的时间。
主固定周期任务时，与任务执行时间一致。
固定周期任务时，如下图所示，是指任务执行时间减去 （执行条件成立到执行完成为止执行的）其他任务和系统服务的时间。



*4. Sysmac Studio Ver.1.12 中追加的项目。 NX 系列 CPU 单元中不显示。

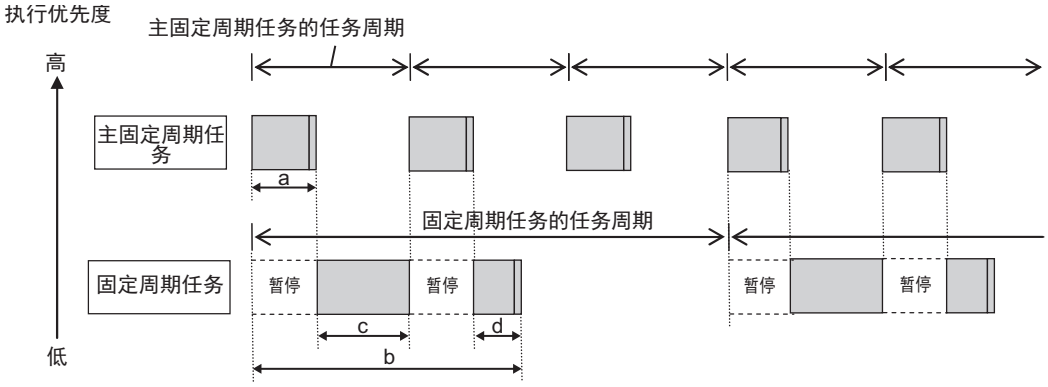
使用注意事项

连接模拟器时，Sysmac Studio 的上述值会与连接实际控制器时略有差异。因此，仅供参考。请务必连接实际控制器并确认，再进行设计研究和正式运行。

任务实际时间和任务实际处理时间的含义

任务的执行时间监视器中显示的“任务执行时间”、“任务执行处理时间”的含义如下所示。“任务实际处理时间”仅在连接模拟器时显示。

显示的时间	含义
任务执行时间	从任务周期开始到任务处理结束为止的时间。 但是，固定周期任务包含执行优先度更高的任务时的暂停时间。
任务实际处理时间	从任务周期开始到任务处理结束为止的时间。 但是，固定周期任务不含执行优先度更高的任务时的暂停时间。



a :主固定周期任务的任务执行时间
b :固定周期任务的任务执行时间
c+d:固定周期任务的任务实际处理时间（仅连接模拟器时）

5-10 任务设计方法及输入输出响应时间

下面介绍任务设计的指南、任务执行时间的预估方法、系统服务监视设定的确认、任务设计示例及输出输出响应时间。

NJ/NX 系列控制器中，主固定周期任务或固定周期任务按设定的周期动作。
实际的执行时间超过设定的周期时，执行各种错误处理。

下面以 1 个主固定周期任务 （单任务）构成时为例，介绍预估方法和评估方法。

安全要点

在实际的控制器上，因用户程序中搭载的逻辑是否动作、是否发生通信指令、数据链接动作、是否使用数据追踪等各种功能不同，任务执行时间会发生变动。
请务必使用实际机器在各种条件下确认性能，确认不会发生任务周期超限，确保良好的通信响应性。

5-10-1 任务执行时间的确认

请确保这里预估的任务执行时间平均值和最大值在设定的任务周期内。

● 上机计算时

参考「A-2 NX 系列中任务实际处理时间的计算标准」（P.A-13）及「A-3 NJ 系列中任务实际处理时间的计算标准」（P.A-24），上机估算 “ 任务执行时间的平均值 ”。
此外，上机时无法预估最大值。

● 用 Sysmac Studio 的模拟器计算时

利用 Sysmac Studio 模拟器的“任务执行时间监视”，确认任务执行时间的预测值（平均值及最大值）。在模拟器上进行动作确认的步骤如下所示。

1 **进行单元 / 从站构成的创建、全局变量 / 设备变量的登录、轴 （轴参数）的创建。**

2 **创建验证用程序。**

3 **进行任务设定并链接。**

4 **以 “ 执行时间预测模式 ” 启动模拟器。**

5 **在显示硬件修订版本的单元中，选择相应的硬件修订版本。**

6 **NJ 系列 CPU 单元中，通过［任务执行时间监视］的［CJ 单元扩展 I/O 点数设定］，用特定的 CJ 高功能单元创建用户定义变量，设定进行 AT （分配目标）指定的扩展区域 （例：DeviceNet 单元的固定分配 I/O 区域）的大小 （OUT 通道数、 IN 通道数）。**
这些大小用于在使用特定高功能单元时计算 I/O 刷新的时间。

7 **在 ［任务执行时间监视］中预估任务执行时间。**

以 “ 执行时间预测模式 ” 执行模拟器时，在 Sysmac Studio 的 “ 任务执行时间监视功能 ” 中，可以确认以下值。

- 任务执行时间 （预测平均值、预测最大值）
- 任务实际处理时间 （预测平均值、预测最大值）
- 表示系统共通处理、 I/O 刷新、用户程序执行时间、运动控制时间明细的柱形图
- CPU 使用率

（注）在 NX 系列 CPU 单元上，仅显示主固定周期任务。

参考

连接 Sysmac Studio 和模拟器时，与连接实际控制器时不同，可确认以下内容。

- “CPU 使用率 ”：
  表示 NJ 系列 CPU 单元的任务处理时间预测最大值、标签数据链接服务执行时间分配、系统服务监视设定中指定的应确保的系统服务处理时间的总和，相对于任务周期的比例。
  若 CPU 使用率超过 100%，表示相对于任务处理时间、系统服务监视设定值，未能确保足够的时间。
  NX 系列 CPU 单元中不显示。
- “ 任务实际处理时间 ”：
  “ 任务实际执行时间 ” 是指从任务开始执行到执行完成为止，该任务本身花费的时间。不含任务开始执行到完成期间执行的其他任务的时间。

● 在实际控制器上计算时

连接实际控制器时，在 Sysmac Studio 的 " 任务执行时间监视功能 " 中，可以确认以下值。

- 任务执行时间 （最小值、平均值、最大值）
- 设定周期
- 任务执行次数
- 周期超过次数
- 任务执行时间比例

Sysmac Studio 中显示的最大值为在实际控制器上动作时的最大值。
如上所述，任务执行时间因实际控制器的内部状态而变化。
所以，可能超出这里监视的最大值。请将监视值或以下计算结果中较大的值作为最大值的标准。

NX 系列 CPU 单元时

（[任务执行时间的平均值] ＋（[任务执行时间的平均值] － [任务执行时间的最小值]））×1.2 ＋ 25μs

NJ 系列 CPU 单元时

- 任务周期为 500μs 时：
  [任务执行时间的平均值] ＋ （[任务执行时间的平均值] － [任务执行时间的最小值]） ＋ 100μs
- 任务周期为 1ms、 2ms、 4ms 时：
  [任务执行时间的平均值] ＋ （[任务执行时间的平均值] － [任务执行时间的最小值]） +120μs

然后，如果是 NJ 系列 CPU 单元，应确认是否能确保足够的系统服务执行时间。
若未能确保足够的系统服务执行时间， Sysmac Studio 上的在线操作或与显示器等外部设备的通信响应性可能会降低。 NJ 系列 CPU 单元的系统服务在任务执行的空闲时间内执行。
请将以下值作为 [任务执行时间比例] 的标准。

CPU 单元的型号	任务执行时间比例的标准
NJ501- □□□□	90% 以下
NJ301- □□□□	80% 以下
NJ101- □□□□	70% 以下

版本相关信息

- [任务执行时间比例] 在 Ver.1.12 以上的 Sycmac Studio 中使用 NJ 系列 CPU 单元时显示。
- CPU 单元 NJ101- □□□□中请使用 Sysmac Studio Ver.1.13 或更高版本。 Sysmac Studio Ver.1.12 或更低版本中无法使用 NJ101- □□□□。

使用注意事项

NX 系列 CPU 单元中，系统服务的执行时间不会发生不足。
因此，不显示 [任务执行时间比例]。

参考

“ 任务执行时间监视功能 ” 中显示的任务执行时间平均值为 10 次任务执行时间的移动平均值。

5-10-2 任务设计示例

本节以 1 个主固定周期任务 （单任务）构成时为例，介绍设计步骤。
但是，因实际的应用和各种条件不同，设计步骤或必须考虑的要素会发生变化，所以仅供参考。

1 **根据装置规格计算必要系统的各输入输出响应时间。**

2 **根据系统的各输入输出响应时间，决定主固定周期任务的 “ 任务周期 ”。**

3 **首先确认是否在 “ 任务执行时间 ” 所确定的 “ 任务周期 ” 内。**
上机或使用 Sysmac Studio 的模拟器，预估任务执行时间的预计平均值或预计最大值。

4 **如果是 NJ 系列 CPU 单元，应确认系统服务时间是否在监视设定范围内。**
Sysmac Studio 的模拟器时，确认 CPU 使用率。

5 **在实际控制器上，确认任务执行时间是否在任务周期内。**
将 Sysmac Studio 与实际控制器在线连接，利用 “ 任务执行时间监视功能 ”，验证任务执行时间。

- **1 个主固定周期任务 （单任务）无法控制在设定的任务周期内时，请从以下角度出发，研究对固定周期任务的任务分配。**
  - NX 系列 CPU 单元时，在装置控制所需的处理中，将需要以最快速度控制的处理分配到主固定周期任务，将其他处理分配到固定周期任务 （执行优先度 5）中。
  - 为了缩短任务执行时间，应使用程序执行指令 （PrgStart）及程序停止指令 （PrgStop），仅在需要主固定周期任务中分配的程序时执行。
  - 实施上述处理后，仍超出主固定周期任务的任务周期时，将主固定周期任务的处理中，不需要高速、高精度处理的部分分配到固定周期任务 （执行优先度 16）中。
    但是，在主固定周期任务和固定周期任务 （执行优先度 16）之间，如果要参照轴参数等数据，主固定周期任务的任务执行时间可能会变长。详情请参考📖《NJ/NX 系列 CPU 单元 用户手册 运动控制篇 （SBCE-CN5-363）》。
  - 执行上述处理后，仍超出主固定周期任务的任务周期时，请修改主固定周期任务的任务周期。

- **按上述方法分配任务后，固定周期任务会因主固定周期任务的空闲时间而大幅变动。**
  固定周期任务时，将计算得到的任务执行时间预测平均值 / 最大值的 2 倍设定为固定周期任务的任务周期，然后进行调谐。

5-10-3 系统输入输出响应时间

系统中从输入到输出为止的时间如下所示。
根据不同情况而不同。
此外，实际情况下，这个时间包括从站 / 单元与外部的输入响应时间、与外部的输出响应时间。

对基本 I/O 单元进行顺序控制时

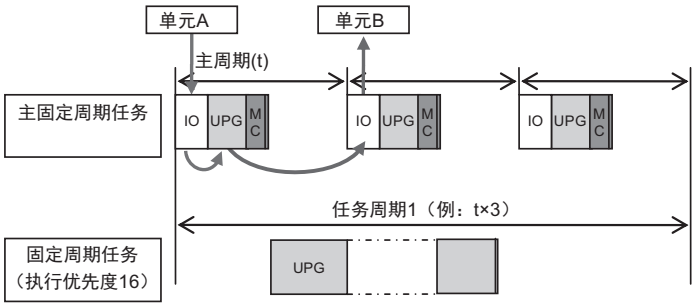
基本 I/O 单元通过分配了 I/O 刷新的任务与外部进行 I/O 刷新。
包含 EtherCAT 通信的输入输出响应时间如下所示。

● 用主固定周期任务的用户程序控制时

按以下输入输出响应时间响应。

最小输入输出响应时间＝“主周期”

例）用主固定周期任务控制单元 A、单元 B 时



注. 上图只记述1次输入输出。

但是，在某些单元的输入时序下，输入输出响应时间如下。

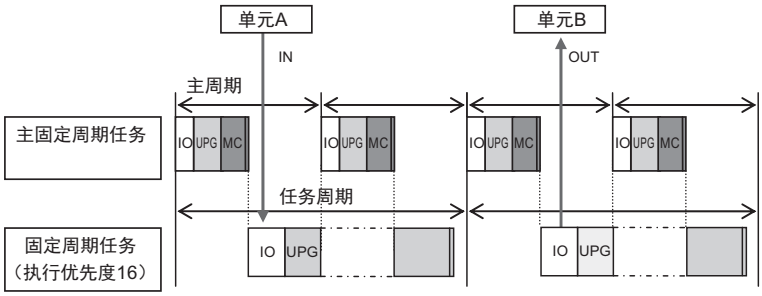
最小输入输出响应时间＝“主周期”×2

● 用固定周期任务 （执行优先度 16）的用户程序控制时

按以下输入输出响应时间响应。

最小输入输出响应时间＝“固定周期任务 （执行优先度 16）的周期”

例）用固定周期任务 （执行优先度 16）控制单元 A、单元 B 时



注. 上图只记述1次输入输出。

但是，在某些单元的输入时序下，输入输出响应时间如下。

最大输入输出响应时间＝“固定周期任务 （执行优先度 16）的周期”×2

对 EtherCAT 从站进行顺序控制时

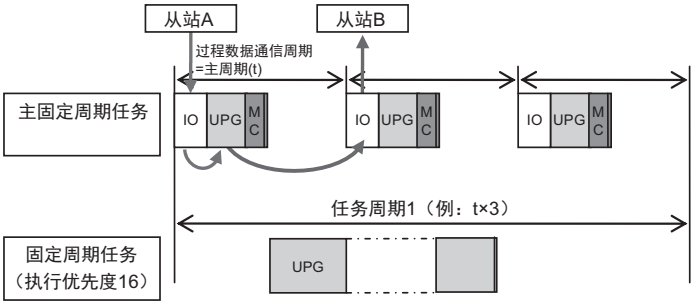
EtherCAT 从站通过主固定周期任务的 I/O 刷新与外部进行 EtherCAT 通信。
包含 EtherCAT 通信的输入输出响应时间如下所示。

● 用主固定周期任务的用户程序控制时

按以下输入输出响应时间响应。

最小输入输出响应时间＝“主周期（＝过程数据通信周期）”

例）用主固定周期任务控制 EtherCAT IN 从站 A、 EtherCAT OUT 从站 B 时



注. 上图只记述1次输入输出。

但是，在某些从站的输入时序下，输入输出响应时间如下。

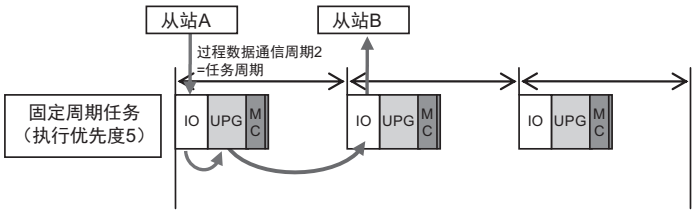
最大输入输出响应时间＝“主周期（＝过程数据通信周期）”×2

● 用固定周期任务（执行优先度 5）的用户程序控制时

用 NX 系列 CPU 单元的固定周期任务（执行优先度 5）的用户程序控制时，按以下输入输出响应时间响应。

最小输入输出响应时间＝“固定周期任务（执行优先度 5）的周期（＝过程数据通信周期 2）”

例）用固定周期任务（执行优先度 5）控制 EtherCAT IN 从站 A、 EtherCAT OUT 从站 B 时



注. 上图只记述1次输入输出。

但是，在某些从站的输入时序下，输入输出响应时间如下。

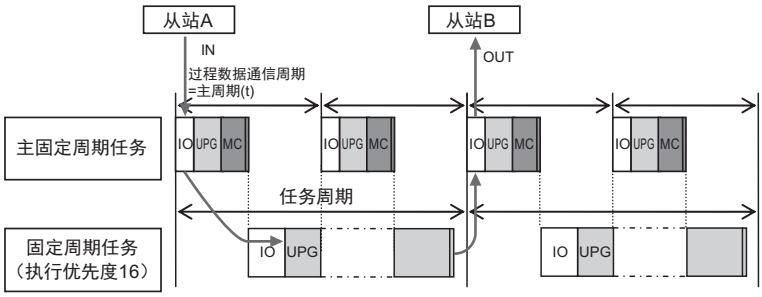
最大输入输出响应时间＝“固定周期任务（执行优先度 5）的周期（＝过程数据通信周期 2）”×2

● 用固定周期任务 （执行优先度 16）的用户程序控制时

按以下输入输出响应时间响应。

输入输出响应时间＝“ 固定周期任务 （执行优先度 16）的周期 ”

例） 用固定周期任务 （执行优先度 16）控制 EtherCAT IN 从站 A、 EtherCAT OUT 从站 B 时



注. 上图只记述1次输入输出。

但是，在某些从站的输入时序下，输入输出响应时间如下。

最大输入输出响应时间＝“ 固定周期任务 （执行优先度 16）的周期 ”×2

用运动控制指令执行运动控制时

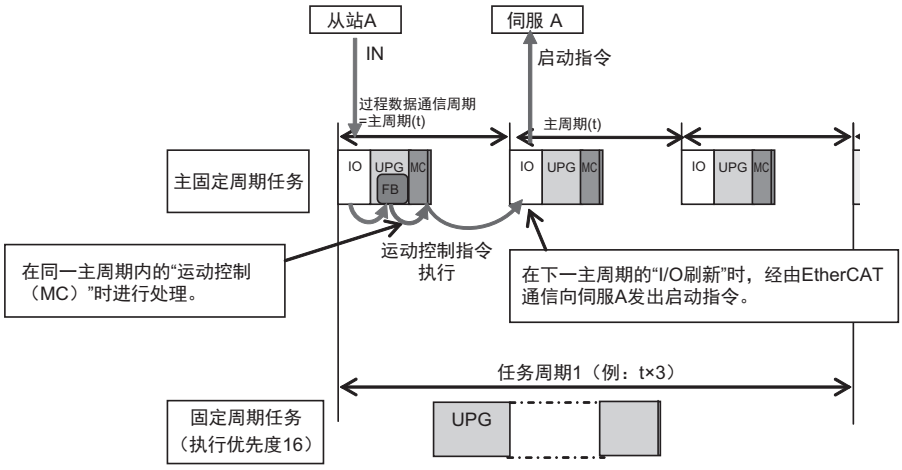
运动控制指令对已分配到 " 轴 " 的伺服 / 编码器输入从站进行访问。
NJ 系列 CPU 单元的运动控制指令可在主固定周期任务或固定周期任务 （执行优先度 16）中记述。
无论是哪个任务，运动控制指令均通过主固定周期任务中的 " 运动控制 （MC） " 执行。
NX 系列 CPU 单元的运动控制指令可在主固定周期任务或固定周期任务 （执行优先度 16）及固定周期任务 （执行优先度 5）中记述。
主固定周期任务或固定周期任务 （执行优先度 16）中记述的运动控制指令通过主固定周期任务中的 " 运动控制 （MC） " 执行。
固定周期任务 （执行优先度 5）中记述的运动控制指令通过固定周期任务 （执行优先度 5）中的 " 运动控制 （MC） " 执行。

包含 EtherCAT 通信的输入输出响应时间如下所示。

● 在主固定周期任务中记述了运动控制指令时

运动控制指令的处理通过主固定周期任务中紧接的 " 运动控制（MC）" 执行。处理结果在下一主固定周期任务的 "I/O 刷新 " 时，经由 EtherCAT 通信输出到 [轴] 中分配的伺服驱动器中。
按以下输入输出响应时间响应。

输入输出响应时间＝ " 主周期 （＝过程数据通信周期） "



注. 上图只记述1次输入输出。

但是，在某些从站的输入时序下，输入输出响应时间如下。

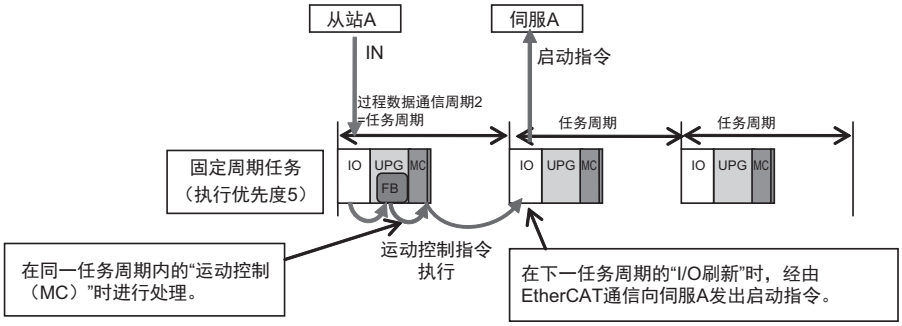
最大输入输出响应时间＝ " 主周期 （＝过程数据通信周期） "×2

● 在固定周期任务 （执行优先度 5）中记述了运动控制指令时

运动控制指令的处理通过固定周期任务 （执行优先度 5）中紧接的 “ 运动控制 （MC） ” 执行。处理结果在下一固定周期任务（执行优先度 5）的 “I/O 刷新 ” 时，经由 EtherCAT 通信输出到 [轴] 中分配的伺服驱动器中。

按以下输入输出响应时间响应。

输入输出响应时间＝ “ 固定周期任务 （执行优先度 5）的周期 （＝过程数据通信周期） ”



注. 上图只记述1次输入输出。

但是，在某些从站的输入时序下，输入输出响应时间如下。

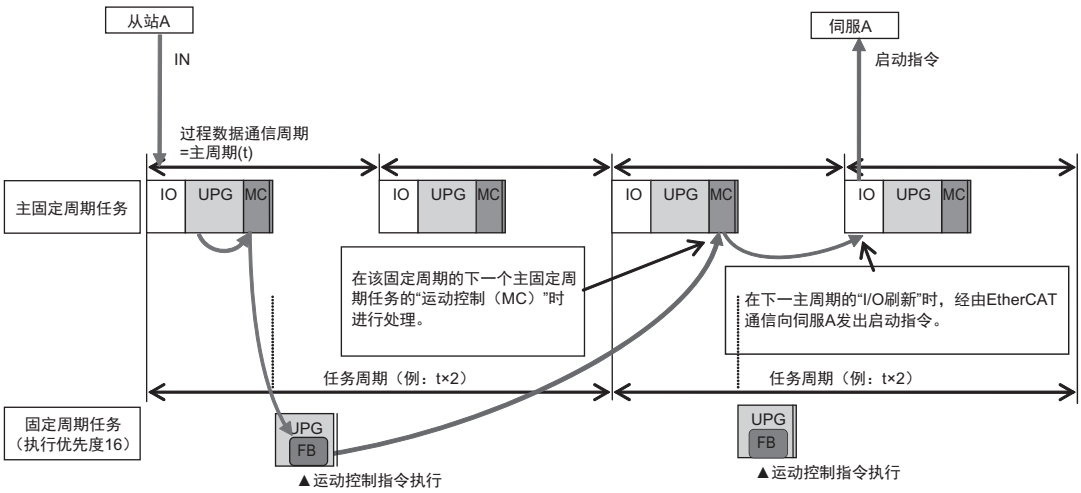
最大输入输出响应时间＝ “ 固定周期任务 （执行优先度 5）的周期 （＝过程数据通信周期 2） ”×2

● 在固定周期任务 （执行优先度 16）中记述了运动控制指令时

运动控制指令的处理通过固定周期任务 （执行优先度 16）的下一主固定周期任务中的 “ 运动控制 （MC） ” 执行。处理结果在下一主固定周期任务的 “I/O 刷新 ” 时，经由 EtherCAT 通信输出到 [轴] 中分配的伺服驱动器中。

因此，无论运动控制指令的执行时序如何，都按以下输入输出响应时间响应。

最小输入输出响应时间＝ “ 固定周期任务 （执行优先度 16）的周期 ” ＋ “ 主周期 （＝过程数据通信周期） ”



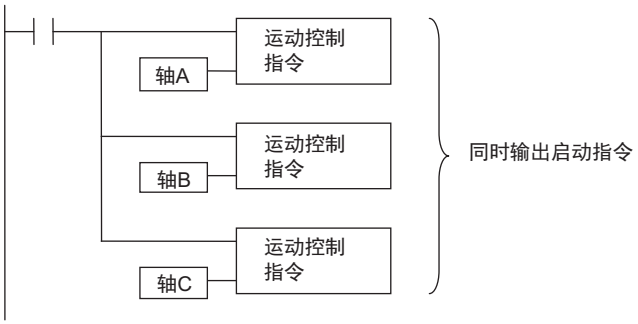
注. 上图只记述1次输入输出。

但是，在某些从站的输入时序下，响应时间如下。

最大输入输出响应时间＝ “ 固定周期任务 （执行优先度 16）的周期 ” ＋ “ 主周期 （＝过程数据通信周期） ”×2

● 多个轴的同时启动

用同一任务中的用户程序在同一任务周期内启动多个轴时，可以同时启动。



参考

在控制轴的任务以外的任务中，可以参照轴变量的值。
使用方法的详情和注意事项请参考《NJ/NX 系列 CPU 单元 用户手册 运动控制篇（SBCE-CN5-363）》。

6

编程

本章介绍程序的详情，如程序的种类、构成程序的变量、指令等。

6-1 编程步骤概要 6-3

6-2 POU （程序构成单位） 6-5

6-2-1 POU 定义 6-5
6-2-2 三个类型 POU 的概要 6-6
6-2-3 程序、功能块、函数的区别 6-7
6-2-4 程序的详情 6-8
6-2-5 功能块 （FB）的详情 6-9
6-2-6 函数 （FUN）的详情 6-18
6-2-7 功能块 （FB）、函数 （FUN）共通动作 6-24
6-2-8 POU 的限制事项 6-26

6-3 变量 6-28

6-3-1 变量定义 6-28
6-3-2 变量的种类 6-28
6-3-3 POU 对应的用户定义变量的种类 6-29
6-3-4 变量的属性 6-30
6-3-5 数据类型 6-31
6-3-6 派生数据型 6-41
6-3-7 数据类型的排列 / 范围指定 6-51
6-3-8 变量的各属性说明 6-57
6-3-9 与状态变化对应的变量值的变化 6-64
6-3-10 功能块 (FB) 的实例 6-76
6-3-11 变量值的监视 6-76
6-3-12 变量名称等程序相关名称的限制 6-77

6-4 常数 （文本） 6-79

6-4-1 常数定义 6-79
6-4-2 各数据类型的记述形式 6-79

6-5 编程语言 6-84

6-5-1 编程语言定义 6-84
6-5-2 梯形图语言 （LD） 6-84
6-5-3 ST 语言 6-90

6-6 指令 6-128

6-6-1 指令定义 6-128
6-6-2 指令的基本知识 6-128
6-6-3 指令错误 6-131

6-7 名称空间 6-137
6-7-1 名称空间定义 6-137
6-7-2 名称空间的规格 6-138
6-7-3 名称空间的使用步骤 6-141

6-8 库 6-142
6-8-1 库的概要 6-142
6-8-2 库的规格 6-143
6-8-3 库部件的规格 6-144
6-8-4 库的使用步骤 6-145

6-9 创建用户程序时的注意事项 6-146
6-9-1 输入变量 / 输出变量 / 输入输出变量为排列指定的变量时 6-146
6-9-2 输入变量 / 输出变量 / 输入输出变量为结构体型变量时 6-146
6-9-3 主控 6-147

6-1 编程步骤概要

下面介绍编程步骤的概要。
在整体步骤中，与以下阴影标示的位置对应。

STEP 1. 软件设计

- STEP1-1 设计 I/O 和处理
- STEP1-2 设计任务
- STEP1-3 设计程序

↓

STEP 2. 软件设定和编程

- STEP2-1 从站 / 单元构成
- STEP2-2 控制器设定
- **STEP2-3 编程**
- STEP2-4 离线调试

↓

STEP 3. 安装和硬件的设定

↓

STEP 4. 配线

↓

STEP 5. 上机确认动作和运行

详情请参考📖「1-3 NJ/NX 系列的整体使用步骤」（P.1-12）。

设计 POU （处理的分割单位）	参考
● **研究和设计用哪个 POU 执行各种处理** 注：函数中不可记述功能块型指令和功能块。	📖「6-2 POU （程序构成单位）」（P.6-5）
● **设计将哪个处理用梯形图、 ST、联机 ST 中的哪种语言创建** 注：“ 联机 ST” 为梯形图中描述 ST 语言的梯形图语言的要素	📖「6-5 编程语言」（P.6-84）

设计变量	参考
● **设计创建哪种用户定义变量**	📖「6-3-1 变量定义」（P.6-28） 📖「6-3-2 变量的种类」（P.6-28）
● **区分 POU 共同使用的变量 （全局变量）和各 POU 单独使用的变量 （本地变量）**	📖「6-3-3 POU 对应的用户定义变量的种类」（P.6-29）
● **设计是自动生成用于访问从站 / 单元的设备变量之变量名称还是由用户定义**	📖「3-3 I/O 端口和设备变量」（P.3-8）
● **设计变量的以下属性** 变量名称、数据类型、 AT （分配目标）指定、初始值、保持、常量、网络公开等 设定变量的数据类型 （排列指定、范围指定、结构体型、列举型）	📖「6-3-4 变量的属性」（P.6-30） 📖「6-3-5 数据类型」（P.6-31） 📖「6-3-6 派生数据型」（P.6-41）
● **设计变量时，请注意以下事项。** • 保持的设计： 接通电源时、变更动作模式时，根据变量值的设定，设计保持属性。 • 结构体型的设计： 在所用指令的变量中使用结构体数据类型时，设计为在输入、输出或输入输出参数中使用该结构体数据类型。 例）通信指令 • 排列指定的设计： 在所用指令的变量中使用排列指定的变量时，设计为在输入、输出或输入输出参数中使用排列指定的变量。 例）移位指令、堆叠 / 表格指令 • AT 指定的设计： • 对部分指令的输入参数变量进行 AT 指定。 例） DeviceNet 单元的固定分配或 I/O 的自由分配 • 网络公开的设计： 设计进行 EtherNet/IP 标签数据链接的变量。	📖「6-3-4 变量的属性」（P.6-30） 📖「6-3-5 数据类型」（P.6-31） 📖「6-3-6 派生数据型」（P.6-41）

6-2 POU （程序构成单位）

NJ/NX 系列 CPU 单元上动作的用户程序全部以 POU （程序构成单位）的形式创建，然后将它们组合起来使用。
下面详细介绍 POU 构成、规格等。
在 Sysmac Studio 上创建 POU 的方法请参考📖《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》。

6-2-1 POU 定义

POU （Program Organization Unit）是指 IEC 61131-3 中用户程序执行模式的单位。作为构成用户程序的基本要素，对本地变量表和算法 （一系列代码 / 逻辑）进行记述。
组合多个 POU，构成整个用户程序。

POU 的构成要素有以下三种。

- 程序
  相当于主例程的、记载有算法的主要要素。
  算法中可记述所有指令、函数、功能块。
- 功能块 （以下简称 “FB”）
  即使输入值相同，也可输出不同值的要素。通过程序或其他 FB 的调用来启动。
- 函数 （以下简称 “FUN”）
  如果输入值相同，即可始终输出相同值的要素。通过程序、 FB 或 FUN 的调用来启动。

POU 由以上三个要素组合构成。此外，还可创建多个 POU。
创建的用户程序将分配到任务后启动。

6-2-2 三个类型 POU 的概要

程序定义

● 启动方法 / 执行条件

- 通过启动任务，启动该任务中分配的程序。
- 始终执行。

● 记述

- POU 中必须记述 1 个以上的程序。 1 个任务的程序数最多为 128 个。

功能块 (FB) 定义

● 启动方法 / 执行条件

- 通过程序、 FB 的调用来启动。
- 始终执行。
- 若要只在条件成立时执行，请定义表示执行条件的输入变量。

● 记述

- 算法中可记述所有指令、用户定义 FUN 及用户定义 FB。
- 可保持内部变量的值。因此，如计时器、计数器等，拥有状态。
- 包括用户定义的功能块和系统定义的功能块。
  用户定义的 FB 称为 “ 用户定义 FB”。系统定义的 FB 称为 “FB 型指令 ”。

FB 的详情请参考📖「6-2-5 功能块 （FB）的详情」（P.6-9）。

函数 （FUN）定义

● 启动方法 / 执行条件

- 通过程序、 FB 或其他 FUN 的调用来启动。
- 执行条件用输入变量 （EN）指定。 EN 为 TURE 时执行一次。

● 记述

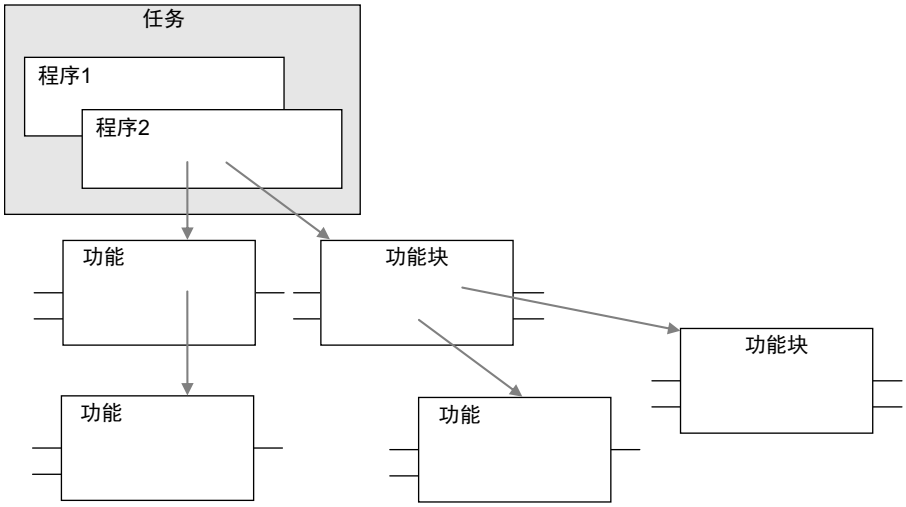
- 算法中不可记述 FB 型指令及用户定义 FB。
- 不可保持内部变量的值。因此，输入值相同时，输出值始终保持一致。
- 包括用户定义的功能块和系统定义的功能块。
  用户定义的 FUN 称为 “ 用户定义 FUN”。系统定义的 FUN 称为 “FUN 型指令 ”。

FUN 的详情请参考📖「6-2-6 函数 （FUN）的详情」（P.6-18）。

6-2-3 程序、功能块、函数的区别

项目 \ POU 种类		程序	功能块 (FB)	函数 （FUN）
启动方法		通过启动分配的任务来启动	程序、其他 FB 的调用	程序、 FB、 FUN 的调用
算法内	所有指令	可	可	不可
	用户定义 FUN	可	可	可
	用户定义 FB	可	可	不可
执行条件		始终执行	始终执行 执行条件用输入变量指定	通过 EN 输入指定

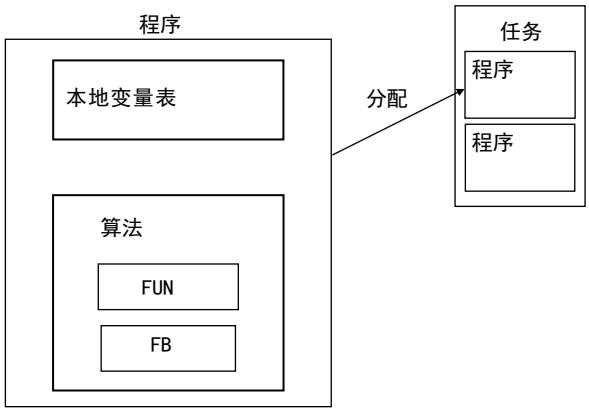
程序、 FB、 FUN 的层级关系如下图所示。



6-2-4 程序的详情

程序的构成

程序由本地变量表和算法部分构成。
算法部分中可记述 FUN 或 FB。



程序不会通过其他 POU 调用。

程序的启动条件

通过启动分配了程序的任务来启动程序。

● 执行顺序

对于任务中分配的程序，可设定在任务中的执行顺序。在 Sysmac Studio 的 [任务设定] ｜ [程序的分配设定] 中指定。

● 相关系统定义变量

程序分配包括以下本地变量的系统定义变量。

变量名称	名称	功能	数据类型	R/W
P_First_Run Mode	运行开始时 1 个周期 ON 标志	控制器的动作模式从程序模式变更为运行模式时，若该程序正在执行，则在 1 个任务周期内为 TRUE。 若该程序不是正在执行，则保持 FALSE。 用于在 CPU 单元开始运行时进行初始处理等情况。	BOOL	R
P_First_Run*1	程序启动时 1 个周期 ON 标志	该程序开始执行后，仅在 1 个任务周期内为 TRUE。*2 用于每次开始执行程序时需要进行初始处理的情况。	BOOL	R
P_PRGER	指令错误发生标志	相应的程序或通过该程序调用的 FB 或 FUN 发生指令错误时，持续变为 TRUE 的标志。一旦变为 TRUE，在通过用户程序变为 FALSE 前，将保持 TRUE。	BOOL	RW
P_CY	进位标志	在部分指令中更新的标志。	BOOL	R

*1. 可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。
*2. 程序的执行 / 停止指令中使用 PrgStart 指令 / PrgStop 指令。PrgStart 指令可设定为 P_First_Run 无需变为 TRUE，即可执行程序。

6-2-5 功能块 （FB）的详情

功能块 (FB) 的创建步骤

FB 由预先创建的 “FB 定义 ” 和实际在程序中使用 FB 所需的 “ 实例 ” 构成。
FB 按以下步骤创建。

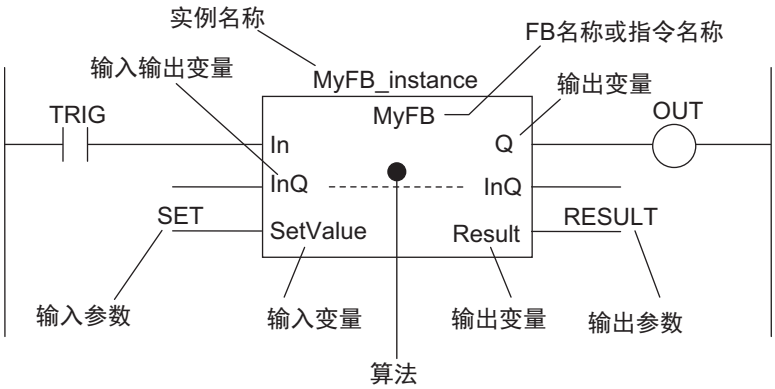
1 创建 FB 定义
创建算法。

2 将 FB 定义作为实例配置到程序中
通过程序或其他 FB 调用 FB 定义并实际配置。
1 个 FB 定义可通过多个程序或其他 FB 调用并配置。
FB 定义在配置到程序或其他 FB 后，将作为实例进行单独处理，可在程序或其他 FB 上进行实际动作。

功能块 (FB) 的构成

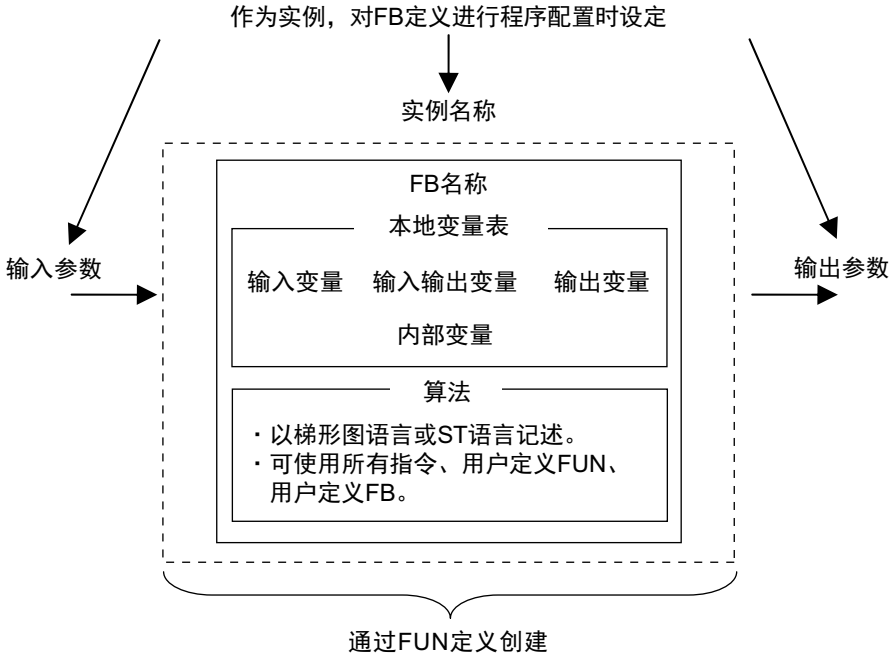
梯形图语言中，FB 用以下长方形表述。ST 语言中的表述请参考后述的📖「 用 ST 语言调用功能块（FB）的表述 」（P.6-11）。
FB 由以下内容构成。

- 用梯形图语言表述的 FB：



- FB 设定的项目

  在 FB 定义和实例中配置时，分别进行如下设定。



● FB 名称或指令名称

创建时命名的 FB 定义名称或指令名称。

● 实例名称

为管理 FB，对各程序中配置的 FB 单独命名的名称。

通过程序或其他 FB 调用 FB 时，设定实例名称。

● 算法

可以用梯形图语言或 ST 语言记述。

算法部分中可记述所有指令、用户定义 FUN 及用户定义 FB。

● 本地变量表

定义输入变量、输出变量、输入输出变量、内部变量、外部变量。

📖请参考「 功能块 （FB）内变量的规格」（P.6-12）。

● 参数

对输入变量的输入参数

开始执行 FB 时，向 FB 内的输入变量传递值。记述变量或常数。

输出变量的输出参数

FB 执行结束时，接收 FB 中输出变量的值。记述变量。

输入输出变量 — 输入输出变量间共享的输入输出参数

在 FB 内变更值。在输入和输出中记述相同的变量。

参考

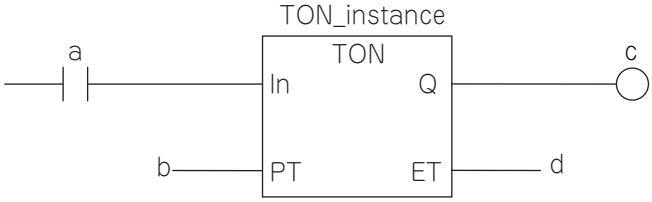
输入参数、输出参数也可省略。动作的详情请参考📖「 参数省略时的动作」（P.6-24）。

用 ST 语言调用功能块 （FB）的表述

ST 语言中， FB 的调用表述如下。

实例名称（输入变量 1:= 输入参数 1, ･･･输入变量 N:= 输入参数 N, 输入输出变量 1:= 输入输出变量 1,･･･输入输出变量 N:= 输入输出参数 N, 输出变量 1=> 输出参数 1,・・・输出变量 N=> 输出参数 N）；

按 FB 定义中规定的顺序记述参数，可省略输入变量名称等，只记述参数。
此时，参数的数量需要与 FB 定义的输入变量的数量一致。



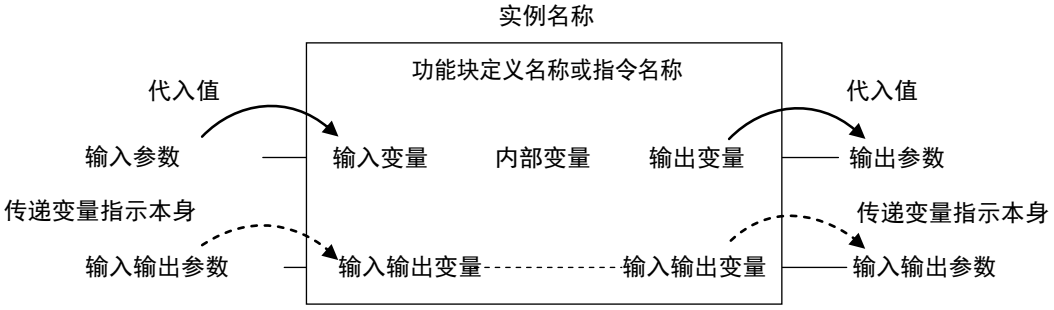
用 ST 语言表述的 FB：

实例名称

```
TON_instance(In:=a, PT:=b, Q=>c, ET=>d);
TON_instance(In:=a, PT:=b, Q=>c); (* 省略了输出ET时 *)
TON_instance(a,b,c,d); (* 省略了输入变量、输出变量时 *)
```

详情请参考后述的📖「FB 调用」（P.6-115）。

功能块 （FB）内变量的规格



FB 变量的规格如下所示。

变量	个数	规格
输入变量[*1]	1 ～ 64	作为 FB 的输入参数，在 FB 内部使用的 （值无法变更）变量。
		• 开始执行 FB 时，将输入参数的值设置为输入变量。 • 输入参数中可记述常数或变量。 • 输入参数的省略： 📖请参考「 参数省略时的动作」（P.6-24）。 • 可指定上升沿 / 下降沿。 • 可从 FB 外部参照或变更值。在 “ 实例名称 . 输入变量名称 ” 中指定。[*2]
输出变量[*3]	1 ～ 64	作为 FB 的输出参数，在 FB 内部使用的变量。
		• FB 执行结束时，将输出变量的值设置为输出参数。 • 输出参数不可指定为常数或带常量属性的变量。 • 输出参数连接可省略。省略时，输出变量的值将不会代入到任何参数中。 • 可从 FB 外部参照值。可通过 “ 实例名称 . 输出变量名称 ” 读取。但是不可写入。
输入输出变量	0 ～ 64	作为 FB 的输入输出，在 FB 内部使用的 （值可变更）变量。
		• 传递或接收 “ 输入输出参数 ”。 • 输入输出参数不可指定为常数或带常量属性的变量。 • 若在 FB 内变更值，输入输出参数的值将在变更时发生变化。 • 不可省略输入输出参数。
内部变量	无限制	FB 内的临时保存用参数。
		• 无论是非执行时还是执行后，都会保持值。 • 可以有保持属性。 • 不可从 FB 外部参照值。
外部变量	无限制	参照全局变量的变量。
EN	0	FB 中无法使用 （用户定义 FB、 FB 型指令亦同）。
ENO	0 或 1	作为 BOOL 型输出变量使用，一般在正常结束时变为 TRUE，异常结束或未启动时变为 FALSE。
		• 用户定义 FB 中可省略。 • 其他详情请参考后述的📖「ENO」（P.6-21）。

*1. 在 Sysmac Studio Ver.1.01 以下的版本中，无论是以梯形图语言使用 FB，还是以 ST 语言使用 FB，都需要 1 个以上的 BOOL 型输入变量。
*2. 在 Sysmac Studio Ver.1.07 以下版本中，不可从 FB 外部变更输入变量的值。但是，可从 FB 外部参照。
*3. 以梯形图语言使用 FB 时，包括 ENO 在内，需要 1 个以上的 BOOL 型输出变量。
以ST语言使用FB时，在Sysmac Studio Ver.1.01以下版本中，包括ENO在内，需要1个以上的BOOL型输出变量。

各变量的变量属性是否可以设定，请参考📖「6-3-4 变量的属性」（P.6-30）。

参考

对于全局变量中外部变量的定义，若用 FB 内的外部变量定义与全局变量同名的变量，将自动设定。

● ENO

ENO 为 FALSE 时，其他输出变量的值将保持上次的值。

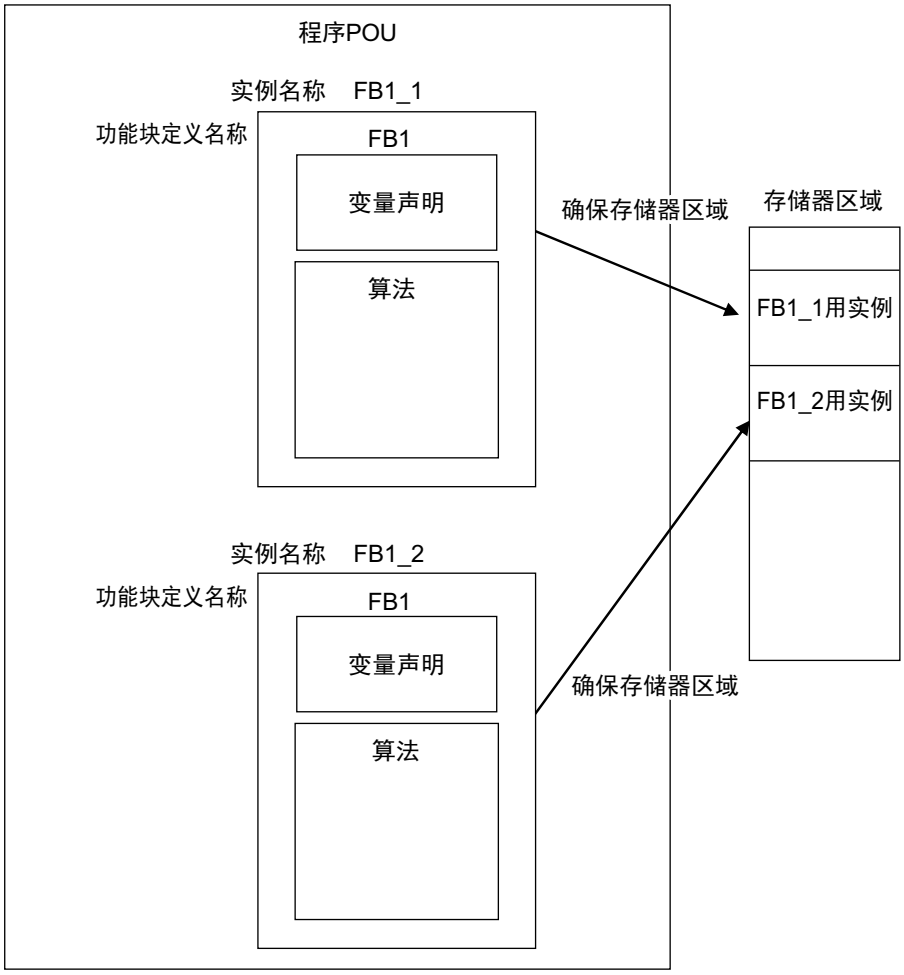
功能块 (FB) 定义和实例

FB 由预先创建的 “FB 定义 ” 和实际在程序中使用 FB 所需的 “ 实例 ” 构成。
重复利用的来源称为 “FB 定义 ”。
“FB 定义 ” 中记载有算法和本地变量表。

● FB 的实例

若将 FB 定义配置到程序或其他 FB 上， FB 定义将作为实际程序或 FB 的一部分处理。
将 FB 定义配置到程序或其他 FB 后，称为 “ 实例 ”。
各实例拥有实例名称作为识别符，分别使用存储区域。

以 1 个 FB 定义为基础，生成多个 “ 实例 ” 后，
可在同一功能中进行不同输入输出数据的处理。



“ 实例 ” 不可通过不同的程序或其他 FB 读取。实例名称相同，但配置到不同的程序或其他 FB 时，将视为不同的实例进行动作。

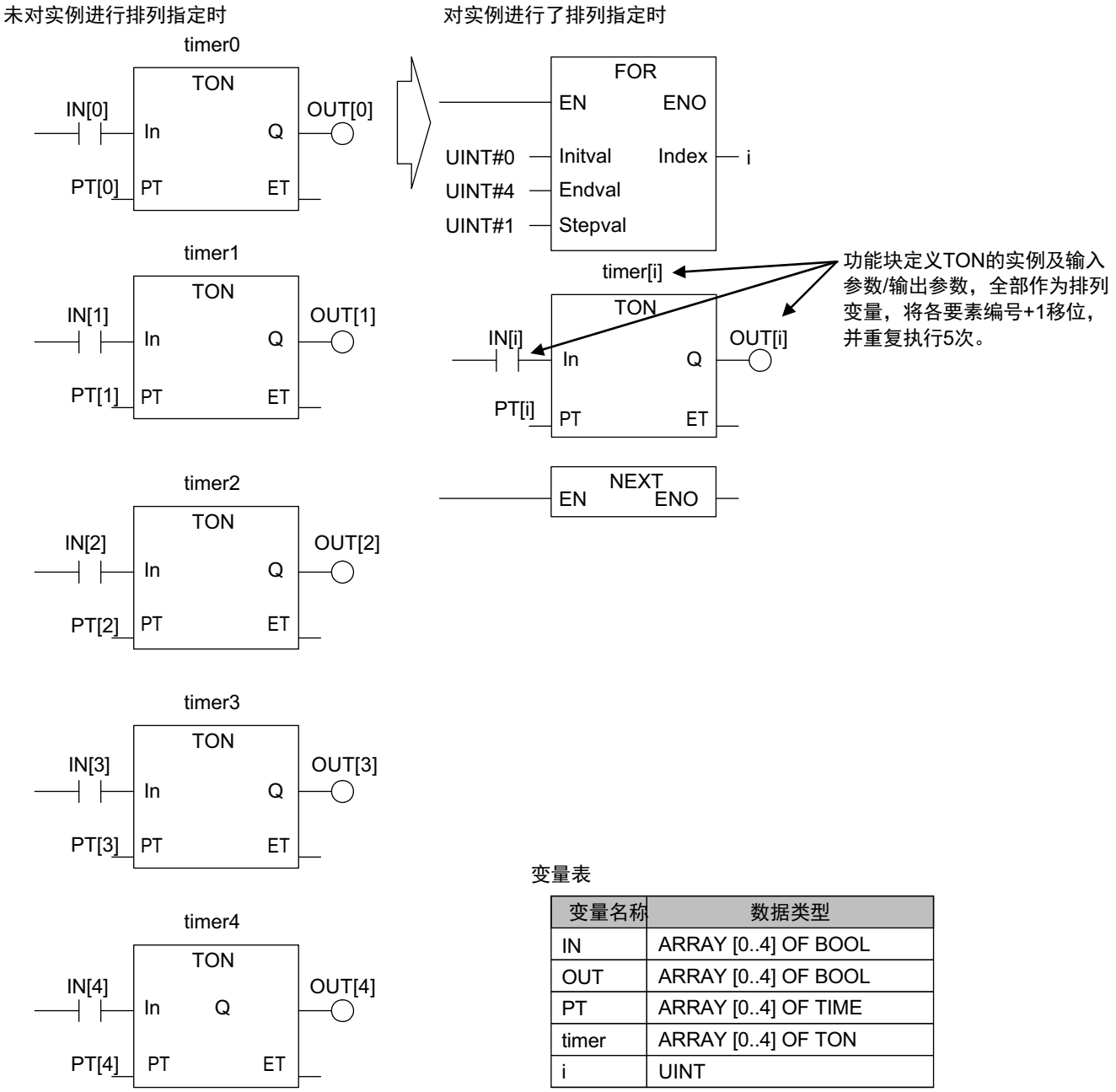
实例的排列指定化

可对实例进行排练指定。

用变量值间接指定要素编号，可用一个实例名称执行多个实例。

此外，输入参数及输出参数也作为排列指定，用共通的变量指定要素编号，可用一个实例名称切换输入源、输出位置，指定多个实例。

例）



变量表

变量名称	数据类型
IN	ARRAY [0..4] OF BOOL
OUT	ARRAY [0..4] OF BOOL
PT	ARRAY [0..4] OF TIME
timer	ARRAY [0..4] OF TON
i	UINT

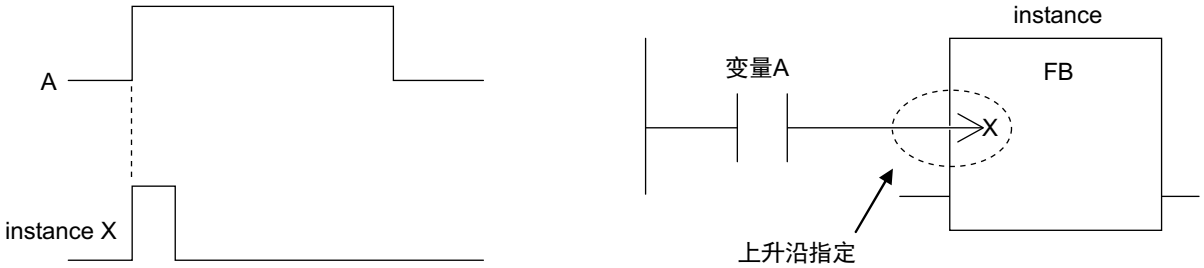
功能块 (FB) 的启动条件

FB 与 FUN 不同，没有 EN 输入，始终 （每个任务周期）执行。

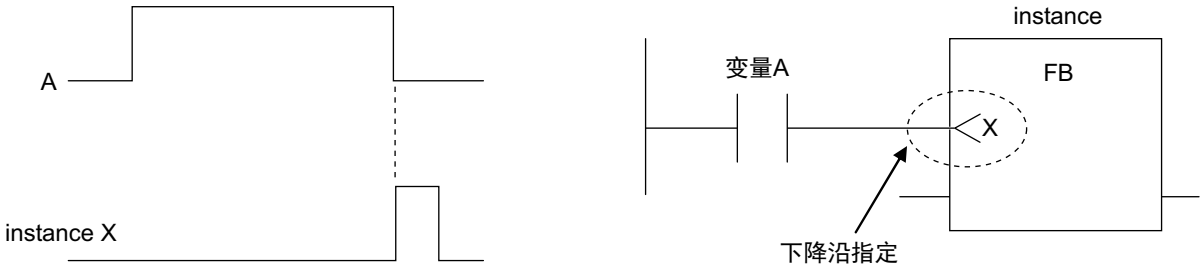
情况	FB 内算法		ENO	ENO 以外的动作
通常时	执行。	正常结束时	TRUE	输出参数： 根据内部算法更新值 输入输出参数： 根据内部算法更新值
		异常结束时	FALSE	输出参数： 保持 输入输出参数： 根据内部算法更新值
主控区域内	在功率流输入为 FALSE 的状态下执行。		用户任意	根据 ENO 的值为上述之一

关于功率流输出、参数输出，请参考「6-5-2 梯形图语言 （LD）」（P.6-84）。

为输入变量指定边缘，该变量在输入参数的上升沿时变为 TRUE。



也可指定下降沿。



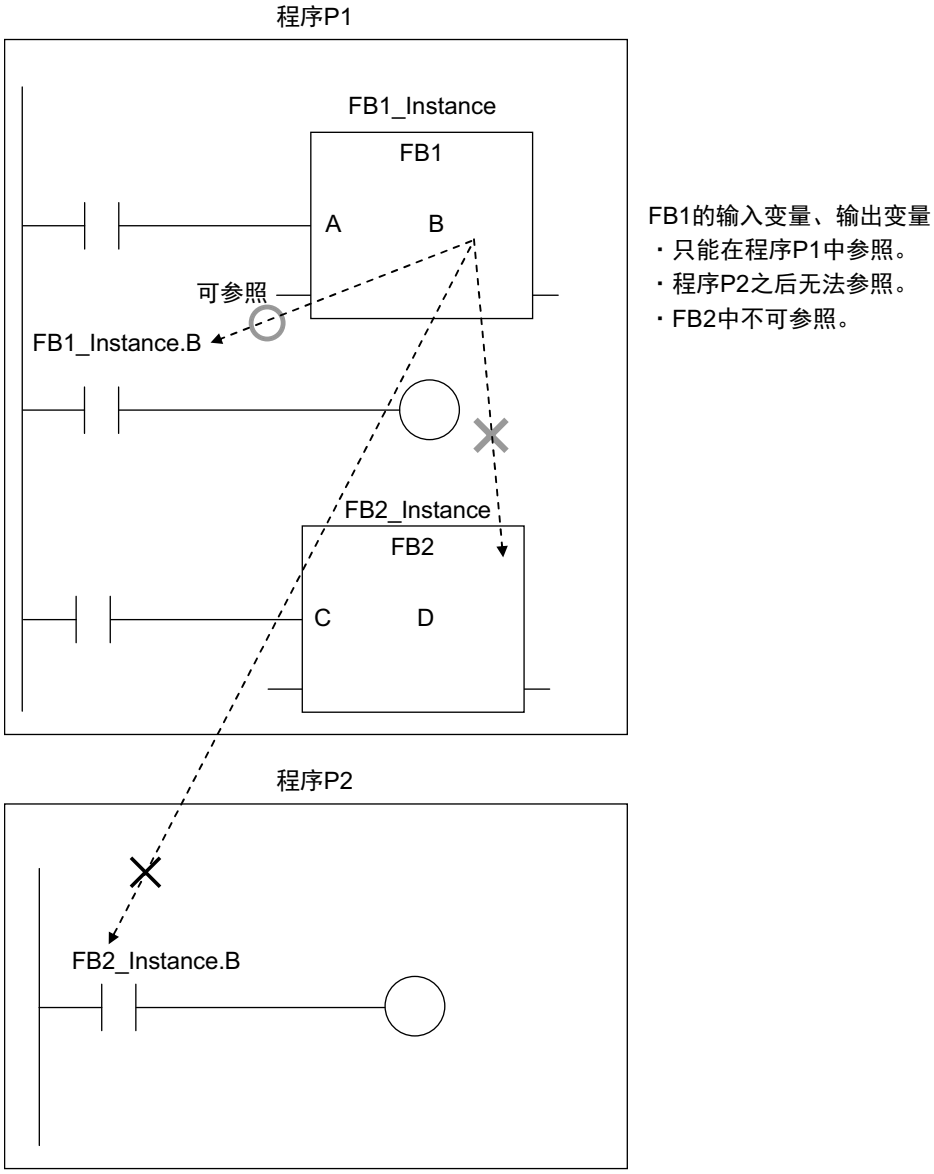
从外部参照功能块 （FB）的变量

FB 的输入变量及输出变量可从 FB 外部参照。变量的表述如下所示。

实例名称 . 变量名称

（例） FB1 的实例 “FB1_Instance” 内的输出变量 “B” 时
FB1_Instance.B

可参照的范围仅限包括该实例的程序内。
不可从该程序中的其他 FB 实例参照。也无法从其他程序参照。



以下参数不可从外部参照。若参照，链接时将发生异常。

- FB 的输入输出变量
- 省略了输入参数时，不应用初始值的 FB 型指令的输入变量

6-2-6 函数 （FUN）的详情

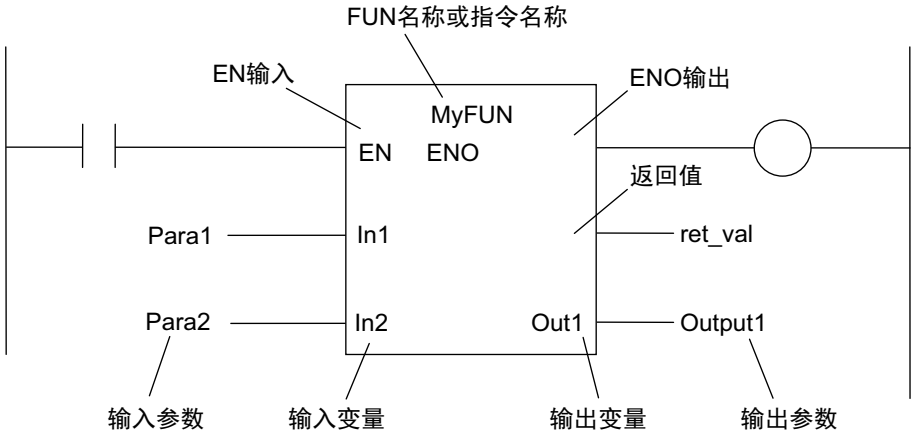
函数 （FUN）的构成

梯形图语言中， FUN 用以下长方形表述。

ST 语言中的表述请参考后述的 📖「 ST 语言中函数 （FUN）的表述」（P.6-19）。

FUN 由以下内容构成。

用梯形图语言表述的 FUN：



● **FUN 定义名称或指令名称**

创建时命名的 FUN 定义名称或指令名称。

● **实例名称**

无。

● **算法**

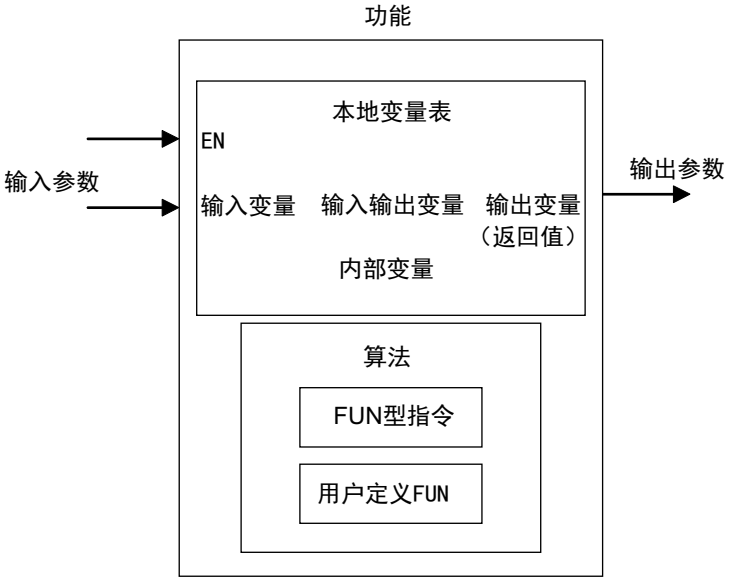
可以用梯形图语言或 ST 语言记述。

算法部分中可记述 FUN 型指令、用户定义 FUN。

不可记述 FB 型指令及用户定义 FB。

不可记述微分指令 （R_TRIG、 UP）。

不可使用用户定义变量 P_First_RunMode、 P_First_Run。



● **本地变量表**

登录输入变量、输出变量、输入输出变量、内部变量、外部变量。

📖请参考「 函数 （FUN）内变量的规格」（P.6-20）。

● **参数**

对输入变量的输入参数

开始执行 FUN 时，向 FUN 内的输入变量传递值。记述变量或常数。

输出变量的输出参数

FUN 执行结束时，接收 FUN 中输出变量的值。记述变量。

输入输出变量 — 输入输出变量间共享的输入输出参数

在 FUN 内变更值。在输入和输出中记述相同的变量。

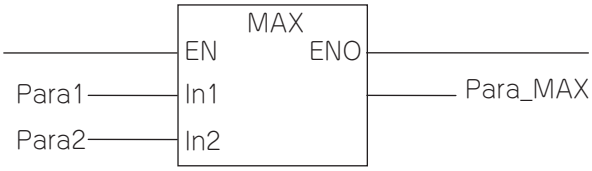
ST 语言中函数 （FUN）的表述

ST 语言中， FUN 表述如下。

返回值 :=FUN 名称（输入变量 1:= 输入参数 1, ･･･输入变量 N:= 输入参数 N, 输入输出变量 1:= 输入输出变量 1,･･･输入输出变量 N:= 输入输出参数 N, 输出变量 1=> 输出参数 1,･･･输出变量 N=> 输出参数 N）; 但是，返回值也可省略。

按 FUN 定义中规定的顺序记述参数，可省略输入变量名称等，只记述参数。此时，参数的数量需要与 FUN 定义的输入变量的数量一致。

用 ST 语言表述的 FUN：

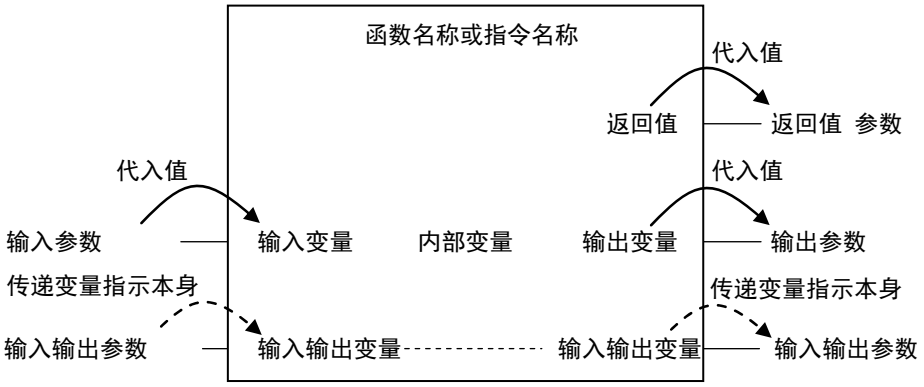


FUN名称

```
Para_MAX := MAX(In1:=Para1, In2:=Para2);
Para_MAX := MAX(Para1, Para2);              (* 省略了输入变量时 *)
```

详情请参考后述的📖「FUN 调用 」（P.6-118）。

函数 （FUN）内变量的规格



FUN 变量的规格如下所示。

变量	个数	规格
输入变量	0 ～ 64	作为 FUN 的输入参数，在 FUN 内部使用的 （值无法变更）变量。
		• 开始执行 FUN 时，将输入参数的值设置为输入变量。 • 输入参数中可记述常数或变量。 • 输入参数的省略： 请参考 「 参数省略时的动作」（P.6-24）。 • 与 FB 不同，不可指定上升沿 / 下降沿。 • 不可从 FUN 外部参照值。 • 欧姆龙提供的指令中包括输入变量的数量可变的指令，但用户创建的 FUN 不可变更输入变量的数量。
输出变量	0 ～ 64	作为 FUN 的输出参数，在 FUN 内部使用的变量。
		• FUN 执行结束时，将输出变量的值设置为输出参数。 • 输出参数不可指定为常数或带常量属性的变量。 • 包括 ENO、返回值在内，至少需要 1 个 BOOL 型输出变量。 • 输出参数连接可省略。 省略时，输出变量的值将不会代入到任何参数中。 • 不可从 FUN 外部参照值。 • 如果是用户定义的 FUN，在 FUN 内的算法中请务必设定输出变量的值。若未在 FUN 内的算法中设定输出变量，输出变量的值将变得不确定。
输入输出变量	0 ～ 64	作为 FUN 的输入输出，在 FUN 内部使用的 （值可变更）变量。
		• 传递或接收 “ 输入输出参数 ” 本身 （变量指示本身）。 • 输入输出参数不可指定为常数或带常量属性的变量。 • 若在 FUN 内变更值，输入输出参数的值将在变更时发生变化。 • 不可省略输入输出参数。 • 不可从 FUN 外部参照值。
内部变量	无限制	FUN 内的临时保存用参数。
		• 执行结束后，值不保持。 • 不可从 FUN 外部参照值。
外部变量	无限制	参照全局变量的变量。
EN	1	启动 FUN 所需的 BOOL 型输入变量。
		• EN 的值为 TRUE 时，执行 FUN。 • 必须存在 1 个 （用户定义 FUN、 FUN 型指令亦同）。

变量	个数	规格
ENO	0 或 1	作为 BOOL 型输出变量使用，一般在正常结束时变为 TRUE，异常结束或未启动时变为 FALSE。
		• 用户定义 FUN 中可省略。 • 其他详情请参考后述的 📖「ENO」（P.6-21）。
返回值	1	FUN 结束算法，向调用源返回作为处理结果的值。
		• 必须存在 1 个。 • 可指定为所有基本数据型、列举型。不可指定为排列、结构体整体、联合体整体。 • 如果是用户定义的 FUN，在 FUN 内的算法中请务必设定返回值。若未在 FUN 内的算法中设定返回值，返回值将变得不确定。 • 其他详情请参考后述的 📖「返回值」（P.6-22）。

各变量的变量属性是否可以设定，请参考📖「6-3-4 变量的属性」（P.6-30）。

参考

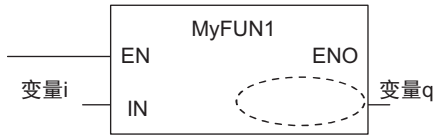
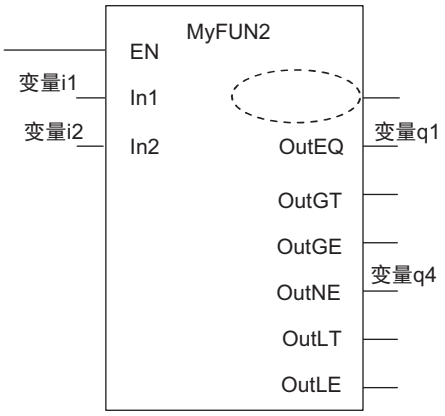
在 FUN 内变量表中将全局变量登录为外部变量后，可参照全局变量。
一般情况下，建议创建函数时，确保输入参数的值与输出值一一对应。在函数内参照全局变量，输出值受此影响的算法并不推荐。因此，用 Sysmac Studio 检查程序时，将显示 “ 不推荐使用函数内的全局变量。 ” 的警告。请在确认后采取必要措施。

● ENO

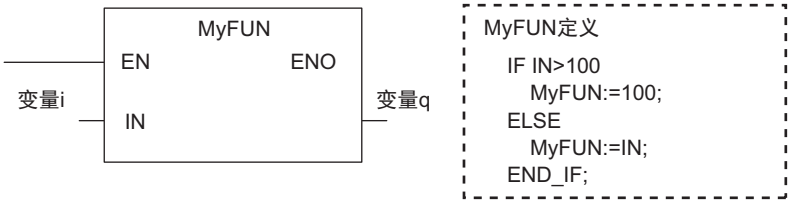
- ENO 为 FALSE 时，其他输出变量的值将保持上次的值。

● 返回值

- 在梯形图语言中，返回值显示为空白。

情况	梯形图语言表述	ST 语言表述
使用返回值时		变量 q := MyFUN1 （变量 i）；
不使用返回值时		MyFUN2 （In1:= 变量 i1, In2:= 变量 i2, OutEQ=> 变量 q1, OutNE=> 变量 q4 ）；

- 梯形图语言、 ST 语言的调用侧都可以不使用返回值。
- 在函数的算法中设置返回值的值时，请将值设置到与函数名称同名的变量中。
  例如，函数为 “MyFUN” 时，返回值为 “MyFUN”。

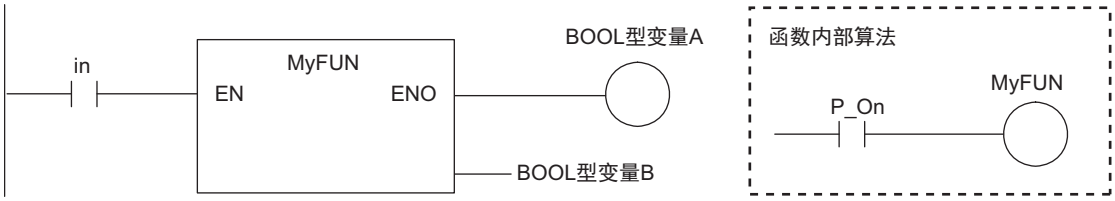


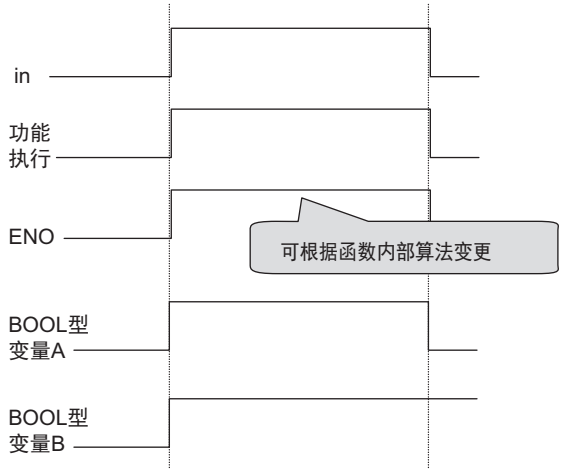
函数 （FUN）的启动条件

EN 变为 TRUE，则 FUN 启动。 EN 变为 FALSE，则 FUN 停止处理。

输入变量	FUN 内算法		ENO	ENO 以外的动作
EN=TRUE 时	执行	正常结束时	TRUE	输出参数： 根据内部算法更新值 输入输出参数： 根据内部算法更新值
		异常结束时	FALSE	输出参数： 保持 输入输出参数： 根据内部算法更新值
EN=FALSE 时	不执行		FALSE	输出参数、输入输出参数： 保持值
主控区域内	不执行		FALSE	输出参数、输入输出参数： 保持值

例）





6-2-7 功能块 （FB）、函数 （FUN）共通动作

EN、 ENO 的使用 / 省略可否

FB /FUN 中 EN、 ENO 的使用 / 省略可否情况如下所示。

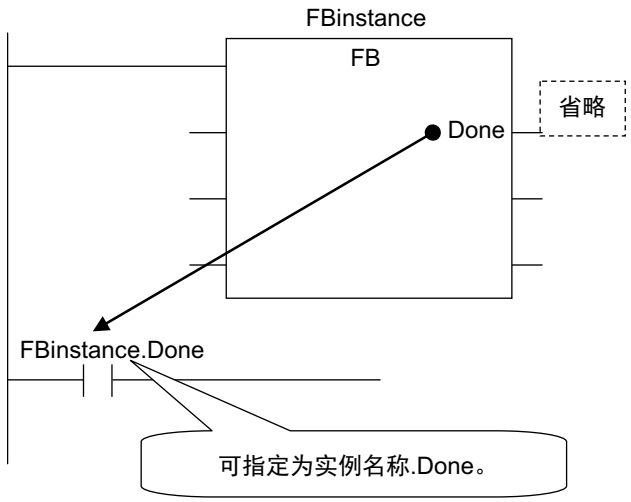
POU		EN	ENO
FB	用户定义	不可使用。 若要通过 Sysmac Studio 定义到变量表中，链接时将发生异常。	可使用 （可省略）。 用户通过 Sysmac Studio 定义为输出变量。
	指令	所有 FB 型指令均不使用 EN。	包括使用 ENO 的指令和不使用的指令。
FUN	用户定义	必须使用。 创建 FUN 时， Sysmac Studio 默认自动定义到变量表中。	可使用 （可省略）。 用户通过 Sysmac Studio 定义为输出变量。
	指令	所有 FUN 型指令均使用 EN。	包括使用 ENO 的指令和不使用的指令。

参数省略时的动作

输入参数、输出参数可以都省略。

	省略时的动作	
省略对象的参数	FB	FUN
对输入变量的输入参数	• 实例首次启动时，按初始值动作。 • 之后按上次值动作 （始终省略输入变量时，按初始值继续动作）。	按初始值动作。
输出变量的输出参数	可省略。 该指令之外，可通过 " 实例名称 . 输出变量名称 " 访问 （*）	可省略，但省略时无法获取动作结果。
对输入输出变量和来自输入输出变量的输入输出参数	不可省略。	不可省略。

* FB 的输入变量 / 输出变量可通过 " 实例名称 . 变量名称 " 从 FB 外部（但仅限该程序）直接参照。FUN 中不可参照。



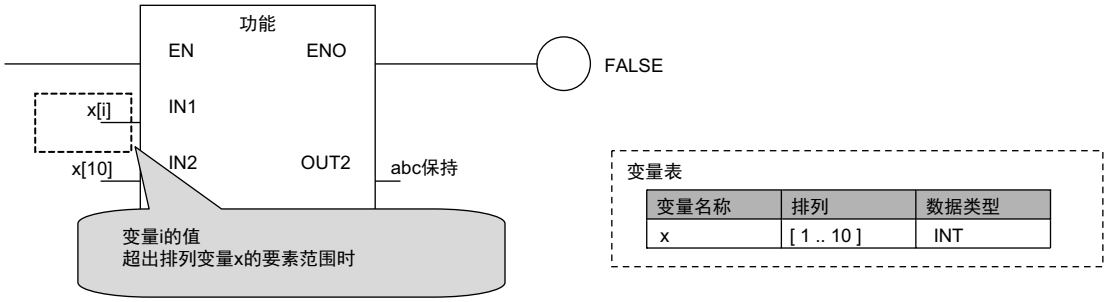
参数异常时的动作

输入参数、输出参数或输入输出参数发生异常时，动作如下。

● 输入发生异常时

检测到输入参数异常时，将不执行 FB/FUN， ENO 变为 FALSE。同时，功率流输出变为 FALSE，其他保持原来的值。

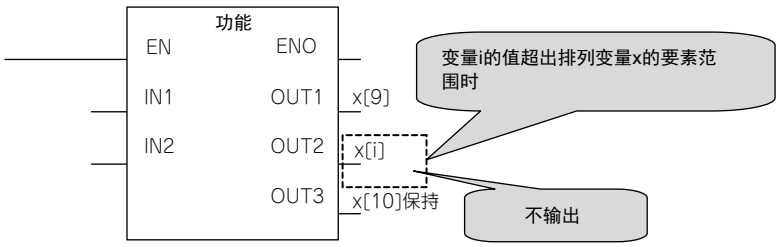
例）



● 输出参数发生异常时

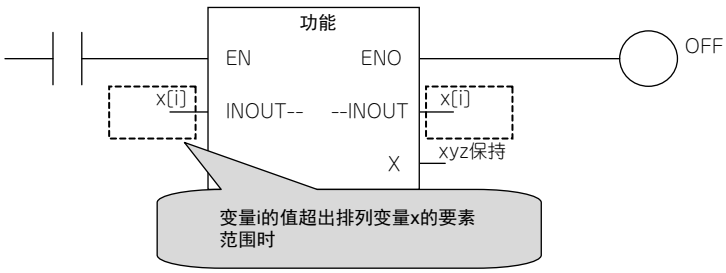
检测到输出参数异常时，该参数之后不输出值，并保持值。

例）



● 输入输出发生异常时

检测到输入输出参数异常时，将不执行 FB/FUN，ENO 变为 FALSE。同时，功率流输出变为 FALSE，其他保持原来的值。



递归调用

FB 及 FUN 均不可进行以下递归调用。此时，用 Sysmac Studio 链接时将发生异常。

- 自身调用
- 在调用的 FB 或 FUN 上，调用自己本身

6-2-8 POU 的限制事项

下面介绍创建 POU 时的各种限制事项。

名称

POU 名称、 FB 的实例名称限制请参考📖「6-3-12 变量名称等程序相关名称的限制」（P.6-77）。

多个参数的传递

向 FB 或 FUN 传递多个参数时，若以排列指定或结构体型传递数据，程序会变得简洁。

此时，与其用输入变量传递，用输入输出变量传递的处理时间更短。但是，使用输入输出变量后，若作为参数传递到 FB 或 FUN 的数据内容发生替换，原来的值不会保留，敬请注意。

参考

将排列指定的变量和结构体型变量的所有成员作为参数时

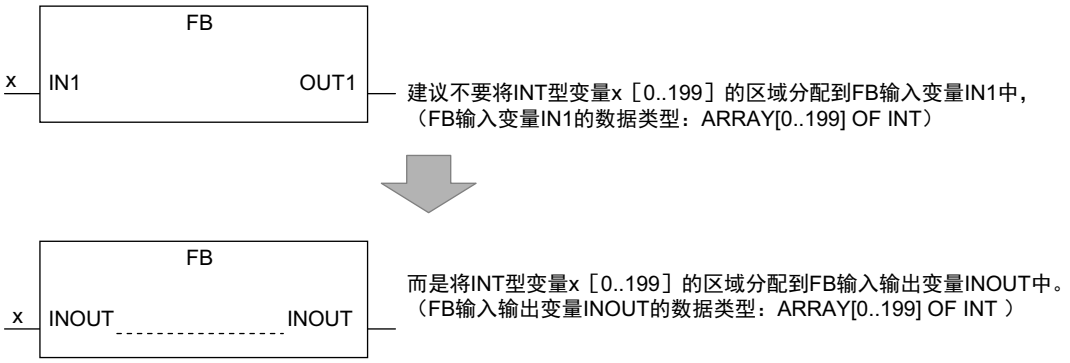
输入参数、输出参数可指定为排列指定的变量或结构体型变量的所有成员。但是，与基本数据型的变量相比，数据的收发时间更长 （取决于大小）。

因此，如果是 FB，收发排列指定的变量和结构体型变量的所有成员时，建议设计为对输入输出变量进行收发。

例 1）指定排列指定的所有要素时

程序的变量表

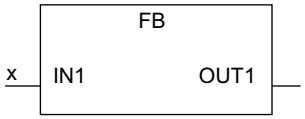
变量名称	数据类型
x	ARRAY[0..199] OF INT



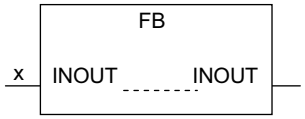
例 2）指定结构体型变量的所有成员时

程序的变量表

变量名称	数据类型
x	MyStructure



建议不要将MyStructure型变量x的整体成员区域分配到FB输入变量IN1中，
（FB输入变量IN1的数据类型：MyStructure）



而是将MyStructure型变量x的的整体成员区域分配到FB输入输出变量INOUT中。
（FB输入输出变量INOUT的数据类型：MyStructure）

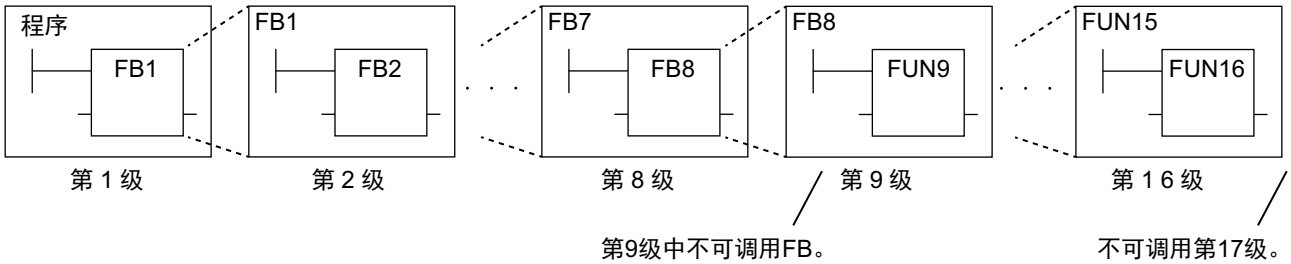
嵌套的层级

用程序调用的 FB 或 FUN，接着调用其他 FB 或 FUN 的操作称为嵌套。通过用户定义的 FB 或 FUN 可调用的 POU 和嵌套层级的深度有以下限制。若超出限制，链接时将发生异常。

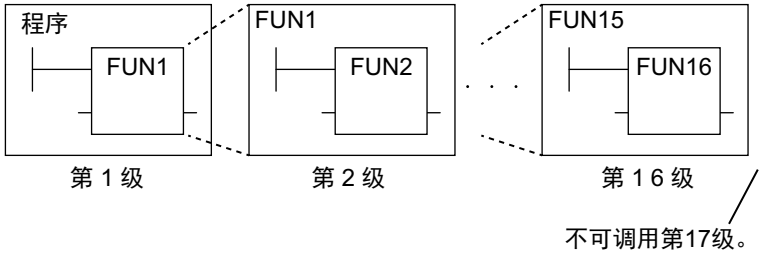
POU	可调用的 POU	层级深度
FB	FB、 FUN	最多 8 层
FUN	FUN	最多 16 层*1

*1. 可在 CPU 单元版本 1.03 以上和 Sysmac Studio Ver.1.04 以上的组合中使用。其他最多为 8 层。

例 1）可用程序最多调用第 8 层 FB。从这里开始，可以继续调用到第 16 层 FUN。



例 2）可用程序最多调用第 16 层 FUN。



6-3 变量

NJ/NX 系列系统中，与外部的输入输出信息交换或数据运算等处理全部通过变量执行。
下面详细介绍变量的规格。
在 Sysmac Studio 上设定变量的方法请参考📖《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》。

6-3-1 变量定义

变量是指保存与外部进行过交换的输入输出数据和 POU 内部处理时的临时数据的容器。换言之，就是带名称或数据类型等属性的数据之容器。

无需指定变量和存储器地址的分配。此外，还可将变量分配到特定的存储器地址中（*）。NJ/NX 系列 CPU 单元将自动分配变量所需的存取区域。

* 在部分 CJ 系列高功能单元中使用特定功能时使用。需要通过变量属性的 AT（分配目标）指定，指定 CJ 单元用存储器的地址。详情请参考📖「AT（分配目标）指定」（P.6-57）。
* 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

6-3-2 变量的种类

变量大致可分为以下三种。

● 用户定义变量

所有属性由用户定义的变量。
本章后面的说明中，将介绍 “ 用户定义变量 ”。

● 准用户定义变量

访问特定设备 / 数据所需的变量。
包括设备变量和凸轮数据变量。
设备变量请参考📖「2-3-1 变量的种类」（P.2-10）及📖「3-3-1 I/O 端口」（P.3-8）。

● 系统定义变量

名称等所有属性由 NJ/NX 系统事先定义，且分配有特定功能的变量。各功能模块分别准备。详情请参考📖「A-4 系统定义变量一览」（P.A-38）。

关于变量分类的详情，请参考📖「2-3-1 变量的种类」（P.2-10）。

6-3-3 POU 对应的用户定义变量的种类

用户定义变量根据在 POU 上的功能大致可分为以下 6 种。

○：可定义

用户定义变量的种类		POU 类型		
		程序	FB	FUN
本地变量	内部变量	○	○	○
	输入变量	无	○	○
	输出变量	无	○	○
	输入输出变量	无	○	○
全局变量		○ （*）	○ （*）	○ （*）
外部变量		○	○	○

* 将全局变量定义为外部变量后，可通过外部变量，参照全局变量。

本地变量

只能通过 POU （程序、 FB、 FUN）读写的变量。
POU 为程序时，本地变量等于内部变量。 POU 为 FB 或 FUN 时，为内部变量、输入变量、输出变量、输入输出变量、外部变量的总称。

● 内部变量

只在 POU 内部使用的变量。
在 POU 的本地变量表中定义 （声明）。
不可从 POU 外部参照内部变量的值。
即使在不同的 POU 中声明相同名称的内部变量，各变量也会分配到不同的分配目标中。

● 输入变量

调用 POU 时，通过调用源代入输入参数值的变量。在 POU 的本地变量表中定义 （声明）。

● 输出变量

POU 的处理结束前，在输出参数中代入该值，并返回到调用源的变量。在 POU 的本地变量表中定义（声明）。

● 输入输出变量

调用 POU 时，通过调用源代入输入输出参数本身 （变量指示本身）的变量。若在 POU 内替换，输入输出参数的值将在替换时发生变化。
在 POU 的本地变量表中定义 （声明）。

● 外部变量

通过 POU 进行外部参照的变量。
通过 POU 参照登录的全局变量。

全局变量

在全局变量表中定义 （声明）。

根据从站 / 单元构成自动生成的设备变量以及根据全轴设定表自动生成的 “ 轴 / 轴组变量 ” 将自动登录到全局变量中。

6-3-4 变量的属性

变量中可设定或附加以下属性。

变量的属性一览和各变量种类的属性有无

● **变量的属性一览**

属性	说明	指定	初始设定
变量名称	用于识别变量的名称。	—	—
数据类型	对变量中保存的数据定义格式。	—	BOOL
AT （分配目标）指定	将 CJ 单元的特定地址作为变量时，指定该变量的分配位置。*1	• 不指定 • 指定	不指定
保持	指定在以下任意情况下是否保持值。 • 断电后接通电源时 • 切换到运行模式时 • 发生全部停止故障等级的控制器异常时	• 保持：保持左侧情况时的值 （仅限有电池时） • 不保存：变为初始值	不保存：变为初始值
初始值	可选择初始值设定 “ 无 ” 和 “ 有 ”。 初始值设定为 “ 有 ” 时：无保持指定，指定以下任意情况下变量的值。 • 电源接通时 • 动作模式变更时 • 发生全部停止故障等级的控制器异常时 初始值设定为 “ 无 ” 时：不保持值	初始值设定 • 有 • 无	取决于数据类型。 （请参考📖「 初始值」（P.6-60）。）
常量	在下载变量时设定初始值，之后无法替换值。	设为 / 不设为常量	—
网络公开	可使用 CIP 通信或数据链接功能，从控制器外部读写变量。	• 不公开 • 仅公开 • 输入 • 输出	不公开
边缘	检测 FB 输入参数的上升沿、下降沿。 仅可设定为 BOOL 型输入变量。	• 无 • 上升沿 • 下降沿	无

*1. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

参考

任务间排他控制功能
设为只能通过一个任务读写全局变量，防止在处理过程中全局变量的值发生变化。不是作为变量的属性，而是在任务设定中指定。

● 各变量种类的属性有无

变量种类		变量名称	数据类型	AT （分配目标）指定	保持	初始值	常量	网络公开	边缘
全局变量		可设定	可设定	可设定	可保持	可设定	可指定	可公开	不可
程序	内部变量	可设定	可设定	可设定	可保持	可设定	可指定	不可公开	不可
	外部变量	不可设定	不可设定	不可设定	不可保持	不可设定	可指定	不可公开	不可
功能块	内部变量	可设定	可设定	可设定	可保持	可设定	可指定	不可公开	不可
	输入变量	可设定	可设定	不可设定	可保持	可设定	可指定	不可公开	可以
	输出变量	可设定	可设定	不可设定	可保持	不可设定	不可指定	不可公开	不可
	输入输出变量	可设定	可设定	不可设定	不可保持	不可设定	可指定	不可公开	不可
	外部变量	不可设定	不可设定	不可设定	不可保持	不可设定	可指定	不可公开	不可
功能	内部变量	可设定	可设定	不可设定	不可保持	可设定	可指定	不可公开	不可
	输入变量	可设定	可设定	不可设定	不可保持	可设定	可指定	不可公开	不可
	输出变量	可设定	可设定	不可设定	不可保持	不可设定	不可指定	不可公开	不可
	输入输出变量	可设定	可设定	不可设定	不可保持	不可设定	可指定	不可公开	不可
	外部变量	不可设定	不可设定	不可设定	不可保持	不可设定	可指定	不可公开	不可

6-3-5 数据类型

数据类型是指规定了表现变量值的形式和范围的属性。

根据数据类型不同，变量声明时确保的存储器大小也不同。存储器大小越大，可表现的值的范围越大。

此外，指令的输入 / 输出 / 输入输出变量的数据类型因各指令而不同。指令的参数 —— 输入 / 输出 / 输入输出参数的数据类型请根据对应的输入 / 输出 / 输入输出变量的数据类型设定。

基本数据型和派生数据型

数据类型包括已事先规定规格的基本数据型和由用户定义规格的派生数据型。

● 基本数据型的种类

基本数据型可进行以下分类。

分类	定义
布尔型	值为 0 或 1 中任意一个的数据类型。
位列型	以位列表现值的数据类型。
整数型	值为整数值的数据类型。
实型数据	值为实数值的数据类型。
持续时间型	值为时间 （日・时・分・秒・毫秒）的数据类型。
时刻型	值为时刻 （时・分・秒）的数据类型。
日期型	值为日期 （年・月・日）的数据类型。
日期时刻型	值为日期时刻 （年・月・日・时・分・秒・毫秒）的数据类型。
字符串型	值为字符串的数据类型。

基本数据型有 20 种。各规格如下所示。

下表的 “ 数据大小 （容量值） ” 及 “ 校准 （临界值） ” 的含义为

- 数据大小 （容量值）： 实际值的容量。
- 校准 （临界值）： 确保存储器时的单位。

分类	数据类型	数据大小 （容量值）	校准 （临界值）	值的范围	记述方法
布尔型	BOOL	16 位	2 字节	FALSE、 TRUE	BOOL#0、 BOOL#1 FALSE、 TRUE
位列型	BYTE	8 位	1 字节	BYTE#16#00 ～ FF	BYTE#2#01011010 BYTE#2#0101_1010 BYTE#16#5A （还可使用分隔符 “_”。）
	WORD	16 位	2 字节	WORD#16#0000 ～ FFFF	
	DWORD	32 位	4 字节	DWORD#16#00000000 ～ FFFFFFFF	
	LWORD	64 位	8 字节	LWORD#16#0000000000000000 ～ FFFFFFFFFFFFFFFF	
整数型*1	SINT	8 位	1 字节	SINT#-128 ～ +127	100 INT#2#00000000_01100100 INT#8#144 INT#10#100 INT#16#64 -100
	INT	16 位	2 字节	INT#-32768 ～ +32767	
	DINT	32 位	4 字节	DINT#-2147483648 ～ +2147483647	
	LINT	64 位	8 字节	LINT#-9223372036854775808 ～ +9223372036854775807	
	USINT	8 位	1 字节	USINT#0 ～ +255	
	UINT	16 位	2 字节	UINT#0 ～ +65535	
	UDINT	32 位	4 字节	UDINT#0 ～ +4294967295	
	ULINT	64 位	8 字节	ULINT#0 ～ +18446744073709551615	
实型数据	REAL	32 位	4 字节	REAL#-3.402823e+38 ～ -1.175495e-38 0 1.175495e-38 ～ 3.402823e+38 + ∞ /- ∞	REAL#3.14 LREAL#3.14 3.14 -3.14 1.0E+6 1.234e4
	LREAL	64 位	8 字节	LREAL#-1.79769313486231e +308 ～ -2.22507385850721e-308 0 2.22507385850721e-308 ～ 1.79769313486231e+308 + ∞ /- ∞	
持续时间型 *1*2	TIME	64 位	8 字节	T#-9223372036854.775808ms （T#-106751d_23h_47m_16s_854.775808ms）～ T#+9223372036854.775807ms （T#+106751d_23h_47m_16s_854.775807ms）	T#12d3h3s T#3s56ms TIME#6d_10m TIME#16d_5h_3m_4s T#12d3.5h T#10.12s T#61m5s （与 T#1h1m5s 相同） TIME#25h_3m
日期型	DATE	64 位	8 字节	D#1970-01-01 ～ D#2106-02-06 （1970 年 1 月 1 日～ 2106 年 2 月 6 日）	开头加 'DATE#' 或 'date#'、或 'D#' 或 'd#'， 以 ' 年 - 月 - 日 ' 记述。 （例） d#1994-09-23
时刻型 *2	TIME_OF_DAY	64 位	8 字节	TOD#00:00:00.000000000 ～ TOD#23:59:59.999999999 （0 点 0 分 0.000000000 秒 ～ 23 点 59 分 59.999999999 秒）	开头加 'TIME_OF_DAY#' 或 'time_of_day#'、或 'TOD#' 或 ' tod #'， 以 ' 时 : 分 : 秒 ' 记述。 （例） tod#12:16:28.12

分类	数据类型	数据大小（容量值）	校准（临界值）	值的范围	记述方法
日期时刻型 *2	DATE_AND_TIME	64 位	8 字节	DT#1970-01-01-00:00:00.000000000 ～ DT#2106-02-06-23:59:59.999999999 （1970 年 1 月 1 日 0 点 0 分 0.000000000 秒～ 2106 年 2 月 6 日 23 点 59 分 59.999999999 秒）	开头加 'DT#''dt#'，以 ' 年 - 月 - 日 - 时 : 分 : 秒 ' 记述。 （例） dt#1994-09-23-12:16:28.12
字符串型	STRING	（半角字符数 +1）×8 位*3	1 字节	字符编码：UTF-8。 0 ～ 1986 字节 （半角英文数字， 0 ～ 1985 字符、日文约 0 ～ 661 字符）。*4 默认大小为 256 字节。	用 “” （半角单引号）括起来记述。 （例） 'OMRON'　'PLC'

*1. 持续时间型和整数型的转换请使用 NanoSecToTime 指令或 TimeToNanoSec 指令。指令的详细规格请参考 📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。
*2. 比较指令中变量的比较以 ns 为单位进行。变更比较的单位时，请使用 TruncTime 指令 / TruncDt 指令 / TruncTod 指令。指令的详细规格请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。
*3. 字符串末尾添加 NULL （1 字节）字符。因此，请确保要处理的最大字符数 +1 字符的存储空间。例如，半角英文和数字的最大字符数为 10 时，请定义为 11 字符 （11 字节）的 STRING 型变量。 STRING[11]
*4. 记载制表符或换行代码等时，记载以 “$” （半角美元符号）为前缀的转义字符。转义字符请参考📖「字符串型」（P.6-82）。

使用注意事项

所有变量需要的存储容量与各变量的数据大小 （容量值）总和不一致，敬请注意。
因为数据配置到存储器的开头位置将自动配置到各数据类型的 “ 校准 （临界值） ” 倍数的位置，因此各数据类型间可能出现空白。
例如，即使数据类型相同，因数据类型的顺序不同，总占用大小也可能不同。

通过结构体型变量等与其他设备进行数据交换时，各数据类型的 “ 校准 （临界值） ” 必须与数据位置一致，需要注意。
详情请参考📖「A-8 变量的存储器确保方法」（P.A-140）。

参考

- 位列数据型（BYTE、WORD、DWORD、LWORD）不可进行大小比较。进行大小比较时，请使用 WORD_TO_UINT 指令等数据类型转换指令，将变量的数据类型转换为整数。

例）
BCD_data : WORD
IF WORD_BCD_TO _UINT （BCD_data） > UINT#1234 THEN

- 整数的数据类型（SINT、INT、DINT、LINT、USINT、UINT、UDINT、ULINT）不可进行逻辑运算。进行逻辑运算时，请使用 INT_TO_WORD 指令等数据类型转换指令，将变量的数据类型转换为位列。

例）
以下示例程序中，对 INT 型变量 a 的值为奇数时，在 a 上加 1。

```
IF (INT_TO_WORD （a）AND WORD#16#0001) = WORD#16#0001 THEN
   a = a+1;
END_IF;
```

● 派生数据型的种类

派生数据型是指由用户定义规格的数据类型。将登录到 Sysmac Studio 的 " 数据类型 " 视图中。派生数据型中包括以下内容。

种类	内容
结构体型	将多个数据类型按层级整合为一个的数据类型。
联合体型	同一数据可根据不同情况作为不同的数据类型处理的数据类型。
列举型	将事先准备的名称列表中的一个作为值的数据类型。

详情请参考后述的 📖「6-3-6 派生数据型」（P.6-41）。

● 指定数据类型

可对各数据类型进行以下排列指定、范围指定。

种类	内容
排列指定	汇总相同数据类型的要素，从开头起使用编号 （后缀）指定。 可指定为基本数据型及派生数据型。
范围指定	明确表示只能提取预先设定的范围内的数值。 可指定为基本数据型的整数型。

详情请参考后述的 📖「6-3-7 数据类型的排列 / 范围指定」（P.6-51）。

参考

数据类型除了基本数据型和派生数据型外，还有 POU 实例型。 POU 实例型为 FB 实例的数据类型。要生成 FB 的实例，需要将实例名称作为变量、将 FB 定义名称作为数据类型登录到变量表中。

数据类型的使用限制

各 POU 中不可使用的数据类型如下所示。

POU 种类	变量种类	不可使用的数据类型	
		基本数据型	派生数据型
程序	内部变量	无	
	全局变量	无	
FUN	输入变量、输出变量、输入输出变量	无	联合体型
	内部变量	无	
	返回值	无	结构体型、联合体型
FB	输入变量、输出变量、输入输出变量	无	联合体型
	内部变量	无	

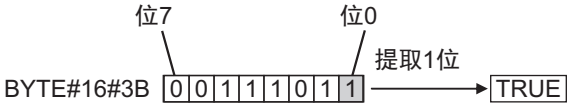
位列型、实型数据、字符串型的数据格式

下面介绍位列型、实型数据、字符串型的数据格式。

● 位列型的数据格式

位列型变量的位位置中，最低位为 “ 位 0”。

此外，位值以 1 或 0 表述，但用 BOOL 型变量表述 1 位的值时， 1 表示 TRUE， 0 表示 FALSE。



● 实型数据 （REAL 型、 LREAL 型）

实型数据 （REAL 型、 LREAL 型）是指拥有实实数型数据格式的数据类型。

下面介绍其表述方法和运算处理。

数据大小

REAL 型的数据大小为 32 位， LREAL 型的数据大小为 64 位。

数据格式

浮动小数点格式是指用符号、指数、尾数表示实数的格式。

用以下公式表示某实数时， s 的值为符号， e 的值为指数， f 的值为尾数。

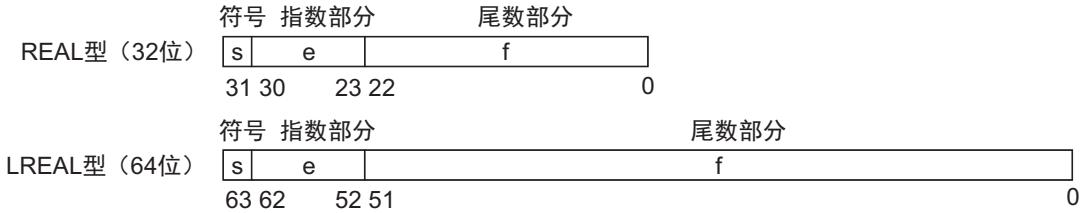
- REAL 型

  数值 = （-1）$^{s}2^{e-127}$ （1+f × 2^{-23}）

- LREAL 型

  数值＝ （-1）$^{s}2^{e-1023}$ （1+f × 2^{-52}）

浮动小数点格式的数据格式符合 IEEE754 标准。分别如下所示。



（例） REAL 型，表示 -86.625 时

1 **由于为负数，所以 s=1。**

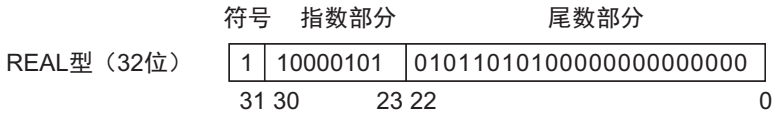
2 **若用 2 进制数表示 86.625，则为 1010110.101。**

3 **若将上述值进行标准化，则变为 1.010110101×2^{6}。**

4 **根据上述公式， e-127=6， e=133 （2 进制数为 10000101）。**

5 **1.010110101的小数点以下为010110101。在此基础上将尾数部23位的不足部分用0填充，则为f。因此， f=01011010100000000000000。**

所以， -86.625 可以如下图表示。

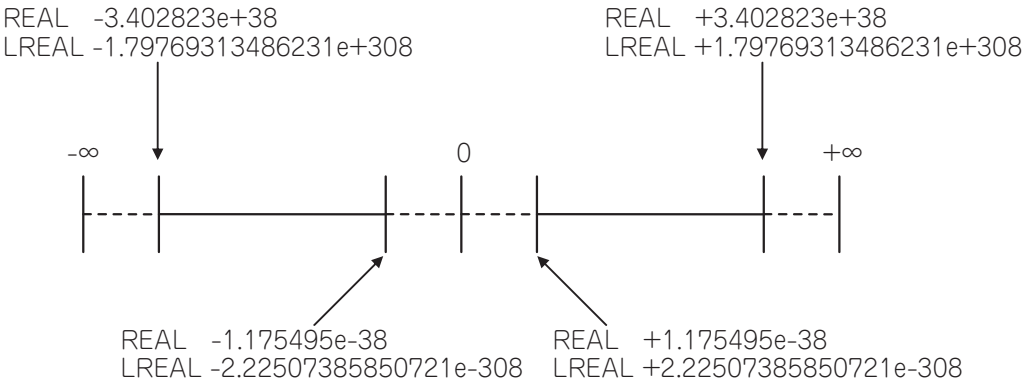


有效范围

REAL 型、 LREAL 型的有效范围如下表所示。

0 附近存在无法表示的值的范围。

数据类型	- ∞	负数	0	正数	+ ∞
REAL	- ∞	-3.402823e+38 ～ -1.175495e-38	0	+1.175495e-38 ～ +3.402823e+38	+ ∞
LREAL	- ∞	-1.79769313486231e+308 ～ -2.22507385850721e-308	0	+2.22507385850721e-308 ～ +1.79769313486231e+308	+ ∞



特殊值

+ ∞、 - ∞、 +0、 -0、非数值称为特殊值。

非数值是指无法视为浮动小数点数的数据，不作为数值处理。

+0 和 -0 在数学上都是 0 的意思，但运算处理内容不同。详情信息后面再介绍。

特殊值的符号 s、指数 e、尾数 f 的值如下表所示。

数据类型	特殊值	符号 s	指数 e	尾数 f
REAL	+ ∞	0	255	0
	- ∞	1	255	0
	+0	0	0	0
	-0	1	0	0
	非数值	-	255	0 以外

数据类型	特殊值	符号 s	指数 e	尾数 f
LREAL	+ ∞	0	2047	0
	- ∞	1	2047	0
	+0	0	0	0
	-0	1	0	0
	非数值	-	2047	0 以外

非标准数

浮动小数点格式无法表示 0 附近（绝对值非常小）的值。为了扩大 0 附近的有效范围而设计了非标准数。与正常数据格式表示的值 （称为标准数）相比，非标准数可以表示绝对值较小的值。

指数 e ＝ 0 且尾数 f ≠ 0 的数值将视为非标准数，值的表示如下。

- REAL 型

  数值 = （-1）$^{s}2^{-126}$ （f × 2^{-23}）
- LREAL 型

  数值＝ （-1）$^{s}2^{-1022}$ （f × 2^{-52}）

（例）REAL 型，表示 0.75 × 2^{-127} 时

1 **由于为正数，所以 s=0。**

2 **若用 2 进制数表示 0.75，则为 0.11。**

3 **（0.11）$_2$×2^{-127}=2^{-126} （f×2^{-23}），所以 f= （0.11）$_2$×2^{22}。**

4 **根据以上公式， f=01100000000000000000000。**

所以， 0.75 × 2^{-127} 可以如下图表示。

	符号	指数部分	尾数部分
REAL型（32位）	0	00000000	01100000000000000000000
	31	30 23	22 0

非标准数比标准数的有效位数少。因此，标准数的运算结果为非标准数时，或在运算过程中结果变为非标准数时，运算结果的有效位数可能小于标准数的有效位数。

运算处理

浮动小数点格式为近似值表现，因此与真值存在误差。此外，值的有效范围也有限制。因此，运算时，需要进行取整、溢出、下溢的处理。

取整

真值超出尾数的有效位数时，按照以下规格取整数值。

- 在浮动小数点格式可表现的值中，将最接近真值的值作为运算结果。
- 在浮动小数点格式可表现的值中，最接近真值的值有 2 个时（真值为 2 个近似值的中间值时），将尾数最低位的位值有 0 的值作为运算结果。

使用注意事项

验证 2 个值是否相等时，对与真值的误差的考虑

实数以浮动小数点格式表现，因此与真值存在误差。验证 2 个值是否相等时，受这个误差影响，可能无法得到预期的结果。

例如，执行 0.1+0.2 和 0.3 的比较 boolv := (0.1 + 0.2 = 0.3); 时，BOOL 型变量 boolv 的值不变为 TRUE，而是 FALSE。

为了避免这样的情况，验证 2 个实数的值是否相等时，请勿使用 EQ、 =、 NE、 <>。请使用大小比较指令，验证两者差的绝对值是否在足够小的容许范围内。

例如，验证 REAL 型变量 real_a 和 real_b 的和是否等于 0.3 时，请如下记述。若 boolv 的值为 TRUE，判断为两者相等。

```
将 boolv := (ABS((real_a + real_b) - 0.3) < 0.000001);   // 0.000001 以内的误差
                                                         // 视为容许范围时
```

溢出、下溢

真值的绝对值大于浮动小数点格式可表现的值时，称为溢出。

反之，小于浮动小数点格式可表现的值时，称为下溢。

- 溢出时，如果真值的符号为正，则运算结果为 + ∞。如果真值的符号为负，则运算结果为 - ∞。
- 下溢时，如果真值的符号为正，则运算结果为 +0。如果真值的符号为负，则运算结果为 -0。

特殊值的运算

特殊值 （+ ∞、 - ∞、 +0、 -0、非数值）的运算按以下规则进行。

- + ∞和 - ∞的相加结果为非数值。
- 符号相同的无限大值的相减结果为非数值。
- +0 和无限大、 -0 和无限大的相乘结果为非数值。
- +0 的相除结果、 -0 的相除结果、无限大的相除结果为非数值。
- +0 和 -0 的相加结果为 +0。
- +0 的相减结果、 -0 的相减结果为 +0。
- 包含非数值的四则运算结果为非数值。
- 比较指令 （Cmp 指令等）中， +0 和 -0 判断为相等。
- 比较指令中，如果比较对象中包含非数值，将始终判断为不相等。

● 字符串型的数据格式

字符串型变量的末尾附加 NULL 字符 （字符编码 BYTE#16#00），以表示结尾。

数据类型的转换

使用了不同数据类型的变量时，可能自动发生数据类型的转换。此外，用户还可使用类型转换指令来转换。

什么是数据类型的转换

所有的变量都有 “ 数据类型 ”，因此在程序中需要按照数据类型进行操作。

例如，作为代入式，左边和邮编的数据类型要相同。但是，实际情况下，为了实现功能，可能需要代入不同数据类型的变量。

例）

var3 := var1; ———————— 代入到不同的类型中

var1为INT型变量

var3为REAL型变量

此时，需要将 var1 的数据转换为适合 var3 的数据类型。这样的转换成为 “ 数据类型转换 ”，或简称为 “ 类型转换 ”。

● 发生数据类型转换时

数据类型的转换在以下 2 种情况下进行。

(1) 使用数据类型转换指令，由用户明示转换

(2) 通过代入 / 指令自动转换

- ST 语言的代入
- 梯形图语言中的连接

参考

整数型和 TIME 型的转换请使用时间 / 时刻指令的 NanoSecToTime 指令和 TimeToNanoSec 指令。详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

6-3-6 派生数据型

派生数据型为根据基本数据型对构成进行定义的数据类型。
派生数据型中包括以下内容。

- 结构体型
- 联合体型
- 列举型

创建派生数据型时，数据类型名称的字符数等限制请参考「6-3-12 变量名称等程序相关名称的限制」（P.6-77）。

参考

NJ/NX 系列中，备有以下 3 种类型作为系统定义的派生数据型。

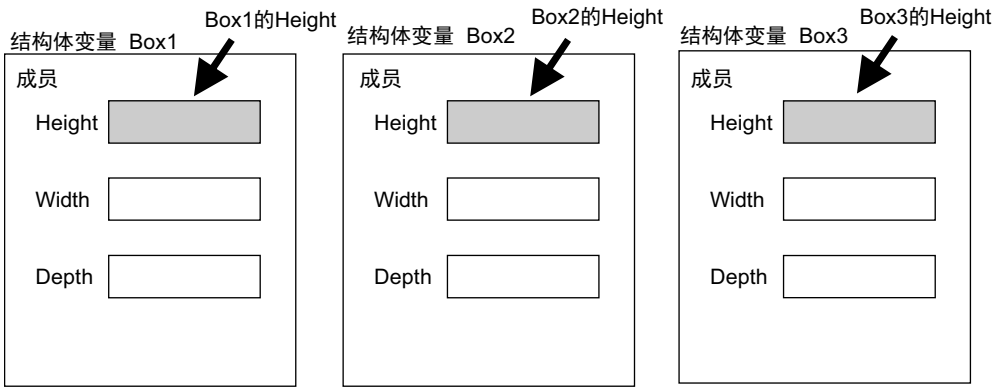
- 系统定义变量的结构体型
- 指令的输入 / 输出 / 输入输出变量使用的结构体型等派生数据型
- 高功能单元的扩展存储用结构体型 （但在单元编辑器上登录后可使用）

结构体型

结构体型是指将数据类型相同或不同的多个数据汇总为 1 个进行处理的派生数据型。
将数据设为结构体型，可方便地进行数据的登录、变更等管理。

例如，定义名为 “Box” 的结构体型时，将 “Width”、“Height”、“Depth” 这 3 个数据作为成员组合并定义。使用该结构体型数据类型，登录名为 “Box1” 的变量后，可在程序中如 “Box1． Width”、“Box1． Height” 所示，用 “．（句点）” 按层级指定数据。

需要其他方框中的新数据时，按照与 “Box1” 相同的要领，将 “Box2” 登录到变量表中。



在所用指令的变量中使用结构体型时，需要在输入、输出或输入输出参数中选择该结构体型，然后登录变量。
例）通信指令

● 结构体型变量的表现

各成员的指定

构成结构体型的数据称为 “ 成员 ”。
结构体型的成员在表示整个结构体型的变量名称后面加半角句点作为成员名称。
此外，结构体型的成员本身为结构体型亦可。

例） abc.x 结构体型变量 abc 的成员 x
abc.Order.z 结构体型变量 abc 的成员 （结构体型） Order 的成员 z

所有成员 （整个结构体）的指定

指构成结构体的所有成员，称为 “ 整个结构体 ”。
整个结构体仅用结构体的变量名称表示。
上述示例中，以 “abc” 表示。

● 结构体型的创建方法

1 在数据类型表中，创建结构体型的数据类型。

指定数据类型名称、成员及数据类型。

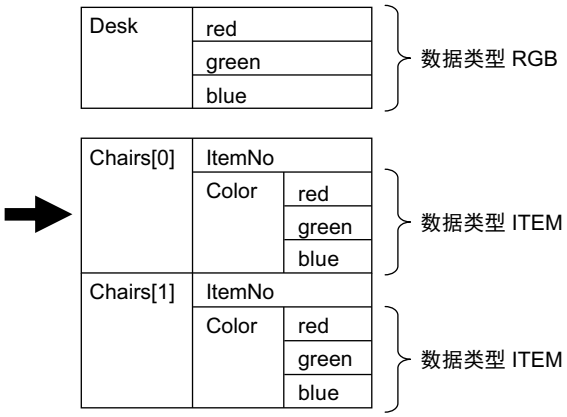
数据类型表

名称	成员	数据类型
RGB	red	INT
	green	INT
	blue	INT
ITEM	ItemNo	INT
	Color	RGB

2 在变量表中，指定上述结构体型数据类型、成员名称作为数据类型，然后登录变量。

变量表

变量名称	数据类型
Desk	RGB
Chairs	ARRAY[0..1]OF ITEM



● 结构体型的规格

结构体型的规格如下所示。

项目	规格
结构体型名称	字母不区分大小写。 禁用字符和字符数限制与变量名称相同。
成员的数据类型	请参考后述 “ 结构体成员的数据类型 ” 中的表。
成员的属性	成员名称 注释
成员数	1 ～ 2048
结构体型的层级深度	最多 8 层 （但是，包括成员名称在内变量名称应在 511 字节以内）
1 个结构体变量的最大大小	NX701- □□□□ : 8MB NJ501- □□□□ : 4MB NJ301- □□□□ : 2MB NJ101- □□□□ : 2MB

结构体型成员的数据类型

分类	数据类型	使用
基本数据型	布尔型、位列型、整数型、实数型、持续时间型、日期型、时刻型、日期时刻型、字符串型	可以
	布尔型、位列型、整数型、实数型、持续时间型、日期型、时刻型、日期时刻型、字符串型的排列指定	可以
派生数据型	结构体型 （*）、联合体型、列举型 * 递归或循环的数据不可 （程序检查时将发生错误）。	可以
	结构体型、联合体型、列举型的排列指定	可以
POU 实例型		不可

● 排列指定和结构体型

还可设定为以结构体型为要素的排列指定。此外，还可设定为以排列指定为成员的结构体型。

● 结构体成员的偏置指定

可通过成员的偏置指定，对各结构体型任意设定成员的存储器配置。这样与 CJ 系列 CPU 单元或外部设备进行标签数据链接时，可以将两者的结构体型成员的存储器配置统一起来。

结构体型成员的偏置种类可从 “NJ”、“CJ”、“ 任意 ” 中选择。选择了 “NJ” 时，将自动向 NJ/NX 系列配置最佳的存储器。关于 NJ/NX 系列的存储器配置，请参考📖请参考 「A-8 变量的存储器确保方法」（P.A-140）。关于 CJ 系列 CPU 单元和标签数据链接的示例，请参考📖「A-8-2 应注意的案例」（P.A-149）。

偏置种类的含义如下所示。

偏置种类	含义
NJ	自动向 NJ/NX 系列配置最佳的存储器，以最高速度动作。
CJ	自动变为与 CJ 系列相同的存储器配置。 可与 CJ 系列 CPU 单元的存储器配置统一。
任意	可任意设定各成员的存储器配置。 与 CJ 系列以外的外部设备统一存储器配置时使用。

版本相关信息

可否使用成员的偏置指定，根据 CPU 单元的单元版本与 Sysmac Studio 版本的组合，发生如下变化。

CPU 单元的单元版本	Sysmac Studio 的版本		
	Ver.1.01 以下	Ver.1.02	Ver.1.03 以上
Ver.1.01 以上	不可	可*1	可
Ver.1.00	不可	不可	不可

*1. 无偏置类型的选择功能。任意设定成员的偏置。

偏置的设定

偏置种类设定为 “ 任意 ” 时，可对各结构体型的成员设定存储器配置的偏置。偏置包括 “ 字节偏置 ”、“ 位偏置 ”。偏置种类为 “NJ”、 “CJ” 时，将自动决定存储器配置，因此无需设定偏置。

偏置的含义如下所示。

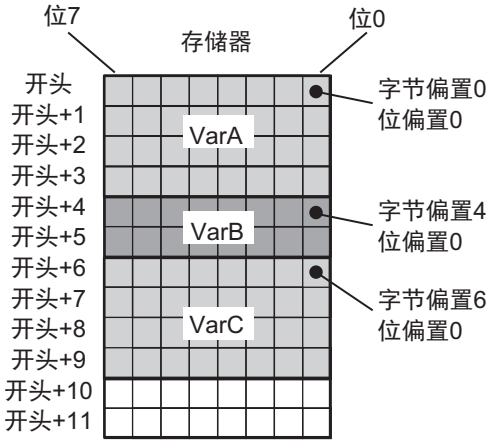
偏置	含义	单位	值的范围
字节偏置	设定各成员从结构体型开头起的偏置。 可使用基本数据型和派生数据型的所有数据类型。	字节	*1
位偏置	设定各成员从字节偏置中设定的字节位置开头起的偏置。	位	0 ～ 63

*1. 使用 NX 系列 CPU 单元时为 0 ～ 8191，使用 NJ 系列 CPU 单元时为 0 ～ 1023。

例）

表示结构体型编辑器的设定为以下情况时的存储器配置。

名称	数据类型	偏置种类	字节偏置	位偏置
StrA	STRUCT	任意		
VarA	DINT		0	0
VarB	INT		4	0
VarC	DINT		6	0



偏置的设定可否

偏置种类设定为 " 任意 " 时，有些数据类型中偏置也可能是固定值。偏置的设定可否如下所示。

分类	数据类型	字节偏置	位偏置
布尔型	BOOL	可设定	可设定
位列型	BYTE,WORD,DWORD,LWORD	可设定	固定
整数型	SINT,INT,DINT,LINT,USINT,UINT,UDINT,ULINT	可设定	固定
实型数据	REAL,LREAL	可设定	固定
持续时间型	TIME	可设定	固定
日期型	DATE	可设定	固定
时刻型	TIME_OF_DAY	可设定	固定
日期时刻型	DATE_AND_TIME	可设定	固定
字符串型	STRING	可设定	固定
排列指定		可设定	仅当要素的数据类型为 BOOL 时可设定
结构体型		可设定	固定
联合体型		可设定	固定
列举型		可设定	固定
POU 实例型		固定	固定

成员的偏置指定限制

进行成员的偏置指定时，有以下限制。结构体型的成员中包括有成员偏置指定的结构体型时，结构体型也有相同的限制。

- 偏置种类设定为 " 任意 " 时，需要对该结构体型的所有成员设定偏置。
- 指定了偏置的结构体型成员不可将初始值设定为任意值。初始值为各数据类型中规定的默认值。详情请参考📖「初始值指定为空白时」（P.6-61）。
- 结构体型的存储大小由各成员的大小、数据类型的校准 （临界值）和存储器配置决定。

成员的偏置指定相关的错误

成员的偏置指定相关的错误如下。

错误名称	含义	偏置种类	处理方法
超出偏置范围错误	偏置中设定的值超出范围。	任意	请变更为合适的偏置值。
偏置未设定错误	有未设定偏置的成员。	任意	请为所有成员设定偏置。
存储器配置重复错误	成员的存储器配置重复。	任意	请变更为合适的偏置值。
初始值设定错误	创建变量表时，欲对指定了偏置的结构体型成员设定初始值。	CJ、任意	请勿设定初始值。

● 以整个结构体为参数的指令

指令中包括将结构体变量作为参数传递的指令。此时，应只将结构体整体指定为输入参数。
例）向 MOVE 指令只传递 1 个结构体型成员时和传递整个结构体变量时

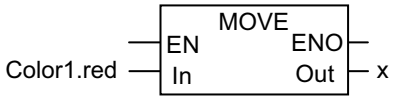
数据类型表

名称	成员	数据类型
RGB	red	UINT
	green	UINT
	blue	UINT

变量表

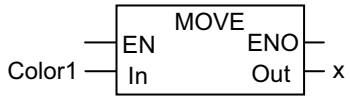
变量名称	数据类型
Color1	RGB

■仅指定结构体型变量的1个成员



仅将RGB的结构体型变量Color1的成员red传送到x

■指定整个结构体型变量



将整个RGB的结构体型变量Color1传送到x

向特定指令的系统定义结构体型输入变量传递值的方法

有些特定指令的输入变量会使用预先准备的结构体型变量。
例）串行通信指令的输入变量 “Port” （目标端口指定）为数据类型名称 “_sPORT” 的结构体。
使用特定指令时，请按以下步骤创建结构体型用户定义变量，指定输入参数。

1 **Sysmac Studio 中，已事先登录指令用的系统定义数据类型。**
在 Sysmac Studio 上选择该系统定义数据类型，将用户定义的结构体型变量登录到变量表中。

2 **通过用户程序或利用初始值设定该结构体型变量中各成员的值。**

3 **将该结构体型变量指定为指令的输入参数。**

联合体型

联合体型是指可通过多个不同数据类型对同一数据进行访问的派生数据型。
指定可访问的多个不同数据类型，如要素数 16 的 BOOL 型排列指定或 16 个 BOOL 型、 WORD 型。

● 联合体型的表现

各成员的指定

定义联合体型时，需要对可访问的各数据类型进行命名。这种名称称为 “ 成员 ”。

联合体型的成员在表示整个联合体型的变量名称后面加半角句点作为成员名称。

例）
如下定义 “My Union” 作为联合体型的数据类型

数据类型定义：

名称	成员	数据类型
My Union	data	WORD
	bit	ARRAY [0..15] OF BOOL

变量表：

变量名称	数据类型
Output	My Union

Output.bit[0]： 将联合体型 Output 视为 16 位 BOOL 型排列指定的变量时，要素编号 0，即第 0 位的值
Output.data： 将联合体型 Output 视为 1 个 WORD 型变量时的值

所有成员 （整个联合体）的指定

指构成联合体的所有成员，称为 “ 整个联合体 ”。
整个联合体仅用联合体的变量名称表示。
上述示例中，以 “Output” 表示。

● 联合体型的创建方法

1 **在联合体型表中，创建联合体型的数据类型。**
将数据类型名称、多个不同的数据类型指定为其成员。

2 **在变量表中，指定上述联合体型作为数据类型，然后登录变量。**

例）
将 OUT16_ACCESS 定义为联合体型的数据类型。
将该联合体型的成员作为要素数 16 的 BOOL 型排列指定及 WORD 型。
登录使用了数据类型 OUT16_ACCESS 的变量 Output。
变量 Output 可对 16 位中的任意 1 位进行 BOOL 型值的读写，也可对 WORD 型的值进行读写。

数据类型定义

名称	成员	数据类型
OUT16ACCESS	BoolData	ARRAY[0..15] of BOOL
	ByteData	ARRAY[0..1] of BYTE
	WordData	WORD

变量名称	数据类型
Output	OUT16ACCESS

BoolData [15]	· · ·	BoolData [8]	BoolData [7]	· · ·	BoolData [0]
ByteData [1]			ByteData [0]		
WordData					

```
Output.WordData := WORD#16#1234;
```

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BoolData	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
ByteData	16#12								16#34							
WordData	16#1234															

```
Output.BoolData[11] :=TRUE;
```

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BoolData	*	*	*	*	1	*	*	*	*	*	*	*	*	*	*	*
ByteData	ByteData[1]的第4位为TRUE								无变化							
WordData	WordData的第11位为TRUE															

● 联合体型的规格

项目	规格
可指定为成员的数据类型	请参考后述 “ 可指定为联合体型成员的数据类型 ” 中的表。
成员数	最大 4
初始值设定	不可。必定变为 0。

可指定为联合体型成员的数据类型

分类	数据类型	使用
基本数据型	BOOL 型、位列型	可以
	BOOL 型、位列型的排列指定	可以
	其他基本数据型	不可
派生数据型	结构体型、联合体型、列举型的排列指定	不可
POU 实例型	—	不可

● 限制事项

- 联合体的初始值固定为 0。
- 不可传送整个联合体。
- POU 的参数不可指定为整个联合体。

列举型 （ENUM）

列举型是指以名为 “ 列举值 ” 的字符表示变量值的派生数据型。
需要预先将该变量可获得的值设定为列举值 （字符）。
若使用列举型，变量值的含义会更容易理解。

● 列举型的表现

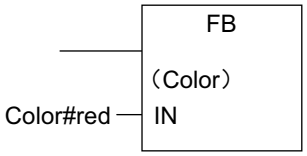
定义列举型时，将变量可获得的值命名为列举值。

● 列举型的创建方法

1 在列举型表中，创建列举型的数据类型。
设定与可获得的列举值对应的值。

2 在变量表中，指定上述列举型作为数据类型，然后登录变量。

例）
创建 Color 作为列举型的数据类型。可获得的列举值为 “red”、“yellow”、“green” 三个。列举值对应的数值为 red=0、 yellow=1、 green=2。
登录使用了数据类型 Color 的变量 DiscColor。变量 DiscColor 可获得的值为 red(0)、 yellow(1)、green(2) 中的一个。



列举型列表

数据类型	
Color	ENUM
列举值	**值**
red	0
yellow	1
green	2

变量表

变量名称	数据类型
DiscColor	Color

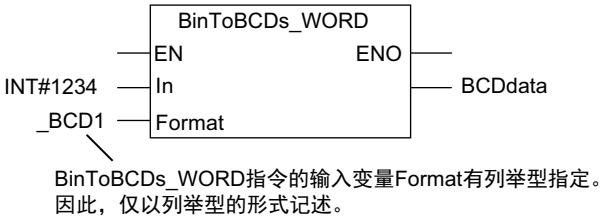
● 列举型的规格

项目	规格
列举值	只含半角英文和数字，字母不区分大小写。禁用字符与变量名称相同。 若重复指定了相同的列举值，链接时将发生异常。 指定为与用户程序中的变量相同的列举值时，或指定为与其他列举型中存在的列举值相同的名称时，链接时将发生异常。
值	有效范围为 -2147483648 ～ +2147483647 的整数值。 不是连续值亦可。 若重复指定了相同的值，链接时将发生异常。 注 . 列举型变量不可进行大小比较。只能比较一致 / 不一致。
列举值的数量	1 ～ 2048 个

● 将列举值作为 FB、 FUN 参数时的记述

将列举值作为 FB、 FUN 的参数传递时，有以下 2 种记述方法。

仅列举值的记述
如果是参数指定为列举型的 FB、 FUN，只能记述列举值。
例）向 BCDToBins_WORD 指令传递列举值时

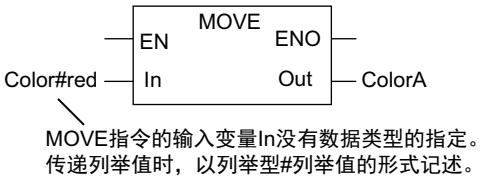


列举型 # 列举值的记述

如果是参数的数据类型没有特别指定的 FB、 FUN，只有列举值的记述为无效。链接时将发生异常。

为了明确表示参数为列举值，以列举型 # 列举值的形式记述。

例）向 MOVE 指令传递列举值时



参考

如果是参数指定为列举型的 FB、 FUN，以列举型 # 列举值的形式记述参数亦可。
因此，如果是上述BinToBCDs_WORD指令，传递到Format变量的参数记述为_eBCD_FORMAT#_BCD1 亦可。

● 值的检查

通过指令执行，将值写入列举型变量时，即使该值不是定义为列举型的列举值，也不会发生错误。因此，需要检查值是否是定义为列举型的列举值时，请在用户程序中进行值的检查处理。

6-3-7 数据类型的排列 / 范围指定

各数据类型的变量中可指定以下属性。

- 排列指定
- 范围指定

排列指定 （ARRAY[]OF）

排列指定是针对将相同属性的数据汇总起来，作为一个块处理的数据类型的指定。可对基本数据型、派生数据型进行指定。
如运动控制时的坐标值等，处理多个相同概念的数据时非常方便。

● 排列指定的表现

各要素的指定

构成排列指定的各数据称为 " 要素 "。
排列指定的要素通过在表示整个排列指定的排列指定变量名称中，从前往后添加后缀（要素编号）来表示。
后缀用半角方括号 [] 括起来。后缀为常数或变量均可。此外， ST 语言时，后缀还可用公式表示。

例）

变量表

变量名称	数据类型
Mem	ARRAY[0..99] OF INT

x:=10;
Mem[x]　　排列指定的变量 Mem 的要素编号 x（变量 x 的值为 10，所以要素编号为 10）

变量表

变量名称	数据类型
Data	ARRAY[0..99] OF INT

x:=10;
y:=20;
Data[x+y]　排列指定的变量 Data 的要素编号 （x+y）（变量 x 的值为 10、变量 y 的值为 20，所以要素编号为 30）

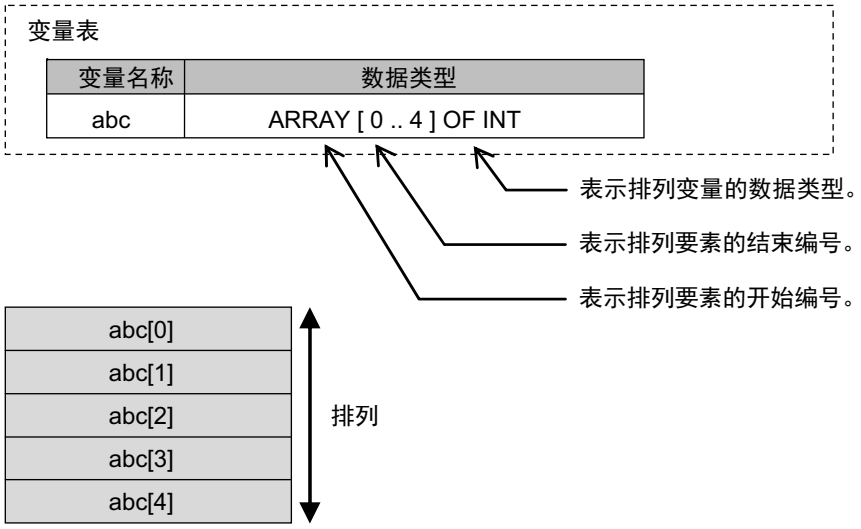
所有要素 （排列整体）的指定

指构成一个排列指定的所有要素，称为 " 整个排列 "。
整个排列仅用排列指定的变量名称表示。
上述示例中，以 "Mem"、 "Data" 表示。

● 排列指定的创建方法

1 **在变量表的数据类型栏中输入 “A”，从候选的数据类型名称中选择 “ARRAY[?..?] OF ?”。**

2 **在[?..?]左侧的?处输入要素的开始编号，右侧的?处输入结束编号，OF ?的?处输入数据类型，然后登录变量。**



● 排列指定的规格

项目	规格
排列指定的变量最大要素数	65,535
要素编号	0 ～ 65535 要素的开始编号不是 0 亦可。
后缀	常数：0 ～ 65535 的整数值 变量： 分类 / 数据类型 / 使用 基本数据型 / 整数型 / SINT、INT、 DINT、USINT、UINT 、 UDINT / 可以 基本数据型 / 整数型 / LINT、 ULINT / 不可 基本数据型 / 布尔型、位列型、实数型、持续时间型、日期型、时刻型、日期时刻型、字符串型 / 不可 派生数据型 / 结构体型、联合体型、列举型 / 不可 POU 实例型 / / 不可 运算式：仅在 ST 语言时可指定运算式　　例） y := x[a+b];
1 个排列变量的最大大小	NX701- □□□□ : 8MB NJ501- □□□□ : 4MB NJ301- □□□□ : 2MB NJ101- □□□□ : 2MB

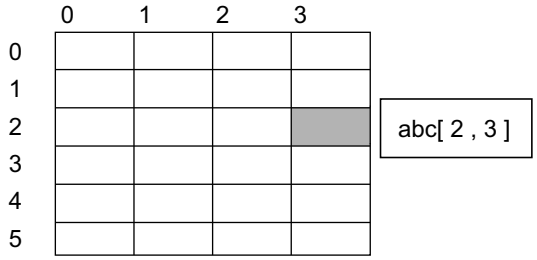
● 排列指定的维数

1 维排列指定可以视为将要素排列在 1 列的 1 维数据。
同样，也可设定 2 维或 3 维排列指定。
此时，排列指定要素中将标示与排列指定的变量名称相同维数的后缀表示。维数的最大值为 3。

2 维排列指定

变量表

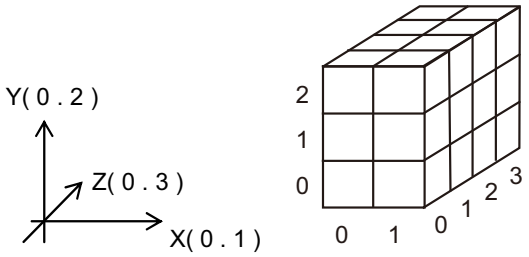
变量名称	数据类型
abc	ARRAY [0..5 , 0..3] OF INT



3 维排列指定

变量表

变量名称	数据类型
ITEM	ARRAY [0..1, 0..2, 0..3] OF INT



● 排列指定和结构体型

还可设定为以结构体型为要素的排列指定。
此外，还可设定为以排列指定为成员的结构体型。

以结构体型为要素的排列指定

数据类型表

数据类型	成员	成员
Str	x y	INT DINT

变量表

变量名称	数据类型
abc	ARRAY [1..4] OF Str

变量

abc[1].x
abc[1].y
:
abc[4].x
abc[4].y

以排列指定为成员的结构体型

数据类型表

数据类型	成员	数据类型
Str	x	ARRAY [0..1] OF INT
	y	DINT

变量表

变量名称	数据类型
abc	ARRAY [1..4] OF Str

变量

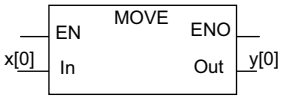
abc[1].x[0]
abc[1].x[1]
abc[1].y
:
abc[4].x[0]
abc[4].x[1]
abc[4].y

● 以排列指定为参数的指令

指令中包括将整个排列作为参数传递的指令。此时，应只将排列指定的变量名称指定为输入参数。

例）向 MOVE 指令只传递 1 个排列指定要素时，传递整个排列

■指定排列的1个要素

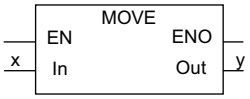


将INT型变量x[0]传送到y[0]

变量表

变量名称	数据类型
x	ARRAY [0 .. 199]OF INT
y	ARRAY [0 .. 199]OF INT

■指定整个排列



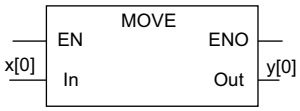
将INT型变量x[0..199]传送到y[0..199]

· 限制事项：
指定整个排列时，包括要素编号的范围在内，均需为同一数据类型。

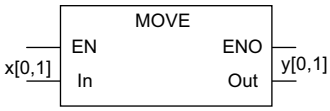
参考

不可将多维排列的部分排列指定为参数。

×



×



变量表

变量名称	数据类型
x	ARRAY [0..9 , 0..9 , 0..9]OF INT
y	ARRAY [0..9 , 0..9 , 0..9]OF INT

● 排列指定的保护功能

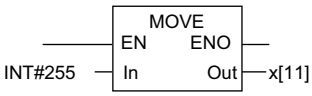
若要超出排列的要素数进行访问，将发生以下错误。

后缀为常数时

在 Sysmac Studio 中输入时，以及程序检查时将显示错误。

变量表

变量名称	数据类型
x	ARRAY [1..10] OF INT

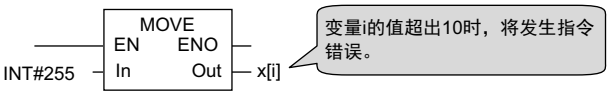


后缀为变量时

在执行指令后，将输出参数代入输出变量时，CPU 单元进行要素数超出检查。指定的后缀变量超出排列变量的要素数时，将视为指令错误。此时，指令的内部处理已正常结束，因此ENO的值变为TRUE。敬请注意。

变量表

变量名称	数据类型
x	ARRAY [1..10] OF INT



范围指定 （(..)）

范围指定是指针对以下整数型变量，限定其值的范围。

分类	数据类型
整数型	SINT、 INT、 DINT、 LINT、 USINT、 UINT、 UDINT、 ULINT

以下情况下，可对输入值判断是否在容许范围内。

- 指定变量的初始值时
- 通过 CIP 信息通信写入了值时

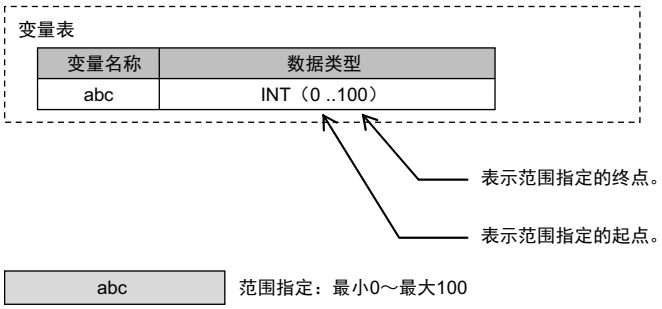
● 范围指定的创建方法

在变量表的数据类型栏中，在数据类型名称后面输入 （起点 .. 终点）。

起点：变量中可保存的最小值

终点：变量中可保存的最大值

例）



● 范围指定的规格

项目	规格
可指定的数据类型	变量型
尝试写入超出范围的值时的动作	

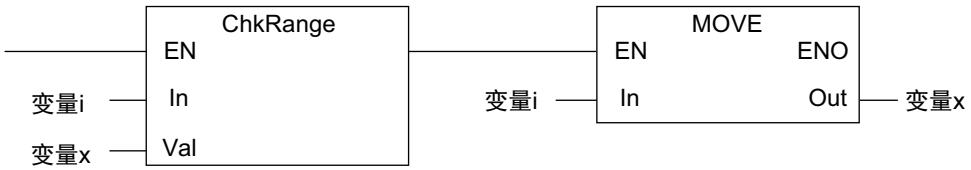
情况		动作	
用户程序		不发生错误，替换值。 执行指令后变量的值发生变化时， CPU 单元不进行范围检查。	
通信	通过 Sysmac Studio 写入、CIP 信息	整数型时	发生指令错误。
		整数型排列变量的要素时	
		整数型的结构体型成员时	
		整数型的整个结构体时	不发生指令错误，替换值。
		整个整数型排列时	
	标签数据链接（内置 EtherNet/IP、EtherNet/IP 单元亦同）	对指定了范围的成员单独进行写入时将发生错误。 对整个结构体 （包含指定了范围的成员）进行写入时，不会发生错误。	
从站 / 单元的 IN 刷新		不发生错误，替换值。	
强制值刷新功能的强制值		不发生错误，替换值。	

使用注意事项

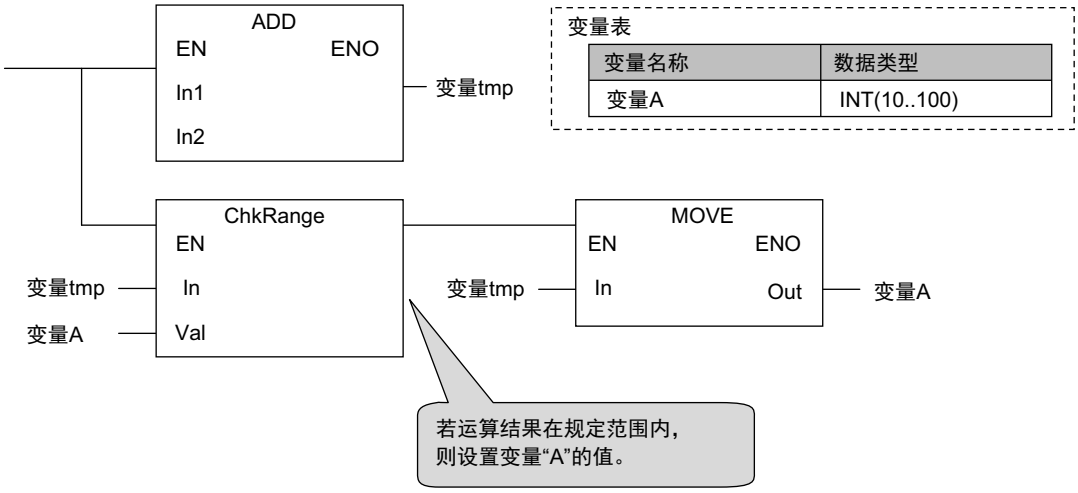
变量的范围指定在通过用户程序执行指令而导致变量的值发生变化前，不会检查。要检查因用户程序执行而发生变化的变量值范围时，请如下所示，使用范围检查指令。

变量表

变量名称	数据类型
变量x	INT(10..100)



此外，设置四则运算结果等时，因无法进行事先检查，所以请如下所示，对四则运算（例：ADD）后值的范围进行检查。



使用范围指定时，请注意初始值。
Sysmac Studio 的初始值列为空白时，初始值为 0。将不含 0 的范围设为有效范围时也为 0。

6-3-8 变量的各属性说明

下面介绍数据型以外的变量各属性。

变量名称

变量名称是指用于识别变量的名称。
在一个 POU 中，不可重复声明相同变量名称的变量。但是，在不同的 POU 中，可以声明相同变量名称的本地变量，并作为不同的变量处理。

变量名称的限制请参考📖「6-3-12 变量名称等程序相关名称的限制」（P.6-77）。

AT （分配目标）指定

本变量用于将变量的分配存储器地址指定为 CJ 单元用存储器。
主要在指定以下高功能单元的分配任意区域时指定。

- 使用 DeviceNet 单元时的固定分配区域
- 使用DeviceNet单元或CompoNet主机单元（通信模式No.8）时，在CX-Integrator上指定的自由分配区域
- 使用高速计数器单元时的扩展存储器
- 使用过程输入输出单元时的扩展存储器
- 使用 EtherNet/IP 单元时的目标节点信息或用户定义状态区域用数据区域

未设定本属性时，变量将自动分配到存储器的任意地址 （变量存储器）中。

使用注意事项

可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

参考

设备变量时，将向 I/O 端口分配，自动在内部进行 AT （分配目标）指定。

● 分配目标区域

可指定为分配目标的区域如下所示。

区域	表现
CIO	0 ～ 6143
WR	W0 ～ 511
HR	H0 ～ 1535
DM	D0 ～ 32767
EM	NJ501- □□□□：E0_0 ～ E18_32767 NJ301- □□□□：E0_0 ～ E3_32767 NJ101- □□□□：E0_0 ～ E3_32767

不同变量数据类型的数据分配如下所示。

变量的数据类型	分配位置
BOOL	能以位为单位指定
BYTE/SINT/USINT	可指定将数据的分配开始位置用指定的 CJ 地址的 bit0 或 bit8 中的哪个配置。 例 1） AT 指定为 D100 的 bit0 时 （%D100） D100：16#**12....1 字节的数据 （12）从 bit0 开始保存。 例 2） AT 指定为 D100 的 bit8 时 （%D100.8） D100：16#12**....1 字节的数据 （12）从 bit8 开始保存。
WORD/INT/UINT	从指定的 CJ 地址的 bit0 开始，根据数据大小依次保存。
DWORD/DINT/UDINT **REAL**	
LWORD/LINT/ULINT **LREAL**	
STRING	可指定将数据的分配开始位置用指定的 CJ 地址的 bit0 或 bit8 中的哪个配置。
TIME **DATE** **TIME_OF_DAY** **DATE_AND_TIME**	从指定的 CJ 地址的 bit0 开始，根据数据大小依次保存

● 可进行 AT 指定的变量种类

以变量为单位指定。排列变量、结构体变量、联合体变量将对整体进行指定。

● AT （分配目标）指定的变量属性

	指定	备注
名称	可	
数据类型	可	
保持	可	不符合 AT （分配目标）指定目标的 CJ 单元用存储器的属性时，将发生错误。
初始值	可	直接使用存储器的值时，设定为 “ 无初始值指定 （None） ”。
常量	可	不可通过指令写入值。
网络公开	可	
边缘	不可	（仅可对功能块的输入变量）

● AT （分配目标）指定的输入和显示

进行 AT （分配目标）指定时，在 Sysmac Studio 的变量表 “ 分配目标 ” 列中输入以下内容。变量表及 I/O 映射的 “ 分配目标 ” 列中将显示以下内容。

变量种类	“ 分配目标 ” 列中的输入和显示	例
字地址中 AT （分配目标）指定的用户定义变量	%[通道地址]	%D100
位地址中 AT （分配目标）指定的用户定义变量	%[通道地址].[位位置]	%W0.00

此外，以下变量已在内部进行 AT （分配目标）指定，在同一 “ 分配目标 ” 列中，将如下显示。

变量种类	“ 分配目标 ” 列中的显示	例
CJ 单元设备变量	IOBus://rack#[装置编号]/ slot#[插槽编号]/ [I/O 端口名称]	基本 I/O 单元时： IOBus://rack#0/slot#1/Ch1_In/Ch1_In00 高功能单元时： IOBus://rack#0/slot#1/PeakHoldCmd/ch1_PeakHoldCmd
EtherCAT 从站设备变量	EtherCAT 从站终端上的 NX 单元时： ECAT://node#[节点地址 .NX 单元编号]/ [I/O 端口名称]	ECAT://node#[10.15]/Input1
	上述以外时： ECAT://node#[节点地址]/ [I/O 端口名称]	ECAT://node#1/Input1
轴变量	MC://_MC_AX[] MC://_MC1_AX[][*1] MC://_MC2_AX[][*1]	MC://_MC_AX[1]
轴组变量	MC://_MC_GRP[] MC://_MC1_GRP[][*1] MC://_MC1_GRP[][*1]	MC://_MC_GRP[1]

*1. 只能在 NX 系列 CPU 单元中可使用的系统定义变量。

使用注意事项

同一地址中可分配多个变量，但考虑到程序的可读性和调试的难度，并不推荐。此时，设定初始值的变量请只设定为 1 个。若将各变量设定为不同的初始值，初始值将变得不确定。

保持

本属性用于指定在以下情况时是否保持变量的值。

- 断电后接通电源时
- 动作模式切换时
- 发生全部停止故障等级的控制器异常时

未设定本属性时，变量的值在上述情况下为初始值。

用于保存生产数量等（CPU 单元的电源断开后仍需保持的数据）的变量可设定为有保持，这样比较方便。

AT （分配目标）指定的变量需要根据分配目标区域统一保持 / 不保持。
（保持区域：HR、 DM、 EM
不保持区域：CIO、 WR）

● 将保持设为有效的必要条件

请在 CPU 本体上安装电池

● 要将初始值反映到保持变量时

传送用户程序时，勾选 “ 初始化保持变量的当前值 ”。

● 有 / 无保持的动作区别

与保持有关的情况以及此时变量中设定的值之间的关系如下所示。

情况		变量的值	
		有保持	无保持
断电后接通电源时		保持	不保持
动作模式切换时			
发生全部停止故障等级的控制器异常时			
用户程序传送时	“ 不初始化保持变量 ” 设定		
	“ 初始化保持变量 ” 设定	不保持	

● 可进行保持指定的变量种类

以变量为单位指定。排列变量、结构体变量、联合体变量将对整体进行指定。

初始值

初始值是指以下情况下设定的变量值。

- 电源接通时
- 切换为运行模式⇔程序模式时
- 设定为传送用户程序时 “ 初始化保持变量 ” 时
- 发生全部停止故障等级的控制器异常时

设定初始值后，可省略对变量进行初始化的程序。例如，可预先将配方等数据设定为初始值。
此外，初始值不设定亦可。

● 可设定初始值的变量种类

根据变量的种类，有可设定和不可设定初始值的变量。以下表示其列表。

变量的种类	初始值设定
全局变量	可
内部变量	
输入变量	
输出变量	
FUN 的返回值	不可
输入输出变量	
外部变量	

● 初始值的有 / 无选择

创建变量时可选择初始值的有 / 无。

有初始值指定

初始值

或
(空白)

初始值
3.14

无初始值指定

初始值
None

各类型的选择可否如下所示。

种类		例	初始值的有 / 无选择
基本数据型变量		aaa	可以
排列指定的变量	排列整体	bbb	可以
	要素	bbb[2]	不可
结构体型变量	结构体整体	ddd	可以
	成员	ddd.xxx	不可
联合体型变量	联合体整体	eee	不可 （初始值固定为 0）
	成员	eee.word	初始值必定为 0
列举型变量		ccc	可以
POU 实例型		instance	不可

参考

部分基本 I/O 单元中，一个 I/O 端口可以有多种访问方法，如位列型、布尔型等。对本 I/O 端口使用初始值时，请将一方设定为 “None （无初始值指定） ”。

● 设定初始值的时间

在设定了初始值的变量中实际代入初始值的时间如下所示。

- 电源接通时
- 动作模式从程序切换为运行、从运行切换为程序时
- 指定 “ 初始化变量 ”，传送了用户程序时
- 发生全部停止故障等级的控制器异常时

● 初始值指定为空白时

初始值指定为空白时，变量的初始值如下所示。

数据类型		初始值
布尔型 / 位列型	BOOL, BYTE, WORD, DWORD, LWORD	0
整数型	SINT, INT, DINT, LINT USINT, UINT, UDINT, ULINT	0
实型数据	REAL, LREAL	0.0
持续时间型、日期型、时刻型、日期时刻型	TIME	T#0S
	DATE	D#1970-01-01
	TIME_OF_DAY	TOD#00:00:00
	DATE_AND_TIME	DT#1970-01-01-00:00:00
字符串型	STRING	" （空字符）

● 排列指定的变量初始值

数据类型	初始值的规格
排列指定	• 可对各要素设定初始值。 • 指定初始值时，需要对所有要素设置值或设为空白。

● 派生数据型的初始值

不是对数据类型本身指定初始值，而是指定各变量的初始值。

数据类型	初始值的规格
结构体型	• 可对各成员设定初始值。 • 指定初始值时，需要对所有要素设置值或设为空白。
联合体型	• 不可指定。始终为 0。
列举型	• 可指定。

● 不应用初始值的变量

如果是以下变量，接通电源时不会应用初始值，而是保持断电前的值。

- 保持属性的变量
- AT （分配目标）指定的变量 （仅限指定保持区域、 DM、 HR/EM 区域时）

使用注意事项

CPU 本体中未安装电池时，上述变量的值也为初始值。

常量

常量是指无法通过指令、ST 语言的运算符或 CIP 信息通信进行值写入的属性。因此，可以防止程序对变量进行值的写入。
如果是这种属性的变量，设置初始值后，可通过指令禁止写入。
若 POU 中存在向有常量属性的变量写入值的指令，链接时将发生异常。

● 动作

POU 中存在向有常量属性的变量写入值的指令或运算符时，动作如下所示。

写入源		尝试写入值时的动作
用户程序		程序检查时将发生错误。 Sysmac Studio 会在链接时进行检查，发生链接异常。
通信	通过 Sysmac Studio 写入	不可
	CIP 信息	发生指令错误。
	标签数据链接	开始标签数据链接时会发生错误，但标签数据链接继续。 但是，带常量属性的变量值不会替换。
从站 / 单元的 IN 刷新		不发生错误，替换值。
强制值刷新		

● 可进行常量指定的单位

以变量为单位指定。排列变量、结构体变量、联合体变量将对整体进行指定。

参考

“ 常量 ” 属性也禁止通过用户程序写入。

网络公开

网络公开属性是指利用 CIP 信息通信或标签数据链接功能，可在外部 （其他控制器或上位计算机）读写变量 （这称为 “ 网络公开 ”）的属性。
未设定本属性时，可通过声明了该变量的控制器来读写变量，不可在外部 （其他控制器或上位计算机）读写。

网络公开的变量称为 “ 网络变量 ”。

● 网络公开的规格

网络公开包括 “ 仅公开 ”、 “ 输入 ”、 “ 输出 ” 指定。各规格如下所示。

网络公开		规格
不公开		无法从外部访问。 无论是否公开，都可通过支持软件访问。
公开	仅公开	可通过 CIP 通信从外部访问的变量。 不可进行标签数据链接。
	输入	可通过 CIP 通信或标签数据链接从外部访问的变量。 标签数据链接时，为数据的输入用 （其他 CPU 单元➞本 CPU 单元）变量。
	输出	可通过 CIP 通信或标签数据链接从外部访问的变量。 标签数据链接时，为数据的输出用 （本 CPU 单元➞其他 CPU 单元）变量。

● 可进行网络公开指定的单位

以变量为单位指定。排列变量、结构体变量、联合体变量将对整体进行指定。

边缘

边缘属性是指 BOOL 型变量的值从 FALSE 上升为 TRUE 时，或从 TRUE 下降为 FALSE 时，向 FB 传递 TRUE 值的属性。
仅可设定为 FB 的 BOOL 型输入变量。

● 用途

想要用 FB 仅接收输入参数的上升沿 / 下降沿时使用。
例如，用于仅在输入参数发生变化时启动 FB 等时。

● 动作

- 若指定为“上升沿”，则仅在与该输入变量连接的输入参数的值从 FALSE 上升为 TRUE 时，输入变量的值变为 TRUE。
- 若指定为“下降沿”，则仅在输入参数从 TRUE 下降为 FALSE 时，输入变量的值变为 TRUE。

指定	输入参数的值	变量的值
上升沿	FALSE → TRUE	TRUE
	其他	FALSE
下降沿	TRUE → FALSE	TRUE
	其他	FALSE
无	—	每次根据输入参数的值变化

6-3-9 与状态变化对应的变量值的变化

接通电源、变更动作模式、下载变量表时、发生全部停止故障等级的控制器异常时、在线编辑时， CPU 单元内部的变量值将如下表所示发生变化。

变量的保持属性	变量种类		电源接通时	动作模式变更时	下载后	
					“初始化保持变量”设定时	“不初始化保持变量”设定时
非保持	用户定义变量		• 有初始值设定：指定的初始值。 • 无初始值设定 (None)：16#00*1			
		AT（分配目标）指定为 CJ 单元用存储器的 CIO/WR 区域	• 有初始值设定：指定的初始值。 • 无初始值设定 (None)：16#00*1		• 有初始值设定：指定的初始值。 • 无初始值设定 (None)：保持值。	
	EtherCAT 从站设备变量		• 有初始值设定：指定的初始值。 • 无初始值设定 (None)：16#00*1			
	CJ 单元设备变量		• 有初始值设定：指定的初始值。 • 无初始值设定 (None)：16#00*1		• 有初始值设定：指定的初始值。 • 无初始值设定 (None)：保持值。	
	CJ 单元用存储器 CIO/WR 区域*2		16#00		保持值。	

<table>
<tr><th rowspan="2">变量的保持属性</th><th colspan="2" rowspan="2">变量种类</th><th rowspan="2">电源接通时</th><th rowspan="2">动作模式变更时</th><th colspan="2">下载后</th></tr>
<tr><th>“初始化保持变量”设定时</th><th>“不初始化保持变量”设定时</th></tr>
<tr><td rowspan="6">保持</td><td rowspan="3">用户定义变量</td><td>满足保持条件*3</td><td rowspan="6">无变化
（保持断电前的值）</td><td rowspan="6">无变化
（保持动作模式变更的值）</td><td rowspan="2">• 有初始值设定：指定的初始值。
• 无初始值设定(None)：16#00*1</td><td>保持下载前的值</td></tr>
<tr><td>不满足保持条件 *3</td><td>• 有初始值设定：指定的初始值。
• 无初始值设定(None)：16#00*1</td></tr>
<tr><td>AT（分配目标）指定为 CJ 单元用存储器的 HR/DM/EM 区域</td><td>• 有初始值设定：指定的初始值。
• 无初始值设定(None)：遵照 AT（分配目标）指定的存储器值。</td><td></td></tr>
<tr><td colspan="2">CJ 单元用设备变量</td><td>• 有初始值设定：指定的初始值。
• 无初始值设定(None)：遵照 AT（分配目标）指定的存储器值。</td><td>遵照 AT（分配目标）指定的存储器值。</td></tr>
<tr><td colspan="2" rowspan="2">CJ 单元用存储器 HR/DM/EM 区域 *2</td><td>保持值。
• 在“设定用数据”（DM 区域）中分配了 CJ 单元设备变量时，遵照变量的初始值设定。
• 分配了用户定义变量时，遵照变量的初始值设定。</td><td>保持值。</td></tr>
</table>

*1. 因数据类型不同，可能变为 16#00 以外的值。详情请参考「初始值指定为空白时」（P.6-61）。
*2. 已对用户定义变量或 CJ 单元设备变量进行 AT（分配对象）指定时除外。
*3. 保持条件请参考「保持条件」（P.6-66）。

变量的保持属性	变量种类	发生全部停止故障等级的控制器异常时	在线编辑时	
			追加到在线编辑对象内的 POU 的变量	在线编辑对象内的 POU 中已存在的变量
非保持	用户定义变量 / 设备变量	• 有初始值设定：指定的初始值 • 无初始值设定时：16#00	• 有初始值设定：指定的初始值 • 无初始值设定时：16#00	无变化
	CJ 单元用存储器 CIO/WR 区域	16#00	无变化	
保持	用户定义变量 / 设备变量	无变化 （保持异常发生前的值）	• 有初始值设定：指定的初始值 • 无初始值设定时：16#00	
	CJ 单元用存储器 HR/DM/EM 区域		无变化	
其它	强制值刷新状态	解除	无变化	

使用注意事项

可使用 CJ 单元及 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

保持条件

保持条件是指下载前后满足以下所有条件。

- 变量名称相同。
- 数据类型名称相同。
- 保持属性为 “ 有保持 ”。

关于保持属性为 “ 有保持 ” 的变量，变更了 POU 名称变量名称时变量值的详情请参考「 变更 POU 名称或新建 POU 等时保持变量的值 」（P.6-67），变更了数据类型时变量值的详情请参考「 变更了保持变量的数据类型时变量的值 」（P.6-69）。

版本相关信息

CPU 单元的单元版本为 1.10 以下时，是指下载前后满足以下所有条件。

- 变量名称相同。
- 数据类型名称及数据类型的大小相同。
- 保持属性为 “ 有保持 ”。

变更 POU 名称或新建 POU 等时保持变量的值

将 Sysmac Studio 中创建的项目下载到控制器后，控制器将以与 Sysmac Studio 中显示的 POU 名称不同的 POU 名称处理。因此，即使 Sysmac Studio 上显示相同的 POU 名称，控制器可能会视为不同的 POU 进行处理。

在 Sysmac Studio 上进行 POU 名称变更或 POU 新建等操作后，执行了下载时，对控制器处理的 POU 名称和保持属性为 " 有保持 " 的本地变量值进行说明。

● 用 Sysmac Studio 操作前的状态

Sysmac Studio 中创建的项目 "Project_A" 将下载到控制器中。"Project_A" 中包含 "POU_A"，"POU_A" 中包含本地变量 VarA。
Symac Studio 中显示为 "POU_A" 的 POU，在控制器上作为 "POU_XXX" 处理。同时，假设控制器上的本地变量 VarA 的当前值已变为 "20"。

Sysmac Studio 侧	
项目名称	: Project_A
POU 名称	: POU_A
变量名称	: VarA
VarA 的初始值	: 0

下载 →

控制器侧	
项目名称	: Project_A
POU 名称	: POU_XXX
变量名称	: VarA
VarA 的当前值	: 20

● 变更了 POU 名称时

即使在 Sysmac Studio 上将 "POU_A" 的名称变更为 "POU_B" 并下载，控制器侧的 POU 名称及本地变量 VarA 的当前值不会发生变更。

Sysmac Studio 侧	
项目名称	: Project_A
POU 名称	: POU_B
变量名称	: VarA
VarA 的初始值	: 0

下载 →

控制器侧	
项目名称	: Project_A
POU 名称	: POU_XXX
变量名称	: VarA
VarA 的当前值	: 20

● 新建了 POU 时

在 Sysmac Studio 上新建包含本地变量 VarA 的 "POU_C" 并下载时，控制器上将作为新的 "POU_YYY" 及新的本地变量 VarA 处理。
因此，新的本地变量 VarA 的值为初始值 "0"。

Sysmac Studio 侧	
项目名称	: Project_A
POU 名称	: POU_A
变量名称	: VarA
VarA 的初始值	: 0
POU 名称	: POU_C
变量名称	: VarA
VarA 的初始值	: 0

下载 →

控制器侧	
项目名称	: Project_A
POU 名称	: POU_XXX
变量名称	: VarA
VarA 的当前值	: 20
POU 名称	: POU_YYY
变量名称	: VarA
VarA 的当前值	: 0

● 复制 POU 并创建了 POU 时

在 Sysmac Studio 上复制 “POU_A” 并创建 “POU_A_copy” 然后下载时，控制器上将作为新的 “POU_ZZZ” 及新的本地变量 VarA 处理。
因此，新的本地变量 VarA 的值为初始值 “0”。

Sysmac Studio 侧	
项目名称	: Project_A
POU 名称	: POU_A
变量名称	: VarA
VarA 的初始值	: 0
POU 名称	: POU_A_copy
变量名称	: VarA
VarA 的初始值	: 0

下载 →

控制器侧	
项目名称	: Project_A
POU 名称	: POU_XXX
变量名称	: VarA
VarA 的当前值	: 20
POU 名称	: POU_ZZZ
变量名称	: VarA
VarA 的当前值	: 0

● 复制了导出的项目文件时

即使在 Sysmac Studio 上复制 “Project_A” 并创建 “Project_A_copy” 然后下载，控制器侧的 POU 名称及本地变量 VarA 的当前值不会发生变更。

Sysmac Studio 侧	
项目名称	: Project_A_copy
POU 名称	: POU_A
变量名称	: VarA
VarA 的初始值	: 0

下载 →

控制器侧	
项目名称	: Project_A_copy
POU 名称	: POU_XXX
变量名称	: VarA
VarA 的当前值	: 20

将复制 “Project_A” 时未存在的本地变量 VarB 追加到 “Project_A” 中并下载，然后将相同定义的本地变量 VarB 追加到 “Project_A_copy” 中并下载时，控制器上的本地变量 VarB 的当前值不会发生变更。
从 “Project_A” 复制本地变量 VarB，或合并变量表时，控制器上的本地变量 VarB 的当前值同样不会发生变更。

Sysmac Studio 侧	
项目名称	: Project_A_copy
POU 名称	: POU_A
变量名称	: VarA
VarA 的初始值	: 0
变量名称	: VarB
VarB 的初始值	: 0

下载 →

控制器侧	
项目名称	: Project_A_copy
POU 名称	: POU_XXX
变量名称	: VarA
VarA 的当前值	: 20
变量名称	: VarB
VarB 的当前值	: 50

● 在另外创建的项目中创建同名的 POU

在 Sysmac Studio 上创建 “Project_B”，其中包含与 “Project_A” 的 POU 名称同名的 “POU_A”，然后下载时，控制器上将作为新的 “POU_YYY” 及新的本地变量 VarA 处理。因此，新的本地变量 VarA 的值为初始值 “0”。

Sysmac Studio 侧	
项目名称	: Project_B
POU 名称	: POU_A
变量名称	: VarA
VarA 的初始值	: 0

下载 →

控制器侧	
项目名称	: Project_B
POU 名称	: POU_YYY
变量名称	: VarA
VarA 的当前值	: 0

● 将原项目的 POU 复制到其他项目时

在 Sysmac Studio 上创建 “Project_B”，从 “Project_A” 复制 “POU_A” 并粘贴，然后下载时，控制器上将作为新的 “POU_YYY” 及新的本地变量 VarA 处理。因此，新的本地变量 VarA 的值为初始值 “0”。

Sysmac Studio 侧	
项目名称	: Project_B
POU 名称	: POU_A
变量名称	: VarA
VarA 的初始值	: 0

下载 →

控制器侧	
项目名称	: Project_B
POU 名称	: POU_YYY
变量名称	: VarA
VarA 的当前值	: 0

变更了保持变量的数据类型时变量的值

在 Sysmac Studio 上变更保持属性为 “ 有保持 ” 的变量数据类型后，对包含 “ 有保持 ” 的变量当前值执行恢复、传送、下载等操作时，控制器上变量值的情况。

● 操作和变量的值

保持属性为 “ 有保持 ” 的变量动作有以下两种模式。操作和动作模式的关系如下表所示。

操作	变量值的动作模式
• 利用 Sysmac Studio 控制器备份功能恢复 • 利用 SD 存储卡备份功能恢复 • SD 存储卡自动传送功能 • SD 存储卡程序传送功能	模式 1
• 利用 Sysmac Studio 变量 / 存储器的备份功能恢复变量	模式 2
• 从 Sysmac Studio 下载	模式 2*1

*1. 单元版本 1.10 以下的 CPU 单元中为模式 1。两者的不同取决于 CPU 单元的单元版本。与项目的设备中设定的单元版本无关。

下面根据不同的数据类型介绍模式 1 和模式 2 的动作。

● 基本数据型的变更

用 Sysmac Studio 变更前的变量定义如下。

变量名称	数据类型	分配目标
ABC	INT	—
DEF	INT	%D100

假设各变量的保持属性为 “ 有保持 ”，当前值为以下值。

变量 / 存储器	当前值
ABC	10
DEF	20

假设 CJ 单元用存储器的 DM100 和 DM101 的当前值为以下值。

变量 / 存储器	当前值
D100	20
D101	50

数据类型的变更内容和控制器上变量动作的关系如下表所示。

变更内容	模式 1	模式 2
无变更	保持变量的值。 ABC = 10 DEF = 20	
变量名称从 “ABC” 变更为 “GHI”	变量的值为初始值。 GHI = 初始值	
变量 “ABC” 的数据类型从 INT 变更为 DINT	变量的值为初始值。 ABC = 初始值	
变量 “ABC” 的保持属性变更为 “ 不保持 ”	变量的值为初始值。*1 ABC = 初始值	
变量 “DEF” 的分配目标从 “D100” 变更为 “D101”	变量的值为变更后的分配目标值。 DEF = 50	
删除变量 “DEF” 的分配目标	变量的值为初始值。 DEF = 初始值	

*1. 保持属性为 “ 无保持 ” 的变量变更为 “ 有保持 ” 时，变量的值也变温初始值。

● 结构体型的变更

用 Sysmac Studio 变更前的结构体型定义如下。

数据类型名称	成员	成员的数据类型
STR_1	x	INT
	y	INT

假设变量 ABC 的数据类型为 “STR_1”、保持属性为 “ 有保持 ”，各成员的当前值为以下值。

成员	当前值
ABC.x	100
ABC.y	-200

数据类型的变更内容和控制器上变量动作的关系如下表所示。
同时符合保持的条件和变为初始值的条件时，变量的值变为初始值。

变更内容	模式 1	模式 2
无变更	保持成员的值。 ABC.x = 100 ABC.y = -200	
变量名称从 “ABC” 变更为 “DEF”	成员的值为初始值。 DEF.x = 初始值 DEF.y = 初始值	
成员 “x” 变更为 “x1”	保持成员的值。 ABC.x1 = 100 ABC.y = -200	变更后的成员值变为初始值。 ABC.x1 = 初始值 ABC.y = -200
成员 “x” 的数据类型从 INT 变更为 WORD*1 （数据大小相同）	保持成员的值。 ABC.x = WORD#16#0064 (= 100) ABC.y = -200	
成员 “y” 的数据类型从 INT 变更为 UINT*1 （数据大小相同，从有符号变更为无符号）	保持成员的值。变更后的成员将直接代入，因此绝对值和符号会发生变化。 ABC.x = 100 ABC.y = 65336	
成员 “y” 的数据类型从 INT 变更为 DINT*1 （数据大小扩大）	成员的值为初始值。 ABC.x = 初始值 ABC.y = 初始值	保持成员的值。变更后的成员发生符号扩展，绝对值和符号无变化。 ABC.x = 100 ABC.y = -200
成员 “y” 的数据类型从 INT 变更为 UDINT*1 （数据大小扩大，从有符号变更为无符号）	成员的值为初始值。 ABC.x = 初始值 ABC.y = 初始值	保持成员的值。变更后的成员发生符号扩展，绝对值和符号发生变化。 ABC.x = 100 ABC.y = 4294967096
成员 “y” 的数据类型从 INT 变更为 SINT*1 （数据大小缩小）	成员的值为初始值。 ABC.x = 初始值 ABC.y = 初始值	数据大小缩小后，成员的值变为初始值。 ABC.x = 100 ABC.y = 初始值
成员 “x” 变更为 “y”，成员 “y” 变更为 “x”	成员的值保持不正确。 ABC.y = 100 ABC.x = -200	保持成员的值。 ABC.y = -200 ABC.x = 100
在成员 “y” 后面追加 INT 的成员 “z”	成员的值为初始值。*2 ABC.x = 初始值 ABC.y = 初始值 ABC.z = 初始值	保持现有成员的值。追加成员的值变为初始值。 ABC.x = 100 ABC.y = -200 ABC.z = 初始值

变更内容	模式 1	模式 2
在成员 “x” 和成员 “y” 之间追加 INT 的成员 “z”	成员的值为初始值。*2 ABC.x = 初始值 ABC.y = 初始值 ABC.z = 初始值	保持现有成员的值。追加成员的值变为初始值。 ABC.x = 100 ABC.z = 初始值 ABC.y = -200
成员 “x” 的排列要素数从 “1” 变更为 “3”	成员的值为初始值。*2 ABC.x[0] = 初始值 ABC.x[1] = 初始值 ABC.x[2] = 初始值 ABC.y = 初始值	保持现有成员的值。变更了排列要素数的成员将从头开始保持，扩展成员的值变为初始值。 ABC.x[0] = 100 ABC.x[1] = 初始值 ABC.x[2] = 初始值 ABC.y = -200

*1. 变更数据类型时，遵照默认的 CAST 执行。关于默认的 CAST，请参考📖「默认的 CAST」（P.6-121）。
*2. 下载前先上传变量的值，下载后再下载变量的值，可保持现有的成员值。

● 联合体型的变更

用 Sysmac Studio 变更前的联合体型定义如下。

数据类型名称	成员	成员的数据类型
UNI_1	x	WORD
	y	DWORD

假设变量 ABC 的数据类型为 “UNI_1”、保持属性为 “ 有保持 ”，各成员的当前值为以下值。

成员	当前值
ABC	WORD#16#0100

数据类型的变更内容和控制器上变量动作的关系如下表所示。
同时符合保持的条件和变为初始值的条件时，变量的值变为初始值。

变更内容	模式 1	模式 2
无变更	保持成员的值。 ABC = WORD#16#0100	
变量名称从 “ABC” 变更为 “DEF”	成员的值为初始值。 DEF = 初始值	
成员 “x” 的数据类型从 WORD 变更为 BYTE*1 （作为联合体型的数据大小相同）	保持成员的值。 ABC = WORD#16#0100	
成员 “x” 的数据类型从 WORD 变更为 LWORD*1 （作为联合体型的数据大小扩大）	成员的值为初始值。 ABC = 初始值	保持成员的值。 ABC = WORD#16#0100
成员 “y” 的数据类型从 DWORD 变更为 BYTE*1 （作为联合体型的数据大小缩小）	成员的值为初始值。 ABC = 初始值	
成员 “x” 变更为 “y”，成员 “y” 变更为 “x”	保持成员的值。 ABC = WORD#16#0100	
在成员 “y” 后面追加 BYTE 的成员 “z” （作为联合体型的数据大小相同）	保持成员的值。 ABC = WORD#16#0100	
在成员 “y” 后面追加 LWORD 的成员 “z” （作为联合体型的数据大小扩大）	成员的值为初始值。 ABC = 初始值	保持成员的值。 ABC = WORD#16#0100
在成员 “y” 后面追加 BYTE 的成员 “z”，删除成员 “y” （作为联合体型的数据大小缩小）	成员的值为初始值。 ABC = 初始值	

*1. 变更数据类型时，遵照默认的 CAST 执行。关于默认的 CAST，请参考📖「默认的 CAST」（P.6-121）。

● 排列变量的变更

用 Sysmac Studio 变更前的排列变量定义如下。

变量名称	数据类型
ABC	ARRAY[1..3] OF INT

假设变量 ABC 的保持属性为 “ 有保持 ”，各要素的当前值为以下值。

成员	当前值
ABC[1]	100
ABC[2]	200
ABC[3]	300

数据类型的变更内容和控制器上变量动作的关系如下表所示。
同时符合保持的条件和变为初始值的条件时，变量的值变为初始值。

变更内容	模式 1	模式 2
无变更	保持要素的值。 ABC[1] = 100 ABC[2] = 200 ABC[3] = 300	
变量名称从 “ABC” 变更为 “DEF”	要素的值为初始值。 DEF[1] = 初始值 DEF[2] = 初始值 DEF[3] = 初始值	
要素的数据类型从 INT 变更为 UINT*1 （数据大小相同）	要素的值为初始值。 ABC[1] = 初始值 ABC[2] = 初始值 ABC[3] = 初始值	
要素的数据类型从 INT 变更为 DINT*1 （数据大小扩大）	要素的值为初始值。 ABC[1] = 初始值 ABC[2] = 初始值 ABC[3] = 初始值	
要素的数据类型从 INT 变更为 SINT*1 （数据大小缩小）	要素的值为初始值。 ABC[1] = 初始值 ABC[2] = 初始值 ABC[3] = 初始值	
排列要素的结束编号从 “3” 变更为 “4”*2 （要素数从 “3” 扩展至 “4”）	要素的值为初始值。 ABC[1] = 初始值 ABC[2] = 初始值 ABC[3] = 初始值 ABC[4] = 初始值	保持现有要素编号的要素值。新要素编号的要素值为初始值。 ABC[1] = 100 ABC[2] = 200 ABC[3] = 300 ABC[4] = 初始值
排列要素的开始编号从 “1” 变更为 “0”*2 （要素数从 “3” 扩展至 “4”）	要素的值为初始值。 ABC[0] = 初始值 ABC[1] = 初始值 ABC[2] = 初始值 ABC[3] = 初始值	保持现有要素编号的要素值。新要素编号的要素值为初始值。 ABC[0] = 初始值 ABC[1] = 100 ABC[2] = 200 ABC[3] = 300
排列要素的结束编号从 “3” 变更为 “2”*2 （要素数从 “3” 缩小为 “2”）	要素的值为初始值。 ABC[1] = 初始值 ABC[2] = 初始值	保持现有要素编号的要素值。 ABC[1] = 100 ABC[2] = 200

变更内容	模式 1	模式 2
排列要素的开始编号从 “1” 变更为 “2”*2 （要素数从 “3” 缩小为 “2”）	要素的值为初始值。 ABC[2] = 初始值 ABC[3] = 初始值	保持现有要素编号的要素值。 ABC[2] = 200 ABC[3] = 300
排列要素的开始编号从 “1” 变更为 “3”，结束编号从 “3” 变更为 “5”*2 （要素数相同）	要素数相同时，要素值从头开始保持。*3 ABC[3] = 100 ABC[4] = 200 ABC[5] = 300	保持现有要素编号的要素值。新要素编号的要素值为初始值。 ABC[3] = 300 ABC[4] = 初始值 ABC[5] = 初始值

*1. 变更数据类型时，遵照默认的 CAST 执行。关于默认的 CAST，请参考「默认的 CAST」（P.6-121）。
*2. 无论变量数据为基本数据型还是派生数据型，结果都一样。
*3. 要素数发生扩展或缩小时，所有要素值均变为初始值。

● 列举型的变更

用 Sysmac Studio 变更前的列举型定义如下。

数据类型名称	列举值	值
ENU_1	x	1
	y	2

假设变量 ABC 的数据类型为 “ENU_1”、保持属性为 “ 有保持 ”，变量的当前值为以下值。

成员	当前值
ABC	1 (= x)

数据类型的变更内容和控制器上变量动作的关系如下表所示。

变更内容	模式 1	模式 2
无变更	保持变量的值。 ABC = 1 (= x)	
变量名称从 “ABC” 变更为 “DEF”	变量的值为初始值。 ABC = 初始值	
在列举值 “y” 后面追加值为 “3” 的列举值 “z”	保持变量的值。 ABC = 1 (= x)	
删除列举值 “x”	列举值不存在时，保持变量的值。 ABC = 1	
列举值 “X” 的值从 “1” 变更为 “5”	即使变更了列举值，仍保持变量的值。 ABC = 1	

6-3-10 功能块 (FB) 的实例

实例化的 FB 为数据类型的一种，登录到本地变量表中并显示。

参考

FB 的实例视为记述了实例的程序中的本地变量，登录到该程序的本地变量表中并显示。不会变成全局变量，敬请注意。

6-3-11 变量值的监视

变量的值可通过 Sysmac Studio 的监视窗口进行监视。

1 **选择［显示］｜［监视窗口］。**
显示监视窗口。

2 **与控制器在线连接，通过以下任意方法登录变量。**

(1) 在监视窗口的名称单元格中输入变量名称。

(2) 从编辑器或变量表拖放到监视窗口中登录。
显示该变量的当前值。

6-3-12 变量名称等程序相关名称的限制

程序相关名称的限制如以下列表所示。

字符限制

<table>
<tr><th>程序相关的名称</th><th>可使用的字符</th><th>保留字</th><th>多字节字符使用可否</th><th>大小写的区分</th><th>最大大小（不含 NULL）</th><th>字符编码</th></tr>
<tr><td>变量名称
（含 POU 实例名称）</td><td rowspan="15">● 可使用的字符
• 0 ～ 9 、 A ～ Z 、 a ～ z
• 半角假名
• _ （下划线）
• 多字节字符 （例：日语）
保留字请参考后述的 “ 保留字 ”。

● 不可组合使用的字符
• 以数字 （0 ～ 9）开始的字符串
• 以 P_ 开始的字符串
• 以下划线开始的字符串
• 有 2 个以上连续下划线（_）的字符串
• 以下划线 （_）结束的字符串
• 构成识别符的字符串的前缀或后缀中包含 1 字节以上的扩展空白符 （全半角空格等，扩展至 Unicode 的空白字符）的字符串</td><td rowspan="15">参考后述的 “ 保留字 ”</td><td rowspan="15">可</td><td rowspan="13">否</td><td rowspan="2">127 字节</td><td rowspan="15">UTF-8*1</td></tr>
<tr><td>POU 定义名称</td></tr>
<tr><td>数据类型名称</td></tr>
<tr><td>结构体成员名称
联合体成员名称</td></tr>
<tr><td>列举值</td></tr>
<tr><td>任务名称</td><td>63 字节</td></tr>
<tr><td>名称空间名</td><td>93 字节</td></tr>
<tr><td>变量名称的完整路径</td><td>网络变量：255 字节
其他：511 字节</td></tr>
<tr><td>设备名称</td><td rowspan="4">127 字节</td></tr>
<tr><td>段落名称</td><td>是</td></tr>
<tr><td>轴名称</td><td rowspan="3">否</td></tr>
<tr><td>轴组名称</td></tr>
<tr><td>凸轮表名称</td></tr>
</table>

*1. UTF-8 中， 1 个字符的半角英文和数字使用 1 字节，多字节字符使用多个字节。日文字符一般为 3 字节。

保留字

以下名称在程序检查时将发生错误。

- 与 《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》的指令相同的名称
- 与 《NJ/NX 系列 指令基准手册 运动篇 （SBCE-CN5-364）》的指令相同的名称
- 系统内部使用的保留字

名称不可重复

以下名称不可重复。程序检查时将发生错误。

- 在同一 CPU 单元中，全局变量名称重复
- 在同一 POU 中，变量名称重复
- 在同一 POU 中，段落名称重复
- 在同一结构体或联合体中，成员名称重复
- 在同一列举型中，列举值重复
- 本地变量名称和全局变量名称重复
- POU 名称和数据类型名称重复
- 数据类型名称和变量名称重复
- 列举值和其他列举型的列举值重复
- 列举值和变量名称重复

6-4 常数（文本）

下面详细介绍常数。

6-4-1 常数定义

变量的值因变量中代入的数据不同而变化。与之相对的是值不会变化的常数。
常数与变量不同，不是存储器中保存的信息。常数不需要声明，也可在 POU 的算法部分中记述。
NJ/NX 系列中，常数与变量一样，都有 " 数据类型 " 的概念。

6-4-2 各数据类型的记述形式

各数据类型的常数记述形式如下所示。若用规定以外的形式记述常数，链接时将发生异常。

布尔型

布尔型的数据类型名称为 "BOOL"。此外，值除了 1 或 0 以外，还可使用 "TRUE" 和 "FALSE"。记述形式及含义如下所示。

记述形式	含义
TRUE 或 FALSE	TRUE、 BOOL#1、 BOOL#TRUE、 1 为等价。 FALSE、 BOOL#0、 BOOL#FALSE、 0 为等价。
BOOL#1 或 BOOL#0	
BOOL#TRUE 或 BOOL#FALSE	
1 或 0*1	

*1. 1 或 0 的记述可在 Sysmac Studio Ver.1.03 以上中使用。若在 Sysmac Studio Ver.1.02 以下中记述 1 或 0，链接时将发生异常。

位列型

位列型使用 "BYTE"、 "WORD"、 "DWORD"、 "LWORD" 中的任意一个作为数据类型名称。此外，使用 "2"、 "8"、 "10"、 "16" 中的任意一个作为基数。记述形式和记述示例如下所示。

记述形式	记述示例
{ 数据类型名称 }#{ 基数 }#{ 数值 }	WORD#16#0064
{ 基数 }#{ 数值 }*1	16#0064
{ 数值 }*1*2	100

*1. 省略数据类型名称的记述可在 Sysmac Studio Ver.1.03 以上中使用。若在 Sysmac Studio Ver.1.02 以下中记述省略数据类型名称的内容，链接时将发生异常。
*2. 基数为 10，即解释为 10 进制数。

版本相关信息

可在 Sysmac Studio Ver.1.03 以上中将 "10" 记述为位列型基数。若在 Sysmac Studio Ver.1.02 以下中将基数设为 "10"，链接时将发生异常。

整数型

整数型使用 “SINT”、“USINT”、“INT”、“UINT”、“DINT”、“UDINT”、“LINT”、“ULINT” 中的任意一个作为数据类型名称。此外，使用 “2”、 “8”、 “10”、 “16” 中的任意一个作为基数。
记述形式和记述示例如下所示。

记述形式	记述示例
{ 数据类型名称 }#{ 基数 }#{ 数值 }	INT#10#-1
{ 基数 }#{ 数值 }*1	10#-1
{ 数据类型名称 }#{ 数值 }*2	INT#-1
{ 数值 }*1*2	-1

*1. 省略数据类型名称的记述可在 Sysmac Studio Ver.1.03 以上中使用。若在 Sysmac Studio Ver.1.02 以下中记述省略数据类型名称的内容，链接时将发生异常。
*2. 基数为 10，即解释为 10 进制数。

实型数据

实型数据使用 “REAL” 和 “LREAL” 中的一个作为数据类型名称。此外，实型数据中可使用的基数仅限 “10”。记述形式和记述示例如下所示。

记述形式	记述示例
{ 数据类型名称 }#{ 基数 }#{ 数值 }	LREAL#10#-3.14
{ 基数 }#{ 数值 }*1	10#-3.14
{ 数据类型名称 }#{ 数值 }*2	LREAL#-3.14
{ 数值 }*1*2	-3.14

*1. 省略数据类型名称的记述可在 Sysmac Studio Ver.1.03 以上中使用。若在 Sysmac Studio Ver.1.02 以下中记述省略数据类型名称的内容，链接时将发生异常。
*2. 基数为 10，即解释为 10 进制数。

持续时间型

持续时间型使用 “TIME” 和 “T” 中的一个作为数据类型名称。记述形式和记述示例如下所示。

记述形式	记述示例
TIME#{ 日 }d{ 时 }h{ 分 }m{ 秒 } s{ 毫秒 }ms	TIME#61m5s
T#{ 日 }d{ 时 }h{ 分 }m{ 秒 }s{ 毫秒 }ms	T#61m5s

持续时间型常数的记述有以下规则。

- { 日 }、 { 时 }、 { 分 }、 { 秒 }、 { 毫秒 } 不需要全部记述。记述其中的一个或几个即可。
- 如 “TIME#12d3.5h”，可用小数记述。
- 记述内容可超出时间的有效范围。例如， “T#-61m5s” 与 “T#-1h1m5s” 的含义相同。
- 数值解释为 10 进制数。若记述了 10 进制数以外的数值，链接时将发生异常。
- 记述时可改变时间的顺序。例如， “T#1h2d” 与 “T#2d1h” 的含义相同。

日期型

日期型使用 “DATE” 和 “D” 中的一个作为数据类型名称。记述形式和记述示例如下所示。

记述形式	记述示例
DATE#{ 年 }-{ 月 }-{ 日 }	DATE#2010-1-10
D#{ 年 }-{ 月 }-{ 日 }	D#2010-1-10

日期型常数的记述有以下规则。

- {年}、{月}、{日} 的开头分别可添加 1 个以上的 0。例如，“DATE#2010-01-10” 与 “D#2010-1-10” 的含义相同。
- 若记述内容超出日期的有效范围，链接时将发生异常。例如， “D#2010-01-35” 时将发生异常。
- 数值解释为 10 进制数。若记述了 10 进制数以外的数值，链接时将发生异常。

时刻型

时刻型使用 “TIME_OF_DAY” 和 “TOD” 中的一个作为数据类型名称。记述形式和记述示例如下所示。

记述形式	记述示例
TIME_OF_DAY#{ 时 }:{ 分 }:{ 秒 }	TIME_OF_DAY#23:59:59.999999999
TOD#{ 时 }:{ 分 }:{ 秒 }	TOD#23:59:59.999999999

时刻型常数的记述有以下规则。

- {时}、{分}、{秒}的开头分别可添加 1 个以上的 0。例如，“TOD#23:01:01” 与 “TOD#23:1:1” 的含义相同。
- 若记述的数值超出时分秒的有效范围，链接时将发生异常。例如， “TOD#24:00:00” 时将发生异常。
- 数值解释为 10 进制数。若记述了 10 进制数以外的数值，链接时将发生异常。

日期时刻型

日期时刻型使用 “DATE_AND_TIME” 和 “DT” 中的一个作为数据类型名称。记述形式和记述示例如下所示。

记述形式	记述示例
DATE_AND_TIME#{ 年 }-{ 月 } -{ 日 }-{ 时 }:{ 分 }:{ 秒 }	DATE_AND_TIME#2010-10-10-23:59:59.123
DT#{ 年 }-{ 月 }-{ 日 }-{ 时 }:{ 分 } :{ 秒 }	DT#2010-10-10-23:59:59.123

日期时刻型常数的记述有以下规则。

- {年}、{月}、{日}、{时}、{分}、{秒}的开头分别可添加 1 个以上的 0。例如，“DT#2010-01-10-23:01:01” 与 “DT#2010-1-10-23:1:1” 的含义相同。
- 若记述的数值超出年月日时分秒的有效范围，链接时将发生异常。例如， “DT#2010-01-35-00:00:00” 或 “DT#2010-01-30-24:00:00” 时将发生异常。
- 数值解释为 10 进制数。若记述了 10 进制数以外的数值，链接时将发生异常。

字符串型

字符串型用两个半角单引号将字符串括起来记述。同时，可使用 “STRING” 作为数据类型名称。记述形式和记述示例如下所示。

记述形式	记述示例
'{ 字符串 }'	' 这是字符串 '
STRING#'{ 字符串 }'*1	STRING#' 这是字符串 '

*1. 可在 Sysmac Studio Ver.1.08 以上中使用 “STRING” 进行记述。

字符串的记述有以下规则。

- 还可记述字符数为 0 的字符串。此时记述为 “''”。
- 如下所示，若跨多行记述字符串，链接时将发生异常。
  strVar := 'ABC
  DEF'
- 记述制表符或换行代码等时，使用开头带 “$” 的转义字符。转义字符的名称和含义如下所示。

转义字符	名称	含义
$$	半角美元符号	半角美元符号字符 （$：字符编码 0x24）
$'	半角单引号	半角单引号符号字符 （'：字符编码 0x27）
$"	半角双引号	半角双引号符号字符 （"：字符编码 0x22）
$L 或 $l	换行	将光标移动到下一行。 LF 控制字符 （Line Feed：字符编码 0x0A）
$N 或 $n	换行	将光标移动到下一行。 NL 控制字符 （New Line：字符编码 0x0A）
$P 或 $p	翻页	将光标移动到下一页。 FF 控制字符 （Form Feed：字符编码 0x0C）
$R 或 $r	复位	将光标移动到行的开头。 CR 控制字符 （Carriage Return：字符编码 0x0D）
$T 或 $t	制表符字符	显示制表符。 制表符字符 （字符编码 0x09）
$ （字符编码）	直接指定字符编码	用 2 位的 16 进制数直接指定字符编码。字符编码的范围为 00 ～ FF。 例如， ‘$L’ 与 ‘$0A’ 的含义相同。 此外， 2 字节以上的字符每个字节都会带 $。 例如，字符编码为 0xE38182 平假名 “ あ ”，记述为 “$E3$81$82”。

- 还可直接指定字符编码后记述字符。此时，在开头加 “$”。用 2 位的 16 进制数记述字符编码。例如，换行 “$L” 的字符编码为 0x0A，因此可记述为 “$0A”。
- 直接指定字符编码，记述 2 字节以上的字符时，每个字节加 “$”。例如，字符编码为 0xE38182 平假名 “ あ ”，记述为 “$E3$81$82”。

列举型

列举型中记述列举型的数据类型名称和列举值。记述形式和记述示例如下所示。

记述形式	记述示例
｛列举型的数据类型名称｝ # ｛列举值｝	_eDAYOFWEEK#_WED

参考

向参数的数据类型指定为列举型的 FB 或 FUN 传递列举值时，可省略列举型的数据类型名称，只记述列举值。
例如，BinToBCDs 指令的输入变量 “Format” 指定为列举型 _eBCD_FORMAT。因此，除了记述列举型的数据类型名称和列举值的“_eBCD_FORMAT#_BCD0”之外，还可省略列举型的数据类型名称，记述为 “_BCD0”。

6-5 编程语言

下面详细介绍各编程语言。
用 Sysmac Studio 输入编程语言的方法请参考📖《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》。

6-5-1 编程语言定义

表示 POU （程序、 FB、 FUN）中算法的语言称为编程语言。
NJ/NX 系列中可使用的编程语言分为梯形图语言（LD）、ST 语言（Structured Text: 结构化文本）两种。

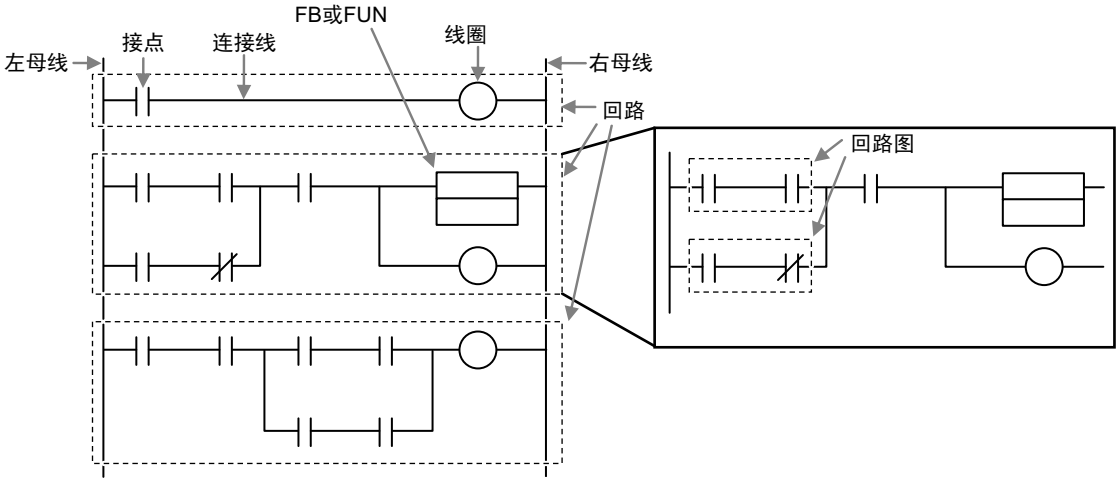
6-5-2 梯形图语言 （LD）

梯形图语言是以电气回路示意图记述的图形编程语言。包括 FB 或 FUN 在内，各部分的处理全部以图形表示，然后用直线将它们连接起来，构建一系列的算法。用梯形图语言撰写的算法称为梯形图。

梯形图语言的构成要素

梯形图语言由左右母线、连接线、梯形图语法 （接点、线圈）、 FB 及 FUN 构成 （*）。
* 只有 JMP 指令及 Label 指令时，以表示 JMP 和 Label 的符号记述。

算法由连接以上要素的多个回路构成。
回路是指将左母线到右母线之间的构成要素连接起来。此外，回路由以 （表示逻辑启动的） LD/LDN 指令开头的回路图构成。



● **母线**

梯形图左端和右端的竖线称为母线。
各母线处于 TRUE 或 FALSE 中的任意状态。这表示将梯形图视为电气回路时的通电状态和非通电状态。
启动写有梯形图的 POU 后，左母线的值变为 TRUE。同时，针对与左母线连接的接点等构成要素，也会传达 TRUE 的状态。按这些构成要素的动作， TRUE 的状态依次向右传达。
这种 TRUE 状态的流动称为 “ 功率流 ”。母线可以说是功率流的源泉。

● **连接线**

水平方向连接母线和各种构成要素的直线称为连接线。
连接线处于 TRUE 或 FALSE 中的任意状态，起到向右传达功率流的作用。

● **接点**

接点配置在连接线上，通过接收功率流来动作。
接点有多个种类，将根据各自的规格将左侧传来的功率流向右传递，或将其阻断。
接点向右传达功率流后，接点右侧的连接线状态将变为 TRUE，若阻断，则变为 FALSE。
各接点规格的详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

● **线圈**

线圈配置在连接线上，通过接收功率流来动作。
线圈的功能是向变量写入 TRUE 或 FALSE 的值。
线圈也有多个种类。各线圈规格的详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

● **FB、 FUN**

FB、 FUN 配置在连接线上，通过接收功率流来动作。
各指令的详细规格请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

梯形图的执行顺序

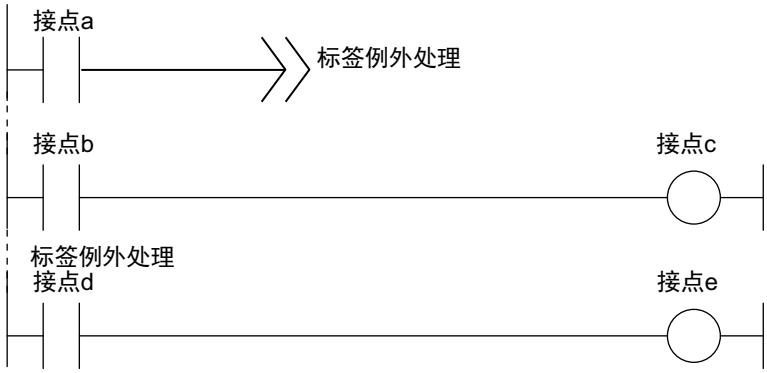
接点、线圈、 FB、 FUN 通过接收功率流来执行。
执行顺序为梯形图从上往下。此外，如果时高度相同的要素，则从左往右执行。

梯形图的结束

梯形图从上往下执行，执行到最底下，则结束处理。
但是，若中途有 END 指令或 RETURN 指令，将在该时间点结束处理。上述指令后面的处理将不执行。

梯形图的执行控制

梯形图基本上从上往下执行，但如果使用执行控制指令，可以更改执行顺序。
例如，以下情况时，若接点 a 变为 TRUE，则转移到有 “ 例外处理 ” 标签的位置并执行。



梯形图中 FB、 FUN 的连接

● 连接形式

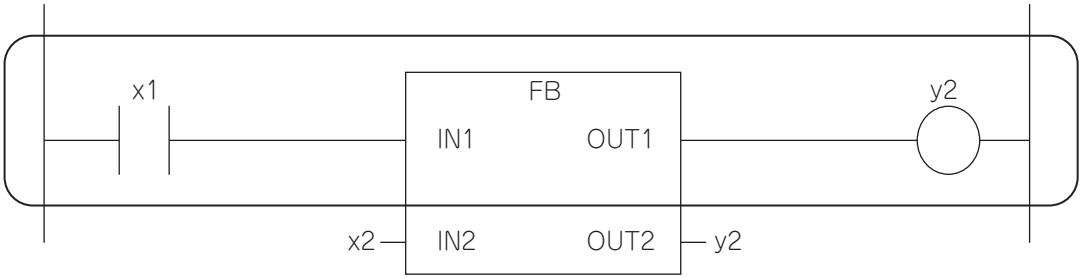
FB 及 FUN 中，有以下两种连接形式。

(1) 功率流输入、输出

在梯形图语言中，是指FB及FUN的输入变量与左母线连接、输出变量与右母线连接时BOOL型的输入、 BOOL 型的输出。

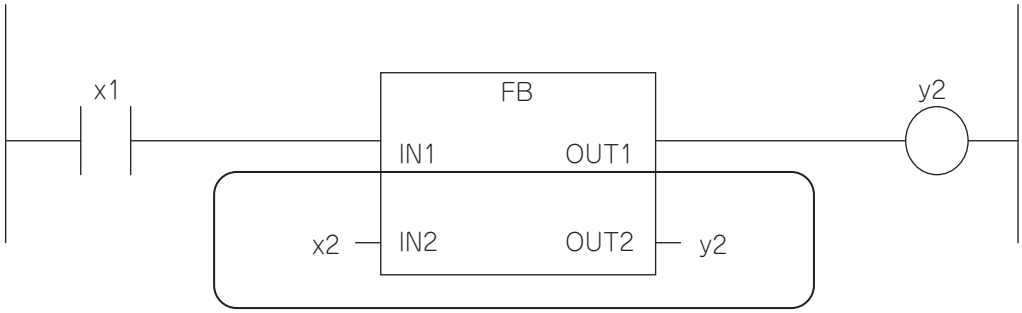
例）

左母线上连接的功率流与接点 （CONTACT）连接，右母线与线圈 （COIL）连接。



(2) 参数输入、参数输出

在梯形图语言中，是指 FB 及 FUN 的输入变量、输出变量不左母线和右母线连接的情况。



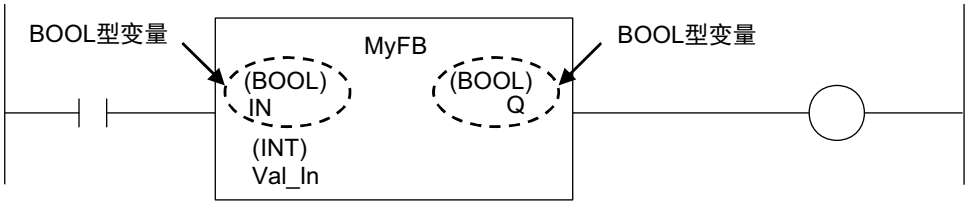
输入及输出参数中可如下指定变量或常数。

FB/FUN 侧变量	输入参数	输出参数
输入变量	可指定变量或常数	—
输出变量	—	只能指定变量
输入输出变量	只能指定变量	只能指定变量

● BOOL 型变量的数量

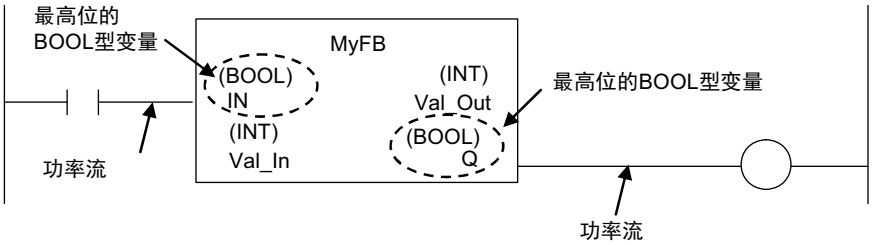
FB 及 FUN 的输入及输出必须有一个以上的 BOOL 型变量 （含 EN、 ENO）。

例）

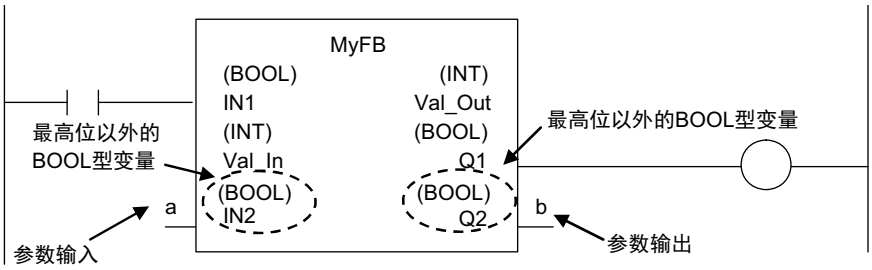


● BOOL 型变量不同位置的连接状态

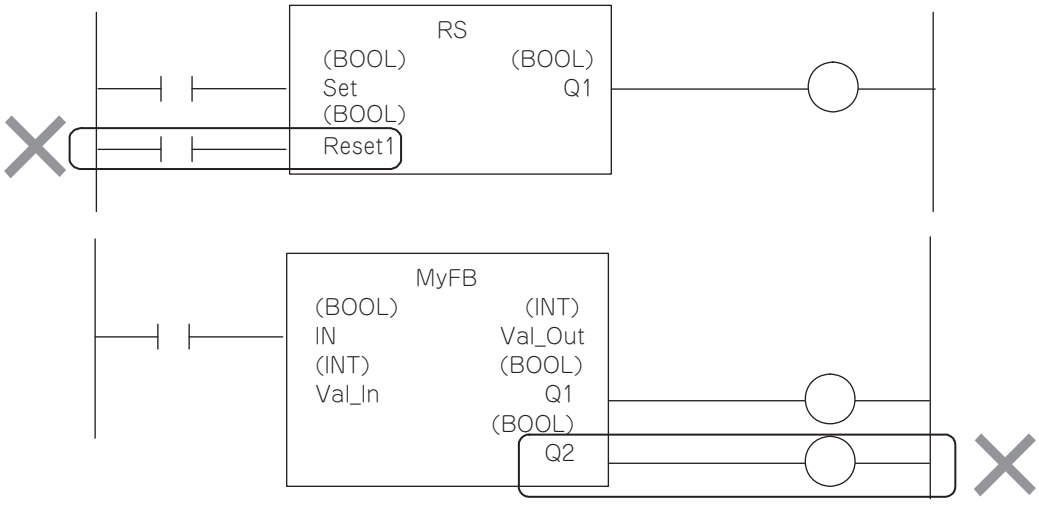
最高位的 BOOL 型变量与左母线及右母线连接。即为功率流输入及功率流输出。



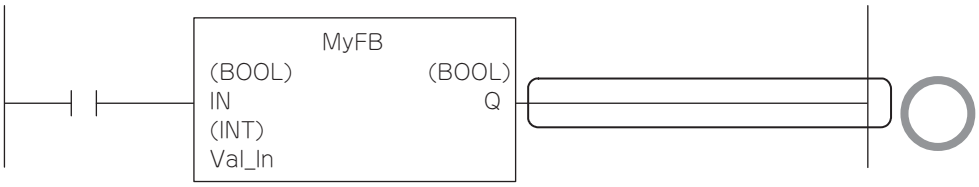
与 FB 及 FUN 连接的功率流输入、功率流输出分别为 1 个。
最高位以外的 BOOL 型变量为参数输入及参数输出。



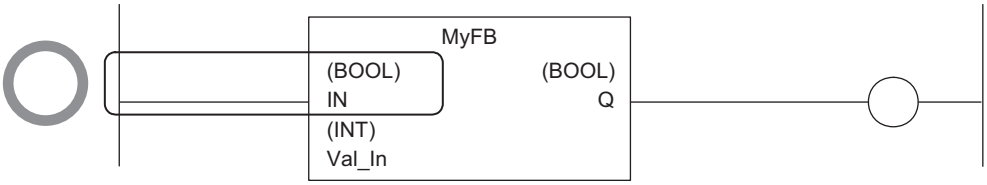
因此，不能像下面这样，将多个 BOOL 型变量连接到左母线或右母线上。



没有 OUT 指令等，与右母线直接连接亦可。

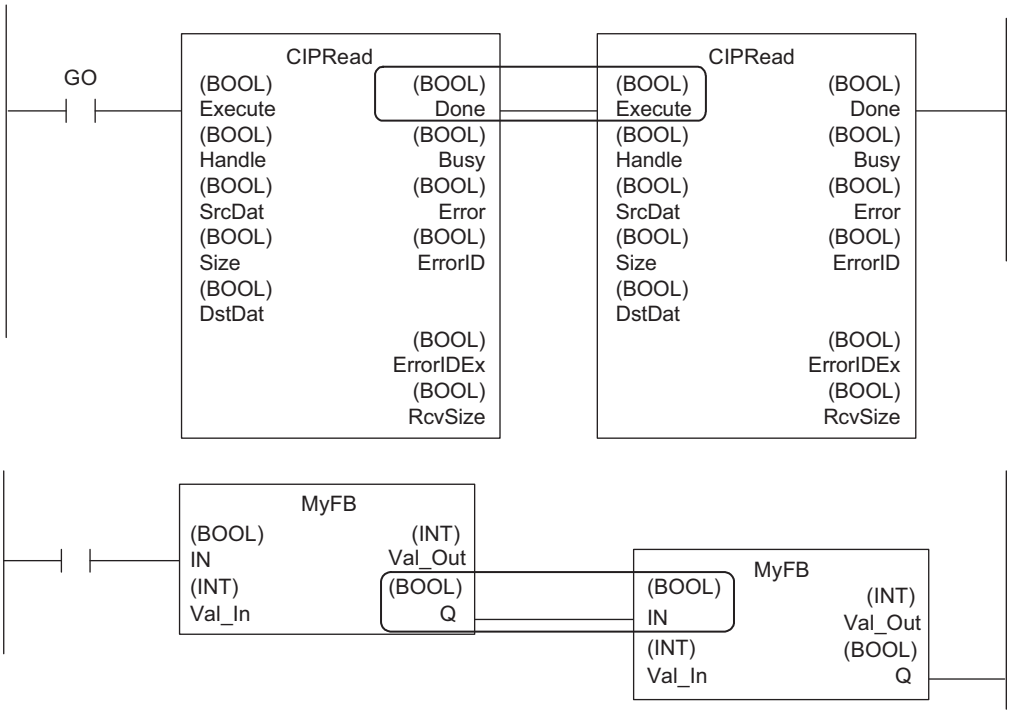


没有 LD 指令等，与左母线直接连接亦可。



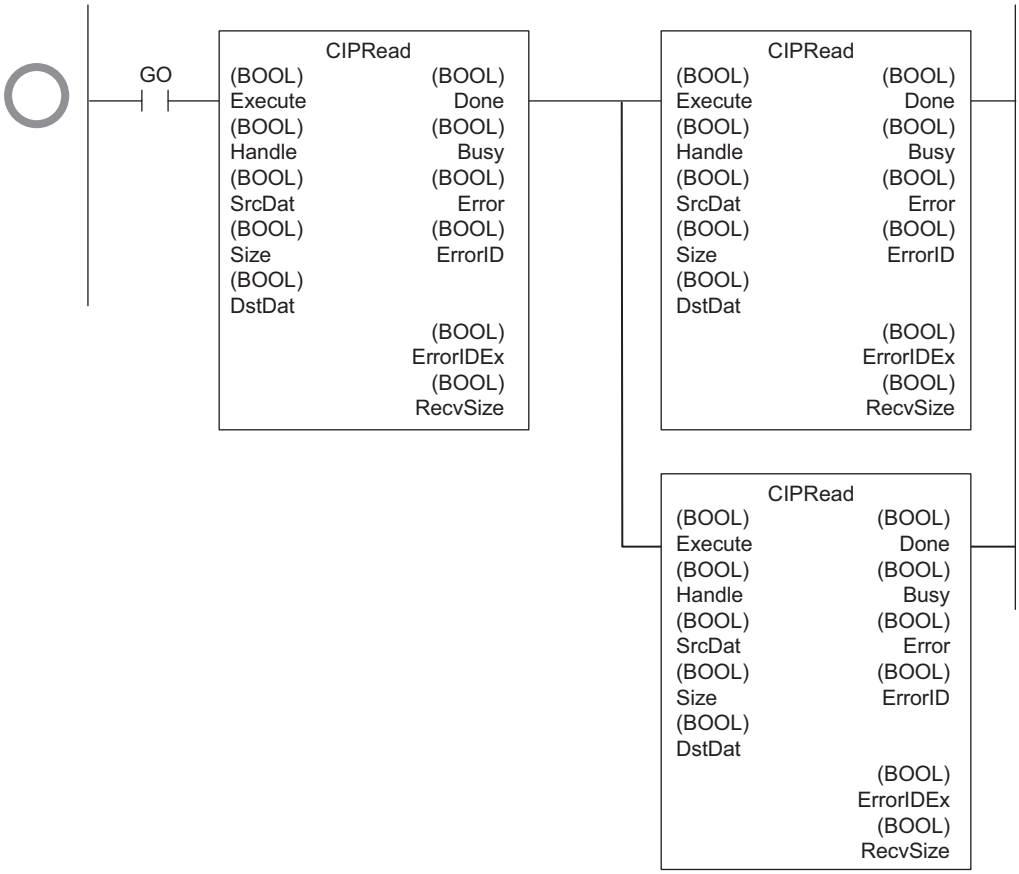
● 级联连接

将 FB/FUN 的输出作为其他 FB/FUN 输入的级联连接只能在功率流输出和功率流输入之间进行。



- 功率流输出可以分支。

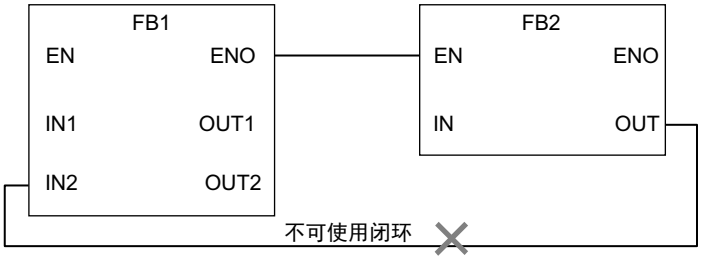
例）



限制

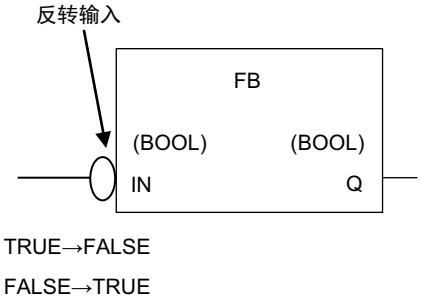
- 不可形成闭环或连接线交叉。

  例）



● 输入的反转

对于 BOOL 型输入变量，可以将值反转后输入。

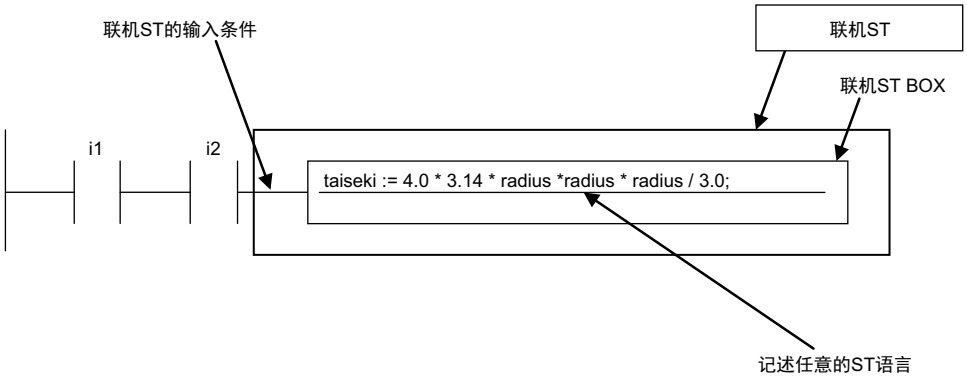


联机 ST

● 概要

联机ST是指在名为“联机ST BOX”的方框（空白的文本输入区域）中记述ST语言的梯形图语言要素。这样，可以方便地在梯形图语言中记述数值运算处理或字符串操作处理。

“ 联机 ST BOX” 的连接线变为输入条件， ST BOX 内的 ST 语言按该输入条件执行。具体如下所示。



此外，联机 ST 将作为以梯形图语言记载的回路要素的一种。所以，与 FB 或 FUN 不同，没有输入变量、输出变量、输入输出变量。

● 联机 ST 中的 ST 语言限制

联机 ST BOX 中，可记述任意的 ST 语言程序。

● 联机 ST 的执行条件

联机 ST 的执行条件如下所示。

状态	动作
输入条件 TRUE 时	按输入条件动作。 输入条件可直接与功率流或左母线连接。 上升沿或下降沿指定在输入条件的接点中指定。
输入条件 FALSE 时	无操作
通过主控执行复位时	无操作

● 联机 ST 中变量的范围

联机 ST 可访问的变量范围与记述有联机 ST 的梯形图程序的 POU 相同。

● 联机 ST 的限制规格

项目	内容
每条回路的联机 ST 数	1 个

6-5-3 ST 语言

ST （Structured Text）语言是指 IEC 61131-3 规定的工业控制 （主要为可编程控制器）的高级语言。使用预备的控制语句、运算符、函数，适用于对梯形图语言中难以记述的数值运算处理进行部分处理。ST 语言有以下特点。

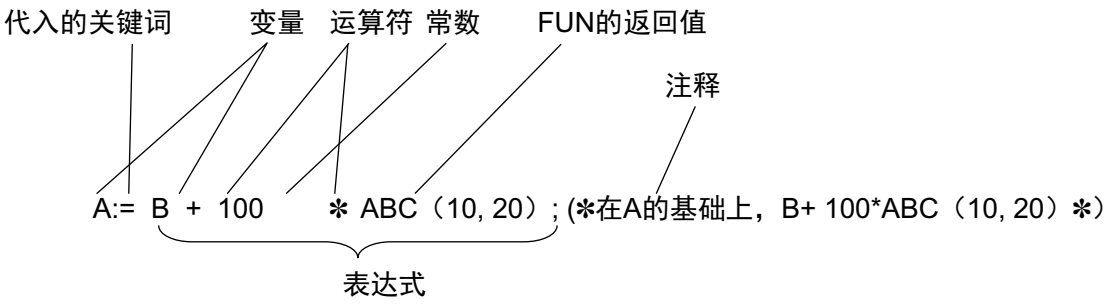
- 备有循环语句、 IF THEN ELSE 等控制语句。
- 利用与 C 语言等高级语言相同的记述或包括汉字在内的文字注释，可创建简单易懂的程序。

```
1
2       // TableNo決定
3       FOR i:=0 TO ItemNum DO
4
5           IF (MinNo[i] <= ItemBox[i]) AND (ItemBox[i] <= MaxNo[i]) THEN   // 範囲内
6               TableNo[i] := ItemBox[i];
7               RangeOK[i] := TRUE;
8
9           ELSIF (ItemBox[i] > MaxNo[i]) THEN                              // 上限超え
10              TableNo[i] := MaxNo[i];
11              RangeOK[i] := FALSE;
12
13          ELSE                                                            // 下限超え
14              TableNo[i] := MinNo[i];
15              RangeOK[i] := FALSE;
16
17          END_IF;
18
19      END_FOR;
```

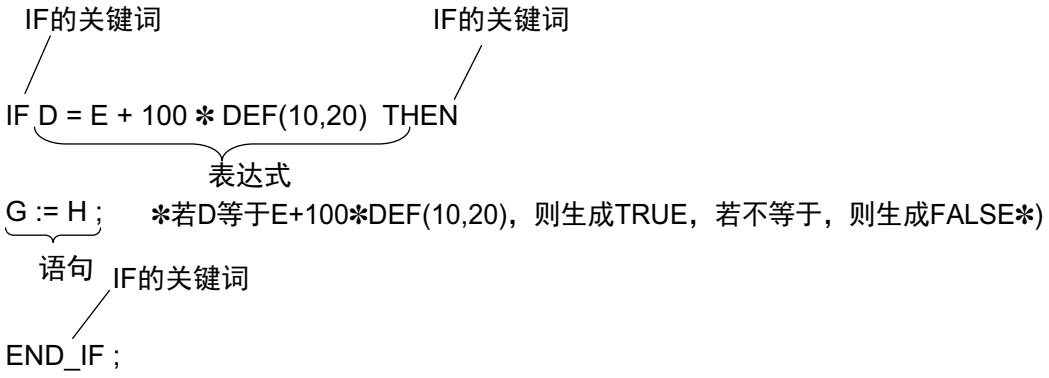
ST 语言的构成

ST 语言的程序由 1 个以上的语句构成。 1 个语句表示 1 个处理。
语句从上往下逐行执行， ST 语言程序 （ST 程序）的处理逐步推进。
语句由关键词和表达式构成。关键词是指表示代入或执行控制的符号或字符串。表达式是指用变量、常数、 FUN 的返回值或将它们与运算符组合，来生成数值的记述。即，语句表示单独结束的处理，而表达式可将生成的数值和关键词组合起来，构成语句。

代入语句的示例：



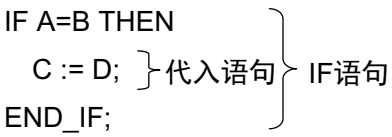
IF 语句的示例：



ST 语言的表现

● 语句的分隔

- 语句必须以半角分号 （;）结束。只是换行将不视为语句的结束。因此，可以多行记述较长的语句。
- 1个语句中带1个半角分号（;）。以下示例中，1个IF语句中有1个代入语句，因此要在各语句的最后记述半角分号 （;）。



● 注释

- 为了便于阅读者更容易理解程序内容，可在程序中记述任意注释。
- 记述为注释的语句不会执行。
- 注释的记述有以下两种。

注释的记述	例	备注
用半角括号和半角星号（* 和 *）括起来。	（*　多行注释 IF ErrCode = 3 THEN 　Value := 1000; END_IF; 为止　*）	可记述多行注释。 注释不可嵌套。
以两个半角斜杠 // 开始，以换行结束。	// 注释 //　A :=　SIN （X） ^2 ；	只能逐行注释掉代码。

● 空格 （空白）、换行、制表符

- 可将空格 （空白）、换行、制表符或它们的任意组合以任意的数量记述在任意的位置中。
  因此，考虑到可读性，可在运算符 / 关键词和表达式之间换行，或在语句前面输入空格或制表符。
- 运算符 / 关键词和变量名称之间请务必输入空格 （空白）、换行、制表符。

例）■表示应该输入空格 （空白）、换行、制表符的地方。

```
IF ■ A>0 ■ THEN ■ X:=10;
ELSE
X:=0;
END_IF;
```

● 小写 / 大写字母、半角 / 全角

- 运算符 / 关键词及变量名称不区分大小写 （哪种都可以）。
- 请输入半角的运算符 / 关键词。若以全角输入，将发生语法错误。

● 变量名称的禁用字符

变量名称的限制请参考📖「6-3-12 变量名称等程序相关名称的限制」（P.6-77）。

● 字符串

字符串的限制请参考📖「6-3-12 变量名称等程序相关名称的限制」（P.6-77）。

ST 语言的关键词和运算符

● 语句的关键词

关键词	含义	例
:=	代入	d := 10;
	FB/FUN 的调用	FBname （para1 := 10, para2 := 20） ; 详情请参考📖「FB 调用」（P.6-115）
RETURN	返回	
IF	选择	IF d < e THEN f := 1; ELSIF d = e THEN f :=2; ELSE f := 3; END_IF;
CASE	多方向分支	CASE f OF 1: g :=11; 2: g :=12; ELSE g :=0; END_CASE;
FOR	重复 （1）	FOR i := 100 TO 1 BY -1 DO Val[i] := i; END_FOR;
WHILE	重复 （2）	WHILE Val < MaxVal DO Val := Val + 1; END_WHILE;
REPEAT	重复 （3）	REPEAT Val := Val + 1; UNTIL （ Val > 4 ） END_REPEAT;

EXIT	循环的中断	FOR i := 1 TO 100 DO
		  FOR j := 1 TO 10 DO
		  IF Val[i, j]>100 THEN EXIT;
		  END_IF;
		  END_FOR;
		END_FOR;

;	空语句	Val[i] := i
		  ; （* 空语句 *）
		
		WHILE （Var <>0）DO
		  ; （* 空语句 *）
		END_WHILE;

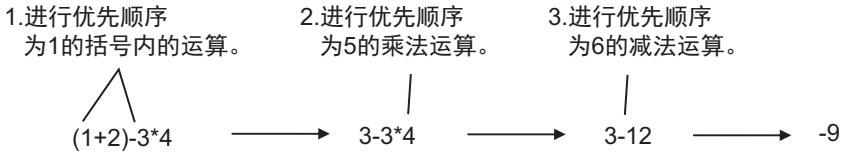
（* 文本 *）	注释	（* 注释掉多行 IF MyFun （ErrorCode）= 3 THEN ReturnValue := GetDetail （） ; END_IF; 为止 *）
// 文本	注释	A := SIN （ X ）^ 2 + COS （ Y ）^2 + 10; // A := SIN （ X ）^ 2 + COS （ Y ）^2 + 5;

● 运算符

下表表示运算符的种类和优先顺序。

一个表达式中有不同优选顺序的运算符时，先执行优先顺序高的运算。一个表达式中最多可记述 64 个运算符。

例） X:= （1+2） -3*4; 时，变量 X 中代入 -9。



运算	运算符	记述示例和评估值	优先顺序
括号	（）	（1+2） * （3+4） 值 21	1
FB、 FUN 的调用		FUN1(FUN2(Var2A, Var2B), Var1B) 分层调用了 FB、 FUN 时，将先执行下面的层级。上述示例中，先执行 FUN2，再执行 FUN1。	2
符号	＋ , －	+100 -100	3
否定	NOT	NOT TRUE 值 FALSE	
乘幂	**	-2**2 值 4 负号的优先顺序高于乘幂运算符。因此，上述示例中，-2**2=(-2)**2，值为 4。 2**3**2 值 64 乘幂运算符有多个重叠时，从左开始运算。上述示例中，2**3**2=(2**3)**2，值为 64。	4
乘法	*	100*200 值 20000	5
除法	/	100/200 值 0.5	
余数	MOD	10 MOD 7 值 3 -17 MOD 6 值 -5 -17 MOD （-6） 值 -5 17 MOD 6 值 5 17 MOD （-6） 值 5	
加法	＋	100+200 值 300	6
减法	－	100-200 值 -100	

运算	运算符	记述示例和评估值	优先顺序
比较	＜，＞，＜＝，＞＝	100<200 若比较结果正确，则值为 TRUE，若不正确，则值为 FALSE。 上述示例中， 100<200 是正确的，所以值为 TRUE	7
等于	＝	100=200 若两个值相等，则值为 TRUE，若不相等，则值为 FALSE。 上述示例中， 100=200 是错误的，所以值为 FALSE	8
不等于	＜＞	100<>200 若两个值不相等，则值为 TRUE，若相等，则值为 FALSE。 上述示例中， 100<>200 是不相等的，所以值为 TRUE	
逻辑与	AND,&	所有位执行 1 个位的逻辑与。 1 个位的逻辑与结果如下所示。 0 AND 0 = 0 0 AND 1 = 0 1 AND 0 = 0 1 AND 1 = 1 0101 AND 1100 值 0100	9
异或	XOR	所有位执行 1 个位的异或。 1 个位的异或结果如下所示。 0 XOR 0 = 0 0 XOR 1 = 1 1 XOR 0 = 1 1 XOR 1 = 0 0101 XOR 1100 值 1001	10
逻辑或	OR	所有位执行 1 个位的逻辑或。 1 个位的逻辑或结果如下所示。 0 OR 0 = 0 0 OR 1 = 1 1 OR 0 = 1 1 OR 1 = 1 0101 OR 1100 值 1101	11

使用注意事项

若分层记述同一个函数，可能不按预期动作。请务必如以下示例所示，分开记述语句。

```
例 ) ×    out    := MyFunc( In1:=x1, In2:=MyFunc( In1:=x2, In2:=x3 ) );

     ○    tmp    := MyFunc( In1:=x2, In2:=x3 );
          out    := MyFunc( In1:=x1, In2:=tmp );
```

使用注意事项

运算符的优先顺序可能因规格或制造商不同而异。特别是乘幂运算符的优先顺序，需要注意。为了确保按计划的顺序进行运算，建议使用括号标明各运算的优先顺序。
例） X:= -2**3**4; 记述为 X:= ((-2)**3)**4; 。

参考

根据数据类型进行运算。
例如，整数型之间的运算以整数型进行。因此，如果以 INT 型变量 A ＝ 3、B ＝ 2 进行 A/B 运算，运算结果不是 1.5，将舍去小数点后的数位，变为 1。所以，（A/B）*2 的运算结果不是 3，而是 2。

● 各运算符的操作对象中可使用的数据类型

1 个运算符中使用的操作对象全部为同一数据类型时，可将以下数据类型设定为操作对象。
但是，若运算符中设定了不同数据类型的操作对象，将发生默认的 CAST。
关于默认的 CAST，请参考📖「默认的 CAST」（P.6-121）。

数据类型	代入运算符	参数设定运算符	数值运算符	余数运算符	乘幂运算符	比较运算符	等价运算符	逻辑运算	正负符号
	:=	:= =>	+ - * /	MOD	**	< <= => >	= <>	NOT AND & OR XOR	+ -
BOOL 型	○	○	×	×	×	×	○	○	×
位列型	○	○	×	×	×	×	○	○	×
整数型	○	○	○	○	○	○	○	×	○
实型数据	○	○	○	×	○	○	○	×	○
持续时间型	○	○	×	×	×	×	×	×	○
日期型	○	○	×	×	×	×	×	×	×
时刻型	○	○	×	×	×	×	×	×	×
日期时刻	○	○	×	×	×	×	×	×	×
字符串型	○	○	×	×	×	× （*）	× （*）	×	×
列举型	○	○	×	×	×	×	○	×	×
母结构体	○	○	×	×	×	×	×	×	×
母排列	○	○	×	×	×	×	×	×	×

○ : 可以
× : 链接时将发生异常。
*字符串型变量的比较请使用指令 （EQascii 等）执行，而不是运算符。

ST 语言的语句

● 代入

概要：

将右边 （表达式的值）代入左边 （变量）。

保留字：

:=

冒号 （:）和等号 （=）的组合

语句的结构：

```
< 变量 >:=< 表达式、变量或常数 >;
```

用途：

将值代入变量时使用。
例如，用于初始值的设定、计算结果的保存等。

说明：

将 < 表达式的值 > 代入 （保存到） < 变量 > 中。

例：

例 1）将表达式 X+1 的结果代入变量 A 中

```
A:=X+1;
```

例 2）将变量 B 的值代入变量 A 中

```
A:=B;
```

例 3）将常数 10 代入变量 A 中

```
A:=10;
```

注意事项：

- 代入源的数据类型需要与代入目标的数据类型一致，或使用默认 CAST 中允许的数据类型组合。如果不是，链接时将发生异常。
- 代入源为字符串时，代入目标 STRING 型变量的大小应大于代入源字符串的大小。若非如此，将发生异常。
- 变量为 STRING 型时，如果（左边的变量大小）＞（右边变量中保存的字符串大小），则可以代入。例）

以下情况下可以代入。

- 变量表：

变量名称	数据类型	大小
Var1	STRING	10
Var2	STRING	20

- 用户程序：
  Var2 :='ABC';
  Var1 := Var2;

不可代入到整个联合体型变量中。请以成员为单位代入。

● RETURN

概要：

根据使用 ST 的程序种类，如下所示。
＜ ST 程序＞
中途结束 ST 程序，执行下一个程序。
＜在 FB 的实例内 FUN 中使用的 ST ＞
中途强制结束 FB、 FUN，执行 FB、 FUN 调用源的下一个指令。
＜联机 ST ＞
结束包括 RETURN 语句的联机 ST 的 POU。

保留字：

RETURN

语句的结构：

```
RETURN;
```

用途：

强制结束正在执行的程序、 FB 或 FUN 时使用。

● IF （1 个条件）

概要：

某个条件成立时，执行指定的语句。未成立时，执行其他语句。
条件成立、未成立时，表现如下。
真：条件成立时
假：条件未成立时

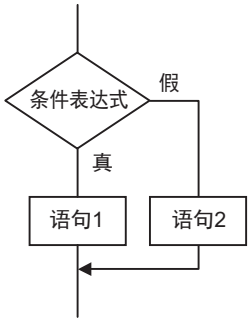
保留字：

IF、 THEN、（ELSE）、 END_IF
注：ELSE 可省略。

语句的结构：

```
IF < 条件式 >  THEN
  < 语句 1>;
ELSE
  < 语句 2>;
END_IF;
```

处理流程图：



用途：

根据某个条件 （条件式）是否成立来区分时使用。

说明：

< 条件式 > ＝真时，执行 < 语句 1>。
< 条件式 > ＝假时，执行 < 语句 2>。

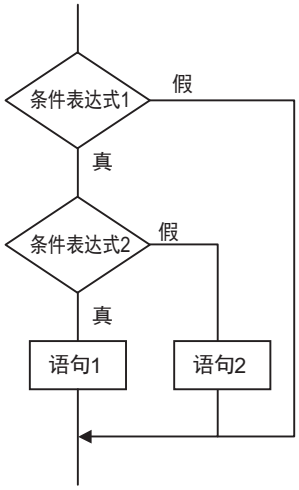
注意事项：

- IF 必须与 END_IF 成对使用。
- < 条件式 > 中可记述评估结果为真或假的表达式 （例如 IF A>10）、 BOOL 型变量 （例如 IF A）。
- < 语句 1>、 < 语句 2> 中可跨多行记述程序。此时，语句和语句之间用 ; 分隔。

例） 在 < 语句 1> 中添加了 IF 语句时

```
IF < 条件式 1>  THEN
    IF < 条件式 2> THEN
            < 语句 1>;
    ELSE
            < 语句 2>;
            END_IF;
    END_IF;
```

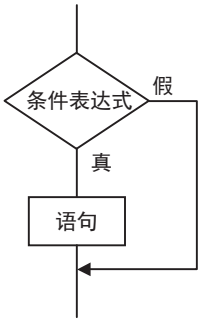
处理流程图：



如上所示， ELSE 与之前的 THEN 对应。

- < 语句 1>、 < 语句 2> 中，均可执行 1 个以上的语句。此时，语句和语句之间用 ; 分隔。
- ELSE 语句可省略。省略时，如果 < 条件式 >= 假，将不执行任何内容。

处理流程图：



例：

例 1）变量 A>0 为真时，在变量 X 中代入数值 10。变量 A>0 为假时，在变量 X 中代入数值 0。

```
IF A>0 THEN
    X:=10;
ELSE
    X:=0;
END_IF;
```

例 2）变量 A>0 且变量 B>1 为真时，在变量 X 中代入数值 10，在变量 Y 中代入数值 20。变量 A>0 且变量 B>1 为假时，在变量 X 中代入数值 0，在变量 Y 中代入数值 0。

```
IF A>0 AND B>1 THEN
    X:=10;Y:=20;
ELSE
    X:=0;Y:=0;
END_IF;
```

例 3）BOOL 型变量 A ＝ TRUE 时，在变量 X 中代入数值 10。变量 A=FALSE 为假时，在变量 X 中代入数值 0。

```
IF A THEN X:=10;
ELSE X:=0;
END_IF;
```

● IF （多个条件）

概要：

某个条件成立时，执行指定的语句。未成立时且其他条件成立时，执行其他语句。均未成立时，执行其他语句。

条件成立、未成立时，表现如下。

真：条件成立时

假：条件未成立时

保留字：

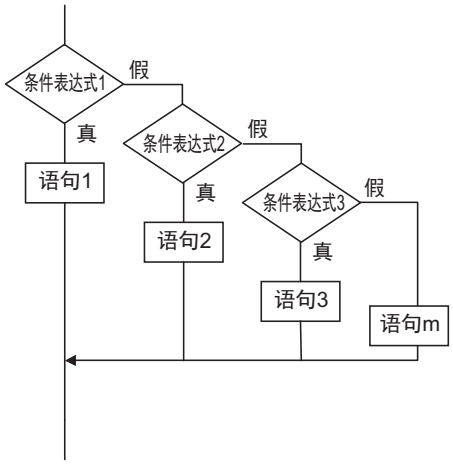
IF、 THEN、 ELSIF、（ELSE）、 END_IF

注：ELSE 可省略。

语句的结构：

```
IF < 条件式 1>  THEN < 语句 1>;
    ELSIF < 条件式 2>  THEN < 语句 2>;
    ELSIF < 条件式 3>  THEN < 语句 3>;
            ·
            ·
            ·
    ELSIF < 条件式 n>  THEN < 语句 n>;
ELSE< 语句 m>;
END_IF;
```

处理流程图：



用途：

根据多个条件 （条件式）是否成立来区分时使用。

说明：

< 条件式 1> ＝真时，执行 < 语句 1>。
< 条件式 1> ＝假时，
如果 < 条件式 2> ＝真，执行 < 语句 2>。
< 条件式 2> ＝假时，
如果 < 条件式 3> ＝真，执行 < 语句 3>。
・
・
・
如果 < 条件式 n> ＝真，执行 < 语句 n>。
不符合任一条件时，执行 < 语句 m>。

注意事项：

- IF 必须与 END_IF 成对使用。
- < 条件式□ > 中记述评估结果为真或假的表达式。例） IF （A>10）
  此外，< 条件式□ > 中除了表达式外，还可只指定 BOOL 型变量（包括 FUN 的 BOOL 型返回值）。
  此时，变量的值为 TRUE ＝评估结果真， FALSE ＝评估结果假。
- < 语句□ > 中可跨多行记述程序。此时，语句和语句之间用 ; 分隔。
- ELSE 语句可省略。省略时，若不符合任一条件，则不执行任何内容。

例：

变量 A>0 为真时，在变量 X 中代入数值 10。
变量 A>0 为假时，如果变量 B ＝ 1 为真，则在变量 X 中代入数值 1。
变量 A>0 为假时，如果变量 B ＝ 2 为真，则在变量 X 中代入数值 2。
不符合任一条件时，在变量 X 中代入数值 0。

```
IF A>0 THEN X:=10;
ELSIF B=1 THEN X:=1;
ELSIF B=2 THEN X:=2;
ELSE X:=0;
END_IF;
```

● CASE

概要：

执行拥有与整数式的值一致的整数选择值的语句。

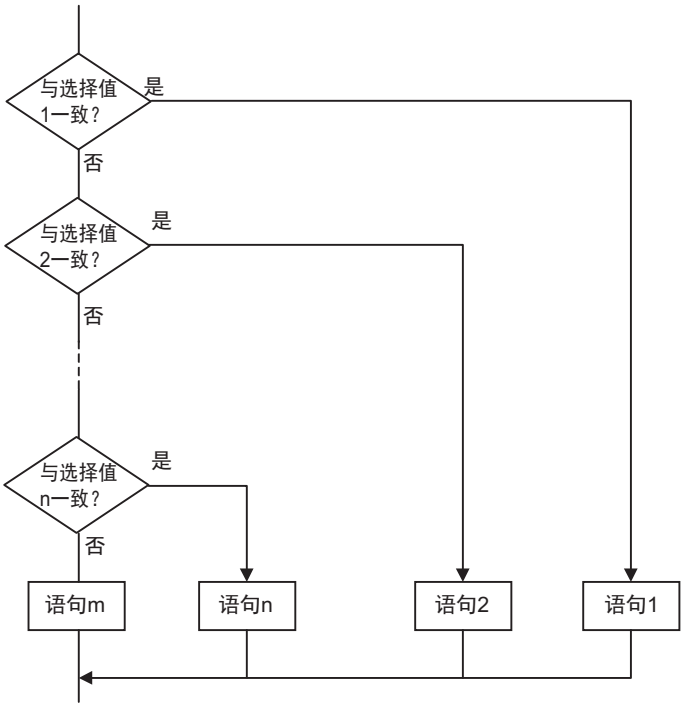
保留：

CASE

语句的结构：

```
CASE < 整数式 > OF
    < 整数式的值 1>:< 语句 1>;
    < 整数式的值 2>:< 语句 2>;
        ·
        ·
        ·
    < 整数式的值 n>:< 语句 n>;
ELSE< 语句 m>;
END_CASE;
```

处理流程图：



用途：

根据某个整数的值进行区分时使用。

说明：

< 整数式 > 与 < 整数式的值 n> 一致时，执行 < 语句 n>。

< 整数式 > 与任一 < 整数式的值 n> 都不一致时，执行 < 语句 m>。

注意事项：

- CASE 必须与 END_CASE 成对使用。
- 在 < 整数式 > 中记述
  - ・整数型或列举型变量　（例：abc）
  - ・整数式　（例：abc+def）
  - ・将整数值作为返回值的 FUN　（例：xyz （））。
- < 语句□ > 中可跨多行记述程序。此时，语句和语句之间用 ; 分隔。
- 在 < 整数式的值 n> 中指定多个整数的 OR 逻辑时，数值用 ,（逗号）分隔。指定连续的整数时，开始整数和结束整数之间以 ..（2 个句点）分隔。

例 1）可为特定的整数值指定条件，或为多个整数值指定相同的条件。

```
CASE A OF
    1: X:=1;  ------------------  变量A为1时，在变量X中代入数值1
    2: X:=2;  ------------------  变量A为2时，在变量X中代入数值2
    3: X:=3;  ------------------  变量A为3时，在变量X中代入数值3
ELSE
    X:=0; ----------------------  不符合上述任一情况时，在变量X中代入0
END_CASE;
```

```
CASE A OF
    1: X:=1;  ------------------  变量A为1时，在变量X中代入数值1
    2,5: X:=2; -----------------  变量A为2或5时，在变量X中代入数值2
    6..10: X:=3; ---------------  变量A为6～10中的任意一个时，在变量X中代入数值3
    11,12,15..20: X:=4; --------  变量A为11、12或15～20时，在变量X中代入数值4
ELSE
    X:=0;  ---------------------  不符合上述任一情况时，在变量X中代入0
END_CASE;
```

例 2）< 整数式 > 中可记述整数型变量・表达式・FUN 的返回值，以及列举型变量・FUN 的返回值。示例如下。

- 整数 · 列 · 型变量的示例

```
CASE ColorVar OF
    RED:
        X := 0;
    BLUE:
        X := 1;
    ELSE
        X := 2;
END_CASE;
```

- 整数式示例

```
CASE （a1 + a2）OF
    0:
        X := 0;
    1:
        X := 1;
    ELSE
        X := 2;
END_CASE;
```

- 整数 · 列举型 FUN 的返回值示例

```
CASE FUN( ) OF
    0:                    根据FUN( )的返回值分支
        X := 10;
    1:
        X := 11;
    ELSE
        X := 12;
END_CASE;
```

CASE 中可使用的数据类型

分类	数据类型		< 整数式 >
基本数据型	整数型		○
	布尔型、位列型、实数型、持续时间型、日期型、时刻型、日期时刻型、字符串型		×
数据类型的指定	排列指定	排列整体	×
		要素	○ 仅可指定整数型、列举型
派生数据型	结构体型	结构体整体	×
		成员	○ 仅可指定整数型、列举型
	联合体型	联合体整体	×
		成员	○ 仅可指定整数型、列举型
	列举型		○

● FOR

概要：

变量 （以下称 “ 反复变量 ”）从某个值变为某个值前，重复执行某语句。
条件成立、未成立时，表现如下。
　真：条件成立时
　假：条件未成立时

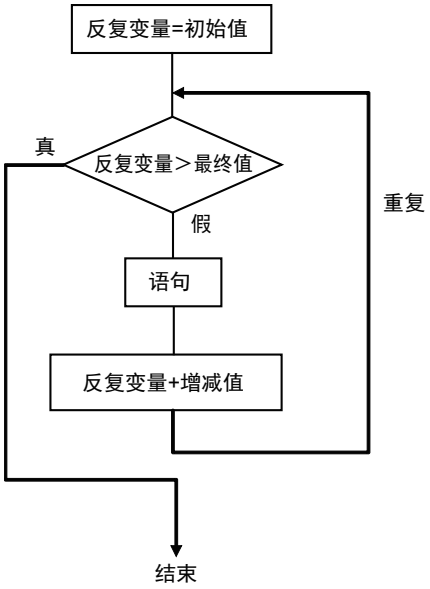
保留字：

FOR、 TO、（BY）、 DO、 END_FOR
注：BY 可省略。

语句的结构：

```
FOR < 反复变量 > := < 初始值 >  TO < 最终值 >  BY < 增减值 >  DO
  < 语句 >;
END_FOR;
```

处理流程图：



用途：

已事先知道重复次数时使用。
特别是根据某个反复变量的值切换排列指定的变量各要素并指定时，为有效。

说明：

评估 < 初始值 >、 < 最终值 >、 < 增减值 > 并决定。
< 反复变量 > 为 < 初始值 > 时，执行 < 语句 >。
执行后，在<反复变量>中加上<增减值>中的值，如果<反复变量>＜<最终值>的值，则执行<语句>。
执行后，在<反复变量>中加上<增减值>中的值，如果<反复变量>＜<最终值>的值，则执行<语句>。
重复上述内容。
如果 < 反复变量 > > < 最终值 >，则结束。
此外， < 增减值 > 为负数时，大小比较符号的方向相反。

注意事项：

- 反复变量为带符号整数型时， < 增减值 > 可指定为负值。
- FOR 必须与 END_FOR 成对使用。
- 反复变量中，以最终值执行处理后，变为 （最终值 + 增减值），结束反复处理。
  例）在以下 ST 语句中， FOR 语句结束时， i 的值为 101。

```
FOR i:=0 TO 100 DO
    X[i]:=0;
END_FOR;
// 此时 i 为 101
```

- 请勿记述要在 FOR 语句内直接变更反复变量的处理。可能发生意外动作。
  例）

```
FOR i:=0 TO 100 BY 1 DO
    X[i]:=0;
    i:=i+INT#5;
END_FOR;
```

- < 语句 > 中可跨多行记述程序。此时，语句和语句之间用 ; 分隔。
- BY< 增减值 > 可省略。省略时， BY 作为 1 执行。
- < 初始值 > 、< 最终值 >、< 增减值 > 中可指定整数型（SINT、INT、DINT、LINT、USINT、UINT、UDINT 或 ULINT）的变量或整数值。此外，也可指定 FUN 的整数型返回值。
  例 1）反复变量从 n ＝ 0 开始，以 5 为单位直至 50，不断重复，在排列指定变量 SP[n] 中代入 100。

```
FOR n := 0 TO 50 BY 5 DO
    SP[n] := 100;
END_FOR;
```

例 2）计算排列指定的变量 DATA[n] 的要素 DATA[1] 到 DATA[50] 的合计值，代入变量 SUM 中。

```
IF a THEN
    FOR n := 0 TO 50 BY 1 DO
        DATA[n]:= 1 ;
    END_FOR;

    FOR n := 0 TO 50 BY 1 DO
        SUM:= SUM + DATA[n] ;
    END_FOR;

    a:=FALSE;
END_IF;
```

例 3）计算排列指定的变量 DATA[n] 的要素 DATA[1] 到 DATA[50] 的合计值，代入变量 SUM 中。最大值代入变量 MAX 中，最小值代入变量 MIN 中。此外， DATA[n] 的值在 0 ～ 1000 的范围内。

```
MAX :=0;
MIN :=1000;
FOR n :=1 TO 50 BY 1 DO
    IF DATA[n] > MAX THEN
        MAX :=DATA[n];
    END_IF;
    IF DATA[n] < MAX THEN
       MIN :=DATA[n];
    END_IF;
END_FOR;
```

- 在<反复变量>的<初始值>上加<增减值>，超过<最终值>前，若包括FOR语句在内的程序执行时间超出任务周期设定时间，将发生任务周期超限。
  - 逻辑上 < 反复变量 > 不会达到 < 最终值 > 时

例）

```
FOR i := 0 TO 100 BY 1 DO
    intArray[i] := i;
    i := INT#50;  ------------------------  变为无限循环，发生任务周期超限
END_FOR;
```

例）

```
FOR i := 0 TO 100 BY 0 DO        --------  变为无限循环，发生任务周期超限
    ;
END_FOR;
```

- < 反复变量 > 超过 < 最终值 > 时，发生溢出 / 下溢的情况

  例）

  变量 i 为 USINT 型时，可表现最终值 256。将视为 0，变为无限循环。

```
FOR i := 0 TO 254 BY 2 DO
    INTArray[i] := i;
END_FOR;
```

版本相关信息

Sysmac Studio Ver.1.08 以上时， < 最终值 > 和 < 增减值 > 中可记述运算式。
但是，< 最终值 > 和 < 增减值 > 的评估在 FOR 循环内的运算前执行。开始 FOR 循环内的运算后，< 最终值 > 和 < 增减值 > 不会发生变化。
下图情况下，< 最终值 > 为 10、< 增减值 > 为 3。即使开始 FOR 循环内的运算、变量 A 或变量 C 的值发生变化， < 最终值 > 保持 10， < 增减值 > 保持 3。

```
A := INT#1;
B := INT#2;
C := INT#10;

FOR i := 0 TO C BY A+B DO
    INTArray[i] := i;
    A := B + i;
    C := C + i;
END_FOR;
```

Sysmac Studio Ver.1.07 以上时，若在 < 最终值 > 和 < 增减值 > 中记述运算式，链接时将发生异常。敬请注意。

FOR 中可使用的数据类型

分类	数据类型		< 反复变量 >、 < 初始值 >、 < 最终值 >、< 增减值 >*1
基本数据型	布尔型、位列型、实数型、持续时间型、日期型、时刻型、日期时刻型、字符串型		×
	整数型		○
数据类型的指定	排列指定	排列整体	×
		要素	○ 仅可指定整数型、列举型*2
派生数据型	结构体型	结构体整体	×
		成员	○ 仅可指定整数型、列举型 *2
	联合体型	联合体整体	×
		成员	×
	列举型		○ *2

*1. 请统一＜反复变量＞、< 最终值 >、< 增减值 > 三者的数据类型。若数据类型不一致，Sysmac Studio 的程序链接时将发生异常。
*2. ＜反复变量＞、 < 最终值 >、 < 增减值 > 中不可使用列举型。

● WHILE

概要：

某条件式为真期间，重复执行指定的语句。

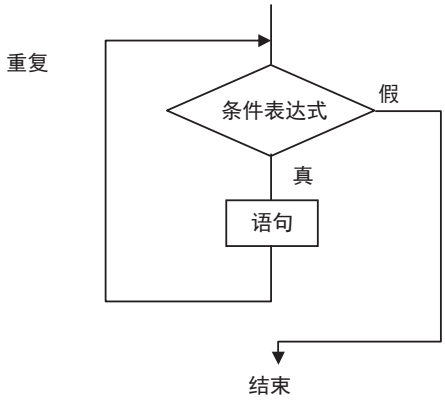
保留字：

WHILE、 DO、 END_WHILE

语句的结构：

```
WHILE < 条件式 > DO
    < 语句 >;
END_WHILE;
```

处理流程图：



用途：

不知道重复次数 （根据条件，不知道几次）时，某条件式成立期间，重复执行某处理时使用。
仅在条件式成立时执行某处理时使用 （前判定的重复）。

说明：

执行 < 语句 > 前，对 < 条件式 > 进行评估。
< 条件式 > 为真时，执行 < 语句 >。然后，重新评估 < 条件式 >。重复执行上述内容。
< 条件式 > 为假时，不执行 < 语句 >，结束 < 条件式 > 的评估。

注意事项：

- WHILE 必须与 END_WHILE 成对使用。
- 执行 < 语句 > 前，若 < 条件式 > 为假，将不执行 < 语句 > 并结束。
- < 语句 1>、 < 语句 2> 中可跨多行记述程序。此时，语句和语句之间用 ; 分隔。
- < 语句 > 中，可执行 1 个以上的语句。此时，语句和语句之间用 ; 分隔。
- < 条件式 > 中除了表达式外，还可只指定 BOOL 型变量 （包括 FUN 的 BOOL 型返回值）。

例 1）计算 7 的倍数且大于 1000 的数字，代入变量 A 中。

```
A := 0;
WHILE A <= 1000 DO
    A := A+INT#7;
END_WHILE;
```

例 2）变量 X ＜ 3000 期间，将 X 的值乘以 2，然后将该值代入排列指定的变量 DATA[1] 中。然后，将 X 的值再乘以 2，将该值代入排列指定的变量 DATA[2] 中。重复上述内容。

```
n := 1;
X := 1;
WHILE X < 3000 DO
    X:= X*INT#10#2;
    DATA[n]:= X;
    n := n+INT#1;
END_WHILE;
```

- < 条件式 > 的撰写不正确时，程序执行时间将延长，可能发生任务周期超限。

例）

```
boolVar := TRUE;
WHILE boolVar DO
intVar := intVar + INT#1;
END_WHILE;
```

● REPEAT

条件成立、未成立时，表现如下。

真：条件成立时

假：条件未成立时

概要：

执行某语句，在某条件式变为真前，重复该语句。

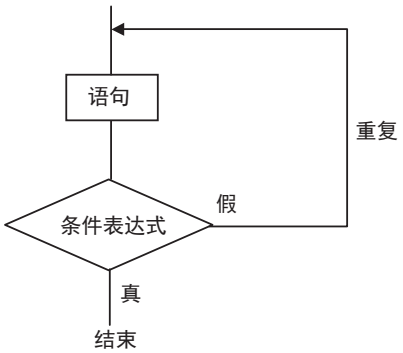
保留字：

REPEAT、 UNTIL、 END_ REPEAT

语句的结构：

```
REPEAT
  < 语句 >;
UNTIL < 条件式 >
END_REPEAT;
```

处理流程图：



用途：

不知道重复次数（根据条件，不知道几次）时，某处理后的条件式成立期间，重复执行该处理时使用。
根据某处理的执行结果，判断是否重复时使用 （后判定的重复）。

说明：

< 语句 > 无条件最先执行。然后，评估 < 条件式 >。
< 条件式 > 为假时，重新执行 < 语句 >。
< 条件式 > 为真时，不执行 < 语句 >，直接结束。

注意事项：

- REPEAT 必须与 END_REPEAT 成对使用。
- 执行 < 语句 > 前，即使 < 条件式 > 为真，也会执行 < 语句 >。
  即， < 语句 > 至少会执行 1 次。
- < 语句 > 中，可跨行记述定为语句关键词的语句。此时，语句和语句之间用 ; 分隔。
- < 条件式 > 中除了表达式外，还可只指定 BOOL 型变量 （包括 FUN 的 BOOL 型返回值）。

例） 1 到 10 的数字相加，然后代入变量 TOTAL 中。

```
A := 1;
TOTAL := 0;
REPEAT
   TOTAL := TOTAL + A;
   A := A+INT#1;
UNTIL A>10
END_REPEAT;
```

- < 条件式 > 的撰写不正确时，程序执行时间将延长，可能发生任务周期超限。

例）

```
intVar := INT#1;
REPEAT
      intVar := intVar + INT#1;
UNTIL intVar = INT#0
END_REPEAT;
```

● EXIT

概要：

仅在重复语句 （FOR 语句、 WHILE 语句、 REPEAT 语句）中使用，结束重复语句。
若在 IF 语句中使用，某条件成立时，可在中途跳过重复语句。

保留字：

EXIT

语句的结构 （例：在 IF 语句中使用）：

```
FOR （WHILE、 REPEAT）语句
            ・
            ・
            ・
IF< 条件式 > THEN EXIT;
END_IF;
            ・
            ・
            ・
END_FOR （WHILE、 REPEAT） ;
```

用途：

结束条件成立前，要强制结束重复处理时使用。

说明 （例：在 IF 语句中使用）：

< 条件式 > 为真时，强制结束重复语句（FOR 语句、WHILE 语句、REPEAT 语句），EXIT 后面记述的语句将不执行。
（注）1 < 条件式 > 中除了表达式外，还可只指定 BOOL 型变量。
2 执行 < 语句 > 前，即使 < 条件式 > 为真，也会执行 < 语句 >。

例：

例 1）变量从 n ＝ 1 开始，以 1 为单位直至 50，不断重复，在排列指定变量 DATA[n] 中加 n。但是，DATA[n] 超过 100 后，将结束。

```
IF A THEN
    DATA[3] :=98;
    FOR n := 1  TO 50 BY 1 DO
        DATA[n] := DATA[n] + n;
        IF DATA[n] > 100 THEN EXIT;
        END_IF;
    END_FOR;
    A :=FALSE;
END_IF;
```

● FB 调用

概要：

调用 FB。

保留字：无

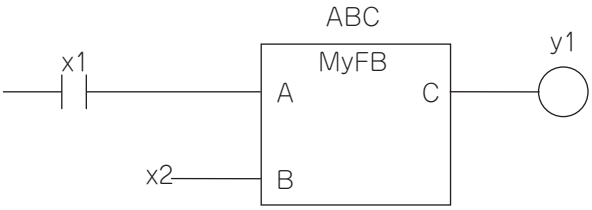
语句的结构：

FB 的实例名称后面，在（ ）内记述参数指定（将指定变量的值传递到被调用方 FB 的输出变量）和返回值指定（将被调用方 FB 的输出变量值代入到指定变量）。
以下①或②的任意记述方式均可。
出于用户程序的可读性考虑，推荐①。

记述方式①：

所调用 FB 的变量名称和调用方的参数名称都要记述时
ABC（A:=x1, B:=x2, C=>y1）；
ABC　　：FB 的实例名称
A、B　：所调用 FB 的输入变量名称或输入输出变量名称
x1、x2 ：调用方的输入参数或输入输出参数（包括常数）
C　　：所调用 FB 的输出变量名称
y1　　：调用方的输出参数

- 梯形图语言中的表现

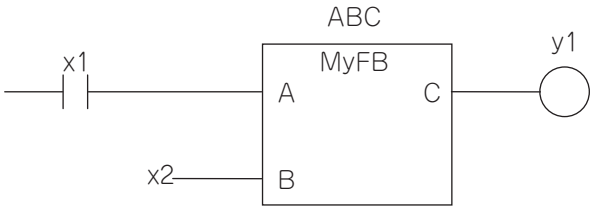


- 各参数或返回值的记述顺序为任意。
- 输入变量名称和输入参数名称可省略。省略时，上次调用时代入到输入变量的值将代入到输入变量中。第一次调用时，将在输入变量中代入初始值。
- 输出变量和输出参数可省略。省略时，输出变量的值将不会代入到任何地方。

记述方式②：

省略所调用 FB 的变量名称，仅记述调用方的参数名称时
ABC （x1, x2, y1）；
ABC　　：FB 的实例名称
A、 B　：省略。（所调用 FB 的输入变量名称或输入输出变量名称）
x1、 x2 ：调用方的输入参数或输入输出参数 （包括常数）
C　　：省略。（所调用 FB 的输出变量名称或常数）。
y1　　：调用方的输出参数

- 梯形图语言中的表现



- 所记述参数的顺序根据 FB 的定义决定。按 FB 的本地变量定义从上往下的顺序。

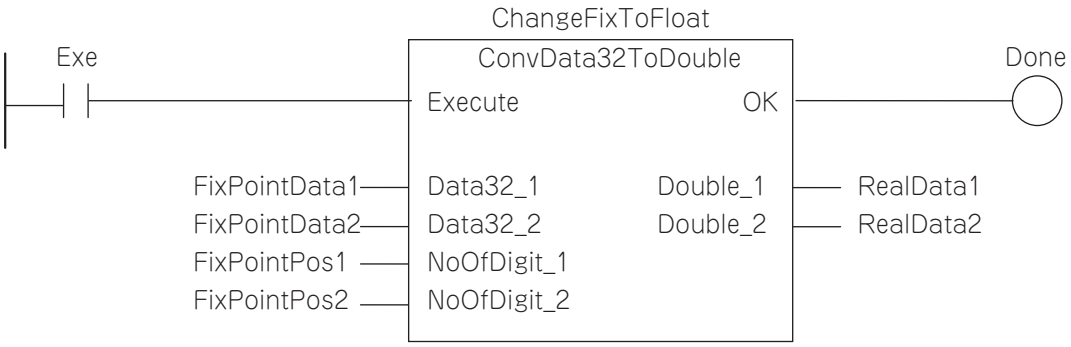
用途：
调用 FB。

例
- 程序

- 记述①
  ChangeFixToFloat （Execute:=Exe,Data32_1:=FixPointData1,
  Data32_2:=FixPointData2,
  NoOfDigit_1:=FixPointPos1,
  NoOfDigit_2:=FixPointPos2,OK=>Done,Double_1=>RealData1,
  Double_2=>RealData2）；

- 记述②
  ChangeFixToFloat （Exe, FixPointData1, FixPointData2, FixPointPos1, FixPointPos2,
  Done, RealData1, RealData2）；

- 梯形图语言中的表现



- 功能块的定义

  功能块名称：ConvData32ToDouble

  功能块的变量

输入 / 输出	变量名称	数据类型
输入变量	Execute	BOOL
	Data32_1	DINT
	Data32_2	DINT
	NoOfDigit_1	INT
	NoOfDigit_2	INT
输出变量	OK	BOOL
	Double_1	LREAL
	Double_2	LREAL

- 程序的变量

变量名称	数据类型	注释
ChangeFixToFloat	ConvData32ToDouble	固定小数点➞浮动小数点转换
Exe	BOOL	启动触发
FixPointData1	DINT	小数点位置指定数据 #1
FixPointPos1	INT	小数点以下位数 #1
FixPointData2	DINT	小数点位置指定数据 #2
FixPointPos2	INT	小数点以下位数 #2
Done	BOOL	正常结束
RealData1	LREAL	浮动小数点数据 #1
RealData2	LREAL	浮动小数点数据 #2

参数的省略可否

调用 FB 时，如果不需要设定参数的值，可省略所记述的参数。

参数的省略可否如下所示。

〇： 可记述。省略时，将代入初始值，以代替参数。

×： 链接时将发生异常。

POU 的种类	所调用 POU 的变量	记述模式		记述的可否
		记述的参数	例	
FB	有记述（记述方式①）	记述所有参数	instance （x:=a,y:=b,z:=c） ;	〇
		记述多个参数	instance （x:=a,y:=b） ;	
		记述 1 个参数	instance （y:=b） ;	
		不记述参数	instance （x:=） ;	×
	不记述（记述方式②）	记述所有参数	instance （a,b,c） ;	〇
		不记述所有参数	instance （） ;	
		只记述开头的参数	instance （a） ;	×
		记述 1 个参数	instance （a, , ） ;	
		记述多个参数	instance （a,b） ;	

● FUN 调用

概要：

调用 FUN。

保留字：无

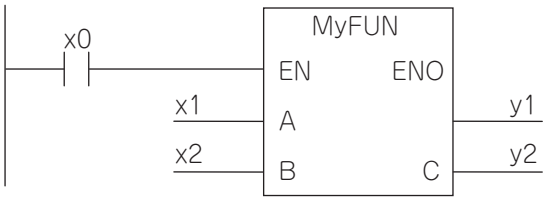
语句的结构：

记述在代入关键词（:=）的左边代入返回值的输出参数。在右边，在 FUN 名称后面的（ ）内记述参数指定（将指定变量的值传递到被调用方 FUN 的输入变量）。
以下①或②的任意记述方式均可。
出于用户程序的可读性考虑，推荐①。

记述方式①：

```
IF （x0=TRUE）THEN
  y1 := MyFUN （A:=x1, B:=x2, C=>y2） ;
END_IF;
```

- 梯形图语言中的表现



MyFUN	:FUN 名称
x0	:FUN 的调用有无
A, B	: 所调用 FUN 的输入变量名称
x1, x2	: 所调用 FUN 的输入参数
C	: 所调用 FUN 的输出变量名称
y1	: 所调用 FUN 的返回值保存位置
y2	: 所调用 FUN 的输出参数

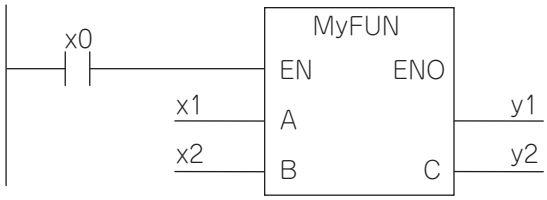
- 各参数的记述顺序为任意。
- 输入变量名称和输入参数名称可省略。省略时，将在输入变量中代入初始值。
- EN 也可省略。省略时，在 EN 中代入 TRUE。

记述方式②：

省略所调用 FUN 的变量名称，仅记述调用方的参数名称时

```
IF （x0=TRUE）THEN
  y1 := MyFUN （x1, x2, y2）；
END_IF;
```

- 梯形图语言中的表现



MyFUN	:FUN 名称
x0	:FUN 的调用有无
A, B	: 所调用 FUN 的输入变量名称
x1, x2	: 所调用 FUN 的输入参数
C	: 所调用 FUN 的输出变量名称
y1	: 所调用 FUN 的返回值保存位置
y2	: 所调用 FUN 的输出参数

- 所记述参数的顺序根据 FUN 的定义决定。按 FUN 的本地变量定义从上往下的顺序。

例

- 程序

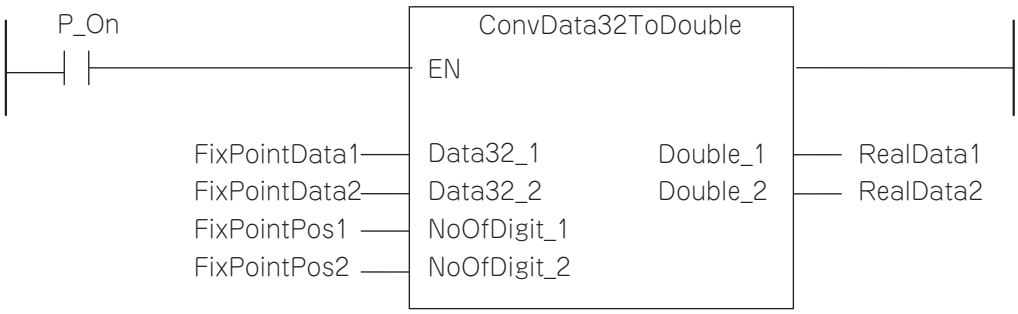
- 记述①

```
ConvData32ToDouble （Data32_1:=FixPointData1,Data32_2:=FixPointData2,
NoOfDigit_1:=FixPointPos1, NoOfDigit_2:=FixPointPos2,
Double_1=>RealData1, Double_2=>RealData2）；
```

- 记述②

```
ConvData32ToDouble （FixPointData1, FixPointData2, FixPointPos1, FixPointPos2,
RealData1, RealData2）；
```

- 梯形图语言中的表现



- 函数的定义
  函数名称：ConvData32ToDouble
  函数的变量

In/Out	变量名称	数据类型
输入变量	Execute	BOOL
	Data32_1	DINT
	Data32_2	DINT
	NoOfDigit_1	INT
	NoOfDigit_2	INT
输出变量	Double_1	LREAL
	Double_2	LREAL
返回值	-	BOOL

- 程序的变量

变量名称	数据类型	注释
ChangeFixToFloat	ConvData32ToDouble	固定小数点➞浮动小数点转换
Exe	BOOL	启动触发
FixPointData1	DINT	小数点位置指定数据 #1
FixPointPos1	INT	小数点以下位数 #1
FixPointData2	DINT	小数点位置指定数据 #2
FixPointPos2	INT	小数点以下位数 #2
Done	BOOL	正常结束
RealData1	LREAL	浮动小数点数据 #1
RealData2	LREAL	浮动小数点数据 #2

用途：

调用 FUN。

参数的省略可否

调用 FUN 时，如果不需要设定参数的值，可省略所记述的参数。
参数的省略可否如下所示。

○： 可记述。省略时，将代入初始值，以代替参数。
×： 链接时将发生异常。

POU 的种类	所调用 POU 的变量	记述模式		记述的可否
		记述的参数	例	
FUN	有记述（记述方式①）	记述所有参数	FUN （x:=a,y:=b,z:=c）；	○
		记述多个参数	FUN （x:=a,y:=b）；	
		记述 1 个参数	FUN （y:=b）；	
		不记述参数	FUN （x:=）；	×
	不记述（记述方式②）	记述所有参数	FUN （a,b,c）	○
		不记述参数	FUN （）；	
		只记述开头的参数	FUN （a）；	×
		记述 1 个参数	FUN （a, , ）；	
		记述多个参数	FUN （a,b）；	

ST 语言相关的注意事项

用 ST 语言创建用户程序，应注意以下事项。

● 默认的 CAST

如下所示，操作对象的数据类型不一致时，将根据默认的 CAST，自动转换数据类型后处理。
此外，若不满足默认的 CAST 规则，链接时将发生异常。

(1) 代入语句右边的表达式内，操作对象的数据类型不一致时

例）

```
A: = INT#10 + SINT#2;
```

(2) 代入语句右边和左边的操作对象数据类型不一致时

例）

```
   A: = B + C;
   /     \ /
REAL    INT
```

(3) 语法的表达式内，操作对象的数据类型不一致时

例）

```
     INT   LINT
       \    /
CASE A+B OF
   INT#1:
      def:=INT#10;
```

下面介绍以下 3 种情况时的 CAST 规则。

在代入语句的右边进行运算时的 CAST 规则

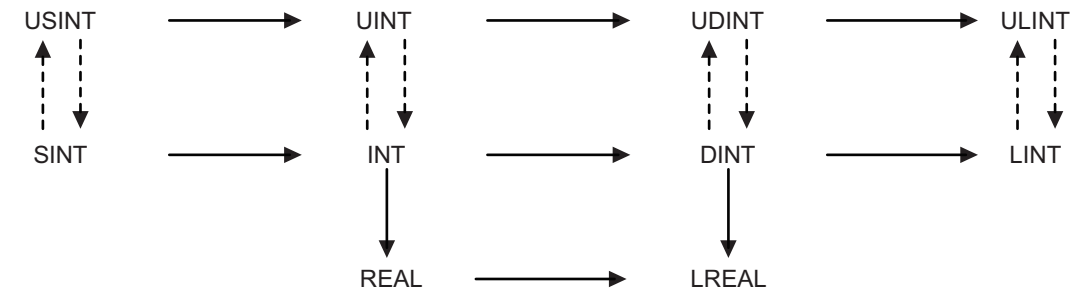
- 如果右边的操作对象为运算符的操作对象中可使用的数据类型且为允许的数据类型，则可以同时存在。
- 在右边的操作对象中，拥有最高等级的数据类型将变为右边的数据类型。
  （数据类型的等级请参考以下 [数据类型等级表]）。

[数据类型等级表]：
等级越高，表示该数据类型可表现的数值范围越大。

等级	数据类型
1	SINT
2	USINT
3	INT
4	UINT
5	DINT
6	UDINT
7	LINT
8	ULINT
9	REAL
10	LREAL
11	BYTE
12	WORD
13	DWORD
14	LWORD

将代入语句右边的值代入到左边时的 CAST 规则

在以下图中，如果代入源的类型和代入目标的类型用箭头连接，将进行 CAST 处理。
若未连接，链接时将发生异常。



→ 代入时，数值的符号和绝对值无变化。

- - - - - → 代入时，数值的符号或绝对值可能发生变化。

例) intVar := -1;　　(* intVar := 16#FFFF *)
　uintVar := 1;
　uintVar := intVar;（* uintVar := 16#FFFF，即代入了-1，结果为65535*)

箭头即使没有直接连接，也可以代入。
例如，SINT → USINT → UINT → UDINT → ULINT 这样连接时，ULINT:= SINT 这样的记述也可以。

使用注意事项

UDINT → ULINT、 DINT → LINT、 DINT → LREAL 的 CAST 时，请注意以下事项。
以上均为从数据大小 32 位的数据类型到 64 位的数据类型的 CAST。此时，代入语句右边的运算结果如果超出 32 位可表现值的范围，可能无法代入正确的数值。

例）在以下用户程序中，第 3 行的相加结果超出 32 位可表现值的范围。将发生溢出，在 LintVar 中代入 0。

```
UdintVar := UDINT#16#FFFF_FFFF;          // 32 位的上限值
DintVar  := DINT#1;                      // 1
LintVar  := (UdintVar + DintVar) / DINT#2;   // (32 位的上限值＋ 1) / 2
```

此时，请先将各数值转换为 64 位的数据类型后再进行运算。上述示例将变更如下。

```
LintTmp1 := UDINT_TO_LINT(UDINT#16#FFFF_FFFF);  // UDINT → LINT 转换
LintTmp2 := DINT_TO_LINT(DINT#1);               // DINT → LINT 转换
LintVar  := (LintTmp1 + LintTmp2) / DINT#2;
```

语法表达式内的 CAST 规则

在语法表达式中，代入语句右边的运算式或右边代入左边时的默认 CAST 规则同样适用。
例）

```
CASE （A+B+C）OF
    Result1:
        ～;
    ResultN:
        ～;
END_CASE;
```

● FUN 的执行顺序

在包含 FUN 的表达式中， FUN 的执行顺序无法完全确定。根据 CPU 单元或 Sysmac Studio 的版本以及记述的方法不同， FUN 的执行顺序也不同。因此，如下所示，结果会因 FUN 的执行顺序不同而变化的表达式请按后述方法记述。

- 包含多个访问同一全局变量的 FUN 之表达式
- 包含因 FUN 及 FUN 的执行其值会变化的变量之表达式

包含多个访问同一全局变量的 FUN 之表达式

以下表达式时，3 个 FUN 的执行顺序与按照运算符的优先顺序决定的运算顺序无关。因此，3 个 FUN 不一定按 FUN2、 FUN3、 FUN1 的顺序执行。

```
result := FUN1() + FUN2() * FUN3();
```

上述表达式中的 3 个 FUN 如果是对同一全局变量进行访问 （包括写入）的 FUN，因 FUN 的执行顺序不同，变量 “result” 的值也可能不同。

此时，为了确定 3 个 FUN 的执行顺序，应分开记述表达式。要以 FUN2、FUN3、FUN1 的顺序执行时，请如下记述。

```
tmp2  := FUN2();
tmp3  := FUN3();
result := FUN1() + tmp2 * tmp3;
```

包含因 FUN 及 FUN 的执行其值会变化的变量之表达式

以下表达式中包含因 FUN 及 FUN 的执行其值会变化的变量。

```
result := varA + FUN4(out => varA);
```

上述表达式时，左右第 1 项变量 “varA” 的评估不一定会在 FUN4 之前执行。因此，根据 “varA” 的评估和 FUN4 的执行顺序不同，变量 “result” 的值可能会不同。

此时，为了确定 “var4” 的评估和 FUN4 的执行顺序，应分开记述表达式。

要在 FUN4 之前执行 “varA” 的评估时，请如下记述。

```
tmp    := varA;
result := tmp + FUN4(out => varA);
```

要在 FUN4 之后执行 “varA” 的评估时，请如下记述。

```
tmp    := FUN4(out => varA);
result := varA + tmp;
```

● 表达式中包含不指定数据类型的常数时，表达式的运算精度

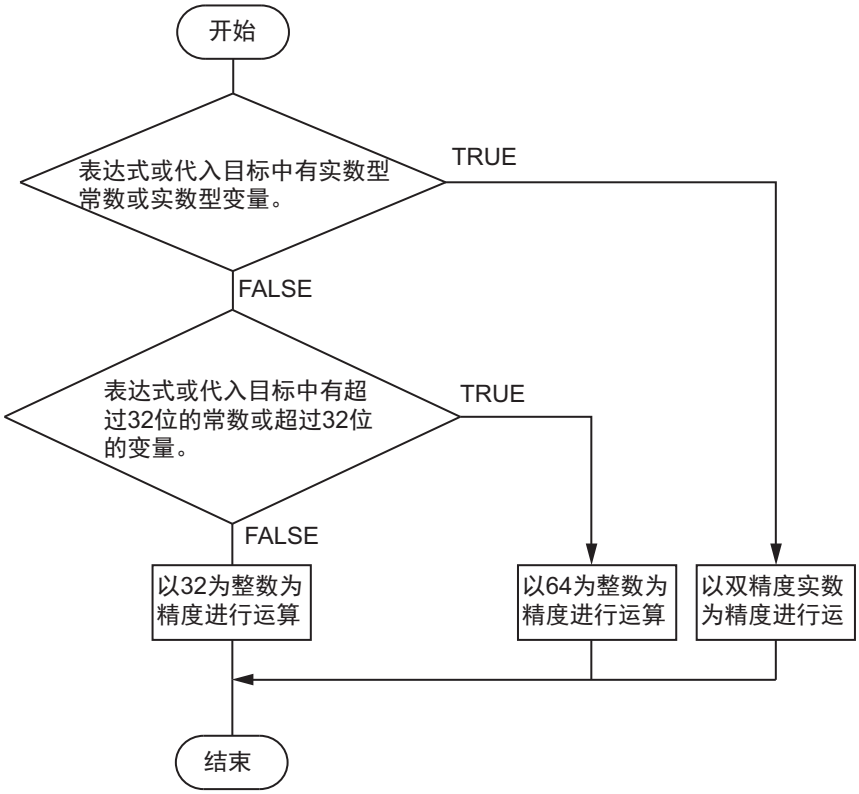
表达式中包含不指定数据类型的常数时，运算精度根据其常数的表述、表达式及代入目标中写入的变量以及常数的数据类型自动决定。

常数的表述

如果常数为 “100” 等没有小数点的表述，作为整数处理。如果为 “100.0” 等带小数点的表述，作为实数处理。

表达式的运算精度

表达式的运算精度根据表达式及代入目标中写入的变量以及常数的数据类型不同，为 32 位整数、 64 位整数、双精度实数中的一个。表达式的运算精度按以下规则决定。



例）realv := 2 + 3 * 4; // 变量 realv 的数据类型为 REAL
变量 realv 为实数型，因此以双精度实数的精度进行运算。运算结果为 14.0。

realv := 2 + 3.0 *4; // 变量 realv 的数据类型为 REAL
常数 3.0 为实数型表述，变量 realv 也是实数型，因此以双精度实数的精度进行运算。运算结果为 14.0。

lintv := 2 + 3 * 4; // 变量 lintv 的数据类型为 LINT
无实数型常数或变量，变量 lintv 为超过 32 位的常数，因此以 64 位整数的精度进行运算。运算结果为 14。

intv := 2 + 3 * 4; // 变量 intv 的数据类型为 INT
无实数型常数或变量，也没有超过 32 位的常数，因此以 32 位整数的精度进行运算。运算结果为 14。

但是，除法的运算精度只根据除数和被除数决定。决定运算精度的规则与前面的流程图相同。

例）realv := 2 / 3 * 4; // 变量 realv 的数据类型为 REAL
除法 2/3 由于被除数和除数都是 32 位以内的整数，因此以 32 位整数的精度进行运算。相除结果为整数 0。
然后，由于变量 realv 为实数型，因此 0*4 以双精度实数的精度进行运算。运算结果为 0.0。

使用注意事项

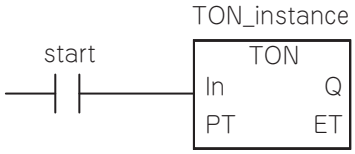
表达式中包含不指定数据类型的常数时，根据其常数的表述、表达式中的变量及常数的数据类型自动决定。因此，可能以用户预期以外的精度进行运算。常数如 REAL#1.0，建议以指定数据类型的方式表述。

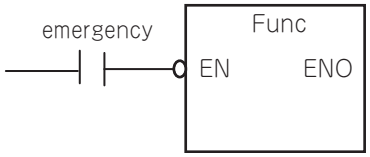
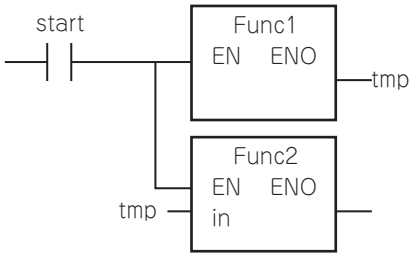
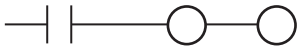
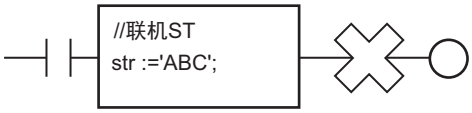
ST 和梯形图的差异

ST 和梯形图之间有差异时的示例如下所示。

项目	梯形图	ST （包含联机 ST）
接点的微分	● **上升沿** ・方法1 start do ----|↑|----------○ ・方法2 R_TRIG_instance R_TRIG start Clk Q do ・方法3 Up start In do	● **上升沿** • 方法 1 R_TRIG_instance （Clk:=start, Q=>do）； ※R_TRIG_instance 为 R_TRIG 指令的实例
	● **下降沿** ・方法1 start do ----|↓|----------○ ・方法2 F_TRIG_instance F_TRIG start Clk Q do ・方法3 Down start In do	● **下降沿** • 方法 1 F_TRIG_instance （Clk:=start, Q=>do）； ※F_TRIG_instance 为 F_TRIG 指令的实例
指令的微分	● **上升沿微分** start @Inst	● **上升沿微分** 无符合的项，用逻辑来实现。例如， • 方法 1 R_TRIG_instance （Clk:=start, Q=>do） ; IF （do = TRUE）THEN Inst （）； END_IF; • 方法 2 IF （start = TRUE）THEN IF （pre_start = FALSE）THEN Inst （）； END_IF; END_IF; pre_start:=start;// 上次值的更新

项目	梯形图	ST （包含联机 ST）

多个任务周期指令的记述	从指令执行到完成需要多个周期，且在功率流 OFF 时进行复位动作的 TON 指令中，用 1 个实例声明兼做计时和复位
	 
		如下所示，声明计时用和复位用的 2 个实例
		IF （start = TRUE）THEN
		TON_instance （In:=TRUE, 后面省略）
		// 定时器启动
		ELSE
		TON_instance（In:=FALSE, 后面省略）;
		// 定时器复位
		END_IF;

FB ／函数参数反转指定	在 BOOL 型参数和 FB ／函数的交点添加○，反转表示 	在参数上添加 NOT 运算符 ※ 除了参数之外， NOT 运算符还可添加到所有 BOOL 型变量中 IF （NOT emergency）THEN Func （）； END_IF;
多段连接		IF （start=TRUE）THEN Func2 （ in := Func1 （））； END_IF;
梯形图的后续	Out 指令后面只能连接 OUT 指令 	联机 ST 的后面不可连接梯形图 
程序的分割单位	有段落功能	无段落功能

6-6 指令

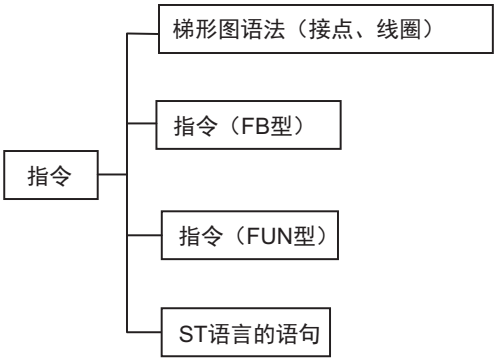
下面介绍预先在 NJ/NX 系列中定义的指令。

指令的详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》及 📖《NJ/NX 系列 指令基准手册 运动篇 （SBCE-CN5-364）》。

6-6-1 指令定义

指令是指 POU 的算法中记载的、欧姆龙指定的最小单位的处理要素。

指令分为以下种类。



程序、用户定义的 FB、用户定义的 FUN 由本指令的组合构成。

6-6-2 指令的基本知识

指令的基本规格遵照 FB、 FUN 的规格。

下面介绍指令特有的规格。

梯形图语法 （接点、线圈）

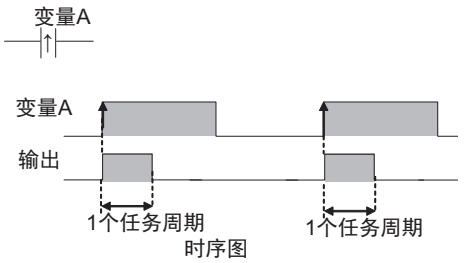
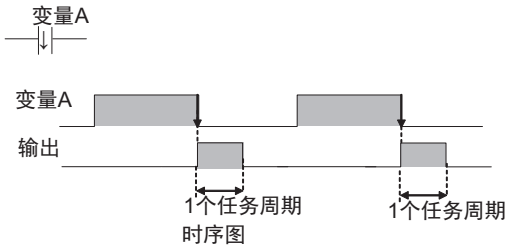
● **可配置位置**

以梯形图语言的接点和线圈形态表现的指令如下所示，有可配置的位置。

分类		可配置位置	图
输入类	逻辑启动型（LD 型）	与左母线直接连接或回路图的起点	
	下一段连接型	逻辑启动型和输出型的中间位置	
输出类		与右母线直接连接	

● 动作选项

梯形图语言的指令（接点）中，也有通过标向上（↑）、向下（↓）后，可进行上升沿微分、下降沿微分的动作。

上升沿 （↑）		该指令为每个任务周期执行接点的读取 / 比较 / 位测试等处理，若结果从 FALSE 变为 TRUE，则输出功率流。 输出的功率流在（1 个任务周期 TRUE 后）下一任务周期变为 FALSE。
下降沿 （↓）		该指令为每个任务周期执行接点的读取等处理，若结果从 TRUE 变为 FALSE，则输出功率流。 输出的功率流在（1 个任务周期 TRUE 后）下一任务周期变为 FALSE。

功能块 (FB) 型的指令

● 执行条件

FB 型指令的执行条件因指令不同而异。

执行条件用的特定输入变量根据各指令定义。

例： Execute：上升沿指定、下降沿指定

Enable：每个任务周期执行

FB 型指令在执行该指令的调用源 POU 期间，将持续执行。

● 动作选项

不可指定。

函数 （FUN）型指令

● 执行条件

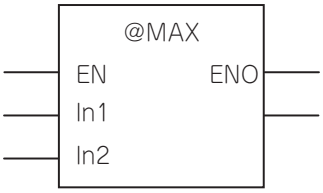
FUN 型指令的执行条件全部为 EN 输入。 EN 为 TRUE 期间，每个任务周期执行。

● 动作选项

梯形图语言时，可添加以下动作选项，将该执行条件作为上升沿指定。 ST 语言中无动作选项。

动作选项		选项符号	
微分选项	上升沿	@	指令为输入上升沿微分型指令。 EN 的上升沿时，执行 1 次指令。

要添加动作选项，需要在指令名称前面添加上述选项符号。
例）



功能块 (FB) 型 / 函数 (FUN) 型的指令共通

● 条件标志

将值作为指令处理的结果代入的系统定义变量称为 “ 条件标志 ”。 NJ/NX 系列的条件标志只有进位标志 （P_CY）。
进位标志的含义为以下之一。

- 判定指令的处理结果是否超过输出变量的数据类型可表现的范围
- 位列数据的位移位指令等，判定是否发生位数溢出的详情，请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》中各指令的说明。

6-6-3 指令错误

在执行指令时发生的异常称为指令错误。下面介绍指令错误的发生时序、指令错误中检测到的异常、发生指令错误时的动作等。

指令错误的发生时序

指令错误的发生时序有以下 3 种。因指令错误的发生时序不同，检测到的异常或发生异常时的动作有所不同。

- 执行指令前，检查输入参数或输入输出参数的值时发生
- 执行指令时，在指令的内部处理时发生
- 执行指令后，检查输出参数的值时发生

指令错误中检测到的异常

指令错误中检测到的异常如下所示。因指令错误的发生时序不同，检测到的异常有所不同。

● 执行指令前、执行指令后检测到的异常

执行指令前、执行指令后检测到的异常如下所示。

- 读写超出排列指定变量的排列区域的要素
- 在 STRING 型变量中，代入比定义字节数更长的字符串
- 在 STRING 型变量中，代入不是以 NULL 字符结束的字符串
- 整数型变量除以 0

● 执行指令时检测到的异常

执行指令时检测到的异常因指令不同而异。各指令中检测到的异常详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

发生指令错误时的动作

因是否发生指令错误而进行不同动作的要素包括输出变量 “ENO”、输出变量 “Error”、输出变量 “ErrorID”、系统定义变量 “P_PRGER”、事件。下面分别介绍。

● 输出变量 “ENO”、输出变量 “Error”、输出变量 “ErrorID”

表示是否发生异常的输出变量包括 “ENO” （允许启用）、 “Error” （错误）、 “ErroriD” （错误代码）。具体是哪个输出变量因指令不同而异。各变量的含义和发生指令错误时的值如下所示。因指令错误的发生时序不同，动作有所不同。

输出变量	数据类型	含义	发生指令错误时的值		
			执行指令前*1	指令执行中	执行指令后*2
“ENO”	BOOL	TRUE：正常结束 FALSE：异常结束、执行中、未执行	FALSE	FALSE	TRUE
“Error”	BOOL	TRUE：异常结束 FALSE：正常结束、执行中、未执行	FALSE	TRUE	FALSE
“ErrorID”	WORD	异常结束时为错误代码， 正常结束时为 WORD#16#0	WORD#16#0	错误代码	WORD#16#0

*1. 如果在执行指令前发生指令错误，则不执行指令。因此，各输出参数的值保持执行指令前的值。
*2. 如果在执行指令后发生指令错误，则判断为指令本身已正常结束。因此，各输出参数中将代入指令的输出变量值。但是，发生异常的输出参数的值保持执行指令前的值。

● 系统定义变量 “P_PRGER”

系统定义变量 “P_PRGER” 为指令错误发生标志。若发生指令错误，无论指令错误发生时序如何，值都变为 TRUE。正常结束时，保持值。 “P_PRGER” 的详情请参考📖「 指令错误发生标志」（P.6-135）。

● 事件

若发生指令错误，将发生针对异常内容的事件。事件的详情请参考📖「 因发生指令错误而发生的事件」（P.6-136）。

版本相关信息

CPU 单元 版本 1.02 以上时，会在发生指令错误时发生事件。

梯形图语言中的输出参数

用梯形图语言记述的指令、用户定义的函数、用户定义的功能块正常结束时，以及发生指令错误时，输出参数的值如下所示。

条件	输出参数的种类	输出参数的值
正常结束时	功率流输出	根据内部算法更新值。
	参数输出 BOOL 型	
	参数输出 BOOL 型以外	
发生指令错误时	功率流输出	变为 FALSE。
	参数输出 BOOL 型	保持之前的值。
	参数输出 BOOL 型以外	

用 ST 语言记述的 POU 中发生语法错误时的动作

● 代入语句中的错误

用 ST 语言记载的代入语句发生错误时，不执行该行。

```
5   a = b / (c + d) + e * f + ABS(g);
6   x := 1;
```

例如，
第5行的b/(c+d)中发生除以0的情况时，
第5行的执行将中断（a的值无变化），
执行第6行。

用户创建函数的输出 ENO 为 OFF 时亦同。

```
5   a = 用户创建函数(b) + c;
6   x := 1;
```

用户创建函数的输出ENO为OFF时，
第5行的执行将中断（a的值无变化），执行第6行。

● IF 语法中的错误

ST 语言的语法发生错误时，根据语法执行错误处理。

以下示例中 （c+d）的值为 0 时，不执行 IF ～ END_IF 之间的内容。

POU“○○”

```
 5   IF a = b / (c + d) THEN
 6     x := 1;
 7   ELSE
 8     x := 2;
 9   END_IF;
10   y := 10;
 :   :
     IF P_PRGER = TRUE THEN
       x := 初始值;  (* 发生异常时的处理 *)
       y := 初始值;
     END_IF;
```

记载发生错误时用户应采取的安全措施。

● ST 语言中的语法错误

ST 语言中的语法错误包括以下情况。

- 超出排列 [i] 的要素数
- 输入输出变量中未设置参数
- 字符串型的代入：左边的字符串型大小 （Byte）＜右边字符串长度 （Byte）时
- 除以 0 （浮动小数点运算除外）

※ 浮动小数点数的值为非数值时，运算结果为非数值，但不视为错误。

● 发生语法错误时的动作

P_PRGER 标志变为 TRUE，具体如下。

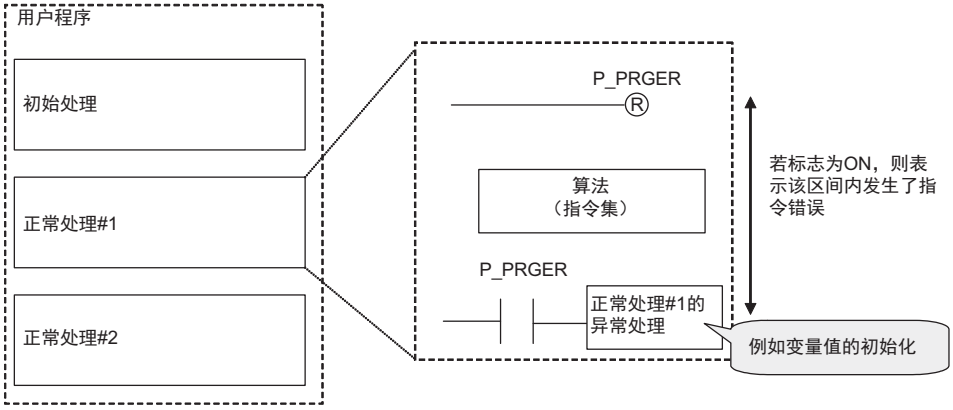
语法	错误发生位置	动作
代入语句		不执行该行
控制语句	IF 条件式内	不执行 IF ～ END_IF 之间
	CASE 条件式	不执行 CASE ～ END_CASE 之间
	FOR 条件式内	不执行 FOR ～ END_FOR 之间
	WHILE 条件式内	不执行 WHILE ～ END_WHILE 之间
	REPEAT 条件式内	不执行 REPEAT ～ END_REPEAT 之间

指令错误发生标志

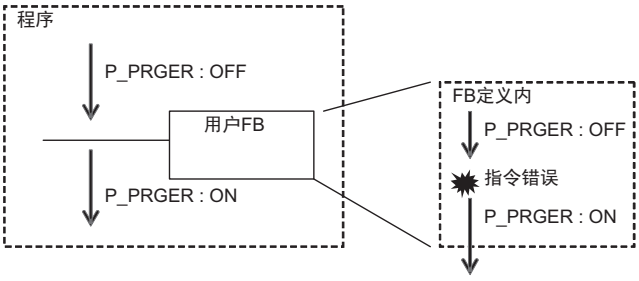
在梯形图算法内发生指令错误或在 ST 算法内发生语法 / 函数错误时，系统定义变量 “P_PRGER”（指令错误发生标志）将变为 TRUE。
P_PRGER 标志为程序的本地变量。程序中发生指令错误时变为 TRUE，在下一任务周期内仍保持值。

变量名称	名称	功能	数据类型	值的范围	初始值	R/W
P_PRGER	发生指令错误标志	发生指令错误时，持续变为 TRUE 的标志。一旦变为 TRUE，在通过用户程序变为 FALSE 前，将保持 TRUE。	BOOL	FALSE/TRUE	FALSE	RW

P_PRGER 标志还可由用户覆盖。因此，可如下所示，当通过用户操作设为 FALSE 时，可在特定范围内判断是否发生指令错误。一旦变为 TRUE，将变更动作模式，或保持值，直至通过用户程序等覆盖。
例）



程序中使用的用户定义 FB 发生指令错误时， P_PRGER 标志也会变为 TRUE。
例）



因发生指令错误而发生的事件

因发生指令错误，发生与之对应的事件。

事件的详细确认方法请参考「8-6　事件日志功能」（P.8-51）。此外，事件的内容请参考《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

使用注意事项

- 发生指令错误时发生事件的情况为在Sysmac Studio的控制器设定中将[事件日志设定]｜[指令错误输出]设定为[是]时。关于 Sysmac Studio 的控制器设定详情，请参考「4-2-2　控制器设定」（P.4-4）及《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362D 以后）》。
- 发生指令错误后若变更用户程序，可能无法在事件日志中正确记录信息。
- 若重复执行有异常的指令，每次都会发生指令错误。因此，事件日志的记录条数可能超出上限。此时，将从旧到新依次覆盖，敬请注意。

版本相关信息

- CPU 单元 版本 1.02 以上时，会在发生指令错误时发生事件。
- 发生指令错误时，切换指令错误输出的是 / 否可在 CPU 单元 版本 1.02 以上和 Sysmac Studio Ver.1.03 以上的组合中使用。

参考

- 若在执行运动控制指令时发生异常，将同时发生指令错误对应的事件和运动控制指令固有的事件。运动控制指令的事件详情请参考《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。
- 在Sysmac Studio的控制器设定中将[事件日志设定]｜[指令错误输出]设定为[否]时，运动控制指令固有的事件仍会发生。

6-7 名称空间

下面介绍名称空间的规格和使用方法。关于操作方法，请参考📖《Sysmac Studio Version 1 操作手册（SBCA-CN5-362C 以后）》。

版本相关信息

名称空间可在 CPU 单元版本 1.01 以上和 Sysmac Studio Ver.1.02 以上的组合中使用。

6-7-1 名称空间定义

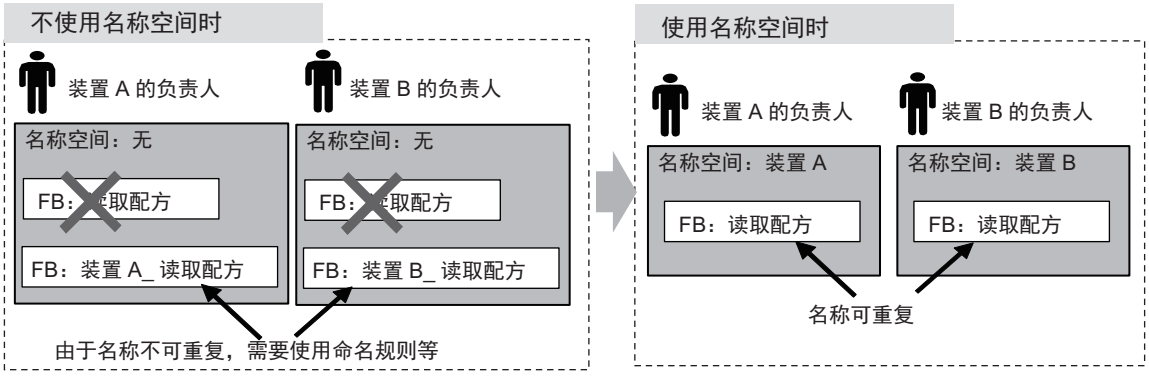
名称空间是指将 FB 定义等分组，以层级结构进行管理的机制。这相当于将文件归类到文件夹，以目录的层级结构进行管理。不使用名称空间时，FB 定义等不可重复使用相同的名称，但如果使用名称空间，设定为不同的名称空间后，可重复使用。
名称空间可使用，也可不用。

名称空间的特点

有以下特点。

● 避免名称的重复

如果是不同的名称空间， FB 定义名称等可以使用相同的名称。
例如，对多个装置进行编程时，如果不同的装置由不同的人员负责编程，FB 定义名称等可能会重复。不使用名称空间时，应利用命名规则等，避免名称重复。如果使用名称空间，为不同的装置设定名称空间，编程时无需顾忌装置的名称会重复。



6-7-2 名称空间的规格

下面介绍名称空间的对象、表现及使用声明。

名称空间的对象

名称空间的对象如下所示。

对象	详细
POU 定义	FB 定义名称、 FUN 定义名称
数据类型	结构体型名称、联合体型名称、列举型名称

名称空间的表现

名称空间以 “\（反斜杠）” 分隔表现。在 POU 的算法中使用时，在名称空间的开头加 “\\（2 个反斜杠）”。“\ （反斜杠） ” 可能因 Windows 的本地设置或字体的关系，显示为 “￥”。

例）

使用场合	表现
在算法以外中使用	装置 A\ 读取配方
在算法内使用	\\ 装置 A\ 读取配方

● 完全限定名称和单纯名称

不省略名称空间名，表现名称空间对象的名称称为完全限定名称。省略名称空间名的名称称为单纯名称。

在 POU 定义的算法中，属于相同名称空间的 POU 定义可用单纯名称记述。

例） 装置 A\ 读取配方

名称空间名　单独名称

完全限定名称

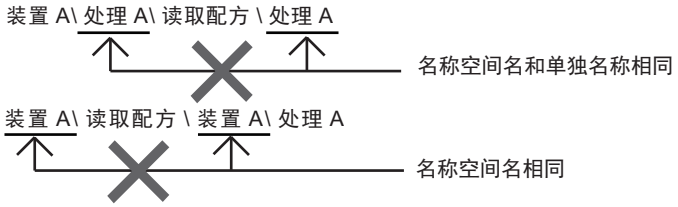
● 名称空间的表现限制

- 可使用的字符与变量名称相同。详情请参考📖「6-3-12 变量名称等程序相关名称的限制」（P.6-77）。
- 名称空间的字符数限制如下所示。

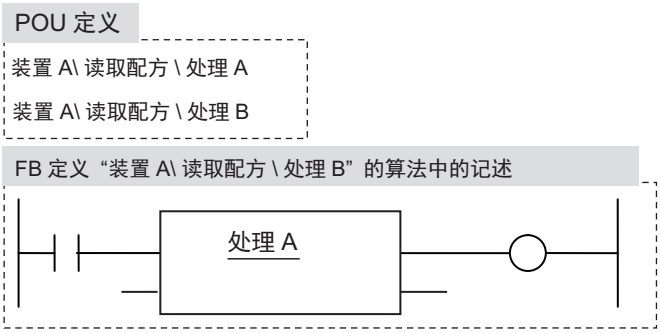
名称	最大大小	字符编码
名称空间名	93 字节	UTF-8
单独名称	127 字节	

使用注意事项

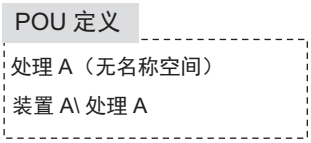
- 如果名称空间名称和单纯名称相同，链接时将发生异常。



- 在 POU 定义的算法中，属于相同名称空间的 POU 定义可用单纯名称记述，但如果该名称空间的上位层级中存在相同单纯名称的 POU 定义或数据类型等，链接时将发生异常。
  如下图所示，存在 POU 定义时，在名称空间 “ 装置 A\ 读取配方 ” 上的 FB 定义 “ 处理 B” 算法的记述中，可用单纯名称调用属于相同名称空间的 “ 装置 A\ 读取配方 \ 处理 A”。



  但是，如下所示，在名称空间 “ 装置 A\ 读取配方 ” 的上位层级中存在 POU 定义 “ 装置 A” 时，由于存在多个单纯名称 “ 处理 A”，因此链接时将发生异常。
  此时，请用完全限定名称记述，或变更单纯名称。



- 与系统内部使用的保留字相同时，程序检查时将发生错误。

名称空间的使用声明

编程时，事先对 POU 定义的算法中使用的名称空间进行声明称为使用声明。
在 POU 定义中进行使用声明后，属于已进行使用声明的名称空间的 POU 定义可用单纯名称表示。进行使用声明后，仍可使用完全限定名称。
在 POU 定义的算法中，使用属于与 POU 定义相同名称空间的 FB 定义或 FUN 定义时，无需进行使用声明，也可使用单纯名称。
1 个 POU 定义可进行多次使用声明。

● 记述示例

下面介绍在创建 FB 定义时，对使用对象 FB 定义的名称空间进行使用声明和不进行使用声明时的记述示例。

例）

在 FB 定义 “ 升降机 ” 的算法内，记述 FB 定义 “ 读取配方 ” 及 FB 定义 “ 上限计算 ” 的示例。

各 FB 定义属于不同的名称空间。 FB 定义 “ 升降机 ” 中，仅对名称空间 “ 装置 C” 进行使用声明。

不属于名称空间 “ 装置 C” 的 FB 定义 “ 读取配方 ” 以完全限定名称记述。属于名称空间 “ 装置 C” 的 FB 定义 “ 上限计算 ” 可用单纯名称记述。

名称空间	对象种类	单独名称
装置A	FB定义	升降机
装置B	FB定义	读取配方
装置C	FB定义	上限计算

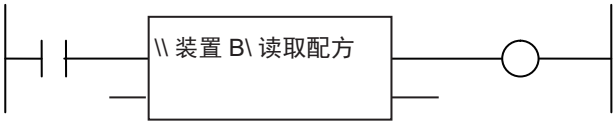
FB 定义 “升降机” 的 [名称空间 - 使用声明] 的记述

装置 C

FB 定义 “升降机” 的算法内的记述

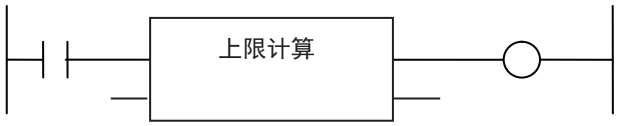
· 不对名称空间进行使用声明时

未对 FB 定义 “读取配方” 所属的名称空间 “装置 B” 进行使用声明，因此以完全限定名称记述。



· 对名称空间进行了使用声明时

已对 FB 定义 “上限计算” 所属的名称空间 “装置 C” 进行了使用声明，因此以单独名称记述。



● 使用声明的限制

单纯名称仅在 POU 定义的算法内可以使用。

使用注意事项

- 以下情况下，在程序检查时将发生错误。
  - 对不存在的名称空间进行了使用声明时
  - 对1个POU定义声明了多个名称空间，各名称空间中存在相同名称的POU定义或数据类型时
- 以下情况下，如果将 POU 定义或数据类型命名为相同的名称，链接时将发生异常。
  - POU 定义所属的名称空间及其上位层级的名称空间
  - 进行了使用声明的名称空间
  - 无名称空间

名称空间	对象种类	单独名称	
装置A\升降机	FB定义	处理A	✕ POU 定义所属的名称空间
装置A	FB定义	处理A	✕ POU 定义所属的名称空间的上位层级
装置B	FB定义	处理B	
		处理A	✕ 进行了使用声明的名称空间
无	FB定义	处理A	✕ 无名称空间

FB 定义“处理 A”的 [名称空间 - 使用声明] 的记述

装置 B

参考

不可对程序名称设定名称空间，但对程序的算法中使用的部件名称空间可以进行使用声明。

6-7-3 名称空间的使用步骤

使用 Sysmac Studio，进行名称空间的设定和使用声明。
创建数据类型或创建 FB 定义、 FUN 定义等部件时，进行步骤 1 和步骤 2 的设定。
使用设定了名称空间的部件时，进行步骤 3 的使用声明。

1 **用数据类型编辑器，为对象数据类型设定名称空间。**

2 **为对象 FB 定义或 FUN 定义的属性设定名称空间。**

3 **在梯形图编辑器或ST编辑器中，针对对象FB定义或FUN定义的属性中设定的名称空间进行使用声明。**

4 **在用户程序中记述对象数据类型或 FB 定义等。**

操作详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362C 以后）》。

6-8 库

下面介绍库的规格。详细的操作方法请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362C 以后）》。

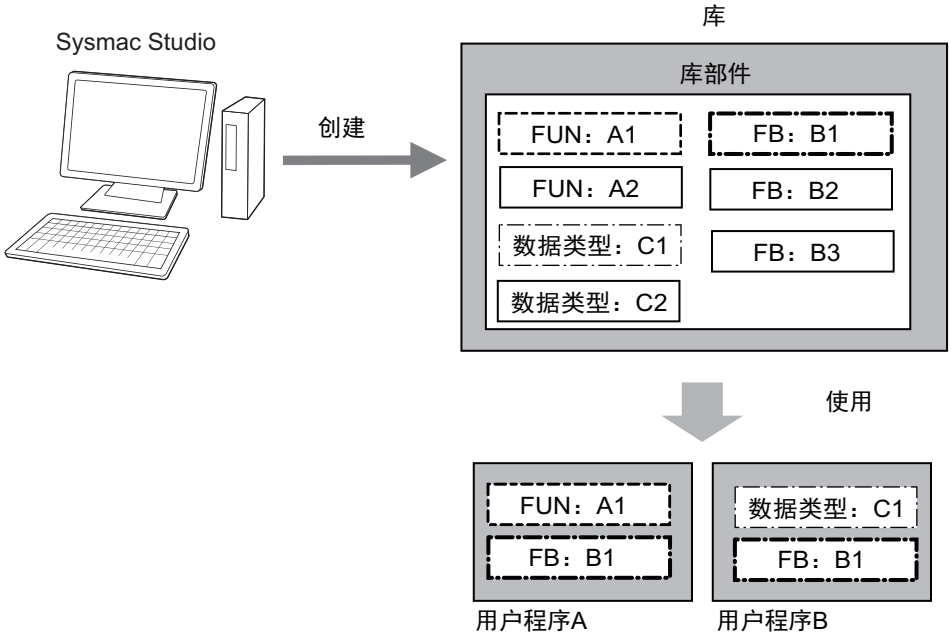
版本相关信息

库可在 CPU 单元版本 1.01 以上和 Sysmac Studio Ver.1.02 以上的组合中使用。

6-8-1 库的概要

将 POU 定义或数据类型汇总为可作为流用所需的部件使用的形式，得到库。库的部件称为库部件。NJ/NX 系列 CPU 单元中，可创建和使用库。

下图中，在库的库部件中，用户程序 A 使用 FUN：A1 和 FB：B1，用户程序 B 使用数据类型：C1 和 FB：B1。



6-8-2 库的规格

下面介绍库的设定和同步。

库的设定

库中可设定以下项目。

设定项目	说明
名称	库的名称。
版本	库的版本。
创建者	库创建者的姓名。（任意）
创建日期	新建库时的日期。
更新日期	更新库时的日期。
注释	库的注释。（任意）
公司名称	库的创建公司名称。（任意）
ID	库核对用的唯一 ID。自动生成，无法变更。
不可显示源代码	可选择源代码的显示或隐藏。*1
附件	可附加多个任意文件。

*1. 库部件设定为数据保护时，若要显示源代码，需要密码。

库部件可参照其他库后创建。此时，选择是否包含参照处的库数据。

参考

如果包含参照处的库，创建参照源的库时，将包含参照处库的副本。因此，库的文件只需 1 个，但如果参照源的库有多个，变更时需要逐个进行。
如果不包含，创建参照源的库时，将不包含参照处的库。因此，库的文件会分割为参照源和参照处多个，如果参照源的库有多个，变更时只需变更参照处的库即可。

● 库部件的选择

选择要包含到库中的库部件。

库的同步

能够以库为单位，从 Sysmac Studio 下载、用控制器上传、与控制器进行核对。

参考

- 向正在传送“显示源代码的库”的控制器，通过Sysmac Studio以“不传送程序的源代码”的设定传送了项目时，将不传送库的源代码。
- 存储器全部清除后，控制器中的库将消失。

6-8-3 库部件的规格

下面介绍库部件的对象和库部件的设定。

库部件的对象

以下数据可视为库部件。

库部件	详细
POU 定义	FB、 FUN
数据类型*1	结构体型、联合体型、列举型

*1. 在 Sysmac Studio 的部件选择画面中，数据类型始终包含在库中。

库部件的设定

各库部件中可设定以下项目。

设定项目	定义
名称	库部件的名称。
名称空间	库部件的名称空间。
版本*1	库部件的版本。
创建者 *1	库部件创建者的姓名。（任意）
创建日期 *1	新建库部件时的日期。
更新日期 *1	更新库部件时的日期。
注释	库部件的注释。（任意）

*1. 在 Sysmac Studio 的各 POU 定义的属性中，只能对 FB 和 FUN 进行设定。

6-8-4 库的使用步骤

库的创建步骤和使用步骤如下所示。

库创建步骤

库作为项目创建。按以下步骤创建库并保存。

1 新建库项目。
在项目窗口的项目种类中选择 “ 库项目 ”，创建项目。

2 新建库部件。
在库项目中创建 POU 定义或数据类型，确认动作。

3 库的设定。
在库项目中，进行库部件的选择、源代码的显示、附件等的设定。

4 保存为库文件。
在 [创建库] 窗口中，保存为库文件。

参考

- 即使是已创建的项目，如果项目中登录的设备只有控制器，可在项目的属性中将种类变更为 “ 库项目 ”。
- 在库项目中，还可创建库部件对象外的数据，但可选择作为库部件的仅限库部件的对象。
- 作为库部件创建的函数、功能块定义、数据类型名称建议使用名称空间，防止多个库之间的名称重复。名称空间请参考「6-7 名称空间」（P.6-137）。

库的使用步骤

将创建的库导入到项目后，可在用户程序中使用。按以下步骤使用库。

1 指定库。
在使用库的项目的 [参照库] 窗口中，指定要使用的库文件。

2 使用库部件进行编程。
在项目中使用所导入库的库部件。
使用方法与普通的函数、功能块定义、数据类型相同。

6-9 创建用户程序时的注意事项

下面介绍创建用户程序时的注意事项。

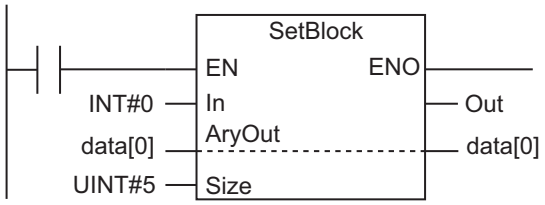
6-9-1 输入变量 / 输出变量 / 输入输出变量为排列指定的变量时

部分指令中存在处理排列变量的指令。

例）

数据类型表

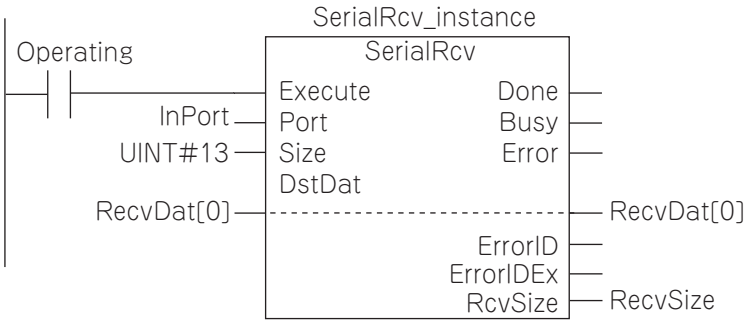
变量名称	数据类型
data	ARRAY[0..4] OF INT



6-9-2 输入变量 / 输出变量 / 输入输出变量为结构体型变量时

部分指令的输入变量、输出变量、输入输出变量的数据类型为结构体型。

例）

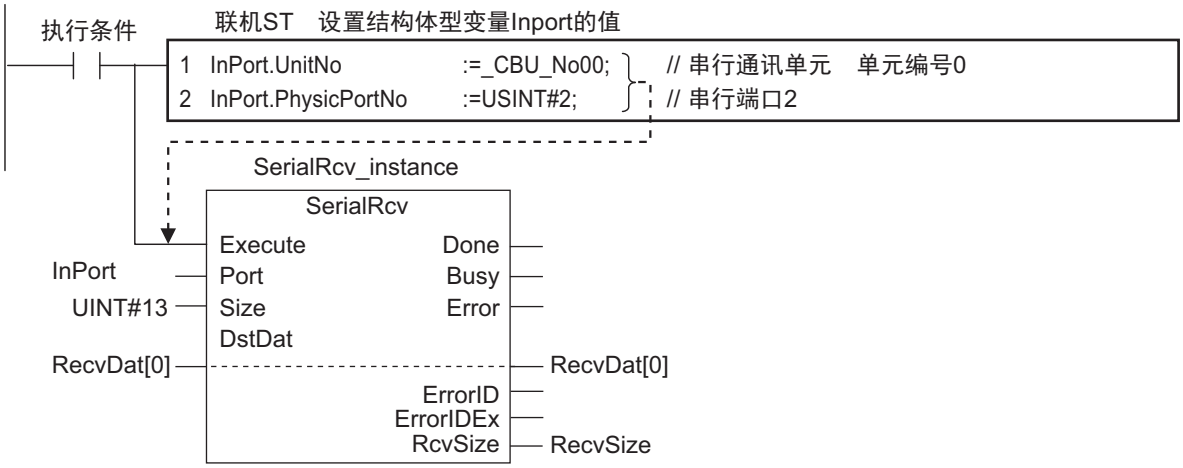


此时，需要事先创建用于输入参数、输出参数、输入输出参数的结构体型变量，并用 MOVE 指令等设置值。

例）

变量表

变量名称	数据类型
InPort	_sPORT



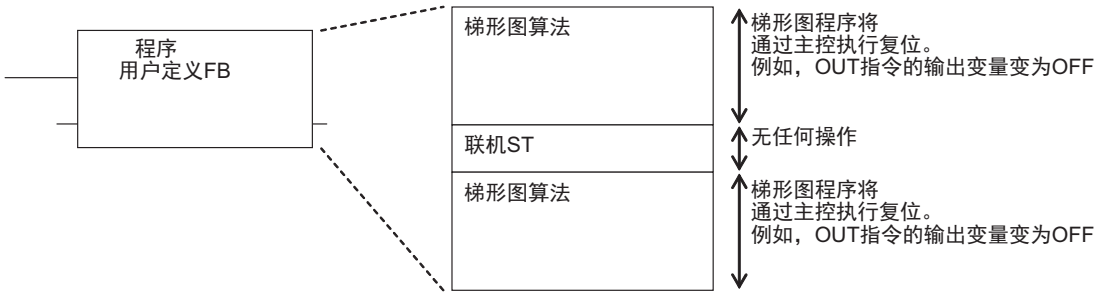
6-9-3 主控

概要

主控是指对 MC （主控开始）指令和 MCR （主控结束）指令之间存在的处理进行 OFF 输出的功能。
使用主控功能，可方便地控制较长的处理执行条件。
详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》的 MC/MCR 指令。

可使用主控的程序语言

主控可在梯形图语言中使用。
ST 语言中无法使用。梯形图程序中的联机 ST 也无法使用。
例）
主控区域内：



主控复位时区域内指令的动作

主控复位时区域内指令的动作请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360） 》的 “MC/MCR 指令 ”。

7

动作确认和运行

本章介绍包括离线调试在内的动作确认内容和动作确认到运行之间的步骤。

7-1 动作确认和运行的步骤概要 7-2

7-2 离线调试 7-3

7-2-1 模拟功能的特点 7-3
7-2-2 模拟功能的执行 7-3
7-2-3 模拟功能的设定 7-6

7-3 上机动作确认和运行 7-8

7-3-1 从上机动作确认到运行开始之间的步骤 7-8
7-3-2 项目的传送 7-9
7-3-3 I/O 的配线确认 7-9
7-3-4 MC 调试 7-9
7-3-5 用户程序的动作确认 7-10
7-3-6 正式运行的开始 7-10

7-1 动作确认和运行的步骤概要

动作确认和运行在整体步骤中，与下图阴影标示的位置对应。STEP2-4 离线调试中，不与控制器在线连接，利用模拟功能进行动作确认。STEP5. 上机进行的动作确认和运行时，与控制器在线连接，上机确认动作，动作确认完成后开始运行。

整体步骤请参考 「1-3 NJ/NX 系列的整体使用步骤」（P.1-12）。

STEP 1. 软件设计

- STEP1-1 设计 I/O 和处理
- STEP1-2 设计任务
- STEP1-3 设计程序

↓

STEP 2. 软件设定和编程

- STEP2-1 从站 / 单元构成
- STEP2-2 控制器设定
- STEP2-3 编程
- **STEP2-4 离线调试**

↓

STEP 3. 安装和硬件的设定

↓

STEP 4. 配线

↓

STEP 5. 上机确认动作和运行

7-2 离线调试

下面介绍离线调试中使用的模拟功能。模拟功能是指在电脑上模拟 NJ/NX 系列的动作，只在电脑上确认用户程序动作的功能。模拟功能中含有实际控制器上无法实现的调试功能，因此可以提高用户程序开发和调试的效率。

7-2-1 模拟功能的特点

与控制器在线连接后进行动作确认的在线调试相比，模拟功能在以下情况下有效。

- 使用断点、逐步执行、暂停等，确认程序的逻辑。
- 选择模拟对象，确认部分程序的动作。
- 通过变更模拟执行速度，以慢于实际情况的速度进行动作确认。
- 利用任务执行时间监视，测量任务执行时间的预计值。
- 用调试程序进行控制器外部输入等。

7-2-2 模拟功能的执行

模拟功能中可执行以下内容。

- 模拟器的启动和停止
- 确认程序的逻辑
- 预测任务执行时间
- 在线连接时的调试功能

模拟器的启动和停止

模拟功能通过 Sysmac Studio 的模拟窗口启动模拟器后执行。利用模拟功能完成动作确认后，停止模拟器。
模拟执行和停止操作的概要如下所示。

1 **在 Sysmac Studio 上，从主菜单中选择［显示］｜［模拟窗口］。**
工具画面右下方将显示模拟窗口。

2 **按下模拟窗口的 [执行] 按钮。**

程序将传送到模拟器中，开始模拟。开始模拟后，编辑器等 Sysmac Studio 的画面将变为与控制器在线连接时相同的状态。

3 **完成动作确认后，按下模拟窗口的 [停止] 按钮，停止模拟器。**

确认程序的逻辑

为了确认程序逻辑的正确性，可暂停模拟器的动作，或逐步执行。利用模拟窗口的按钮，可进行以下操作。

操作名称	操作内容
断点	在程序中指定的位置，暂停执行程序。
逐步执行	若在 ST 语言的程序中执行 1 行、在梯形图语言的程序中执行 1 个指令，将暂停模拟器。
连续逐步执行	以一定间隔连续逐步执行。
暂停	暂停模拟的执行。
Step-In 执行	在功能块或函数的资源内逐步执行。
Step-Out 执行	Step-In 执行中的功能块或函数结束后执行。
1 周期执行	执行 1 个周期当前正在执行的任务。在下一周期的开头暂停。
条件断点	设定断点的停止条件，仅在条件成立时暂停执行程序。

预测任务执行时间

如果以执行时间预测模式执行模拟器，从任务开始执行到完成执行所花费的时间 （任务执行时间）的预测值将显示在任务执行时间监视器中。显示任务执行时间的平均预测值和最大预测值。任务执行时间的显示请参考📖「任务执行时间的监视」（P.5-83）。

使用注意事项

- 在显示硬件修订版本的单元中，请选择相应的硬件修订版本。
- 任务执行时间预测值不一定与实际机器的执行时间一致。因用户程序中所记载逻辑的执行有无、外部的输入条件或通信有无不同，实际机器的执行时间可能会超出预测最大值。因此，预测值请作为任务设计的参考。请务必连接实际控制器后确认任务执行时间，再进行设计研究和正式运行。

在线连接时的调试功能

在模拟器上，只能执行与控制器在线连接时使用的调试功能中的一部分。控制器上的在线调试和模拟器的离线调试的区别如下表所示。各调试功能中控制器和模拟器的动作区别请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

（○：可使用、 ×：不可使用）

调试功能	控制器	模拟器
监控	○	○
通过监视窗口监视	○	○
通过 I/O 映射监视	○	○
微分监控	○	○
接点的设置 / 复位	○	○
接点的强制值刷新 TRUE/FALSE/ 解除	○	○
数据的当前值变更	○	○
存储器全部清除	○	×
交叉参考弹出窗口	○	○
在线编辑	○	○
控制器状态监视	○	×
任务执行状态监视	○	○
轴状态监视 （MC 监视表）	○	○
动作模式的变更	○	×
控制器的重置	○	×
数据追踪	○	○
触发的设定	○	○
采样变量的设定	○	○
追踪的开始和停止	○	○
追踪结果的显示	○	○
追踪结果的导出	○	○
3D 装置模型的创建	○	○
以数字、模拟图表显示	○	○
3D 形式的轨迹显示	○	○
任务执行时间的监视	○	○
执行处理时间预测	×	○
程序的模拟调试	×	○
模拟程序的设定	×	○
模拟执行速度的变更	×	○
断点的设定	×	○
逐步执行	×	○
故障排除	○	○
异常信息监视	○	○
异常记录显示	○	○
事件表设定	○	○
用户存储器的使用量监视	○	○
时钟信息的设定	○	×
访问权的强制释放	○	×

7-2-3 模拟功能的设定

模拟功能中可设定以下内容。

- 模拟程序的设定
- 调试程序的设定
- 模拟执行速度的设定

模拟程序的设定

模拟功能中，能以任务或程序为单位设定模拟对象并执行。在整个程序中，可用模拟器运行模拟对象的部分程序。
设定模拟程序的操作如下所示。

1 **显示模拟窗口。**
多视图浏览器的［编程］｜［任务］下方的程序左侧，将显示模拟对象设定复选框。

2 **以任务或程序为单位勾选模拟对象设定用复选框。**

调试程序的设定

调试程序是指通过离线调试进行动作确认时使用的程序。调试程序中，记述控制器外部的虚拟输入处理和用户异常的强制发生等。调试程序仅在模拟器上可以执行。

创建调试程序的操作概要如下所示。

1 **在多视图浏览器上右击［编程］｜［POU］｜［程序］，从菜单中选择［调试用追加］｜［ST］或［梯形图］。**
将新建调试程序。

2 **在创建的调试程序中，记述测试用的程序代码。**

3 **将调试程序分配到任务中。**

已创建的普通程序也可变更为调试程序。

1 **在多视图浏览器上右击［编程］｜［POU］｜［程序］中的各程序，设定为［调试用设定］｜［有效］。**

模拟速度的设定

用模拟窗口的“模拟速度变更滑片”，可在 0.1 ～ 1 倍的范围内，变更计算机上的执行速度。模拟执行中或停止中均可变更。
要以比实际慢的速度显示模拟器执行时使用。

7-3 上机动作确认和运行

下面介绍从上机动作确认到运行开始之间的步骤概要和操作概要。

7-3-1 从上机动作确认到运行开始之间的步骤

从上机动作确认到运行开始之间的步骤概要如下所示。

STEP1 与 Sysmac Studio 的在线连接和项目的下载	参考
1. 接通控制器的电源。 2. 与 Sysmac Studio 在线连接。 3. 通过 Sysmac Studio 传送项目 （用户程序或单元构成等各种设定）。	《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》

STEP2 上机确认动作	参考
1. 在程序模式下，通过 I/O 映射或梯形图编辑器对实际 I/O 进行强制值刷新，确认 I/O 配线。 2. 进行运动控制时，在 MC 调试画面，进行配线检查，通过点动进给确认电机的方向，通过相对定位确认移动量 （电子齿轮的设定），确认原点复位动作。 3. 设为运行模式，确认用户程序的动作。	「7-3-3 I/O 的配线确认」（P.7-9） 《NJ/NX 系列 用户手册 运动控制篇 （SBCE-CN5-363）》 「7-3-5 用户程序的动作确认」（P.7-10）

STEP3 控制器的正式运行	参考
1. 确认按设计动作后，开始正式运行。	—

参考

要从 Sysmac Studio 下载项目，需要利用 “ 同步功能 ”，从计算机传送到控制器上。“ 同步 ” 的详情请参考《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

7-3-2 项目的传送

从 Sysmac Studio 向实际控制器传送项目。

1 **与控制器在线连接后，从 [控制器] 菜单选择 [同步]。**
计算机上的数据和实际控制器的数据将自动核对。

2 **单击 [传送 [计算机→控制器]] 按钮。**

7-3-3 I/O 的配线确认

通过 Sysmac Studio 的监视窗口等执行强制值刷新，确认 I/O 的配线。I/O 的配线可向进行配线确认的从站或单元的 I/O 写入值，并对结果进行监视。

强制值刷新的详情请参考📖「8-5-1 强制值刷新」（P.8-33）。

7-3-4 MC 调试

MC 调试是指不使用用户程序，主要通过 Sysmac Studio 进行以下操作。

- 配线的确认：伺服驱动器连接器的输入输出信号监视、
  伺服驱动器的状态监视
- 电机的动作及方向确认：伺服 ON、点动进给
- 电子齿轮的设定确认：进行相对值定位，确认 / 变更移动量
- 原点复位确认：确认原点复位动作

通过 Sysmac Studio 与控制器进行在线连接，在 MC 调试画面进行。

详情请参考📖《NJ/NX 系列 CPU 单元 用户手册 运动控制篇 （SBCE-CN5-363）》。

进行以下操作。

1 **进行必要的配线后，通过 Sysmac Studio 与控制器在线连接。**

2 **进行 " 轴 " 的创建、轴的分配、以下轴参数的设定。**
MC 调试所需的轴参数设定：
显示单位选择、电机每转 1 圈的脉冲数及移动量
最高速度、点动最高速度、最大加速度和减速度、
软件限位功能选择、软件限位、计数模式

3 **打开 MC 调试画面，进行以下操作。**
例）
- 监视器、配线确认
- 通过点动进给确认电机的方向
- 通过相对定位确认移动量 （电子齿轮的设定）
- 原点复位动作的确认

7-3-5 用户程序的动作确认

上机进行用户程序的动作确认时，应将 CPU 单元的动作模式设为运行模式。动作确认包括以下项目。

- 确认用户程序的动作
- 通过在线编辑修改用户程序
- 通过数据追踪确认用户程序的动作

确认用户程序的动作

用户程序的动作确认按以下步骤进行。

1 **监视用户程序的执行状态**

2 **更新接点或线圈的状态、变量的值，进行动作确认**

● 监视用户程序的执行状态

对控制器的接点、线圈的 ON/OFF 状态、变量的当前值进行监视。监视可通过 Sysmac Studio 的梯形图编辑器、监视窗口、 I/O 映射进行。

● 更新接点或线圈的状态、变量的值，进行动作确认

更新用户程序的接点、线圈的 ON/OFF 状态、变量的值，通过监视确认是否设计动作。若要变更接点或线圈的状态，则使用 “ 强制值刷新 ”，若要变更变量的值，则使用 “ 变更当前值 ”。

详情请参考📖「8-5-1 强制值刷新」（P.8-33）、📖「8-5-2 当前值变更」（P.8-37）。

通过在线编辑修改用户程序

通过动作确认，如果有判断为需要修改的用户程序，则通过 “ 在线编辑 ” 进行修改。使用 “ 在线编辑 ”，无需停止 CPU 单元的运行，即可变更用户程序。

详情请参考📖「8-5-3 在线编辑」（P.8-39）。

通过数据追踪确认动作

使用 “ 数据追踪 ”，确认接点或线圈的 ON/OFF 时间和变量值的变动等。

详情请参考📖「8-5-4 数据追踪」（P.8-41）。

7-3-6 正式运行的开始

将动作模式设为运行模式，开始正式运行。请对编写完成的用户程序及各种数据、设定值进行充分的动作确认后，再转移到正式运行。

8

CPU 内置功能

本章介绍 CPU 单元中可使用的功能。

8-1 数据管理、时钟、运行相关功能 8-3
8-1-1 存储器全部清除 8-3
8-1-2 时钟功能 8-3
8-1-3 运行中输出功能 8-6

8-2 CJ 单元相关管理功能 8-7
8-2-1 基本 I/O 单元相关 8-7
8-2-2 高功能单元相关 8-8

8-3 SD 存储卡功能 8-10
8-3-1 SD 存储卡功能一览 8-10
8-3-2 支持的 SD 存储卡及文件夹 / 文件规格 8-11
8-3-3 SD 存储卡操作指令 8-13
8-3-4 FTP 客户端通信指令 8-13
8-3-5 FTP 服务器功能 8-14
8-3-6 通过 Sysmac Studio 进行的文件操作 8-14
8-3-7 SD 存储卡的寿命检测功能 8-14
8-3-8 SD 存储卡相关的系统定义变量一览 8-15
8-3-9 SD 存储卡的自检功能 8-16
8-3-10 SD 存储卡中文件访问的排他控制 8-18

8-4 安全功能 8-19
8-4-1 用户程序执行用 ID 认证功能 8-20
8-4-2 无用户程序复原信息传送功能 8-23
8-4-3 项目文件整体的保护功能 8-24
8-4-4 数据保护功能 8-25
8-4-5 操作权限的认证功能 8-27
8-4-6 CPU 单元的写入保护功能 8-29
8-4-7 CPU 单元名称功能及序列 ID 功能 8-30

8-5 调试功能 8-33
8-5-1 强制值刷新 8-33
8-5-2 当前值变更 8-37
8-5-3 在线编辑 8-39
8-5-4 数据追踪 8-41
8-5-5 微分监控 8-46

8-6 事件日志功能 8-51
8-6-1 概要 8-51

8-6-2 事件日志功能的详情 8-52
8-6-3 控制器事件 （控制器异常 / 信息） 8-57
8-6-4 用户事件 （用户异常 / 信息） 8-58

8-7 事件重要程度变更功能 8-64
8-7-1 事件重要程度变更的用途 8-64
8-7-2 可变更事件重要程度的事件 8-64
8-7-3 事件重要程度变更的操作方法 8-65

8-1 数据管理、时钟、运行相关功能

下面介绍数据管理功能、时钟和运行相关功能。

8-1-1 存储器全部清除

可通过 Sysmac Studio，将 CPU 单元内的用户程序、控制器配置和设定、变量、绝对值编码器原点位置偏置初始化为出厂状态。这种操作称为 “ 存储器全部清除 ”。

使用注意事项

- 存储器全部清除仅可在程序模式下执行。
- 设定了安全功能的 “CPU 单元的写入保护 ” 时，无法执行。
- 请勿在存储器全部清除过程中关闭 CPU 单元的电源。

存储器全部清除后，与用出厂状态的 CPU 单元创建系统构成后的动作相同。

● 通过 Sysmac Studio 操作

通过 Sysmac Studio 与 CPU 单元在线连接，从 [控制器] 菜单中选择 [存储器全部清除]。

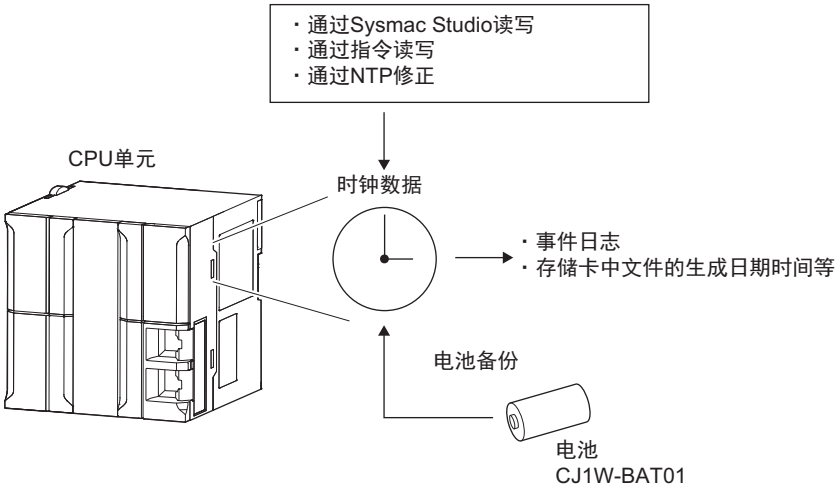
操作详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

8-1-2 时钟功能

概要

CPU 单元内置有时钟 （RTC）。
该时钟的时刻数据将反映到事件日志、 SD 存储卡的文件生成日期时间中。
包括以下功能。

- 通过 Sysmac Studio 读取 / 写入
- 通过指令读取 / 写入
- 通过系统定义变量读取 （不可写入）
- 通过 NTP 进行时钟修正



使用注意事项

时钟的时刻数据在电源 OFF 时由电池保持，因此接通电源时的时刻数据将变得不正确。此外，电源OFF时，可在接通电源后通过EtherNet/IP网络，利用NTP重新设定时刻数据，确保正确的时刻。

● 时刻数据的范围

- NX 系列 CPU 单元时　　：1970-01-01 ～　2069-12-31
- NJ 系列 CPU 单元时　　：1970-01-01 ～　2106/2/6

● 时区和本地时间的设定

首次使用时，请登录时区和本地时间 （时刻数据）。
时区和本地时间在 Sysmac Studio 的时钟设定画面中设定。
EtherCAT 从站或 CJ 单元从 CPU 单元读取的时刻数据及设定的时刻数据为与时区对应的本地时间。

参考

未安装电池或电池用完时，时区设定会保持，但时刻数据不会保持，将变为不正确的值。

时刻数据的设定方法

包括以下方法。

● 通过 Sysmac Studio 变更

使用 Sysmac Studio，能够以与计算机的时刻同步的形式，设定内置时钟的时刻数据。

● 通过指令变更

可使用 SetTime 指令设定时刻数据。

● 通过 NTP 服务器变更

可使用 EtherNet/IP 上的 NTP 服务器设定时刻数据。

通过 NTP 进行时钟修正

● 目的

在网络系统中，系统整体的时刻数据需要共通。为了方便地进行校准，提供 NTP 功能。

● 规格

支持 NTP 客户端功能。

详情请参考📖《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册 (SBCD-CN5-359)》。

时刻数据的读取方法

时刻数据不正确时，将直接读取不正确的值。

● 通过指令读取

在用户程序中，可使用 GetTime 指令读取时刻数据。

● 通过系统定义变量读取 （不可写入）

使用以下系统定义变量，可读取时刻数据。
“_CurrentTime” （系统时刻）

● 通过 Sysmac Studio 读取

可使用 Sysmac Studio，通过 ［控制器］菜单的 ［控制器时钟］，读取时刻数据并显示。

记录

变更了时钟数据时，将记录到事件日志中。
但是，通过 NTP 补偿时，不会记录到事件日志中。

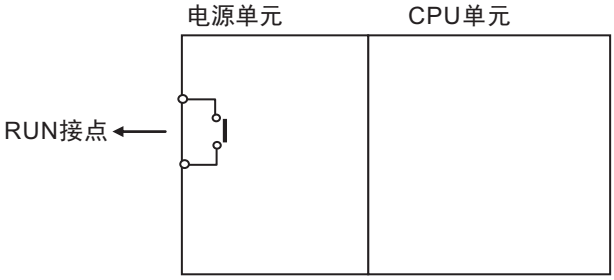
相关系统定义变量

变量名称	名称	功能	数据类型	R/W
_CurrentTime	系统时刻	保存 CPU 单元的内部时钟数据。	DATE_AND_TIME	R

8-1-3 运行中输出功能

概要

从电源单元 NX-PA9001、NX-PD7001、NJ-P □ 3001 输出表示 CPU 单元正在运行的 RUN 接点（运行中输出）信号。



RUN 接点 （运行中输出）动作如下。

状态	动作
运行中 （运行模式）	ON
启动中 （接通电源后，按照接通电源时的动作模式设定变为运行状态前）	OFF
运行停止中 （程序模式）	
发生全部停止故障等级的控制器异常	

电源单元 NX-PA9001、 NX-PD7001、 NJ-P □ 3001 的 RUN 接点 （运行中输出）的额定规格如下。

项目	内容
接点配置	1a
开闭能力	AC250V2A （阻性负载） AC120V0.5A （感性负载） DC24V2A （阻性负载）

用途

- 向上级通知 CPU 单元是否正常动作和运行的存活信号
- 多个 CPU 单元完成启动的同步
- CPU 单元开始运行后解除联锁

安全要点

从接通电源到变为运行模式约需 10 ～ 20 秒。在此期间的输出遵照从站或单元的规格。
请设计故障安全回路，避免在使用电源单元的运行中输出等时，外部设备发生误动作。

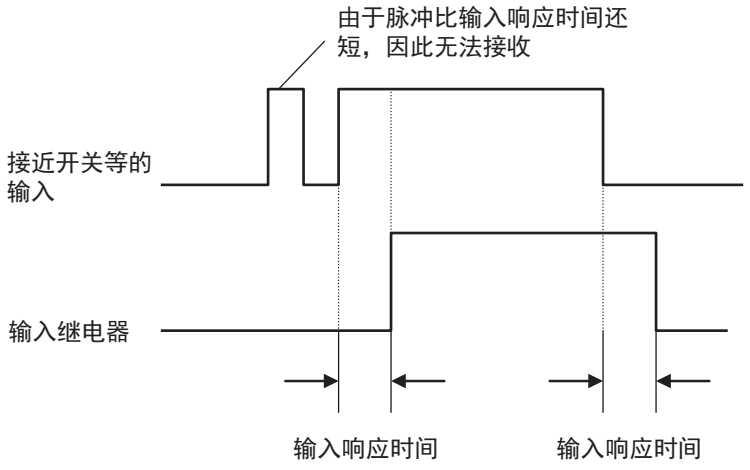
8-2 CJ 单元相关管理功能

下面介绍单元相关的管理功能。
可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

8-2-1 基本 I/O 单元相关

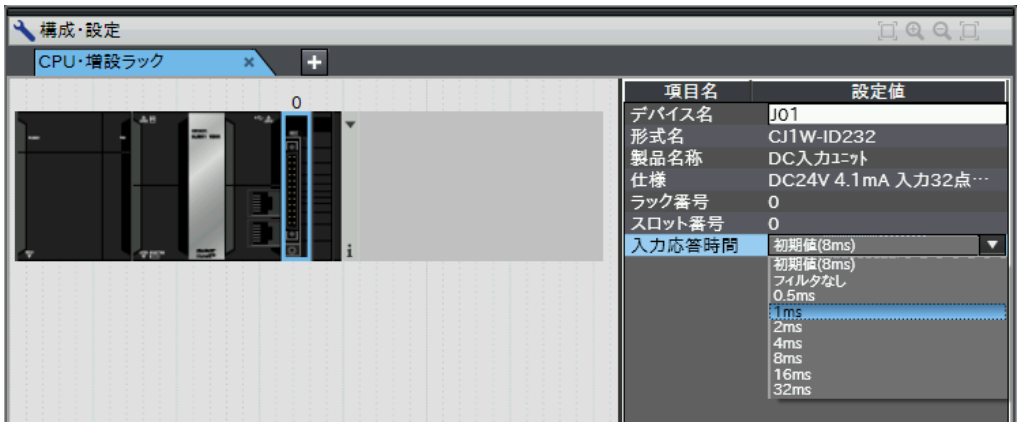
概要

延长输入响应时间，可减少抖动或外部干扰的影响。反之，减小该值（在 ON 时间或 OFF 时间大于刷新时间的条件下），可以接收短脉冲输入。



设定方法

在 Sysmac Studio 的多视图浏览器中双击［配置和设定］｜[CPU・扩展装置]，通过基本 I/O 单元的单元信息 “ 输入响应时间 ”，从下拉菜单中选择输入响应时间。



要启用设定，需要进行以下任一操作。

- 控制器电源的 OFF/ON
- 通过 Sysmac Studio 进行控制器重置 （CPU 单元整体）

相关系统定义变量

基本输入单元的输入响应时间设定值将输出到以下系统定义变量中。

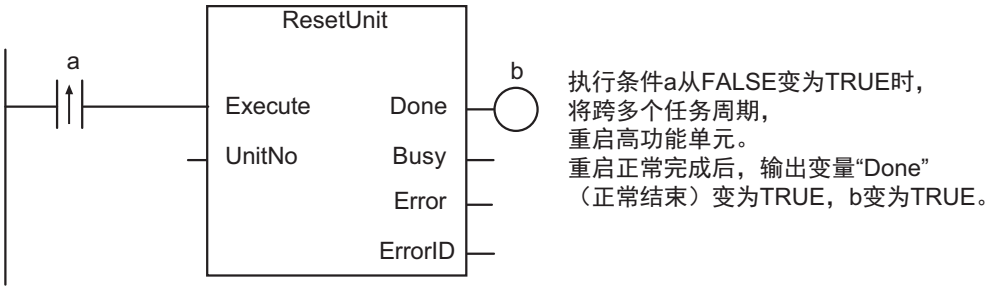
变量名称	名称	功能	数据类型	R/W
_CJB_InRespTm	基本输入单元的输入时间常数	以 0.1ms 为单位表示基本输入单元的输入时间常数。	ARRAY [0..3, 0..9] OF UINT	R

8-2-2 高功能单元相关

高功能单元的重启

不关闭控制器的电源，直接将高功能单元 （高功能 I/O 单元或 CPU 总线单元）的单元设定值设为有效。通过以下 ResetUnit （重启单元）指令执行。

指令名称	指令	功能
重启单元	ResetUnit	重启 CPU 总线单元或高功能 I/O 单元。



通过以下任一方式变更了高功能单元设定时，需要重启单元或将控制器的电源从 OFF 设为 ON。

- 在 Sysmac Studio 的高功能单元设定画面中发生编辑时
- 在 I/O 映射或监视窗口中发生编辑时
- 在用户程序内部设定时

● 相关系统定义变量

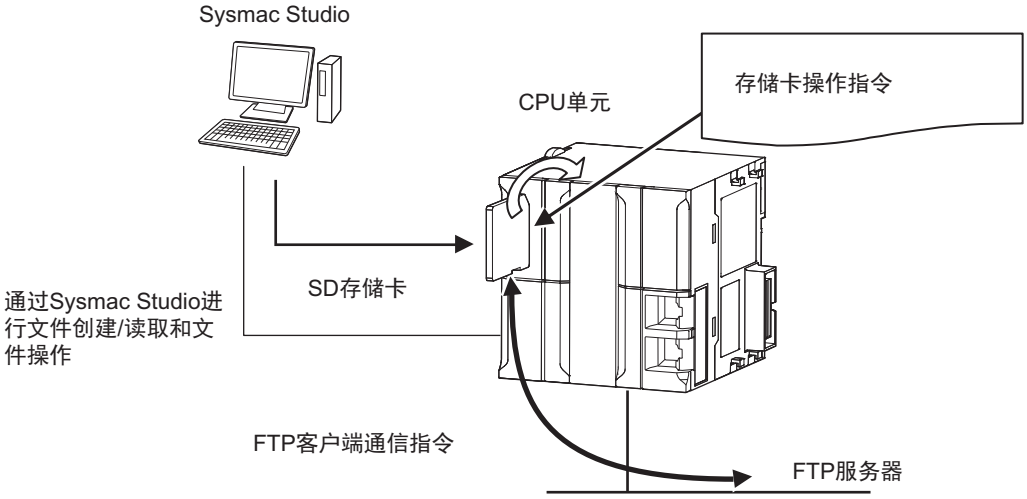
变量名称	名称	功能	数据类型	R/W
_CJB_CBU00InitSta ・ ・ ・ _CJB_CBU15InitSta	CPU 总线单元初始化中标志	CPU 总线单元正在初始化时，变为 TRUE。 初始化完成时，变为 FALSE。 各个位与单元编号对应。	BOOL	R
_CJB_SIO00InitSta ・ ・ ・ _CJB_SIO95InitSta	高功能 I/O 单元初始化中标志	高功能 I/O 单元正在初始化时，变为 TRUE。 初始化完成时，变为 FALSE。 各个位与号机 No. 对应。	BOOL	R
_CJB_CBU00Restart ・ ・ ・ _CJB_CBU15Restart	CPU 总线单元重启标志	上升时，重启 CPU 总线单元 （重启处理后，在系统中变为 FALSE）。 各个位与号机编号对应。 执行指令时，若重启标志变为 TRUE，将在执行下一任务周期的刷新处理时，开始重启处理。	BOOL	RW
_CJB_SIO00Restart ・ ・ ・ _CJB_SIO95Restart	高功能 I/O 单元重启标志	上升时，重启高功能 I/O 单元 （重启处理后，在系统中变为 FALSE）。 各个位与号机编号对应。 执行指令时，若重启标志变为 TRUE，将在执行下一任务周期的刷新处理时，开始重启处理。	BOOL	RW

8-3 SD 存储卡功能

下面介绍 SD 存储卡功能。

8-3-1 SD 存储卡功能一览

NJ/NX 系列 CPU 单元支持的 SD 存储卡功能如下。



功能名称	概要
SD 存储卡操作指令	通过用户程序中的指令访问 SD 存储卡的功能。
FTP 客户端通信指令*1	通过 CPU 单元使用 FTP 向 Ethernet 上的计算机及控制器传输文件所需的指令。
FTP 服务器功能	通过内网的 FTP 客户端，使用 FTP 指令，经由 EtherNet/IP，读写 SD 存储卡中的大容量文件的功能。
通过 Sysmac Studio 进行的文件操作功能	通过 Sysmac Studio 对 CPU 单元中安装的 SD 存储卡进行文件操作的功能。用户不仅可以向 SD 存储卡保存控制器用文件，还可以保存通用文档文件等。
SD 存储卡的寿命检测功能	通过系统定义变量或事件日志通知 SD 存储卡寿命的功能。
SD 存储卡备份功能、 SD 存储卡自动传送功能、 SD 存储卡程序传送功能*2	使用 CPU 单元中安装的 SD 存储卡，对控制器的用户程序和数据进行备份、恢复、核对的功能。详情请参考📖「9-2 SD 存储卡的备份功能」（P.9-12）、「9-4 SD 存储卡自动传送功能」（P.9-26）、「9-5 SD 存储卡程序传送功能」（P.9-28）。

*1. FTP 客户端通信指令可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。
*2. 可在 CPU 单元版本 1.11 以上和 StudioVer.1.15 以上的组合中使用。

8-3-2 支持的 SD 存储卡及文件夹 / 文件规格

SD 存储卡的规格

本产品支持的存储卡为 SD 存储卡及 SDHC 卡。但是，能保证动作的仅限以下欧姆龙生产的 SD 存储卡。若使用动作保证外的 SD 卡或 SDHC 卡，可能无法正常动作。

型号	卡种类	容量 [GB]	格式	重写次数	写保护
HMC-SD291	SD 卡	2	FAT16	10 万次	可通过 SD 存储卡本身的硬件开关设定
HMC-SD491	SDHC 卡	4	FAT32		

以下情况下，系统定义变量 “_Card1Err” （SD 存储卡错误标志）将变为 TRUE （监视信息等级）。

- 格式异常时

此时，正面的 “SD PWR”LED 熄灭， SD 存储卡不可访问。

文件夹 / 文件的规格

● 字符限制

用户命名的内容	可使用的字符	保留字	多字节字符使用可否	大小写的区分	最大大小 （不含末尾的 NULL 字符）
旋钮标签名称	0 ～ 9、A ～ Z、 a ～ z % - _ @ ! ` （ ） ~ = # & + ^ [ ] { } , . ; 半角假名*1	CON, PRN, AUX, CLOCK$, NUL, COM0, COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9, LPT0, LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9	不可*2	否	11 字节
目录名	0 ～ 9、A ～ Z、 a ～ z $ % ' - _ @ ! ` （ ） ~ = # & + ^ [ ] { } , . ; 半角假名				65 字节
文件名					65 字节

*1. 第一个字符不可使用空格。
*2. 多字节字符 （日文等）在计算机上可以使用，但在 CPU 单元上无法使用。

● 子目录层级

最大 5 个层级 （例：f1/f2/f3/f4/f5/abc.txt）

● 可保存的最大文件数

SD 存储卡中可保存的最大文件数因保存文件的目录层级不同而异。各自的最大文件数如下所示。但是，下表的值为文件名 8.3 格式的情况。文件名为长文件名时，可保存的最大文件数会小于下表的值。

目录层级	格式	可保存的最大文件数
根目录	FAT16	511
	FAT32	65,533
子目录	FAT16、 FAT32	65,533

● 1 个文件的最大大小

1 个文件的最大大小为 2,147,483,647 字节 (2GB-1 字节)。

8-3-3 SD 存储卡操作指令

可利用以下指令，对 SD 存储卡进行各种操作。

指令名称	指令	功能
读取变量文件	FileReadVar	读取 SD 存储卡中的文件 （二进制格式），保存到指定的变量 （排列变量或结构体变量亦可）中。
写入变量文件	FileWriteVar	将指定变量 （排列变量或结构体变量亦可）的值以二进制格式写入到 SD 存储卡的文件中。 文件名中指定的目录不存在时，将同时创建该目录。
打开文件	FileOpen	打开指定的文件。
关闭文件	FileClose	关闭指定的文件。
查找文件	FileSeek	设定指定文件的文件位置指示符。
读取文件	FileRead	从指定的文件中读取数据。
写入文件	FileWrite	将数据写入指定的文件中。
读取字符串	FileGets	从指定的文件中读取 1 行字符串。
写入字符串	FilePuts	将 1 行字符串写入指定的文件中。
删除文件	FileRemove	删除 SD 存储卡中的指定文件。
重命名文件	FileRename	变更指定文件的文件名或目录名。
复制文件	FileCopy	将指定的文件复制为其他文件名。
创建目录	DirCreate	在 SD 存储卡中创建目录。
删除目录	DirRemove	删除 SD 存储卡中的指定目录。

8-3-4 FTP 客户端通信指令

FTP 客户端通信指令是指，通过 NJ/NX 系列 CPU 单元使用 FTP 向 Ethernet 上的计算机及控制器传输文件所需的指令。以下指令中，在执行指令时，将对 SD 存储卡中的文件进行读写。

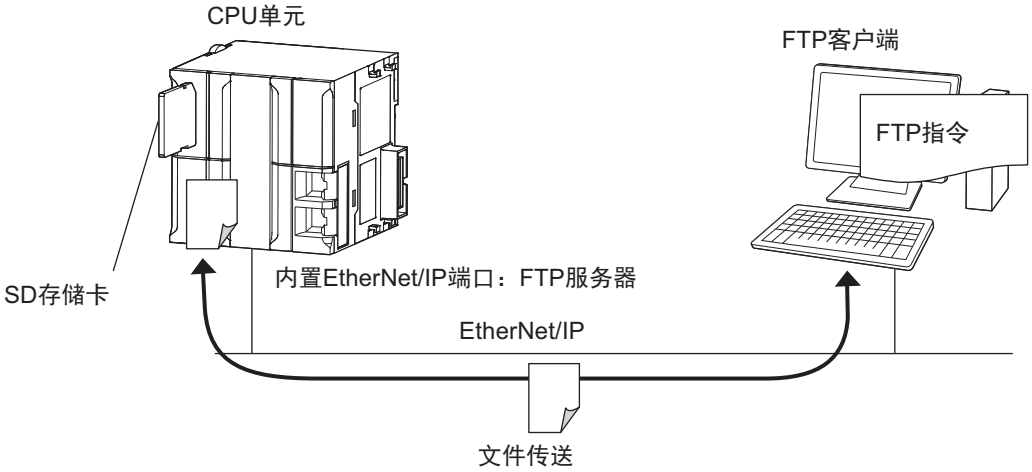
指令名称	指令	功能
向 FTP 服务器上传文件	FTPPutFile	将 FTP 客户端的 SD 存储卡中的文件上传到 FTP 服务器上。
从 FTP 服务器下载文件	FTPGetFile	将 FTP 服务器上的文件下载到 FTP 客户端的 SD 存储卡中。

版本相关信息

FTP 客户端通信指令可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。

8-3-5 FTP 服务器功能

可通过FTP客户端，对内置EtherNet/IP端口发行FTP指令，经由EtherNet/IP，读写SD存储卡中的文件。
详情请参考📖《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册 (SBCD-CN5-359)》。



8-3-6 通过 Sysmac Studio 进行的文件操作

通过 Sysmac Studio 对 CPU 单元中安装的 SD 存储卡进行文件操作。
不仅可以向 SD 存储卡保存控制器用文件，还可以保存文档文件等。

8-3-7 SD 存储卡的寿命检测功能

可在 SD 存储卡因达到寿命而受到物理破坏前，掌握 SD 存储卡的寿命。
可通过以下方式检测 SD 存储卡的寿命。

- 系统定义变量 “_Card1Deteriorated” （SD 存储卡使用寿命警告标志）
- 将 “SD 存储卡使用寿命 （监视信息） ” 登录到事件日志中

检测时间为电源接通时和安装有 SD 存储卡期间 （定期检查）。

检测到 SD 存储卡达到使用寿命时，请保存 SD 存储卡中的数据或更换 SD 存储卡。

8-3-8 SD 存储卡相关的系统定义变量一览

作为 SD 存储卡功能的状态进行通知的系统定义变量如下所示。

变量名称	名称	功能	数据类型
_Card1Ready	SD 存储卡可使用标志	识别到 SD 存储卡时，变为 TRUE。 未识别到时，为 FALSE。 TRUE：可使用 FALSE：不可使用	BOOL
_Card1Protect	SD 存储卡写保护标志	SD 存储卡为写保护时，变为 TRUE。 TRUE：有写保护 FALSE：无写保护	BOOL
_Card1Err	SD 存储卡错误标志	安装了不可使用的 SD 存储卡或格式异常时，变为 TRUE。 TRUE：有错误 FALSE：无错误	BOOL
_Card1Access	SD 存储卡访问中标志*1	正在访问 SD 存储卡时，变为 TRUE。 TRUE：访问中 FALSE：非访问中	BOOL
_Card1Deteriorated	SD 存储卡寿命警告标志	检测到 SD 存储卡达到使用寿命时，变为 TRUE。 TRUE：检测到使用寿命 FALSE：未检测到使用寿命	BOOL
_Card1PowerFail	SD 存储卡访问中断电标志*2	正在访问 SD 存储卡的过程中，若控制器发生断电，则变为 TRUE。 TRUE：正在访问 SD 存储卡时发生断电 FALSE：正常	BOOL

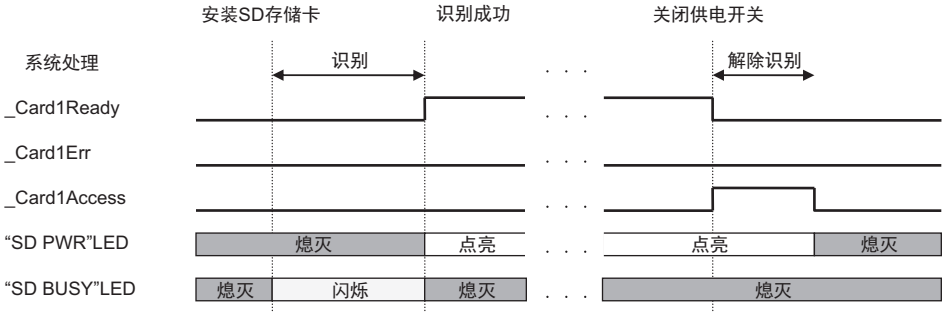
*1. 使用 SD 存储卡访问中标志 （“_Card1Access”）时的注意事项
SD 存储卡访问中标志的目的是向外部进行通知。对 SD 存储卡的访问状态不会实时反映。因此，请勿在用户程序中参照本标志。
由于 SD 存储卡的访问状态不会实时反映，所以可能导致用户程序的动作不正确。

*2. 使用 SD 存储卡访问中断电标志 （“_Card1PowerFail”）时的注意事项
SD 存储卡访问中断电标志为 TRUE 时，请确认 SD 存储卡中是否存在正确的文件，或装置是否正确动作。
不存在正确的文件或装置无法正确动作时，请将正确的文件下载到 SD 存储卡中。请重新接通 CPU 单元的电源或重置控制器，确认装置正确动作。
处理完成后，请将 SD 存储卡访问中断电标志设为 FALSEFALSE。（本标志不会自动变为 FALSE）。

（注）SD 存储卡备份功能相关的系统定义变量请参考「9-2 SD 存储卡的备份功能」（P.9-12）。

参考

SD 存储卡的识别 / 解除安装时的时序图



8-3-9 SD 存储卡的自检功能

电源接通时，可对安装的 SD 存储卡执行自检。

在Sysmac Studio的"配置和设定"－"控制器设定"－"动作设定"中，选择在电源接通时是否执行以下自检。

- 文件系统的检查
- 相当于 CHKDSK 的检查
- 检查 NG 时的修复功能

设定大项目	设定中项目	设定小项目	功能说明	设定值
动作设定－动作设定标签－整体设定	SD 存储卡设定	电源接通时的存储卡诊断*1	设定在电源接通时，是否对安装的 SD 存储卡执行自诊断 （文件系统的检查和修复）。	不检查 检查

*1. SD 存储卡本身锁定 （设定为写保护）时，不执行自检。

● 自检的执行结果

情况	LED			异常的种类	处理	备注
	RUN	SD PWR	SD BUSY			
自检执行中	闪烁	熄灭	点亮	—	—	
自检没有发现问题时	—	点亮	熄灭	正常	无	
SD 存储卡的格式不正确时	—	熄灭	熄灭	监视信息	请用 Sysmac Studio 进行格式化。	
检查文件系统时检测到错误，自动执行了文件系统的修复时	—	熄灭	修复时：闪烁 完成后：熄灭	监视信息	确认是否因 Sysmac Studio 的文件操作或将 SD 存储卡安装到计算机并进行修复后，而删除了文件。	检测到损坏的文件时，尝试修复损坏的文件。
SD 存储卡故障时	—	熄灭	熄灭	监视信息	请更换 SD 存储卡。	

安全要点

电源接通时，如果修复功能动作，变为运行模式需要一定的时间。在此期间，输出变为 OFF，也不能与外部进行通信。
请设计故障安全回路，避免在使用电源单元的 RUN 接点输出等时，外部设备发生误动作。

使用注意事项

即使电源接通时的 SD 存储卡自检设定为有效，正在访问 SD 存储卡时，也不要断开 CPU 单元的电源。
若有损坏的文件，将通过 SD 存储卡修复功能尝试修复，修复失败时，可能在接通电源时自动删除。

8-3-10 SD 存储卡中文件访问的排他控制

SD 存储卡中的文件访问方法有以下 4 种。

(1) FTP 服务器功能

(2) SD 存储卡操作指令

(3) FTP 客户端通信指令

(4) 通过 Sysmac Studio 进行的文件操作

若同时对 SD 存储卡中的同一文件进行访问，可能发生意外处理，如读取正在写入的文件、覆盖正在读取的文件等。因此，需要进行排他控制，防止对同一文件同时执行上述处理。以下表示需要排他控制的处理组合。执行需要排他控制的组合处理时，请在确认一个处理结束后，再执行另一个处理。

<table>
<tr><th colspan="3" rowspan="3"></th><th colspan="6">前面正在执行的访问</th></tr>
<tr><th colspan="2">指令*1</th><th colspan="2">通过 Sysmac Studio 进行的文件操作</th><th colspan="2">FTP 服务器功能</th></tr>
<tr><th>读取</th><th>写入</th><th>读取</th><th>写入</th><th>读取</th><th>写入</th></tr>
<tr><td rowspan="6">之后执行的访问</td><td rowspan="2">指令</td><td>读取</td><td colspan="2" rowspan="2">自动进行排他控制，之后执行的指令将发生错误。</td><td>无需排他控制。</td><td rowspan="2">请执行排他控制。</td><td>无需排他控制。</td><td rowspan="4">请执行排他控制。</td></tr>
<tr><td>写入</td><td>请执行排他控制。</td><td>请执行排他控制。</td></tr>
<tr><td rowspan="2">通过 Sysmac Studio 进行的文件操作</td><td>读取</td><td>无需排他控制。</td><td>请执行排他控制。</td><td>—</td><td>—</td><td>无需排他控制。</td></tr>
<tr><td>写入</td><td colspan="2">请执行排他控制。</td><td>—</td><td>—</td><td>请执行排他控制。</td></tr>
<tr><td rowspan="2">FTP 服务器功能</td><td>读取</td><td>无需排他控制。</td><td>请执行排他控制。</td><td>无需排他控制。</td><td>请执行排他控制。</td><td>—</td><td>—</td></tr>
<tr><td>写入</td><td colspan="4">请执行排他控制。</td><td>—</td><td>—</td></tr>
</table>

*1. 指令包括 SD 存储卡操作指令和 FTP 客户端通信指令。

8-4 安全功能

下面介绍 NJ/NX 系列中可使用的安全功能。

使用安全功能后，可保护控制器中的用户程序和各种数据，以保护资产。此外，为了防止误操作，还可限制 Sysmac Studio 上的操作。

NJ/NX 系列拥有以下安全功能。

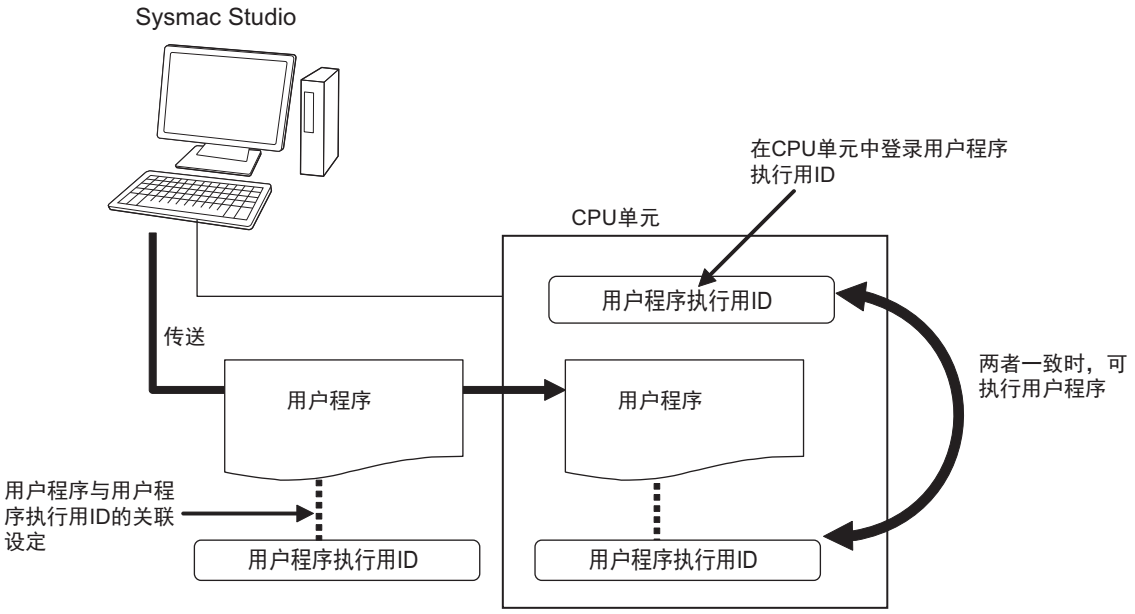
目的	安全功能	功能概要	参考
防止资产盗用	用户程序执行用 ID 认证功能	即使复制了用户程序，也无法在其他 CPU 单元上运行。	「8-4-1 用户程序执行用 ID 认证功能」（P.8-20）
	无用户程序复原信息传送功能	不向 CPU 单元传送程序的复原信息 （源代码），用其他计算机上传时，将不会显示程序。	「8-4-2 无用户程序复原信息传送功能」（P.8-23）
	项目文件整体的保护功能	对项目文件设置密码保护，保护用户资产信息。	「8-4-3 项目文件整体的保护功能」（P.8-24）
	数据保护功能*1	对项目中需要保护的数据进行保护，以保护用户资产信息。	「8-4-4 数据保护功能」（P.8-25）
防止误操作	操作权限的认证功能	在 Sysmac Studio 上对 CPU 单元设定操作权限，限制可操作的功能。	「8-4-5 操作权限的认证功能」（P.8-27）
	CPU 单元的写入保护功能	将无法通过 Sysmac Studio 替换控制器中的数据。	「8-4-6 CPU 单元的写入保护功能」（P.8-29）
防止误连接	CPU 单元名称功能	确认计算机和 CPU 单元的控制器名称或序列 ID 是否一致，防止在线连接错误。	「8-4-7 CPU 单元名称功能及序列 ID 功能」（P.8-30）

*1. 可在 CPU 单元版本 1.01 以上和 Sysmac Studio Ver.1.02 以上的组合中使用。

8-4-1 用户程序执行用 ID 认证功能

概要

事先在 CPU 单元上设定特定的 ID （称为 “ 用户程序执行用 ID”），可只执行与该 ID 关联的用户程序。



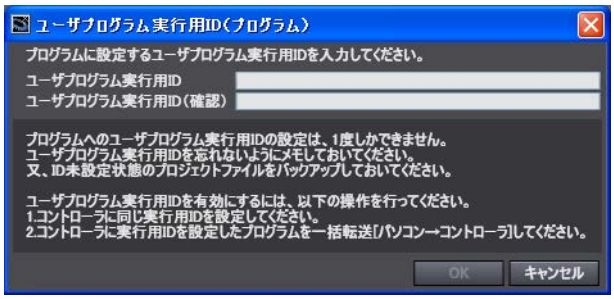
因此，可防止在不同的 CPU 单元 （硬件）上使用某个用户程序。

此外，本功能与保护功能不同，程序仍可显示和编辑。

操作方法

1 **尚未进行用户程序执行用 ID 关联设定的项目文件必须事先备份。**

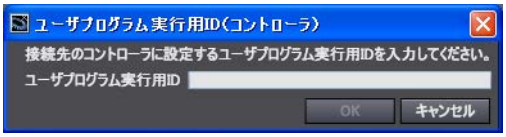
2 **在 Sysmac Studio 上，以离线的状态，在用户程序上进行用户程序执行用 ID 的关联设定。**



使用注意事项

进行了用户程序执行用 ID 关联设定的用户程序将不可进行 ID 的变更及删除。
若要变更，请读取步骤 1 中事先备份的项目文件，再关联为不同的用户程序执行用 ID。若要删除，请使用步骤 1 中事先备份的项目文件。

3 **将 Sysmac Studio 在线连接，在 CPU 单元中登录 2 中关联的用户程序执行用 ID。**



向 CPU 单元登录用户程序执行用 ID 的操作将登录到事件日志中。
此时，即使已登录用户程序执行用 ID，也会覆盖登录。

4 **将关联设定为相同用户程序执行用 ID 的用户程序传送到 CPU 单元。**
如果用户程序侧和 CPU 单元侧的用户程序执行用 ID 不一致，或者一方不存在时，变更为运行模式后，将发生 ID 核对错误 （全部停止故障等级的控制器异常），无法运行。

使用注意事项

进行了用户程序执行用 ID 关联设定的 CPU 单元将不可进行 ID 的删除及读取。如果要删除，请对 CPU 单元进行存储器全部清除。

● 发生 “ID 核对错误 ” 时的操作

CPU 单元侧的用户程序执行用 ID 不正确或未登录时：

通过 Sysmac Studio 与 CPU 单元在线连接，

1 **将正确的用户程序执行用 ID 登录到 CPU 单元中。**

2 **请将控制器电源从 OFF 设为 ON，或通过 Sysmac Studio 进行复位操作。**

用户程序侧的用户程序执行用 ID 不正确或未进行关联设定时：

1 **请用 Sysmac Studio 读取已备份的项目文件，再关联为正确的用户程序执行用 ID。**

2 **请将 Sysmac Studio 与 CPU 单元在线连接，传送用户程序。**

3 **请将控制器电源从 OFF 设为 ON，或通过 Sysmac Studio 进行控制器复位操作。**

● **其他情况时**

要删除用户程序侧的用户程序执行用 ID 时：

请用 Sysmac Studio 读取已备份的项目文件。

要删除 CPU 单元侧的用户程序执行用 ID 时：

请将 Sysmac Studio 与 CPU 单元在线连接，进行存储器全部清除。

要确认用户程序侧的用户程序执行用 ID 时：

出于安全考虑，不可通过 Sysmac Studio 等确认与用户程序关联的用户程序执行用 ID。请根据备份的项目文件，重新进行关联设定。

要确认 CPU 单元侧的用户程序执行用 ID 时：

出于安全考虑，不可通过 Sysmac Studio 等确认 CPU 单元中登录的用户程序执行用 ID。请在进行存储器全部清除后，重新登录。

规格

● **用户程序执行用 ID 的核对规格**

核对时序

CPU 单元将在开始运行时核对 CPU 单元中登录的用户程序执行用 ID 以及与用户程序关联的用户程序执行用 ID。

核对条件

核对条件如下所示。

A、 B 表示 ID。

CPU 单元中登录的用户程序执行用 ID	与用户程序关联的用户程序执行用 ID	异常	运行
A	A	无	可以
无	无		
无	A	ID 核对错误	不可
A	无		
A	B		

核对不一致时的动作

核对不一致时，将发生 “ID 核对错误 ”（全部停止故障等级的控制器异常）， CPU 单元无法运行。要解除异常，需要将控制器的电源从 OFF 设为 ON，或通过 Sysmac Studio 重置控制器。

● **用户程序执行用 ID 中可使用的文字规格**

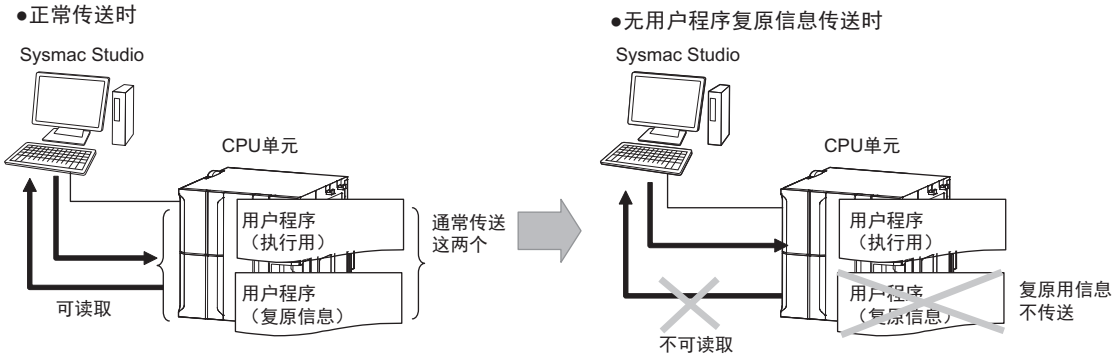
可使用的字符	大小写的区分	最大大小 （不含 NULL）
0 ～ 9、A ～ Z、 a ～ z	是	8 ～ 32 个字符

8-4-2 无用户程序复原信息传送功能

不向 CPU 单元传送程序的复原信息 （源代码），用其他计算机上传时，将不会显示程序。

概要

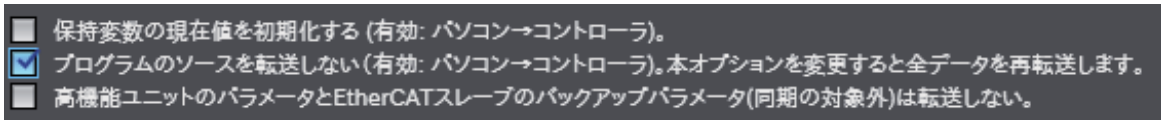
通常，通过 Sysmac Studio 向 CPU 单元传送用户程序时，会同时传送复原所需的信息。
利用本功能，可以不传送复原所需的信息，禁止读取用户程序。



用户无需在现场修改用户程序且需要防止用户程序数据被盗用的情况。

操作方法

向 CPU 单元传送用户程序时，在 Sysmac Studio 的同步窗口，勾选 “ 不传送程序的源代码 ” 设定，单击 “ 传送 ［计算机 | 控制器］ ” 按钮。

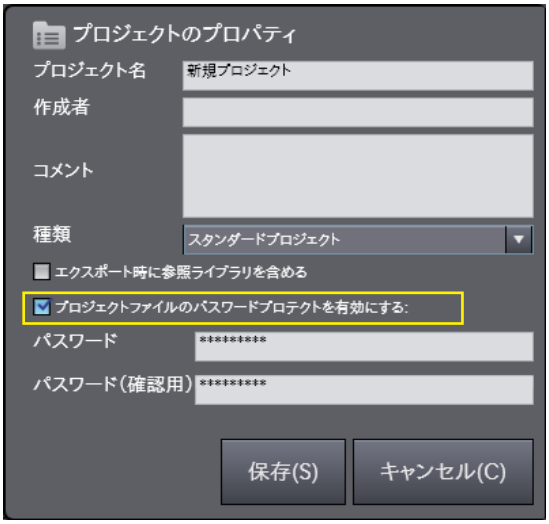


8-4-3 项目文件整体的保护功能

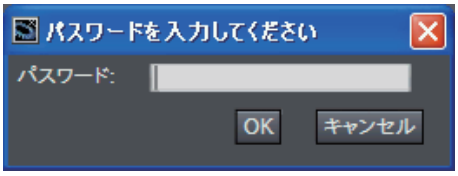
对项目文件设置密码保护，保护用户资产信息。

操作方法

下面介绍为项目设定密码时的操作。
另存项目文件时，如果勾选 “ 开启项目文件的密码保护 ”，则可以设定密码。



下面介绍打开设定了密码的项目时的操作。
打开或导入设定了密码的项目文件时，将显示 [输入密码] 对话框。



本功能的操作详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

8-4-4 数据保护功能

对项目数据中需要保护的数据进行保护，以保护用户资产信息。

版本相关信息

数据保护功能可在 CPU 单元版本 1.01 以上和 Sysmac Studio Ver.1.02 以上的组合中使用。

概要

以项目数据中需要保护的数据为单位设定保护，以限制对数据的访问。
设定保护时，可从 3 种强度的访问限制中选择并设定。
访问设有访问限制的数据时，需要临时解除保护。临时解除的有效期限因执行的操作不同而异。

保护对象数据

可进行保护设定的数据如下所示。

- 梯形图程序 （程序 / 函数 / 功能块共通）。
- ST 程序 （程序 / 函数 / 功能块共通）。
- 凸轮轮廓

访问限制的种类

访问限制的强度可从以下 3 种中选择。但是，凸轮轮廓只能选择禁止变更。

- 复制、显示、禁止变更
- 显示、禁止变更
- 禁止变更

对数据的访问操作以及不同访问限制种类的限制和临时解除保护时的有效期限关系如下。

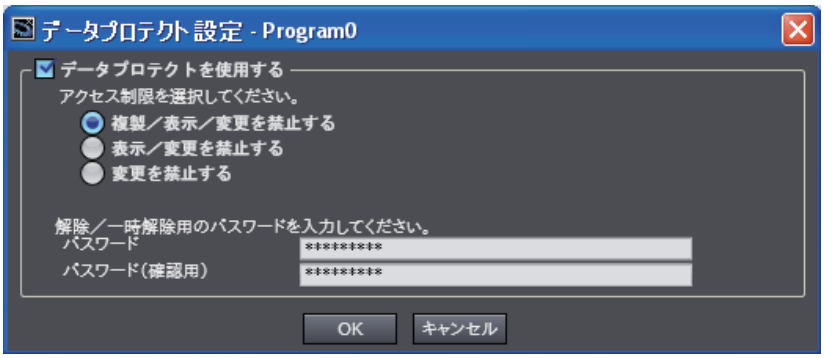
（○：可限制操作、－：不可限制操作）

数据访问的操作	访问限制的种类			临时解除有效期限
	复制、显示、禁止变更	显示、禁止变更	禁止变更	
对象物的显示	○	○	－	已打开项目时
对象物的打印	○	○	－	
对象物的变更	○	○	○	
对象物的复制	○	－	－	每次操作都需要临时解除
核对结果 （概要）的显示	－	－	－	不限制本操作
核对结果 （详情）的显示	○	○	－	已打开项目时
从事件日志或交叉参考跳转到内容	○	○	－	
登录到库	○	－	－	不能临时解除。

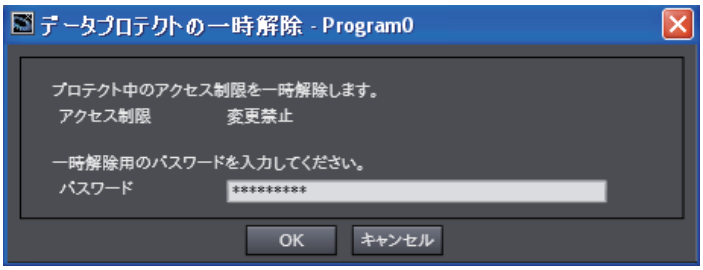
操作方法

本功能可离线操作 Sysmac Studio。
设定内容将保存到项目中。如果用 Sysmac Studio 的同步功能传送项目，传送方 （计算机 / 控制器）对象数据的数据保护设定将传送到接收方 （控制器 / 计算机）。

保护设定请在 Sysmac Studio 的 ［控制器］ | ［安全］ | ［数据保护设定 / 解除］中设定。



要临时解除保护，请选择 ［控制器］ | ［安全］ | ［禁止变更的临时解除］。



要关闭临时解除保护，请选择 ［控制器］ | ［安全］ | ［关闭禁止变更的临时解除］。

本功能的操作详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

8-4-5 操作权限的认证功能

概要

因操作失误可能对装置或人施加危害，根据操作权限限制在线操作功能的功能。
例如以下情况。

- I/O 监视器　　：写入、强制值刷新等
- 控制器操作　　：动作模式变更、在线编辑、轴调试等

通过 Sysmac Studio 在 CPU 单元中登录操作权限的密码，在线连接时输入操作权限的密码后，与操作权限类别对应的在线操作功能即变为可用。

管理员应设定不同操作权限的密码，根据技能水平将操作权限名称和密码告知使用者。

操作权限相关设定的操作详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

操作方法

操作权限的认证功能可选择 Sysmac Studio 的［控制器］｜［安全］｜［操作权限的设定］后执行。
操作方法的详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

规格

● 操作权限的种类

按权限从大到小的顺序， Sysmac Studio 的操作权限种类有以下 5 种。

中文名称	英文名称	密码设定
管理员	Administrator	必须
设计者*1	Planning Engineer	任意*2 设定省略 ［操作权限认证设定］对话框的密码输入时，是否需要密码。
维护者	Maintenance	
操作者 *1	Operator	
观察者 *1	Observer	不需要

*1. 可在 CPU 单元 版本 1.01 以上和 Sysmac Studio Ver.1.02 以上的组合中使用。
*2. 设定省略 ［操作权限认证设定］对话框的密码输入时，是否需要密码。如果是比省略时的操作权限更高级的权限，则需要密码。相同或更低级的权限则不需要密码。

● 对象在线操作功能示例

不同在线操作种类的操作权限示例如下。

详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

（○ : 可操作、△ : 每次操作时需要认证、×: 不可操作）

状态监视功能 （示例）	管理员	设计者	维护者	操作者	观察者
异常监视 （故障排除）	○	○	○	○	○

I/O 监视功能 （示例）	管理员	设计者	维护者	操作者	观察者
I/O 监视 （读取）	○	○	○	○	×
I/O 监视 （写入）	○	○	○	△	×
设置 / 复位	○	○	○	△	×
强制值刷新	○	○	○	×	×

控制器操作功能 （示例）	管理员	设计者	维护者	操作者	观察者
运行模式 / 程序模式	○	○	△	×	×
在线编辑	○	○	△	×	×
控制器重置	○	○	×	×	×
异常解除 （故障排除）	○	○	○	△	×
轴调试、轴调试重新开始	○	○	△	×	×
用户程序执行用 ID （控制器）	○	×	×	×	×
CPU 单元的写入保护	○	○	○	×	×

● 密码规格

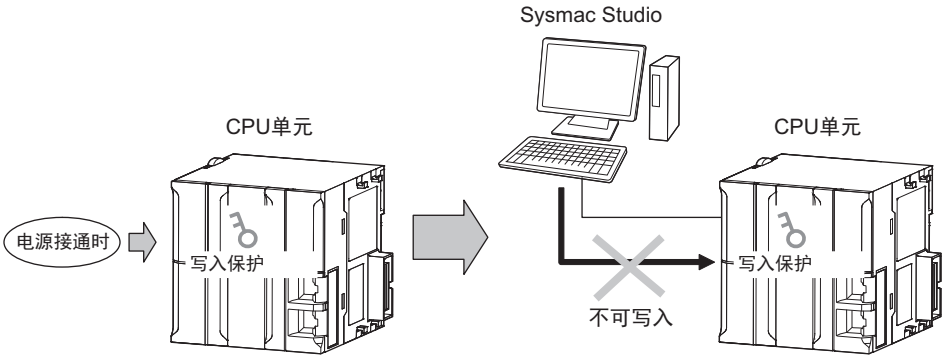
项目	内容
有效字符数	8 个字符以上且 32 个字符以内
可使用的字符	半角英文和数字 （区分大小写）

8-4-6 CPU 单元的写入保护功能

为保护用户程序资产和防止误操作，使 CPU 单元内各种数据的写入变为无效的功能。包括以下 2 种。

● CPU 单元电源接通时的写入保护设定

接通 CPU 单元的电源时，自动启用写入保护的功能。

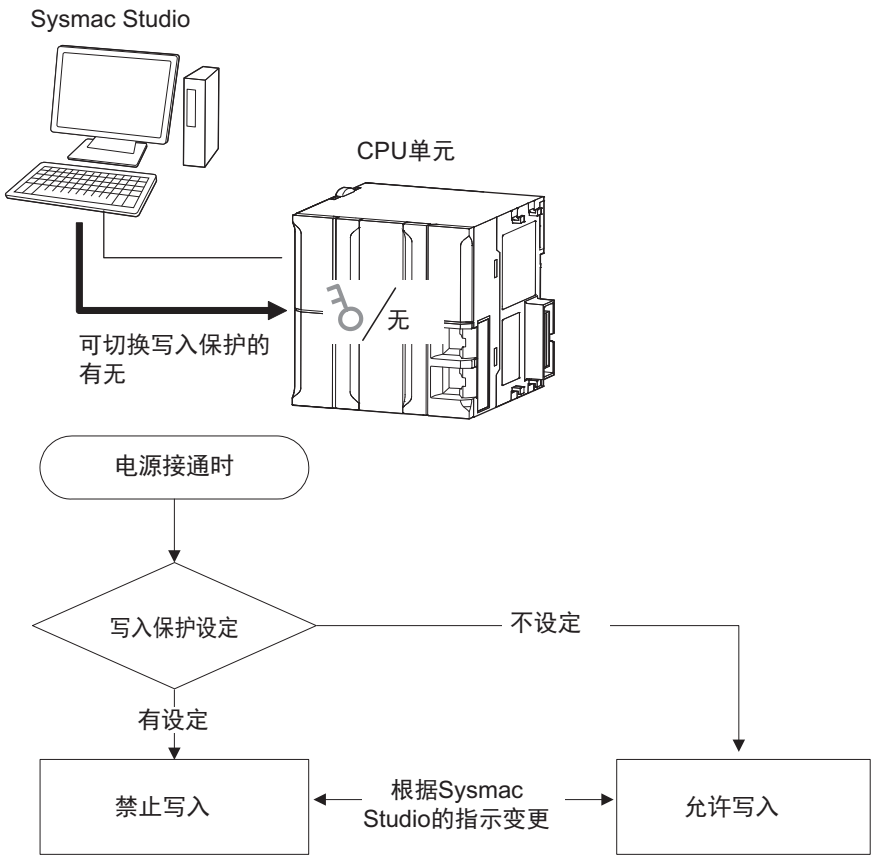


Sysmac Studio 上的操作方法请在 Sysmac Studio 的 “ 配置和设定 ” | “ 控制器设定 ” | [动作设定] 中，设定在接通电源时是否自动进行写入保护。

设定 大项目	设定 中项目	设定 小项目	功能说明	设定值
动作设定－ 动作设定标签－ 整体设定	安全设定	电源接通时的 写入保护	设定启用 / 禁用写入保护。	不保护 保护

● 通过 Sysmac Studio 进行写入保护的 “ 设置 ” / “ 解除 ” 切换

通过 Sysmac Studio 进行在线连接，在 ［控制器］菜单的 ［安全］－ ［CPU 单元的写入保护］中，切换 “ 设置 ” / “ 解除 ”。



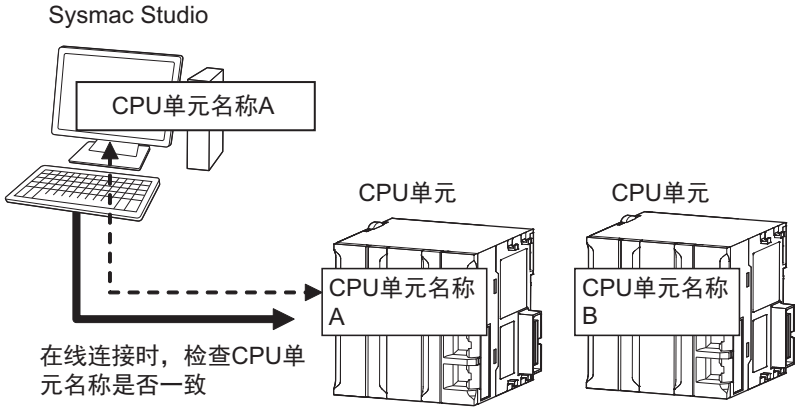
8-4-7 CPU 单元名称功能及序列 ID 功能

概要

请在 CPU 单元中登录 CPU 单元名称。

通过 Sysmac Studio 进行在线连接时，可确认项目中的 CPU 单元名称与连接对象 CPU 单元的 CPU 名称是否一致。

由此，可防止 Sysmac Studio 与错误的 CPU 单元连接。特别是通过 EtherNet/IP 操作时非常有效。



此外，除了 CPU 单元名称之外，还可识别与 CPU 单元的生产信息对应的序列 ID （选项）。

设定方法

1 **在 Sysmac Studio 上创建项目时，设定 CPU 单元名称。**

CPU 单元名称显示如下。

要变更名称时，可右击控制器图标，在 “ 重命名 ” 中编辑。

2 **第一次与 CPU 单元在线连接时，Sysmac Studio 会提醒是否在 CPU 单元中保存 CPU 单元名称。**

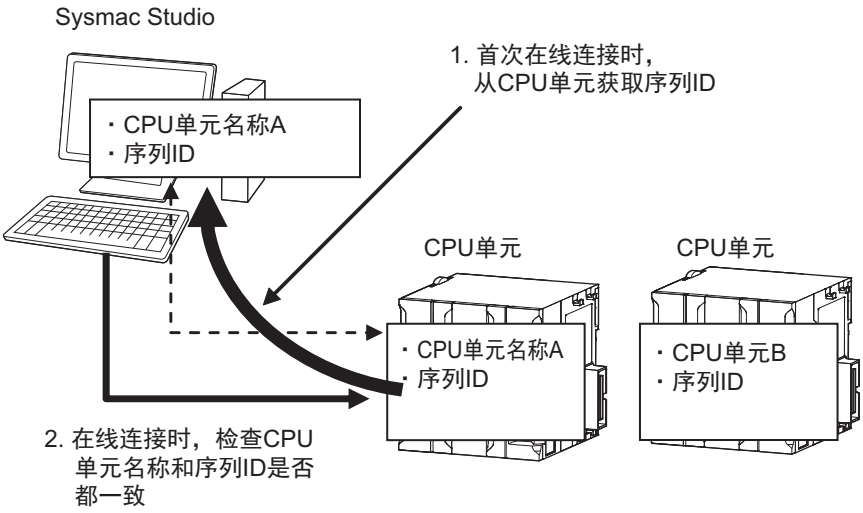
3 **然后，与CPU单元在线连接时，Sysmac Studio会核对项目中的CPU单元名称和要连接的CPU单元名称。**
如果不一致，将显示警告对话框，并提醒是否继续连接。

参考

NetworkConfigurator 中，可对 EtherNet/IP 端口进行命名。

序列 ID 功能

Sysmac Studio 第一次在线连接时，可获取与 CPU 单元的生产信息对应的序列 ID，并保存到项目中。然后， Sysmac Studio 进行在线连接时，将核对 CPU 单元名称和序列 ID。由此，可严密地进行 CPU 单元核对。



8-5 调试功能

下面介绍调试功能。

NJ/NX 系列控制器中有以下调试功能。
- 强制值刷新
- 当前值变更
- 在线编辑
- 数据追踪
- 微分监控

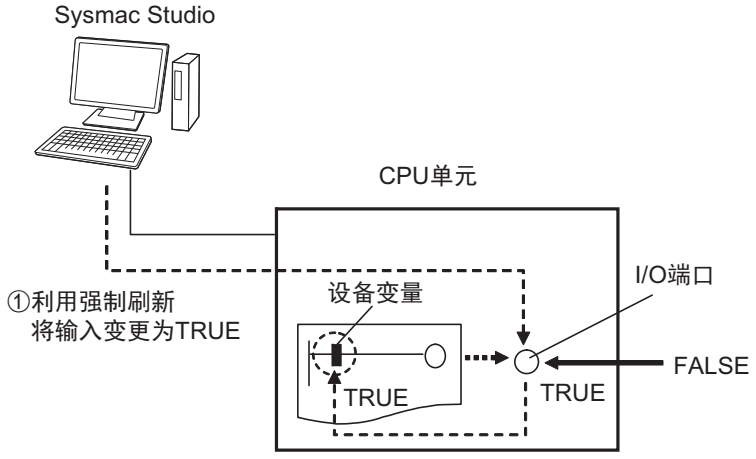
8-5-1 强制值刷新

概要

强制值刷新是指用户进行调试等时，以 Sysmac Studio 中指定的强制值刷新来自外部的输入或向外部的输出。
强制值刷新将针对设备变量中分配的 I/O 端口执行，而不是指定的设备变量。
强制值刷新中指定的状态在通过 Sysmac Studio 发出强制状态解除指示前，将一直保持。（关于对 CPU 单元状态变化的保持 / 解除，请参考后述的 “ 强制值刷新的保持 / 解除 ”。）
此外，异常停止时、存储器全部清除时、模式变更时、断电时、下载时将不保存，直接清除。

● **输入**

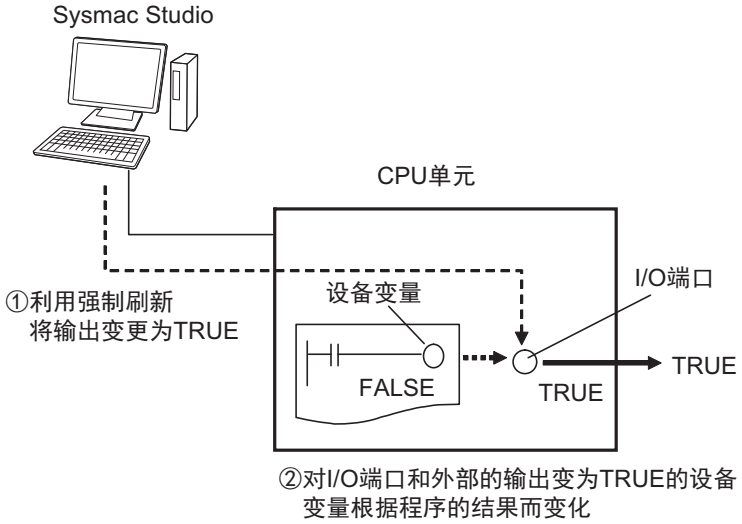
无论来自外部的输入如何， I/O 端口和设备变量将变为强制值刷新中指定的状态。



● **输出**

I/O 端口和向外部的输出将变为强制值刷新中指定的状态。

用户程序中的、 I/O 端口上分配的设备变量的状态将根据用户程序的结果而变化，而不是强制值刷新中指定的状态。



对象区域

强制值刷新可对以下 I/O 端口或 CJ 单元用存储器执行。

- EtherCAT 从站的 I/O 端口
- CJ 系列基本 I/O 单元的 I/O 端口
- CJ 系列高功能单元的 I/O 端口
- 通过变量 AT （分配目标）指定的 DeviceNet 从站或 CompoNet 从站的输入输出继电器

通过梯形编辑器或监视窗口执行强制值刷新时，利用变量变更 I/O 端口或 CJ 用单元存储器的状态。

（注）可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

可同时进行强制值刷新的个数

可进行强制值刷新的个数如下。

- CJ 系列单元：共 64 点
- EtherCAT 从站：共 64 点

上述个数为外部输入输出接点的数量。例如，多个变量的分配目标指定为 1 个外部输入输出接点时，计为 1 个。

用途

● **输入**

- 发出虚拟输入信号，进行用户程序的调试
- 创造出只有故障时才可能出现的状态 （同时控制排他 2 接点的 ON/OFF 等）

● **输出**

- 为检查配线，进行输出的 ON/OFF
- 无论用户程序的结果如何，都将强制关闭不需要的输出。

操作方法

可在以下画面中操作。

- 程序画面 （梯形图语言）
- I/O 映射
- 监视窗口

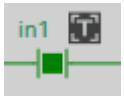
● 梯形编辑器中强制值刷新的操作步骤

1 **在 [控制器] 的 [监视器] 中打开监视器。**

2 **在多视图浏览器的 ［编程］中，双击对象梯形图程序或梯形图功能块 / 函数。**
在梯形图上，回路将以监视状态显示。

3 **选择强制值刷新时打开的接点和线圈，右击选择｜［强制值刷新］｜［TRUE］。或者，选择强制值刷新时要关闭的接点和线圈，右击选择｜［强制值刷新］｜［FALSE］。**

4 **梯形编辑器上的对象接点和线圈将强制变为 On 或 Off 状态，On 时为导通状态，Off 时为非导通状态。**
接点、线圈中将显示如下强制状态的标志。



梯形图

强制状态的 “TRUE”“FALSE” 显示为强制值刷新中指定的状态，并不是显示接点或线圈的当前值。

强制状态标志	操作
T	指定了强制值刷新 （TRUE）时
F	指定了强制值刷新 （FALSE）时

参考

如果有变量指定了与强制值刷新变量的 AT （分配目标）指定位置相同的存储器地址，该变量中也会显示强直状态标志。

动作模式与电源断开的关系

● 可进行强制值刷新的动作模式

可在程序模式、运行模式的任意模式下执行。
但是，发生全部停止故障等级的控制器异常时，将无法强制。

● 动作模式变更时或电源断开时的强制值刷新状态

从运行模式变更为程序模式时或电源断开时，将解除强制值刷新状态。

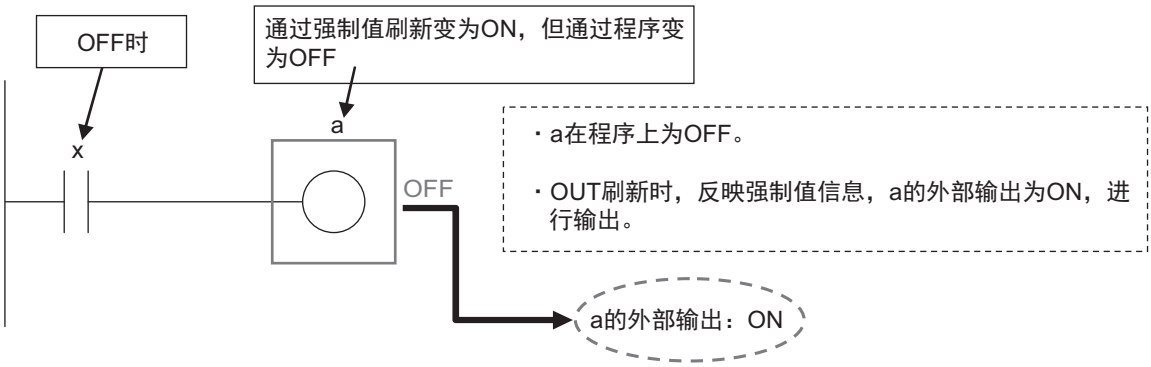
强制值刷新的保持 / 解除

与 CPU 单元的状态变化对应的强制值刷新的保持 / 解除如下。

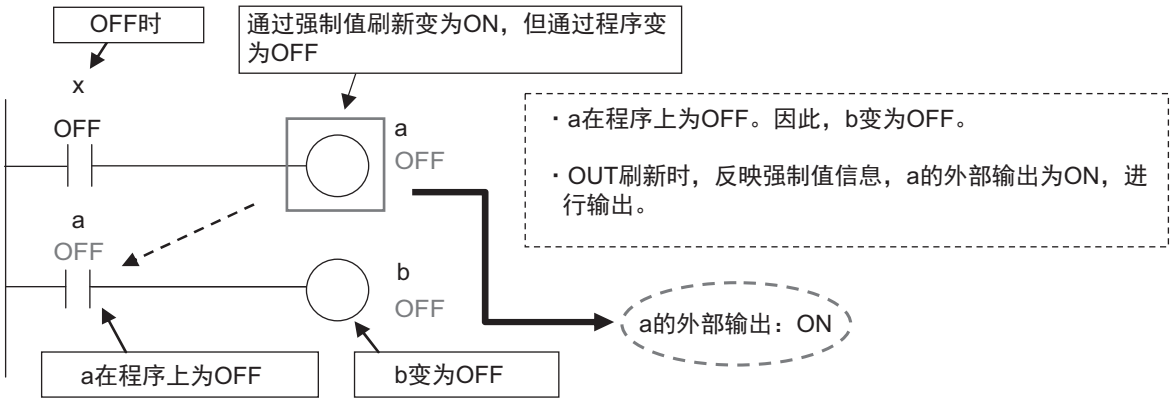
状态变化		强制值刷新状态
电源接通时		解除
动作模式变更时	运行模式→程序模式 程序模式→运行模式	解除
下载后		解除
发生全部停止故障等级的控制器异常时		解除
在线编辑时		保持

在程序中使用强制值时的注意事项

在用户程序上执行强制值刷新时，设定为强制值刷新对象的变量将被用户程序覆盖。因此，不保持强制值刷新中指定的状态。
但是，对外部的刷新与用户程序上的变量状态无关，继续以强制值刷新中指定的值输出。
在程序上使用强制值时，程序上的变量和外部输出的状态有差异，敬请注意。
例）通过强制值刷新将 a 设为 ON 时



然后，接收该接点时：



安全要点

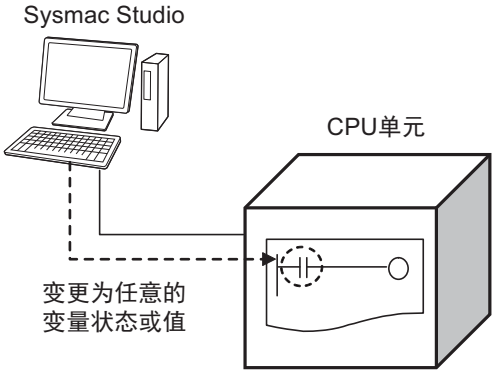
- 进行强制值刷新操作前，请确认其不会对设备造成不良影响。
- 强制值刷新是指与用户程序执行结果无关，始终以设定的值覆盖并执行 I/O 刷新的功能。
  因此， I/O 刷新对象以外的接点会因本功能的影响先变为设定的值，再根据用户程序替换。

因强制状态不同，控制对象可能发生意外动作。

8-5-2 当前值变更

概要

当前值变更是指为了确认程序或设定是否正确动作，可通过 Sysmac Studio 变更程序或设定中使用的变量值或控制接点、线圈的 ON/OFF （TRUE/FALSE）。



安全要点

请在确认不会对设备造成影响后，再变更当前值。

用途

- **任意变更接点或线圈的 ON/OFF （TRUE/FALSE）**

  控制任意 BOOL 型变量状态的 ON/OFF。该状态在由用户程序覆盖后，变为执行状态，通过模式变更和电源重启，恢复为初始值状态。
  可通过梯形编辑器、监视窗口、 I/O 映射进行。

- **变更为任意的变量值**

  将用户定义变量、系统变量、设备变量的当前值变更为任意值并反映。可通过监视窗口进行。

安全要点

请在确认不会对设备造成影响后，再变更当前值。

操作方法

当前值变更可在以下画面中操作。各画面的操作方法详情请参考📖《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》。

- 程序画面 （梯形图语言、 ST 语言）
- I/O 映射
- 监视窗口

通过当前值变更，变更向外部的输出状态时的注意事项

通过当前值变更，变更输出到 CJ 系列基本输出单元或 EtherCAT 输出从站的 I/O 端口的状态时，请注意以下事项。

● 运动模式时，通过 I/O 映射变更当前值

即使通过 I/O 映射变更 I/O 端口的值，也会被用户程序的执行结果覆盖。
因此，不会将当前值变更中变更的值输出到外部。
要将 I/O 端口的值变更为任意值，然后输出到外部时，请通过强制值刷新执行。

● 程序模式时，通过监视窗口变更当前值

即使通过监视窗口变更设备变量 （*）（外部变量、本地变量）的值，也不会输出到外部。
要将任意值输出到外部时，请进行以下操作。

- 进行强制值刷新
- 通过监视窗口，对设备变量 （*）（本地变量）进行当前值变更
  * CJ 系列基本输出单元或 EtherCAT 输出从站的 I/O 端口上的设备变量。此外，分配有 CJ 系列基本输出单元的输出继电器中 AT 指定的全局变量也相同。

● 针对与CJ系列基本输出单元的输出继电器相当的I/O存储器地址，直接从外部写入时的注意事项

通过标签数据链接，直接访问与分配有 CJ 系列基本输出单元的输出继电器相当的 I/O 存储器地址，并变更 I/O 存储器地址的状态，变更的值仍将被用户程序的执行结果覆盖。
因此，不可通过标签数据链接，直接将 I/O 存储器地址中写入的值输出到外部。

（注）可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

8-5-3 在线编辑

下面介绍在线编辑的概要。详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

概要

在线编辑是指通过 Sysmac Studio，直接变更和追加 CPU 单元中部分用户程序的功能。

可选择以下内容，进行在线编辑。

- POU （程序、 FB、 FUN）
  如果是以梯形图语言记述的程序，则选择段落。
- 全局变量

用途

想要不停止 CPU 单元的运行，直接变更用户程序时使用。

Sysmac Studio 上的操作

● 在线编辑的操作

1 **选择在线编辑对象。**

2 **从［项目］菜单中选择［在线编辑］｜［开始］。**

3 **进行变更。**

4 **从［项目］菜单中选择［在线编辑］｜［传送变更］。**

5 **进行反映确认。**

6 **以在线编辑后的用户程序开始动作。**

⚠ 注 意

请确认输入输出时间打乱后也不会受到影响，再执行在线编辑。执行在线编辑后，任务执行时间可能超过任务周期，与外部之间的输入输出无法更新，导致输入信号无法读取、输出时间紊乱。

使用注意事项

- 在线编辑中编辑的回路所含的微分指令的微分状态将初始化。
- 反映在线编辑内容时，任务的执行时间会延长。设定任务周期时，请确保不会因在线编辑而发生 “ 任务周期超限 ”。
- 正在反映在线编辑 (*) 时，如果 CPU 单元发生断电，断电恢复时，将发生全部停止故障等级的控制器异常 （“ 用户程序 / 控制器配置和设定时传送错误 ”、“ 用户程序 / 控制器配置和设定不正确 ”、“ 有非易失性存储器修复 ” 中的一个）。此时，请重新下载用户程序。
- 正在反映在线编辑 (*) 时，请勿执行 MC_SaveCamTable（凸轮表保存）指令。若执行，可能无法正确反映在线编辑的变更。

* 是指在反映确认窗口中选择 [是] 后，关闭 “ 反映确认窗口 ” 之前的时间。但是，在 CPU 单元版本 1.04 以上和 Sysmac Studio Ver.1.05 以上的组合中，是指关闭 “ 反映确认窗口 ” 后面显示的 “ 内置非易失性存储器备份中窗口 ” 为止的时间。

8-5-4 数据追踪

无需程序，对指定变量进行采样的功能。可通过 Sysmac Studio 读取，然后确认并保存文件。可用于装置启动、运用 / 维护。

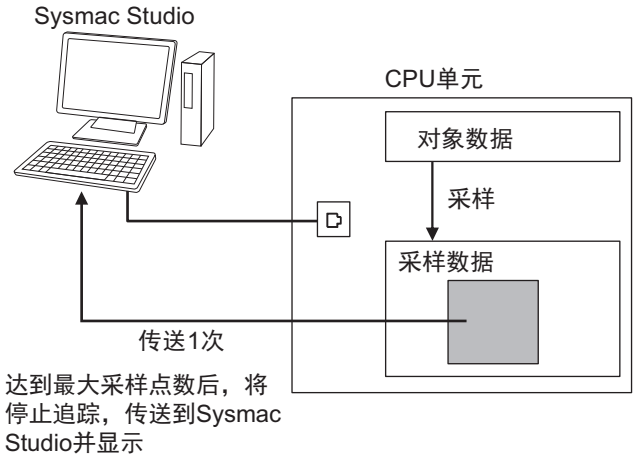
下面介绍数据追踪的概要。操作方法请参考📖《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》。

追踪方法分为以下 2 种。

● 触发跟踪

通过设定触发条件，记录问题发生前后的数据。达到最大采样点数后，自动停止采样。未与 Sysmac Studio 连接时，也可在触发条件成立时进行数据追踪，与 Sysmac Studio 连接后再上传。

- 可根据变量的当前值变更情况确认程序的流程。
- 用于调查变量发生意外变更的原因。

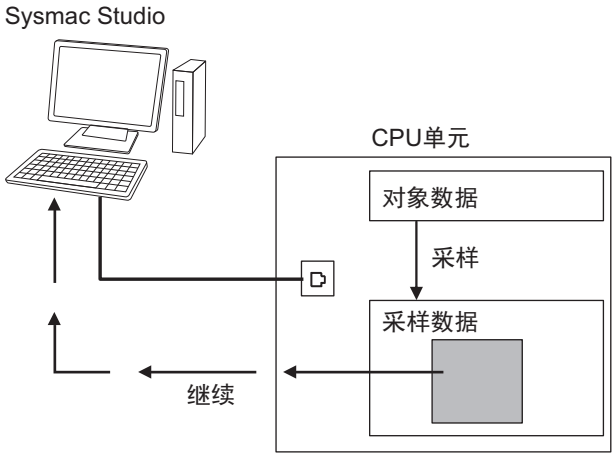


● 连续跟踪

在没有触发的情况下开始采样，超出 10,000 点后继续采样。采样数据将依次传送到计算机上，然后保存到文件中。

达到显示缓存大小后，将自动保存为 CSV 文件。

因此，从开始追踪到停止为止，可将长期的追踪结果数据保存为多个 CSV 文件。



数据追踪的规格

数据追踪的规格如下所示。

项目		内容
数据追踪的种类	触发跟踪	设定开始采样的触发条件，记录条件成立前后的数据。
	连续跟踪	采样数据将依次传送到计算机上，然后保存到文件中。
采样时间	指定任务的周期	指定任务，将该任务的周期作为采样周期。
	固定时间间隔	将输入的时间作为采样周期。但是，时间将取整至主周期任务的整数倍。
	追踪采样指令	在用户程序中执行 TraceSamp 指令时，执行采样。
采样对象设定*1	最大对象数	NX701- □□□□：192 个变量 NJ501- □□□□：192 个变量 NJ301- □□□□：48 个变量 NJ101- □□□□：48 个变量
	数据类型	基本数据型 （字符串除外） 排列 （指定要素） 列举型 结构体型 / 联合体型 （指定成员）
记录最大数		每个变量 10,000 点
触发位置设定		设定所有追踪时间 （数量）中的触发位置。
触发设定	条件的数据类型	基本数据型 （时刻、时间长度、日期、字符串除外） 排列 （指定要素）、结构体 / 联合体 （指定成员）
	条件表达式*2	当以下条件表达式成立时开始追踪。 BOOL：上升沿、下降沿 BOOL 以外：＝、＞、≧、＜、≦、≠*3
	Sysmac Studio 的指示	单击触发 ON 按钮后，开始追踪。
	数据追踪触发条件成立指令	执行 TraceTrig 指令后，开始追踪。
	判定时序	• 采样时间设定为追踪采样指令以外时，在指定的任务周期内执行 1 次触发判定。 • 采样时间设定为追踪采样指令时，在执行指令时执行触发判定。
	延时值	用滑块设定触发成立前 / 成立后的采样数比例。（例：20%/80%）
追踪开始	Sysmac Studio 的指示	按下 Sysmac Studio 的 “ 执行按钮 ” 后开始。
	运行时开始设定	开始运行 （程序模式➜运行模式）时自动开始追踪。
追踪停止	触发跟踪	• 达到采样上限值 10,000 点后停止。 • 按下 Sysmac Studio 的 “ 停止按钮 ” 后停止。
	连续跟踪	• 达到上限时的设定为 “ 追踪停止 ” 时，达到上限大小或上限期间后，则停止。 • 按下 Sysmac Studio 的 “ 停止按钮 ” 后停止。

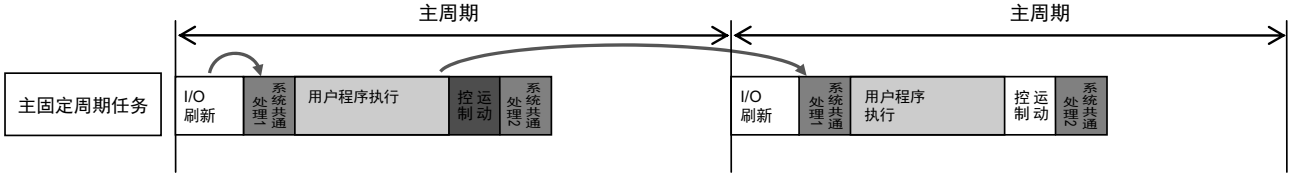
项目		内容
连续跟踪专用设定	数据保存期限上限	设定连续跟踪数据的保存期限上限。
	数据保存大小上限	设定连续跟踪中保存的文件总大小上限。
	每个文件的数据数	设定在连续跟踪时，一个文件中保存的数据数。
	文件保存位置	指定在连续跟踪时，创建数据保存文件的位置。
	文件名前缀	指定自动命名时文件名开头的名称。
	达到上限时的设定	指定在达到保存期限上限或保存大小上限时的动作。 • 追踪停止 • 删除旧文件并继续
追踪结果显示	图表显示	显示的图表为 X 轴：时间、 Y 轴：变量值。可在一个图表中显示 BOOL 型和其他变量。
	表显示	以表格的形式显示各变量的最大值、最小值、平均值及指定时间的值。
	3D 运动监视显示	在 3D 空间内设置虚拟结构模型，根据运动轴的指令位置及当前位置虚拟显示结构的动作。
追踪数据导出	CSV 文件导出	将追踪结果及追踪编号以外的设定保存为 CSV 格式文件。
可同时启动的数据追踪*4		NX701- □□□□：4 个 NJ501- □□□□：4 个 NJ301- □□□□：2 个 NJ101- □□□□：2 个
追踪数据导入		在当前显示的图标中，重叠显示保存为 CSV 格式的追踪结果。
保存		将追踪结果与追踪设定一起，保存到项目中。
打印		打印图表。使用 Sysmac Studio 的打印功能。

*1. 系统定义变量 EN、ENO、P_off、P_on、P_CY、P_First_RunMode、P_First_Run、P_PRGER 及功能块的实例中使用的输入输出变量、函数中的变量不在采样对象范围内。
*2. 开始数据追踪时，即使触发的条件表达式为成立状态，也不会开始数据追踪。
*3. 不可组合多个条件表达式。此外，比较常数的有效范围遵照条件式左边的变量类型文字表现的有效范围。
*4. NX701 及 NJ501 时，追踪编号设定为 0 ～ 3，NJ301 及 NJ101 时为 0 ～ 1。使用该编号，执行指令或访问系统定义变量。

数据追踪的动作

数据追踪处理 （采样及触发判定）在 I/O 刷新处理后、执行用户程序前的 “ 系统共通处理 1” 中执行。

例）采样时间指定为 “ 主固定周期任务 ” 时，按下图 “ 系统共通处理 1” 的时序执行数据追踪。



参考

任务处理中，“I/O 刷新、用户程序、运动控制 ” 为同一任务周期，但在数据追踪的显示中，“ 用户程序、运动控制、下一主周期的 I/O 刷新 ” 作为同一任务周期显示。《NJ/NX 系列 指令基准手册 运动篇 （SBCE-CN5-364）》中记载的时序图以任务周期为基准记载，因此与数据追踪的显示不同。

以下为按指定任务周期采样时，数据追踪的动作及执行结果的显示示例。

例 1）

创建了以下程序时：传感器的输入信号中分配的变量 “Sensor1” 变为 TRUE 后，将变量 “SysRun” 变为 TRUE 的程序

数据追踪的动作及执行结果的显示如下所示。

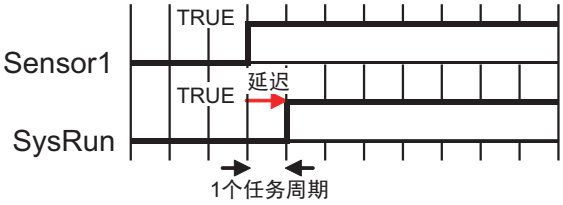
(1)在系统共通处理 1 的数据追踪处理中获取 “Sensor1” 的 TRUE

(2)在用户程序中，将 “SysRun” 设为 TRUE

(3)在下一主周期的系统共通处理 1 的数据追踪处理中获取 “SysRun” 的 TRUE

因此，在数据追踪的显示中，“SysRun” 会比 “Sensor1” 晚一个任务周期变为 TRUE。

数据追踪的显示

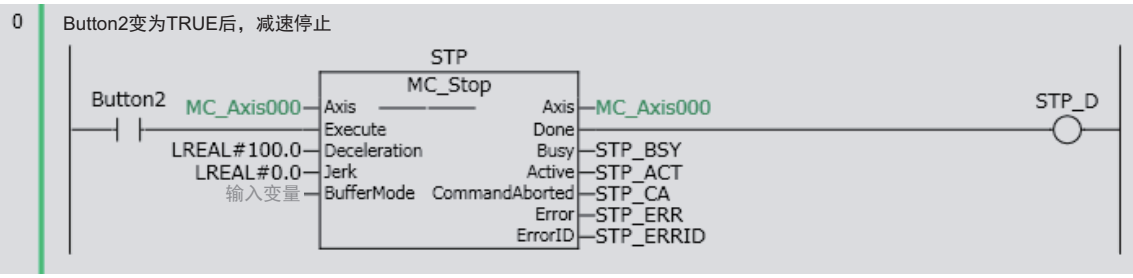


参考

在用户程序中变更了变量的值时，在数据追踪的显示中，值的变化和 I/O 刷新的输出处理在同一任务周期内变化。

例 2）

创建了以下程序时：以速度控制动作时，按钮的输入信号中分配的变量 “Button2” 变为 TRUE 后，使轴 0（“MC_Axis000”）减速停止的程序



数据追踪的动作及执行结果的显示如下所示。

(1)在系统共通处理 1 的数据追踪处理中获取 “Button2” 的 TRUE

(2)在用户程序中将 “STP_BSY” 设为 TRUE，运动控制功能模块进行减速处理

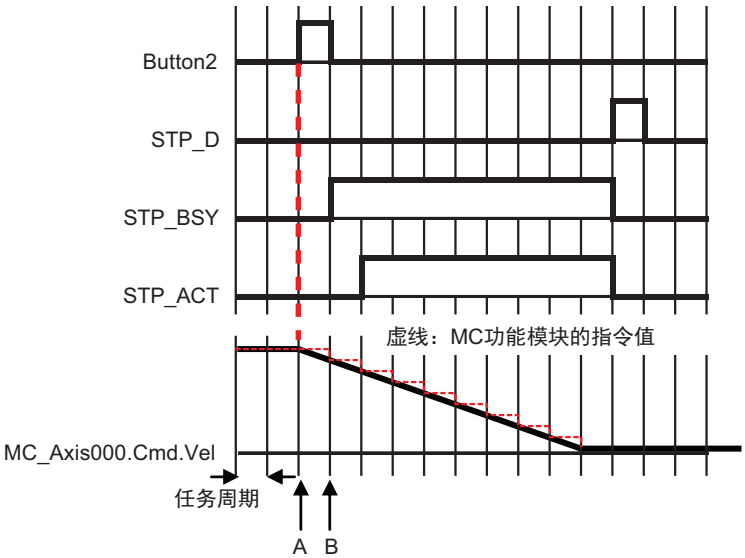
(3)在下一主周期的系统共通处理 1 的数据追踪处理中获取 STP_BSY 的 TRUE 和运动变量

(4)在用户程序中，将 “STP_ACT” 设为 TRUE

(5)在下一主周期的系统共通处理 1 的数据追踪处理中获取 STP_ACT 的 TRUE

运动控制功能模块的指令值在用户程序中的 “STP_BSY” 变为 TRUE，运动控制功能模块开始减速处理时，开始变化 （下图 B 的时间点）。指令值与主固定周期任务同步进行阶段性变化。

数据追踪将对前周期的值和下一周期的值进行直线插补。因此，运动变量的指令速度 “MC_Axis000.Cmd.Vel” 显示为在指令值发生变化前的一个周期，即从 “Button2” 变为 TRUE 开始 （下图 A 的时间点）进行减速。开始减速后一个周期，“STP_BSY”变为TRUE，再一个周期后，“STP_ACT”变为TRUE。



参考

在包含运动控制指令的功能块中，开始执行功能块时，输入参数的值将设置到输入变量中，功能块执行结束时，输出变量的值设置到输出参数中 （参考「 功能块 （FB）内变量的规格」（P.6-12））。数据追踪的显示与输入参数和输入变量、输出参数和输出变量在同一任务周期内变化。

相关系统定义变量

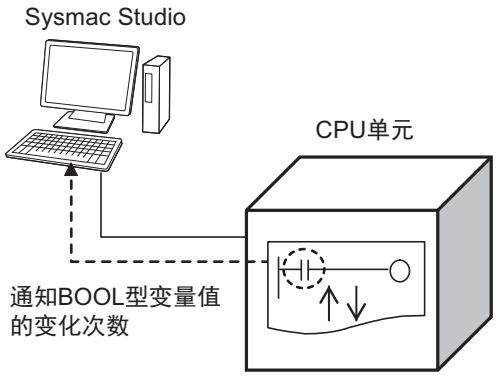
变量名称	成员名称	名称	功能	数据类型	R/W
_PLC_TraceSta[0..3][*1]				_sTRACE_STA	R
	.IsStart	追踪执行中 ON 标志	开始追踪时，变为 TRUE。	BOOL	R
	.IsComplete	追踪完成时 ON 标志	追踪执行完成时，变为 TRUE。 下次开始追踪时，变为 FALSE。	BOOL	R
	.IsTrigger	追踪触发监视标志	追踪触发条件成立时，变为 TRUE。 下次开始追踪时，变为 FALSE。	BOOL	R
	.ParamErr	追踪参数异常标志	开始追踪时，若追踪设定异常，则变为 TRUE。 正常时，为 FALSE。	BOOL	R

*1. 对应数据追踪 0 ～ 3。

（注）本系统定义变量在用户程序中不可使用。通过用户程序参照数据追踪的状态时，请使用 GetTraceStatus 指令。

8-5-5 微分监控

微分监控是指对指定 BOOL 型变量的值与指定条件一致的次数进行通知的功能。按主固定周期任务的任务周期 （称为主周期）执行条件一致判定，并通知目前为止的条件一致次数。



版本相关信息

微分监控可在 CPU 单元版本 1.03 以上和 Sysmac Studio Ver.1.04 以上的组合中使用。

用途

可确认来自外部的输入信号或用户网络中接点的 ON/OFF 有无和次数。在设备启动时以及调查运行过程中发生动作不良的原因时非常有用。

微分监控的规格

微分监控的规格如下所示。

项目		规格
微分监控条件	设定数量	最大 8
	指定变量	• BOOL 型 • BOOL 型排列的 1 个要素 • 结构体、联合体的 1 个 BOOL 型成员
	条件表达式	• 上升沿 (FALSE → TRUE） • 下降沿 (TRUE → FALSE)
条件一致判定	时序	主周期内一次
	次数计算	对各个指定变量分别计算条件一致次数
开始和停止	开始条件	Sysmac Studio 的指示
	停止条件	• Sysmac Studio 的指示 • 发生全部停止故障等级的控制器异常 • 下载用户程序 • 存储器全部清除 • 解除 Sysmac Studio 的在线连接
动作模式		• 运行模式 • 程序模式

微分监控条件

微分监控的设定对象变量或变化种类称为微分监控条件。下面介绍它们的规格。

● 设定数量

最大 8。即，可同时检测 8 个变量值的条件一致次数。

● 指定变量

可指定为微分监控对象的变量数据类型如下所示。

- BOOL 型
- BOOL 型排列的 1 个要素
- 结构体、联合体的 1 个 BOOL 型成员

不可指定为整个排列、整个结构体、整个联合体。

此外，作为变量的种类，可指定以下内容。

变量的种类		可否指定
系统定义变量		可*1
准用户定义变量		可
用户定义变量	全局变量	可
	程序中使用的变量	可
	功能块中使用的变量	可*2
	函数中使用的变量	不可

*1. 以下变量除外。
EN,ENO,P_Off,P_CY,P_First_RunMode,P_First_Run,P_PRGER

*2. 输入输出变量除外。

● 条件表达式

所检测变量的变化条件称为条件表达式。条件表达式有以下 2 种，选择其中之一。条件表达式可按变量指定。

- 上升沿 (FALSE ➞ TRUE)
- 下降沿 (TRUE ➞ FALSE)

使用注意事项

例如，条件表达式设定为 " 上升沿 " 时，开始微分监控功能时，即使指定变量的值为 TRUE，也不会判定为上升沿。变量的值先变为 FALSE，再变为 TRUE 时，判定为上升沿。

条件一致判定时间和次数计算

指定变量的条件一致判定按主周期执行。对上次判定时的变量值和本次判定时的变量值进行比较。所以，如果与指定条件表达式一致的变量值发生变化，将计算其次数。

- 条件一致计算次数按变量执行。
- 计数值会在开始微分监控时重置为 0。
- 不可单独将变量的计数值重置为 0。

使用注意事项

- 在主周期内，即使与条件表达式一致的值发生多次变化，一个主周期将只计算为一次。
- 如果上次判定时和本次判定时变量的值相同，即使在此期间值发生过变化，也不会视为与条件表达式一致。

开始条件和停止条件

要开始微分监控，需要通过 Sysmac Studio 发出指示。

要停止微分监控，通常由 Sysmac Studio 发出指示。但是，以下情况下，也会停止微分监控。

- 发生全部停止故障等级的控制器异常
- 下载用户程序
- 存储器全部清除
- 解除 Sysmac Studio 的在线连接

操作方法

微分监控的操作方法如下所示。详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362E 之后）》。

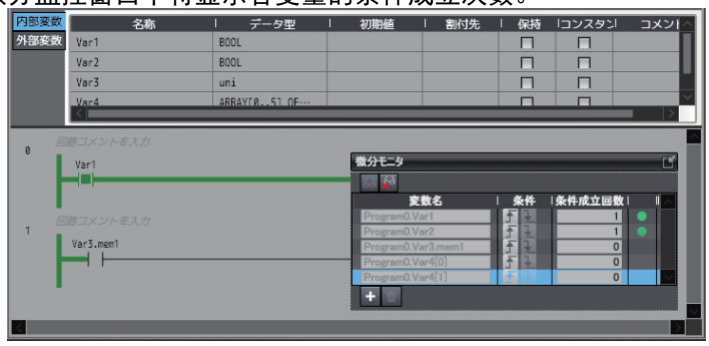
1 **选择 Sysmac Studio 的 [显示] ｜ [微分监控]。**

2 **选择可指定为微分监控对象的变量并右击，选择 [追加到微分监控]。**

3 **在微分监控窗口中，对各变量设定微分监控的条件表达式。**

4 **执行用户程序。**

微分监控窗口中将显示各变量的条件成立次数。



微分监控功能的注意事项

使用微分监控功能时，请注意以下事项。

● 执行微分监控的过程中与 Sysmac Studio 断开通信

执行微分监控的过程中，由于通信电缆断线、Sysmac Studio 异常结束等，与 Sysmac Studio 的通信中断。此时，CPU 单元将继续执行微分监控。重新执行微分监控时，请在重新开始与 Sysmac Studio 的通信后，先停止微分监控。

● 微分监控不可多个执行

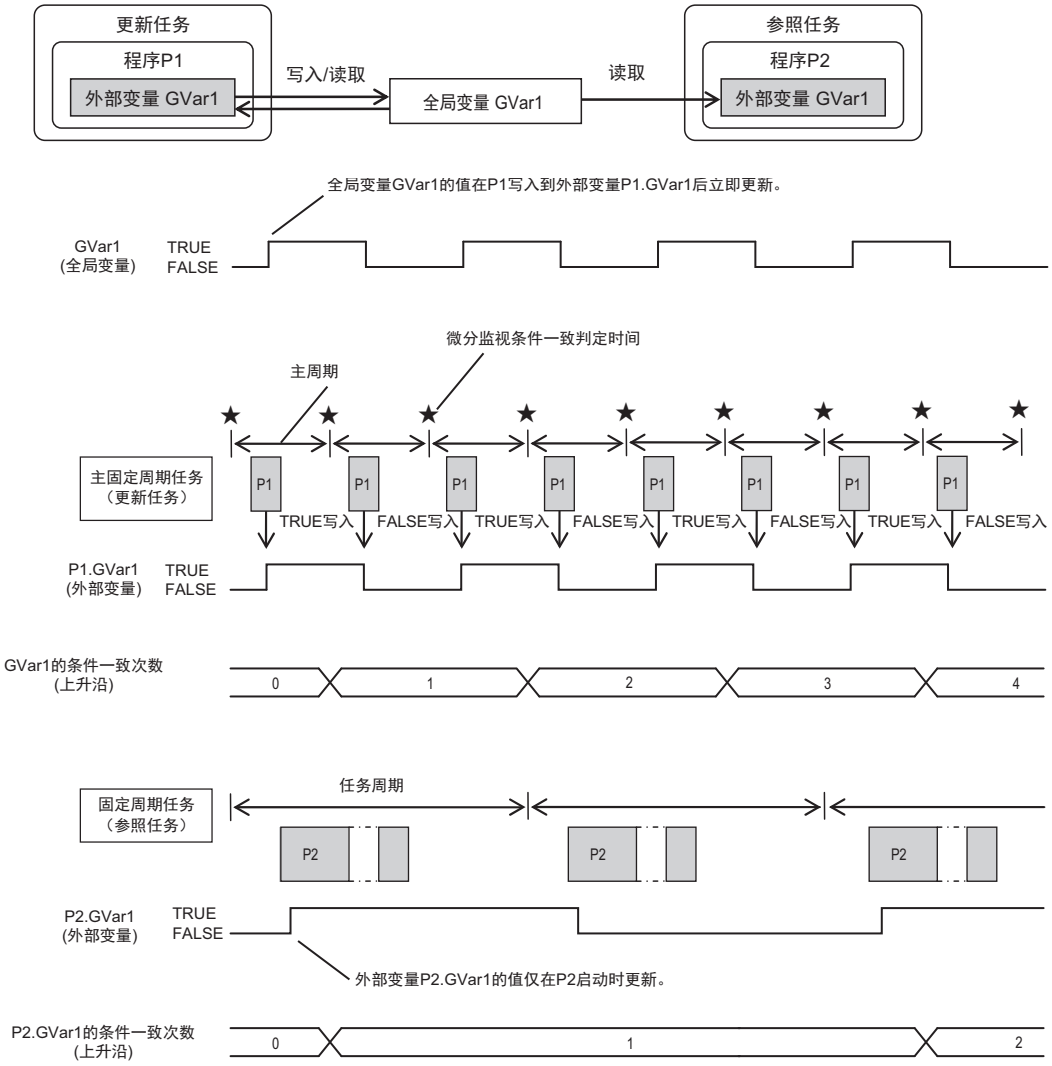
微分监控不可通过多 Sysmac Studio 同时执行。

● 全局变量、外部变量的指定

全局变量以及在 POU 内对其进行指定的外部变量均可设为微分监控的指定变量。但是，全局变量和外部变量的值更新时间不同，敬请注意。
全局变量会在写入值后立即更新。但外部变量只会在对该外部变量进行声明的 POU 启动时更新。
说明如下图。假设监视以下两个变量。

变量名称	变量的种类	要读写的 POU
GVar1	全局变量	分配到主固定周期任务中的程序 P1
P2.GVar1	程序 P2，指定为 GVar1 并进行声明的外部变量	分配到固定周期任务中的程序 P2

全局变量 GVar1 通过主固定周期任务中分配的程序 P1 读写，因此如果每个周期发生写入，将按主周期更新。而外部变量 P2.GVar1 只会在分配到固定周期任务中的程序 P2 启动时更新。即只会按固定周期任务的任务周期更新。固定周期任务的任务周期长于主周期，因此P2.GVar1的更新次数比GVar1少。

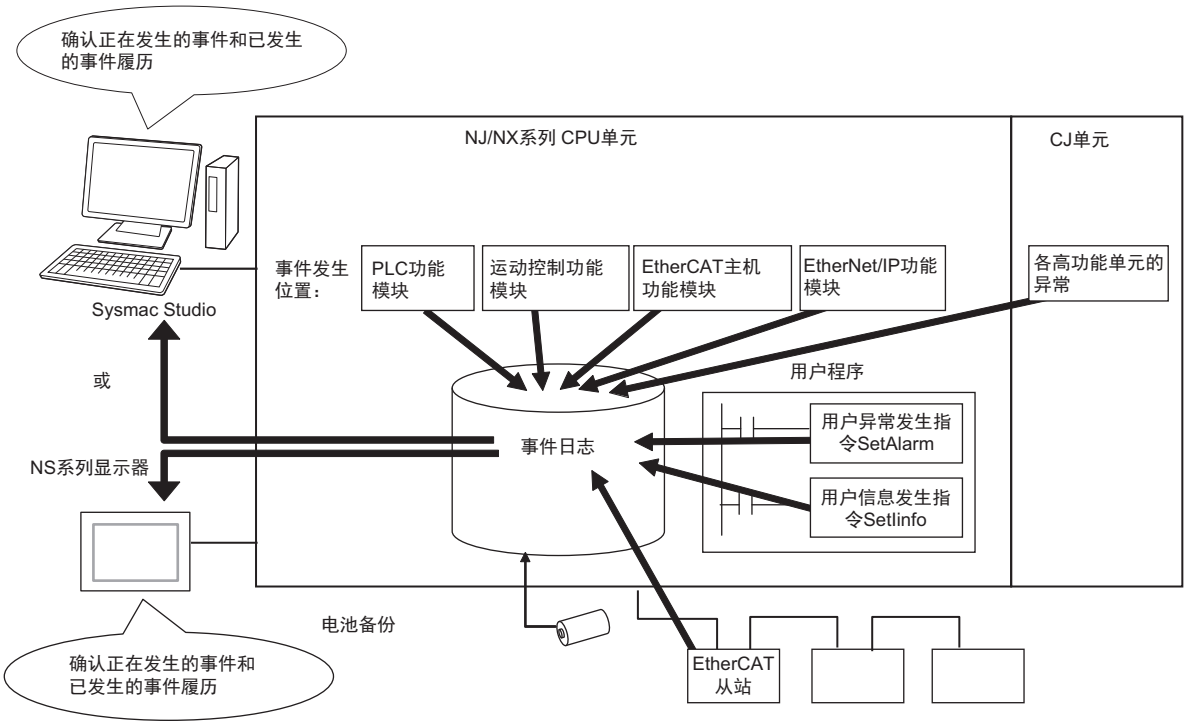


8-6 事件日志功能

下面介绍事件日志功能。

8-6-1 概要

事件日志功能是指对 NJ/NX 系列控制器中发生的异常、状态变化及用户定义现象等事件（*）进行记录的功能。



* “ 事件 ”是指控制器上意外发生的、包括异常在内的现象。异常是指根据控制器或用户定义，包括要向用户通知的非异常信息在内的总称。
包括以下两种四类。
- 控制器事件
  控制器异常
  控制器信息
- 用户事件
  用户异常
  用户信息

通过 NS 系列显示器确认事件时，请通过 CPU 单元的内置 EtherNet/IP 端口连接 NS 系列显示器。

使用注意事项

- 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。
- 关于显示器的故障排除应对范围，请参考📖《NJ/NX 系列 故障排除手册（SBCA-CN5-361）》。

特点

事件日志功能有以下特点。

- 除了异常记录外，还会记录电源接通 / 电源断开或开始运行等各种记录。
- 这些记录可按时间顺序确认，因此在发生故障时，有助于推断异常原因。

可记录内容

事件分为以下种类。

● 系统定义的事件 （控制器事件）

控制器自动检测的事件。CPU 单元的各功能模块、CJ 单元、EtherCAT 从站的事件。包括以下事件。

- 控制器异常
- 控制器信息

● 用户定义的事件 （用户事件）

用户设计的应用上的事件。通过指令执行，可发生以下事件。

- 用户异常
- 用户信息

可通过 Sysmac Studio 或显示器读取事件日志。

8-6-2 事件日志功能的详情

事件发生位置

用于识别在控制器的哪个位置发生了事件。
控制器事件和用户事件各自的发生位置如下。

● 控制器事件发生位置

控制器事件中，发生位置为存在于 CPU 单元中的功能模块。
不同的功能模块有更详细的事件发生位置。这称为 " 事件发生位置详情 "。
包括以下内容。

事件发生位置	事件发生位置详情
PLC 功能模块	I/O 总线主机、 CJ 系列单元 *1
运动控制功能模块	共通、轴、轴组
EtherCAT 主机功能模块	通信端口、 EtherCAT 主机、 EtherCAT 从站
EtherNet/IP 功能模块	通信端口 / 通信端口 1 / 通信端口 2、CIP/CIP1/CIP2、FTP、NTP、 SNMP

*1. CJ 系列的高功能单元 /EtherCAT 从站中的异常记录不在对象范围内。用个别的支持软件读取。

● 用户事件发生位置

用户事件中，事件发生位置为 PLC 功能模块。

类别

表示事件日志的分类信息。
通过 Sysmac Studio 或显示器参照时的单位。

事件的种类	事件日志的类别	内容
控制器事件	系统日志	控制器自身检测到的事件记录。包括 CJ 单元的异常。
	访问日志	因用户操作，对控制器的动作产生影响的事件记录。
用户事件	用户事件日志	用户定义的事件记录。

记录条数

各种事件日志中可记录的条数分别如下。
CPU 单元发生超出以下可记录条数的事件时，最早的信息将被覆盖。

事件的种类	事件日志的类别	每 1 类别的可记录条数
控制器事件	系统日志	NX701- □□□□：2,048 条 NJ501- □□□□：1,024 条 NJ301- □□□□：512 条 NJ101- □□□□：512 条
	访问日志	
用户事件	用户事件日志	

断电保持

NJ/NX 系列 CPU 单元即使发生断电，也可通过电池来保持事件日志。

使用注意事项

事件日志由电池保持。如果没有电池，将无法保持，敬请注意。
必要时，请定期导出事件日志。

事件代码

根据事件种类，由系统事先分配到控制器事件的代码或由用户分配到用户事件的代码。控制器事件的事件代码以 16 进制 8 位表示。
发生中的异常其事件代码可通过异常状态获取指令获取。
用户事件中可以标注 1 ～ 60000 的 10 进制数事件代码。

事件重要程度

各事件有表示重要程度的 “ 事件重要程度 ”。
事件重要程度的分类因事件种类不同而异。控制器事件和用户事件分别独立定义。

● 控制器事件时

根据对控制的影响大小，控制器事件的重要程度分为以下 5 个阶段。

编号	重要程度		分类
1	高	控制器异常	全部停止故障等级
2	↑		部分停止故障等级
3			轻度故障等级
4	↓		监视信息等级
5	低	控制器信息	一般信息等级

异常的重要程度越高，控制器提供的功能会受损，修复的紧急程度也越高。
该分配会在发生异常时显示在 Sysmac Studio 或显示器中。

● 用户事件时

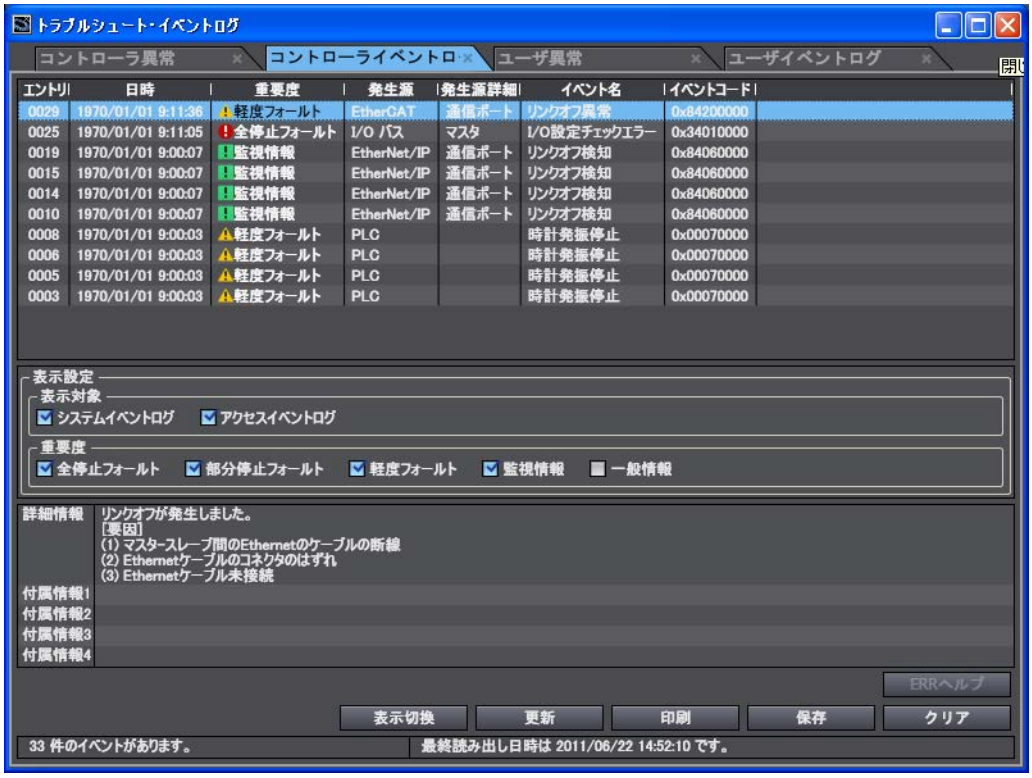
用户事件可按 NJ/NX 系列系统提供的以下重要程度分类。
对用户事件的重要程度进行定义。

编号	重要程度	分类	含义
1	高	User fault Level1	该事件表示应用上用户定义的异常。 用户执行 SetAlarm （发生用户异常）指令后发生事件。
2	↑	User fault Level2	
3		User fault Level3	
4		User fault Level4	
5		User fault Level5	
6		User fault Level6	
7		User fault Level7	
8	↓	User fault Level8	
9	低	User Information	该事件表示应用上用户定义的信息。 用户执行 SetInfo （发生用户信息）指令后发生事件。

事件日志的显示

Sysmac Studio 或显示器根据 “ 控制器日志 ”“ 用户日志 ” 的类别显示事件日志一览。 “ 控制器日志 ” 中显示 “ 访问日志 ” 和 “ 系统日志 ”。
此外， Sysmac Studio 还可显示 CJ 单元中或 EtherCAT 从站中记录的异常记录。

Sysmac Studio 中将如下进行一览显示。从一览中选择事件后，将显示对应的详细信息。



参考

控制器上发生 Sysmac Studio 或显示器的版本不支持的事件时，发生位置显示为 “ 不详 ”，事件名称显示为 “ 不详事件 ”。事件代码和附属信息直接显示。

事件日志的删除

● 通过 Sysmac Studio/ 显示器删除

通过 Sysmac Studio 或显示器，可按事件日志的控制器事件和用户事件分别删除事件日志。

使用注意事项

- 通过 Sysmac Studio ／显示器删除 CPU 单元中的事件日志时，请确认之前获取的所有事件信息均可删除后，再执行。特别是一般信息等级 / 监视信息等级的控制器事件或用户事件，可能尚未确认，敬请注意。
- 通过显示器删除事件日志时的限制事项请参考📖《NJ/NX 系列 故障排除手册（SBCA-CN5-361）》。

● 通过存储器全部清除来删除

通过 Sysmac Studio 对 NJ/NX 系列 CPU 单元进行存储器全部清除操作后，可选择是否删除事件日志。

事件日志的导出

可使用 Sysmac Studio 或显示器将显示的事件日志内容导出为 CSV 文件。

8-6-3 控制器事件 （控制器异常 / 信息）

概要

NJ/NX 系列系统中定义的异常 / 信息。
NJ/NX 系列系统检测到异常原因、信息原因后发生。

● 控制器异常

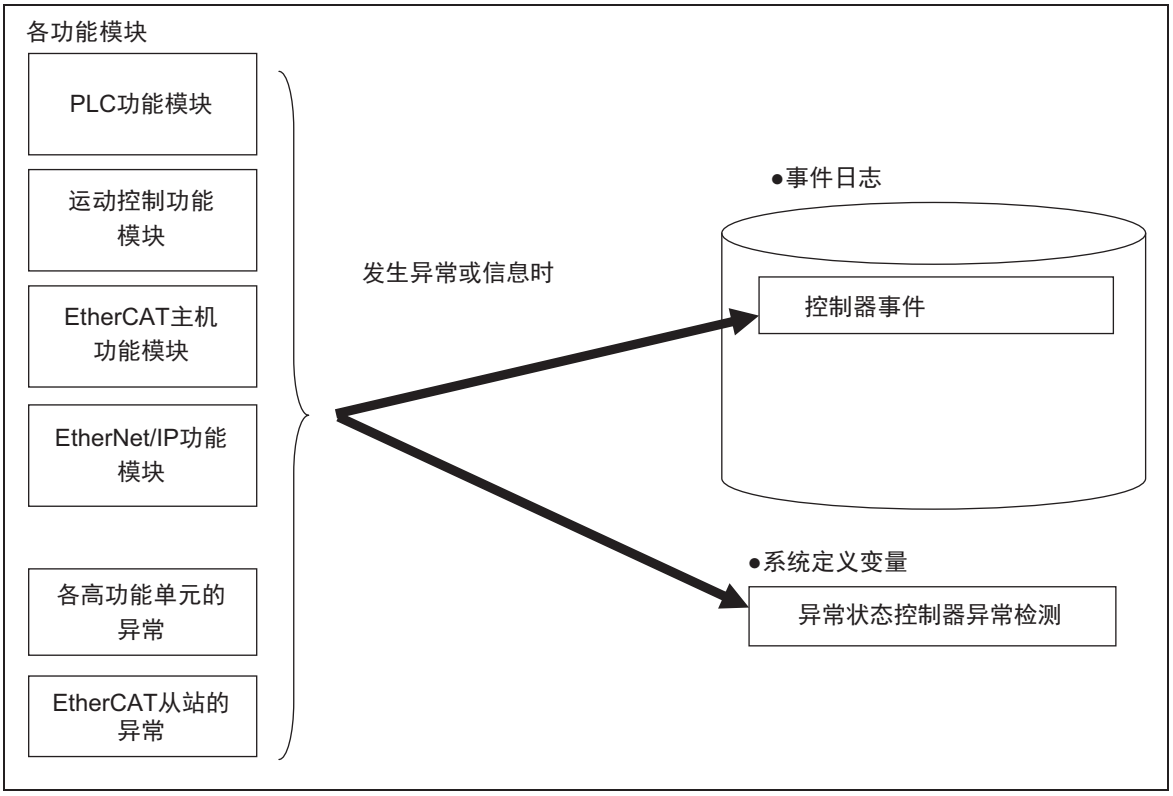
系统定义的异常。
“全部停止故障等级”、“部分停止故障等级”、“轻度故障等级”、“监视信息等级”的控制器事件的总称。
检测 CPU 单元的各功能模块、 CJ 单元、 EtherCAT 从站的各异常。此时，控制器异常将记录到事件日志中。
在用户程序上确认发生的控制器异常状况时，通过异常状态获取指令来参照系统定义变量的 “ 异常状态变量 ”。

（注）可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

控制器异常不会因动作模式的变更而解除。
控制器异常的详情请参考📖《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。

● 控制器信息

系统定义的通知信息，并非异常。“ 一般信息等级 ” 的控制器事件。
例如，电源 ON/OFF、运行开始 / 停止、 Sysmac Studio 的在线连接、用户程序的下载等异常以外的现象。



8-6-4 用户事件 （用户异常 / 信息）

概要

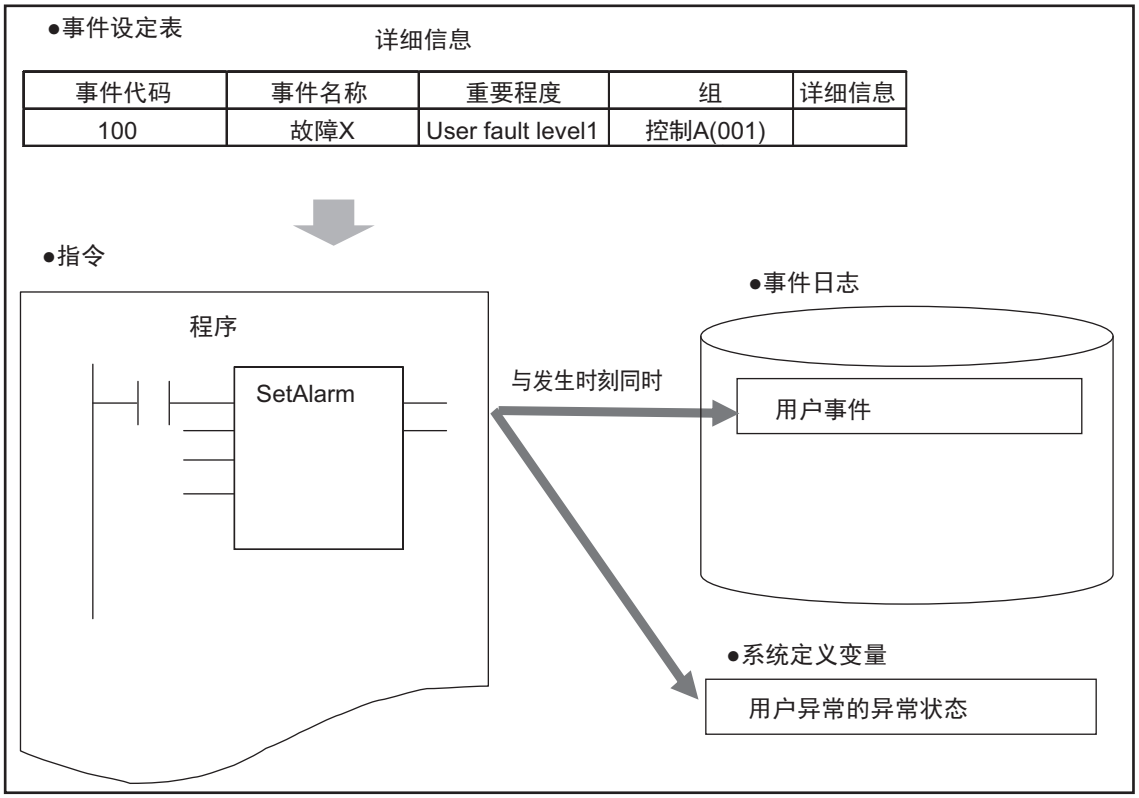
用户定义的异常 / 信息。可通过指令发生。

● 用户异常

用户定义的异常。
通过用户异常发生 （SetAlarm）指令，发生用户异常。此时，用户异常将记录到事件日志中。
对应的系统定义变量变为 TRUE。
用户异常不会因动作模式的变更而解除。

● 用户信息

用户定义的通知信息，并非异常。
通过用户信息发生 （SetInfo）指令，发生用户信息。
此时，用户信息将记录到事件日志中。



使用步骤

按以下步骤使用。

● 用户异常

1. 将 " 用户异常 " 登录到事件设定表

↓

2. 执行用户异常发生指令 SetAlarm
   （指定事件设定表中定义的事件代码）

↓

3. 发生用户异常

↓

4. 对应的系统定义变量 "_AlarmFlag" （用户异常状态）变为 TRUE。以该条件执行任意处理

↓

5. 通过 Sysmac Studio、指令或显示器确认用户异常

● 用户信息

1. 将 " 用户信息 " 登录到事件设定表

↓

2. 执行用户信息发生指令 SetInfo

↓

3. 确认已登录到事件日志中

事件设定表的设定

要发生 “ 用户异常 ” 或 “ 用户信息 ”，首先需要通过 Sysmac Studio，事先在 “ 事件设定表 ” 中登录 “ 用户异常 ” 或 “ 用户信息 ”。

这里设定的用户事件可在 Sysmac Studio 或显示器中显示相同的内容。

事件设定表中可登录的事件总数最多为 5120 条。

事件设定表的设定画面

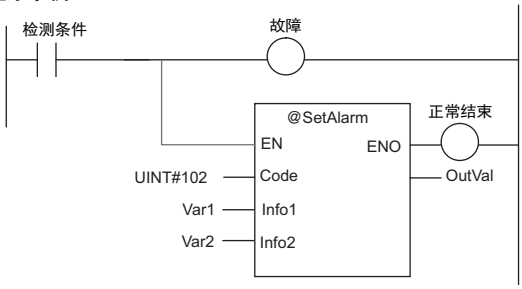
事件设定表

事件代码	事件名称	重要程度	组	详细信息
10001	故障X	User fault Level3	控制A(001)	
:	:	:	:	:

详细信息

■内容
发生故障X。
■处理
请确认安全，然后按照原因代码进行处理。

程序示例



在事件设定表中设定以下内容。

● 事件设定表的内容

项目	内容	值
事件代码	可根据事件重要程度指定识别事件的编号。	用户异常：1 ～ 40000 用户信息：40001 ～ 60000
事件名称	可记载表示事件名称的标题。	最多 128 个字符
重要程度	指定事件的重要程度。 用数值表示重要程度，数值越小，重要程度越高。	用户异常：User fault Level1 ～ 8 用户信息：User Information
组	可指定表示事件发生位置和分类的组名。可根据用户定义对各事件进行分组。	最多 32 个字符 可使用的字符无限制。 区分大小写。 无保留字。
详细信息	可记载对事件内容进行说明的信息。 用户可自由填入任意字符串。在 Sysmac Studio ／显示器上显示事件时使用。	最多 1024 个字符 可使用的字符无限制。 区分大小写。 无保留字。
发生全部停止故障等级的控制器异常时，在显示器中显示的异常详细信息	请参考后述的 “ 参考：发生全部停止故障等级的控制器异常时，在显示器上显示用户信息的功能：”。	最多 128 个字符 可使用的字符无限制。 区分大小写。 无保留字。
注释	可对多个表添加信息。	

参考

可对同一事件代码，向多个区域或多个用户设定最多 9 种语言（9 种）。事件设定表可利用 Sysmac Studio，通过 Microsoft Excel 文件的剪贴板导入。

参考

发生全部停止故障等级的控制器异常时，在显示器上显示用户信息的功能：
发生全部停止故障等级的控制器异常时，停止用户程序的执行。
NJ/NX 系列中，发生全部停止故障等级的控制器异常时，能在显示器上显示面向用户的信息。该信息在 Sysmac Studio 的事件设定表的用户事件一览下段中设定。

● 事件重要程度和事件代码

事件种类	重要程度	事件重要程度的分类名称（*）	对应的事件代码的范围	说明
用户异常	高	User fault Level1	1 ～ 5000	用户从 8 个级别中选择。
	↕	User fault Level2	5001 ～ 10000	
		User fault Level3	10001 ～ 15000	
		User fault Level4	15001 ～ 20000	
		User fault Level5	20001 ～ 25000	
		User fault Level6	25001 ～ 30000	
		User fault Level7	30001 ～ 35000	
	低	User fault Level8	35001 ～ 40000	
用户信息	最低	User Information	40001 ～ 60000	事件种类为 “ 用户信息 ” 的重要程度

* 用户异常的重要程度与控制器异常的重要程度分类不同。

使用注意事项

更新事件设定表，并传送到 CPU 单元时，之前的用户事件的事件日志将以旧内容记录。由于事件设定表发生更新，因此可能发生不匹配。敬请注意。

相关指令

包括发生和解除用户异常的指令以及确认所发生用户异常的指令。

● 发生和解除用户异常

利用以下指令发生 / 解除用户异常、发生用户信息。

可同时变为发生中状态的用户异常数量为每个事件重要程度 32 件，总共最多 256 件。

指令名称	指令	功能
用户异常发生指令	SetAlarm	发生用户异常。
用户异常解除指令	ResetAlarm	解除用户异常。
用户信息发生指令	SetInfo	将任意用户信息记录到事件日志中。

● 确认用户异常

通过用户异常状态获取 （GetAlarm）指令，可获取用户异常的发生有无、发生中的用户异常中最重要的事件重要程度及其事件代码。

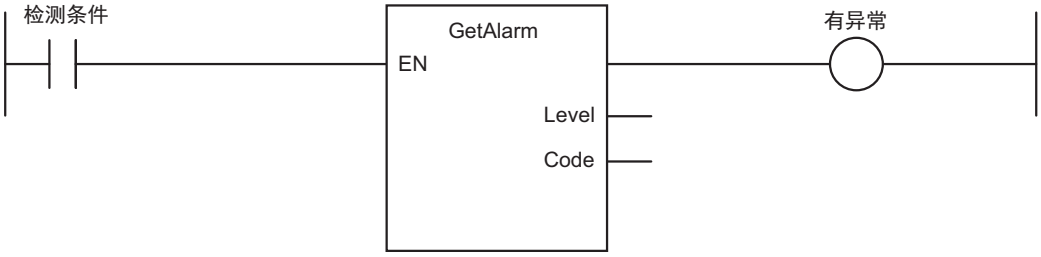
例）

事件设定表

事件代码	事件名称	重要程度	组	详细信息
10001	故障X	User fault Level3	控制A(001)	
:	:	:	:	:

详细信息
■内容 发生故障X。 ■处理 请确认安全，然后按照原因代码进行处理。

实际程序



参考

可根据用户异常，对发生的控制器异常，添加处理方法等信息。
关于发生中的控制器异常，通过 GetPLCError 等控制器异常的状态获取指令，获取异常状态和事件代码，并将该信息作为触发发生用户异常。

例 1
电池电压降低时，获取事件代码 （16#000B0000），显示以下信息。

```
电池用完。
更换电池前，请通电 5 分钟以上，并在电源关闭后
5 分钟内更换新电池。
```

例 2
发生部分停止故障等级的控制器异常时，获取事件重要程度 （最重要的状态：2），显示以下信息。

```
装置发生故障。请咨询经销商。
＜维修联系方式＞
受理时间 AM8：00 ～ PM9：00
TEL ***-****-****
```

用户异常相关的系统定义变量

变量名称	名称	功能	数据类型	R/W
_AlarmFlag	用户异常状态	发生用户异常时，与事件重要程度对应的位将变为 TRUE。 User fault Level1 ～ 8 对应位 00 ～ 07。	WORD	R

记录到事件日志中

发生用户异常或用户信息时，如果通过 ResetAlarm 解除，将记录到事件日志中。此时，事件日志的用户事件日志中将记录发生时刻、事件代码、事件重要程度、附属信息 1、 2。

用户异常的解除

用户异常可通过 NJ/NX 系列控制器电源的 OFF ➜ ON 来解除。
也可通过 Sysmac Studio、指令 （用户异常解除指令ResetAlarm）或显示器解除异常。
详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

8-7 事件重要程度变更功能

NJ/NX 系列控制器中发生的异常、状态变化、用户定义现象等的总称，称为事件。用户可根据 Sysmac Studio 的显示、 CPU 单元正面的 LED 动作，知道发生了什么样的事件。
事件包括由系统事先定义的控制器事件和由用户任意定义的用户事件。各控制器事件分为 5 个事件重要程度。事件重要程度的详情请参考📖「 事件重要程度 」（P.8-54）。
事件重要程度变更功能是指用户任意变更各控制器事件中分配的事件重要程度的功能。

版本相关信息

事件重要程度变更功能可在CPU单元版本1.03以上和Sysmac Studio Ver.1.04以上的组合中使用。

8-7-1 事件重要程度变更的用途

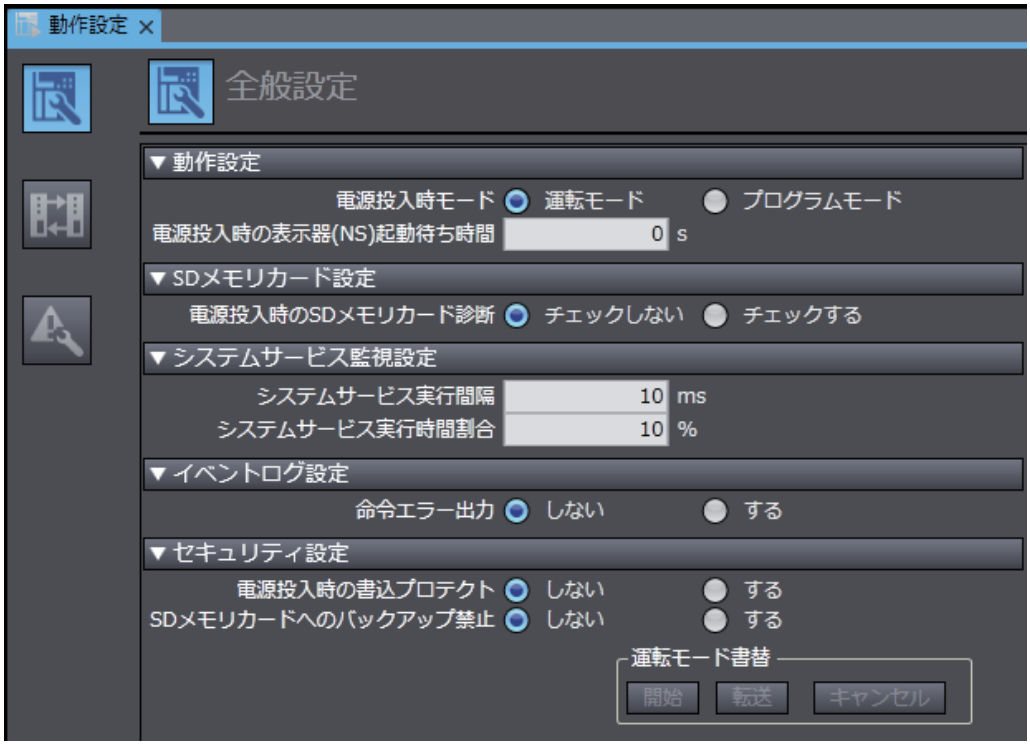
根据发生的控制器事件的重要程度，事先规定 CPU 单元正面 LED 的动作。使用事件重要程度变更功能，可对发生的任意事件，变更控制器的动作。
例如， ERROR LED 在事件重要程度为轻度故障等级时闪烁，监视信息等级时保持熄灭。若使用事件重要程度变更功能，可对发生的任意事件，设定为 ERROR LED 熄灭或闪烁。
控制器的不同事件重要程度的动作请参考📖《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。

8-7-2 可变更事件重要程度的事件

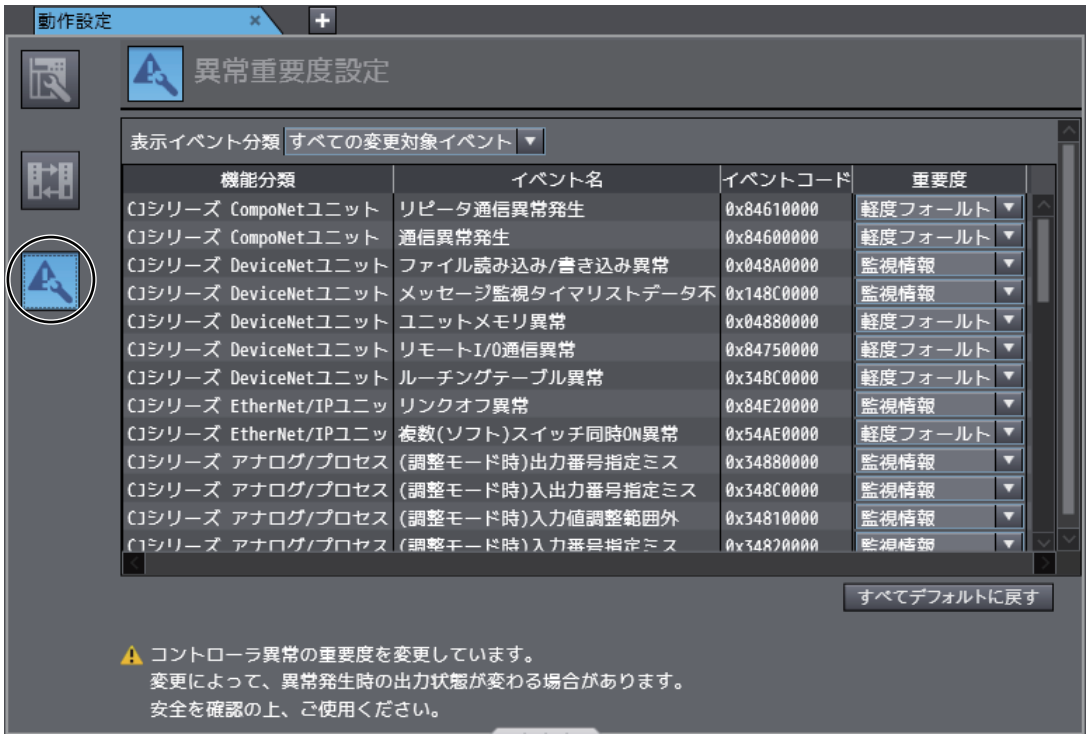
事件的重要程度是否可以变更，取决于各事件。
控制器事件的种类、重要程度、重要程度变更的可否请参考📖《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。

8-7-3 事件重要程度变更的操作方法

1 **双击或右击 Sysmac Studio 的 [配置和设定] | [控制器设定] | [动作设定]，选择 [编辑]。**
编辑窗口中将显示动作设定的 [整体设定] 视图。



2 **单击 [异常重要程度设定] 按钮。**
显示可变更事件重要程度的事件一览。



3 **将要变更重要程度的事件之 [重要程度] 设定为变更目标的重要程度。**

使用注意事项

通过 Sysmac Studio 进行事件重要程度变更设定，并下载到控制器时，如果已发生该事件，开始下载时，将先解除该事件。下载过程中，如果再次发生事件，将根据变更前的重要程度动作。下载完成后，如果发生该事件，将切换为与变更后的重要程度对应的动作。

9

备份功能

本章介 NJ/NX 系列控制器的各种设定数据的备份功能。备份功能包括不同的种类，各自的备份对象数据和保存位置不同。首先，对备份功能进行整体说明，然后对不同种类进行说明。

9-1 备份功能的概要 9-3
9-1-1 备份功能的用途 9-3
9-1-2 备份功能的操作步骤示例 9-4
9-1-3 备份功能的对象数据 9-6
9-1-4 备份功能的种类 9-7
9-1-5 备份功能的种类和数据表的关系 9-9
9-1-6 备份功能的对象范围 9-10

9-2 SD 存储卡的备份功能 9-12
9-2-1 备份 （从控制器到 SD 存储卡） 9-13
9-2-2 恢复 （从 SD 存储卡到控制器） 9-18
9-2-3 核对 （控制器和 SD 存储卡之间） 9-20

9-3 SD 存储卡备份禁止功能 9-25

9-4 SD 存储卡自动传送功能 9-26

9-5 SD 存储卡程序传送功能 9-28

9-6 Sysmac Studio 控制器备份功能 9-34
9-6-1 备份 （从控制器到计算机） 9-35
9-6-2 恢复 （从计算机到控制器） 9-36
9-6-3 核对 （控制器和计算机之间） 9-37

9-7 Sysmac Studio 备份文件的导入 / 导出功能 9-38

9-8 Sysmac Studio 变量 / 存储器的备份功能 9-39
9-8-1 Sysmac Studio 变量 / 存储器备份功能的对象数据 9-39
9-8-2 Sysmac Studio 变量 / 存储器备份功能的操作方法 9-39
9-8-3 CPU 单元型号间的兼容性 9-40

9-9 连接 EtherCAT 从站时的备份功能 9-42
9-9-1 EtherCAT 从站的备份对象数据 9-42
9-9-2 控制器在各种状态下的备份功能执行可否 9-43
9-9-3 EtherCAT 从站的恢复执行条件 9-44
9-9-4 备份功能的对象 EtherCAT 从站 9-45

9-10 连接 EtherCAT 从站终端时的备份功能 9-48

9-10-1 EtherCAT 从站终端的备份对象数据 9-48
9-10-2 EtherCAT 从站终端在各状态下的备份功能执行可否 9-49
9-10-3 EtherCAT 从站终端的恢复执行条件 9-49

9-11 连接 CJ 单元时的备份功能 9-50
9-11-1 CJ 单元的备份对象数据 9-50
9-11-2 控制器在各种状态下的备份功能执行可否 9-50
9-11-3 CJ 单元的恢复执行条件 9-51

9-12 备份相关文件 9-52
9-12-1 备份相关文件的种类 9-52
9-12-2 备份文件的规格 9-53
9-12-3 恢复指令文件的规格 9-54
9-12-4 自动传送指令文件的规格 9-56
9-12-5 控制器核对结果文件的规格 9-58
9-12-6 EtherCAT 核对结果文件的规格 9-59
9-12-7 EtherCAT 从站终端核对结果文件的规格 9-60
9-12-8 CJ 单元核对结果文件的规格 9-61

9-13 备份功能的文件间兼容性 9-63
9-13-1 备份功能间的兼容性 9-63
9-13-2 CPU 单元型号间的兼容性 9-64
9-13-3 CPU 单元的单元版本兼容性 9-65

9-14 无法与备份功能同时执行的功能 9-66

9-1 备份功能的概要

NJ/NX 系列的备份功能是指以下 3 个。

功能	内容
备份	将控制器中的各种设定数据汇总保存到 SD 存储卡或计算机中的功能。要保存的文件称为备份文件。
恢复	将 SD 存储卡或计算机中的备份文件传送到控制器的功能。控制器中的各种设定数据将替换为备份时的数据。
核对	对 SD 存储卡或计算机中的备份文件内容与当前控制器中的各种设定数据是否一致进行判定的功能。

下面介绍整体的备份功能。

项目	内容
备份功能的用途	介绍备份功能的有效使用方法。
备份功能的操作步骤示例	备份功能可通过简单的操作执行。表示其示例。
备份功能的对象数据	介绍在使用备份功能后，可以保存所连接单元或从站中的哪些数据。
备份功能的种类	备份功能因数据保存位置等不同，有以下几种。介绍有哪些种类以及它们的区别。
备份功能的种类和数据表的关系	因备份功能的种类不同，对象数据表也不同。介绍备份功能的种类和数据表的关系。
备份功能的对象范围	介绍在使用备份功能后，可以保存哪个单元或从站的设定数据。

安全要点

若硬件修订版本不同，性能也会不同。向不同硬件修订版本的 CPU 单元传送用户程序及各种数据、设定值时，请进行充分的动作确认后再转移到正式运行。

9-1-1 备份功能的用途

备份功能可在以下用途中使用。

项目	用途
程序、设定变更	变更运行中装置的用户程序或设定时。
硬件更换	更换 CPU 单元、单元、从站的硬件时。
装置异常时的原因调查	装置发生异常时，为分析原因而保存控制器的数据时。
装置的备份和复原	装置发生异常时，将装置恢复为正常的状态数据时。或者事先备份运行中装置的数据时。
装置的制造	为制造相同的装置，将复制源装置的数据传送到购入后初始状态的装置时。

9-1-2 备份功能的操作步骤示例

使用备份功能，只需简单的操作即可进行控制器备份、恢复、核对。
下面为使用 CPU 单元的正面开关，备份到 SD 存储卡以及恢复、核对的操作步骤。

使用注意事项

NX 系列 CPU 单元时，拨动开关为 1 到 8，共 8 个。使用备份功能时，请将 5 到 8 全部设为 OFF。

备份的操作步骤

● 备份准备

1 **将 SD 存储卡安装到 CPU 单元中。**

2 **将 CPU 单元的拨动开关 1 到 4 分别设定为 1:OFF,2:OFF,3:ON,4:OFF。**

● 执行备份

3 **长按 3 秒 SD 存储卡停止供电按钮。**
备份开始。 SD PWR 每亮起 3s 后熄灭 0.5s，以此频率闪烁。
完成备份后， SD PWR 变为亮灯。

● 备份操作结束

4 **将 CPU 单元的拨动开关 1 到 4 全部设为 OFF。**

5 **按下 SD 存储卡停止供电按钮，确认 SD PWR 熄灭。**

6 **拔出 SD 存储卡。**

恢复的操作步骤

● 恢复准备

1 **关闭 CPU 单元及 EtherCAT 从站的电源。**

2 **将保存有备份文件的 SD 存储卡安装到 CPU 单元中。**

3 **将 CPU 单元的拨动开关 1 到 4 分别设定为 1:OFF,2:OFF,3:ON,4:ON。**

● 执行恢复

4 **打开 CPU 单元及 EtherCAT 从站的电源。**
开始恢复。 SD PWR 每亮起 3s 后熄灭 0.5s，以此频率闪烁。
完成恢复后， SD PWR 变为亮灯。

● **恢复操作结束**

5 **按下 SD 存储卡停止供电按钮，确认 SD PWR 熄灭。**

6 **关闭 CPU 单元及 EtherCAT 从站的电源。**

● **正常运行开始**

7 **拔出 SD 存储卡。**

8 **将 CPU 单元的拨动开关 1 到 4 全部设为 OFF。**

9 **打开 CPU 单元及 EtherCAT 从站的电源。**

使用注意事项

连接 EtherCAT 从站时的恢复

- 连接 EtherCAT 从站进行恢复时，恢复完成后，请务必重新接通 CPU 单元及 EtherCAT 从站的电源。若不重新接通电源直接开始运行，控制器可能发生意外动作。
- 连接EtherCAT从站进行恢复，然后进行核对时，请先关闭CPU单元及EtherCAT从站的电源，以安全模式启动后再核对。正常重新接通电源后，控制器会在执行核对前开始运行。因此，可能在核对不一致的状态下开始运行。安全模式的详情请参考📖《NJ/NX 系列 故障排除手册（SBCA-CN5-361）》。

核对的操作步骤

● **核对准备**

1 **将保存有备份文件的 SD 存储卡安装到 CPU 单元中。**

2 **将 CPU 单元的拨动开关 1 到 4 全部设为 OFF。**

● **执行核对**

3 **长按 3 秒 SD 存储卡停止供电按钮。**

开始核对。 SD PWR 每亮起 3s 后熄灭 0.5s，以此频率闪烁。

核对一致且核对完成后， SD PWR 变为亮灯。

如果在核对不一致的状态下完成核对， SD PWR 将 0.5s 亮灯 0.5s 熄灭，变为闪烁。

● **核对操作结束**

4 **按下 SD 存储卡停止供电按钮，确认 SD PWR 熄灭。**

5 **拔出 SD 存储卡。**

9-1-3 备份功能的对象数据

备份功能的对象数据如下所示。
在备份功能的详细说明中，表示对象数据时，使用 “ 数据组 ”。

数据组	数据项目
用户程序设定	EtherCAT 构成 （EtherCAT 从站构成、 EtherCAT 主机设定） 单元构成和单元设定 I/O 映射 控制器设定 （动作设定、内置 EtherNet/IP 端口设定） 运动控制设定 凸轮数据设定 事件设定 任务设置 数据追踪设定 标签数据链接表 控制器名称 操作权限认证 用户程序执行用 ID （用户程序内） 内置时钟 （时区设定） POU 数据 （数据型、全局变量）
变量的当前值	有保持属性的变量值*1
CJ 单元用存储器的当前值	CJ 单元用存储器中的 HR、 DM、 EM 区域的值*2
单元和从站设定	EtherCAT 从站的备份参数*3 CJ 单元内的参数 *2*4
绝对编码器原点位置偏置	用于恢复带绝对编码器的伺服驱动器的当前位置的设定值

*1. 有保持属性的系统定义变量中，包含不在备份功能的对象范围内的系统定义变量。详情请通过「A-5 系统定义变量的个别规格」（P.A-79）确认个别规格。
*2. 可使用 CJ 单元用存储器及 CJ 单元内参数的 CPU 单元仅限 NJ 系列 CPU 单元。
*3. EtherCAT 从站中，有些机型的部分或全部设定参数可能不在备份对象内。备份功能的对象 EtherCAT 从站详情请参考「9-9-4 备份功能的对象 EtherCAT 从站」（P.9-45）。
*4. 备份功能的具体对象数据请参考 《CJ 系列 CJ2 CPU 单元 用户手册 硬件篇 （SBCA-CN5-349）》。

安全要点

绝对编码器原点位置偏置相关的注意事项

CPU 单元利用电池来保持绝对编码器的信息，即 “ 绝对编码器原点位置偏置 ”。满足以下任一条件时，请将绝对编码器原点位置偏置从恢复对象中排除，再执行恢复。然后，请重新设定绝对编码器原点。若未设定绝对编码器原点，控制对象可能发生意外动作。

- 伺服电机或驱动器的个体变为创建备份时的个体。
- 创建备份后，执行了绝对编码器的设置。
- 绝对编码器的绝对值数据将丢失。

9-1-4 备份功能的种类

NJ/NX 系列中可使用的备份功能的数据保存位置分为 SD 存储卡和计算机。操作方法有以下三种：通过 CPU 单元正面开关操作、通过系统定义变量操作、通过 Sysmac Studio 画面操作。

将数据保存到 SD 存储卡的功能

向 SD 存储卡进行数据备份、恢复、核对的功能为 SD 存储卡备份功能。相关功能包括：SD 存储卡备份禁止功能、 SD 存储卡自动传送功能、 SD 存储卡程序传送功能。

功能名称		内容	操作方法			参考
			CPU 单元正面开关	系统定义变量	Sysmac Studio 画面	
SD 存储卡的备份功能	备份	将控制器的数据作为备份文件保存到 SD 存储卡中。	○	○	○	「9-2-1 备份 （从控制器到 SD 存储卡）」（P.9-13）
	恢复	将 SD 存储卡中的备份文件数据传送到控制器中。	○			「9-2-2 恢复 （从 SD 存储卡到控制器）」（P.9-18）
	核对	对控制器的数据和 SD 存储卡的备份文件数据进行核对。	○	○	○	「9-2-3 核对 （控制器和 SD 存储卡之间）」（P.9-20）
SD 存储卡备份禁止功能		禁止向 SD 存储卡备份。			○	「9-3 SD 存储卡备份禁止功能」（P.9-25）
SD 存储卡自动传送功能		接通电源时，将 SD 存储卡中的备份文件数据自动读取到控制器中。读取后 CPU 单元的动作模式取决于 [接通电源时的模式] 的设定。	○			「9-4 SD 存储卡自动传送功能」（P.9-26）
SD 存储卡程序传送功能		根据系统定义变量，将 SD 存储卡中保存的程序传送到控制器中。 传送后 CPU 单元的动作模式取决于 [接通电源时的模式] 的设定。		○		「9-5 SD 存储卡程序传送功能」（P.9-28）

将数据保存到计算机的功能

将控制器的数据向计算机进行备份、恢复、核对的功能为 Sysmac Studio 控制器备份功能。
不使用控制器，通过 Sysmac Studio 的项目将各数据保存、读取到计算机的功能为 Sysmac Studio 备份文件的导入 / 导出功能。
只将电池保存的当前值向计算机进行备份、恢复的功能为 Sysmac Studio 变量 / 存储器的备份功能。

功能名称		内容	操作方法			参考
			CPU 单元正面开关	系统定义变量	Sysmac Studio 画面	
Sysmac Studio 控制器备份功能	备份	将控制器的数据作为备份文件保存到计算机中。			○	「9-6-1 备份 （从控制器到计算机）」（P.9-35）
	恢复	将计算机中的备份文件数据传送到控制器中。			○	「9-6-2 恢复 （从计算机到控制器）」（P.9-36）
	核对	对控制器的数据和计算机的备份文件数据进行核对。			○	「9-6-3 核对 （控制器和计算机之间）」（P.9-37）
Sysmac Studio 备份文件的导入 / 导出功能	导出	不使用控制器，通过 Sysmac Studio 的项目创建备份文件。			○	「9-7 Sysmac Studio 备份文件的导入 / 导出功能 」（P.9-38）
	导入	不使用控制器，将备份文件读取到 Sysmac Studio 的项目中。			○	
Sysmac Studio 变量 / 存储器的备份功能	备份	将电池保持的数据当前值作为文件 （XML 格式）备份（读取）到计算机中。			○	「9-8 Sysmac Studio 变量 / 存储器的备份功能 」（P.9-39）
	恢复	将电池保持的数据当前值由计算机对 CPU 单元进行恢复（写入）。			○	

版本相关信息

- SD存储卡备份功能、SD存储卡自动传送功能、Sysmac Studio控制器备份功能、Sysmac Studio备份文件的导入 / 导出功能可在 CPU 单元版本 1.03 以上和 Sysmac Studio Ver.1.04 以上的组合中使用。
- SD存储卡程序传送功能可在CPU单元版本1.11以上和Sysmac Studio Ver.1.15以上的组合中使用。

参考

备份功能在系统服务中执行。因此，以运行模式执行了备份和核对时， NJ 系列 CPU 单元上的执行完成时间可能会变长。以运行模式执行备份和核对时，请确保足够的系统服务执行时间。以程序模式执行系统服务，可缩短处理时间。

9-1-5 备份功能的种类和数据表的关系

因备份功能的种类不同，对象数据表也不同。备份功能的种类和数据表的关系如下所示。

（○：对象　×：对象外）

备份功能的种类		数据组				
		用户程序设定	变量的当前值	CJ 单元用存储器的当前值*1	单元和从站设定	绝对编码器原点位置偏置
SD 存储卡的备份功能	备份	○	○*2	○*3	○	○
	恢复*4	○	○*2	○*3	○	○
	核对*4	○*5	×	×	○	×
SD 存储卡自动传送功能*6		○	○*2	○*3	×	×
SD 存储卡程序传送功能*7		○	○*2	○*3	×	×
Sysmac Studio 控制器备份功能	备份	○	○*2	○*3	○*8	○
	恢复	○	○*2	○*3	○*8	○
	核对	○*5	×	×	○*8	×
Sysmac Studio 备份文件的导入 / 导出功能	备份文件的导出	○*9	×	×	×	×
	备份文件导入	○*9	×	×	○	×
Sysmac Studio 变量 / 存储器的备份功能	备份恢复	×	○*2	○*3	×	○

*1. 可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。
*2. 备份的对象数据仅在保持属性的变量当前值时处理。
*3. 备份的对象数据仅在保持属性的 CJ 单元用存储器的当前值时处理。
*4. 对象数据组中，用户程序、设定以外的数据仅将恢复指令文件中指定为 “ 恢复 ” 的数据作为对象。
*5. 用户程序、设定数据组中，不核对数据追踪设定。
*6. 对象数据组中，用户程序、设定以外的数据仅将自动传送指令文件中指定为 “ 传送 ” 的数据作为传送对象。
*7. 对象数据组中，仅传送系统定义变量中设定为传送对象的数据。
*8. 备份对象为CJ单元时，将CJ单元中的参数作为对象。EtherCAT从站时，将EtherCAT从站的备份参数作为对象。
*9. 以下数据不在处理对象范围内。内置 EtherNet/IP 用标签数据链接设定、操作权限认证、数据追踪设定。

参考

Sysmac Studio变量 / 存储器的备份功能中处理的文件与其他备份功能中处理的文件不同，因此不兼容。
Sysmac Studio 变量 / 存储器的备份功能详情请参考「9-8 Sysmac Studio 变量 / 存储器的备份功能」（P.9-39）。

9-1-6 备份功能的对象范围

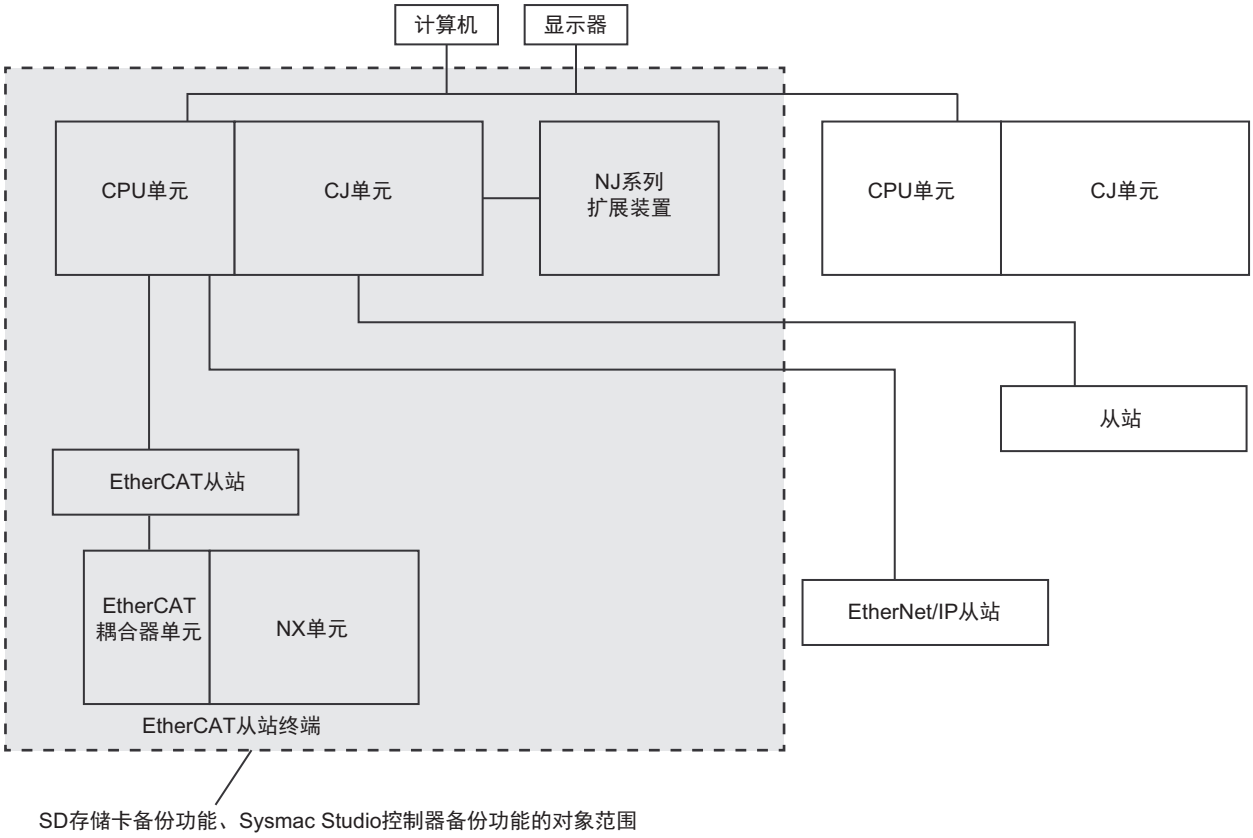
因备份功能不同，对象从站或单元也不同。各备份功能的对象从站、单元如下所示。

（○：对象　×：对象外）

备份功能的种类	从站、单元					
	NJ/NX 系列 CPU 单元	EtherCAT 从站*1	CJ 单元*2		EtherNet/IP 从站	计算机、显示器
			单元、主站单元	从站		
SD 存储卡的备份功能	○	○*3	○	×	×	×
SD 存储卡自动传送功能	○	×	×	×	×	×
SD 存储卡程序传送功能	○	×	×	×	×	×
Sysmac Studio 控制器备份功能	○	○*3	○	×	×	×
Sysmac Studio 备份文件的导入 / 导出功能	○	○*4	×	×	×	×
Sysmac Studio 变量 / 存储器的备份功能	○	×	×	×	×	×

*1. 包括 EtherCAT 从站终端。备份功能的对象为 EtherCAT 从站终端时，EtherCAT 耦合器单元和各 NX 单元均为备份功能的对象。
*2. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。
*3. 安全控制单元不在对象范围内。安全控制单元的设定数据导入 / 导出功能请参考《NX 系列 安全控制单元 用户手册 （SGFM-CN5-710）》。
*4. 只能导入。导出功能不在对象范围内。

SD 存储卡备份功能、 Sysmac Studio 控制器备份功能的对象从站和单元如下图所示。



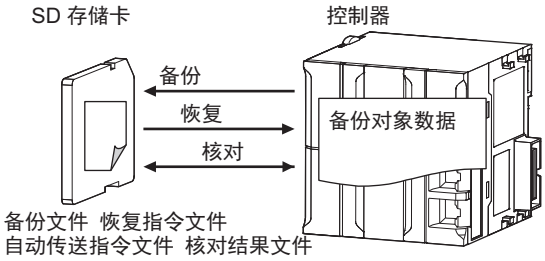
（注）可使用 CJ 单元及 NJ 系列扩展装置的 CPU 单元仅限 NJ 系列 CPU 单元。

版本相关信息

EtherCAT 从站终端的连接可在 CPU 单元版本 1.05 以上和 Sysmac Studio Ver.1.06 以上的组合中使用。

9-2 SD 存储卡的备份功能

使用 SD 存储卡，进行控制器数据的备份、恢复、核对的功能。



执行备份后，将在 SD 存储卡的指定目录下创建 “ 备份文件 ”、“ 恢复指令文件 ”、“ 自动传送指令文件 ”。
此外，若执行核对，将在指定目录下创建 “ 核对结果文件 ”。
这些统称为 “ 备份相关文件 ”。
备份相关文件的作用如下。

文件	作用			
	内容	备份	恢复	核对
备份文件	用备份相关功能中处理的控制器数据制作的文件。	创建	参考	参考
恢复指令文件	执行恢复时，用于指定恢复对象数据组的文件。在计算机的文本编辑器中编辑文件，可指定恢复对象。	创建	参考	参考
自动传送指令文件	执行 SD 存储卡自动传送时，用于指定传送对象数据组的文件。在计算机的文本编辑器中编辑文件，可指定传送对象。	创建	无操作	无操作
核对结果文件	执行核对后，用于保存核对结果的文件。	无操作	无操作	创建

本功能的操作方法、对象目录及可执行的动作模式如下。

操作方法	目录*1	可执行的动作模式		
		备份	恢复	核对
CPU 单元正面开关*2	根目录	运行模式、程序模式	电源接通时	运行模式、程序模式
系统定义变量*3*4	系统定义变量中指定的目录	运行模式、程序模式	不可执行	运行模式、程序模式
Sysmac Studio SD 存储卡画面	画面中指定的目录	运行模式、程序模式	不可执行	运行模式、程序模式

*1. 仅可在 SD 存储卡内指定。
*2. 恢复或核对时，请事先将备份文件和恢复指令文件保存到根目录下。
*3. 通过显示器发出指示时使用。无法通过用户程序访问该系统定义变量。
*4. 备份中或核对中时，请在用户侧进行管理，防止从其他的显示器发出备份或核对指示。若非如此，可能不按预期动作。

9-2-1 备份 （从控制器到 SD 存储卡）

将控制器的数据保存到 CPU 单元上安装的 SD 存储卡中。

处理内容

- 备份对象数据为所有数据组。
- 执行备份后，将在 SD 存储卡的指定目录下创建 “ 备份文件 ”、“ 恢复指令文件 ”、“ 自动传送指令文件 ”。
- 指定目录下已经存在备份相关文件时，将被覆盖。
- 正在向 SD 存储卡写入备份相关文件时如果发生错误，已有的备份相关文件将被删除。然后，也不会创建新的备份相关文件。
- 创建新的备份相关文件前如果发生错误，将保持已有的文件，不创建新文件。
- 正在执行备份时，即使按下 SD 存储卡停止供电按钮，也无效。
- 备份完成后， SD 存储卡将保持安装状态。

操作方法

● 通过 CPU 单元正面开关操作的方法

处理阶段	操作方法
开始指示	将拨动开关 1 到 4*1 分别设定为 1:OFF,2:OFF,3:ON,4:OFF，并长按 SD 存储卡停止供电按钮 3 秒，将开始执行。
执行中	刚开始执行后*2 SD PWR 将以亮灯➞熄灭 0.5s ➞亮灯的模式闪烁 1 次。 备份执行中 SD PWR 每亮起 3s 后熄灭 0.5s，以此频率闪烁。 SD BUSY 闪烁 （闪烁模式不确定）。 系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。
执行结果	正常结束时 SD PWR 变为亮灯。 异常结束时 SD PWR 每亮起 0.5s 后熄灭 0.5s，以此频率闪烁。按下 SD 存储卡停止供电按钮则亮灯。

*1. NX 系列 CPU 单元时，请将拨动开关 5 到 8 全部设为 OFF。
*2. 未安装 SD 存储卡时， SD PWR 保持熄灯状态。

● 通过系统定义变量 _Card1BkupCmd （SD 存储卡备份指示）操作的方法

处理阶段	操作方法
开始指示	在系统定义变量 _Card1BkupCmd.DirName （目录名称）中保存保存位置的目录名称。 （例） 指定目录 dirA 下的子目录 dirB："dirA/dirB" 系统定义变量 _Card1BkupCmd.ExecBkup （备份启动标志）从 FALSE 变为 TRUE，则开始执行。
取消指示	取消备份执行指示。 将系统定义变量 _Card1BkupCmd.CancelBkup(备份取消标志）设为 TRUE 后，备份处理将异常结束。
执行中	系统定义变量 _Card1BkupSta.Active （执行中标志）变为 TRUE。 系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。
执行结果	正常结束时 系统定义变量 _Card1BkupSta.Done （完成标志）变为 TRUE。 异常结束时 系统定义变量 _Card1BkupSta.Err （错误标志）变为 TRUE。

（注）无法通过用户程序访问该系统定义变量。

● 通过 Sysmac Studio 的 SD 存储卡画面操作的方法

处理阶段	操作方法
开始指示	按下 Sysmac Studio SD 存储卡画面的 ［SD 存储卡备份］按钮，指定保存位置的目录名称，发出执行备份的指示。
执行中	显示备份进度状况画面。 系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。
执行结果	显示完成信息，接着确认执行备份数据的核对。

操作的详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

● 通过专用指令操作的方法

处理阶段	操作方法
开始指示	在用户程序中执行 BackupToMemoryCard 指令。
执行中	BackupToMemoryCard 指令的输出变量 Busy 的值变为 TRUE。 系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。
执行结果	正常结束时 BackupToMemoryCard 指令的输出变量 Done 的值变为 TRUE。 异常结束时 BackupToMemoryCard 指令的输出变量 Error 的值变为 TRUE。 BackupToMemoryCard 指令的输出变量 ErrorID 中将保存错误代码。

版本相关信息

BackupToMemoryCard 指令可在 CPU 单元版本 1.08 以上且 Sysmac Studio Ver.1.09 以上中使用。

相关系统定义变量

利用系统定义变量进行备份时，相关的系统定义变量如下。系统定义变量的详情请参考📖「A-5 系统定义变量的个别规格」（P.A-79）。

变量名称 / 成员名称	名称	功能	数据类型	R/W
_Card1BkupCmd*1	SD 存储卡备份指示		_sBKUP_CMD	RW
ExecBkup*1	备份启动标志	向 SD 存储卡备份控制器的数据时，变为 TRUE。	BOOL	RW
CancelBkup*1	备份取消标志	停止向 SD 存储卡备份时，变为 TRUE。	BOOL	RW
DirName*1	目录名	指定备份对象的 SD 存储卡目录名称。	STRING (64)	RW
_Card1BkupSta*1	SD 存储卡备份状态		_sBKUP_STA	R
Done*1	完成标志	备份执行完成时为 TRUE。	BOOL	R
Active*1	执行中标志	正在执行备份时为 TRUE。	BOOL	R
Err*1	错误标志	备份处理异常结束时为 TRUE。	BOOL	R
_BackupBusy	备份相关功能执行中标志	正在执行备份、恢复、核对时为 TRUE。	BOOL	R

*1. 无法通过用户程序访问该系统定义变量。

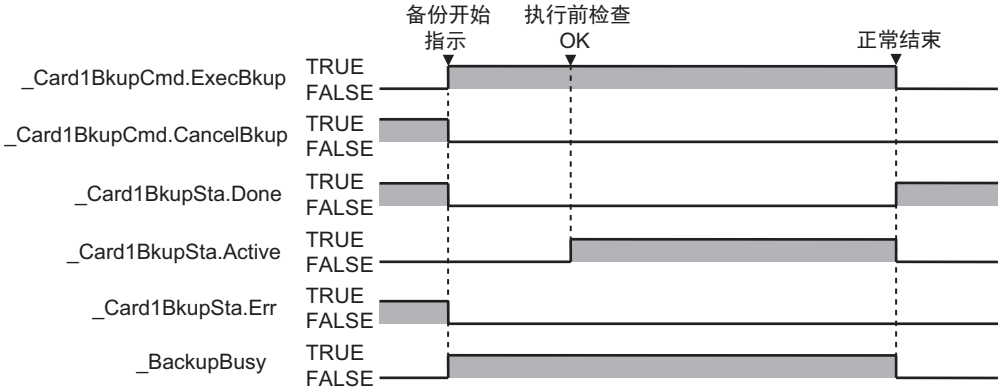
参考

- 连接NJ/NX系列控制器和NA系列显示器使用时，变量映射请参考📖《可编程终端 NA系列 用户手册 软件篇 (SBSA-CN5-546)》。
- 连接NJ/NX系列控制器和NS系列显示器使用时，将这些系统定义变量登录到CX-Designer变量表的方法请参考📖「A-9 CX-Designer 的变量表登录方法」（P.A-153）。

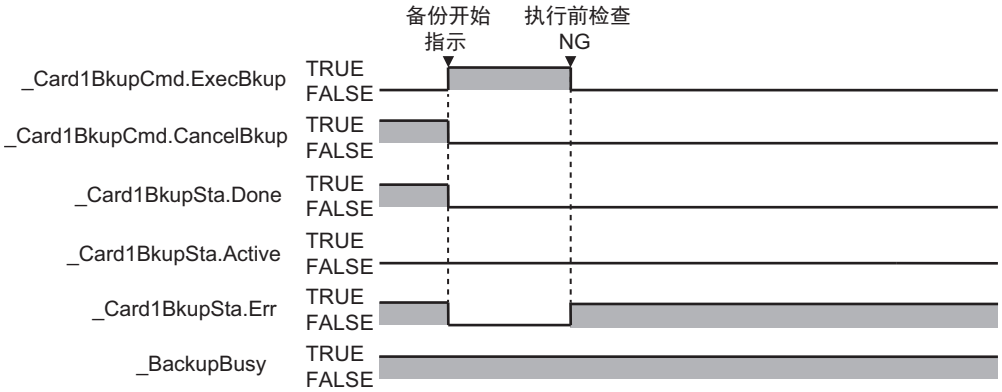
时序图

利用系统定义变量进行备份时，系统定义变量的动作如下。
图中的“执行前检查”是指在开始备份前，检查是否存在 SD 存储卡。_Card1BkupSta.Active（执行中标志）的值变为 TRUE，则表示执行前检查的结果为 OK，已实际开始备份。

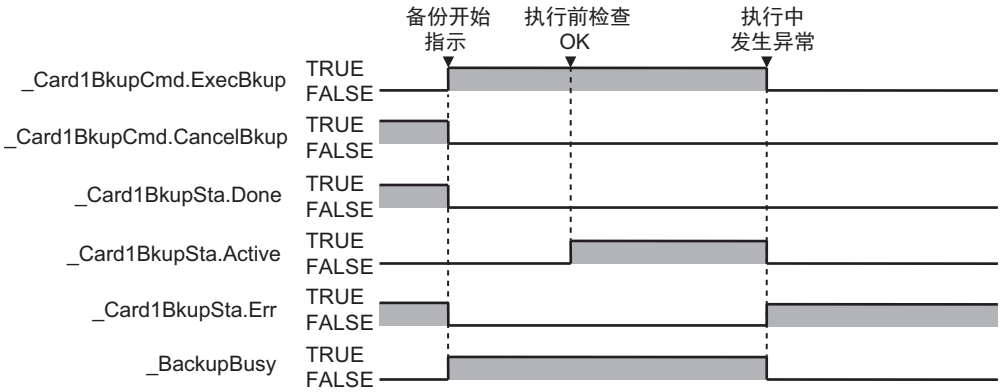
● 正常动作



● 因正在执行其他备份功能，无法开始备份时

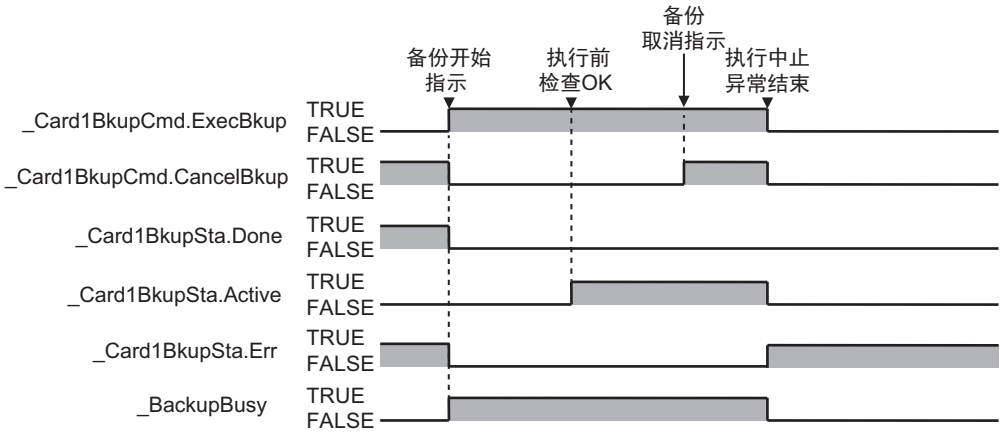


● 开始正常备份后，因某些原因导致备份失败时



● **备份执行中发出取消指示时**

从发出取消指示到停止备份为止的时间因备份处理的进度状况不同而异。



处理时间和备份文件的大小

备份所需的时间因 CPU 单元、动作模式、单元构成、用户程序的内容不同而异。此外，备份文件的大小因单元构成、用户程序的内容不同而异。以下表示备份时间和备份文件大小的参考值。

CPU 单元	动作模式	所连接的 EtherCAT 从站	所连接的 CJ 单元	用户定义的 POU 数	用户程序的存储大小 [MB]	备份时间 [s]	备份文件的大小 [MB]
NX701-□□□□	程序模式	*1	—	113	7.13	约 30	29.45
NJ501-□□□□			*2	53	2.36	约 50	20
NJ301-□□□□		*3		20	0.53	约 30	9.85
NJ101-□□□□		*4		15	0.38	约 30	9.94

*1. AC伺服驱动器 R88D-KNA-ECT:32台、数字I/O终端 GX-ID1611:32台、数字I/O终端 GX-OD1611:32台的构成。
*2. 串行通讯单元 CJ1W-SCU22:4 台、 EtherNet/IP 单元 CJ1W-EIP21:1 台的构成。
*3. AC 伺服驱动器 R88D-KNA-ECT:8 台、数字 I/O 终端 GX-ID1611:8 台、数字 I/O 终端 GX-OD1611:8 台的构成。
*4. AC 伺服驱动器 R88D-KNA-ECT:2 台、数字 I/O 终端 GX-ID1611:2 台、数字 I/O 终端 GX-OD1611:2 台的构成。

9-2-2 恢复 （从 SD 存储卡到控制器）

接通电源时，将 CPU 单元上安装的 SD 存储卡中的备份文件数据传送到控制器中。
该功能的操作只能通过 CPU 单元正面开关操作。
此外，不可指定备份相关文件的读取源目录。要恢复的备份文件需要事先保存到 SD 存储卡的根目录下。

处理内容

- 将 SD 存储卡的根目录下保存的备份文件数据传送到控制器中。
- 恢复对象数据为根目录下保存的恢复指令文件（RestoreCo m mand.ini）中指定为 " 恢复 " 的数据组。恢复指令文件的详情请参考📖「9-12-3 恢复指令文件的规格」（P.9-54）。
- SD 存储卡的根目录下不存在恢复指令文件时，将恢复根目录下存在的所有可恢复的备份文件数据。
- 执行完成后，动作模式变为 " 程序模式 "，不可以这样的状态开始运行。要开始运行，需要将拨动开关全部设为 OFF，然后重新接通控制器的电源，或执行 " 控制器重置 "。
- 恢复执行完成后，请重新接通所有 EtherCAT 从站的电源。
- 正在执行恢复时， CPU 单元的状态为 " 启动中 "。
- 在恢复处理的开始前检查中发生错误时，控制器中的数据将保持恢复前的数据。
- 正在执行恢复处理时如果控制器发生断电，将发生"用户程序/控制器配置和设定传送错误（全部停止故障等级的控制器异常） "。此时，控制器中的数据将变得不确定，请按以下方法排除异常。
  - 重新执行恢复。
  - 通过 Sysmac Studio 执行存储器完全清除、项目的下载。
- 备份文件的单元和从站构成与恢复对象的实际单元和从站构成不同时，若执行恢复，将发生恢复执行错误。
- 有保持属性的变量当前值不是恢复对象时，该变量的当前值保持恢复前的值。但是，不满足维持变量本身的条件时，变量的值将被初始化。维持变量本身的条件如下所示。
  - 恢复前的变量和要恢复的变量的变量名称、数据类型名称、数据类型大小应全部相同。
- 恢复完成后， SD 存储卡将保持安装状态。
- 恢复完成后， CPU 单元的写入保护状态遵照 " 电源接通时的写入保护 " 设定。

操作方法

● 通过 CPU 单元正面开关操作的方法

处理阶段	操作步骤
开始指示	将拨动开关 1 到 4*1 设定为 1:OFF,2:OFF,3:ON,4:ON，接通控制器的电源。
执行中	恢复执行中 SD PWR 每亮起 3s 后熄灭 0.5s， RUN 每亮起 0.5s 后熄灭 0.5s，分别以此频率闪烁。 SD BUSY 闪烁 （闪烁模式不确定）。
执行结果	正常结束时 SD PWR 变为亮灯。 异常结束时 SD PWR 每亮起 0.5s 后熄灭 0.5s，以此频率闪烁。按下 SD 存储卡停止供电按钮则亮灯。*2

*1. NX 系列 CPU 单元时，请将拨动开关 5 到 8 全部设为 OFF。
*2. 未安装 SD 存储卡时， SD PWR 保持熄灯状态。

处理时间

恢复所需的时间因 CPU 单元、单元构成、用户程序的内容不同而异。以下表示恢复时间的参考值。

CPU 单元	所连接的 EtherCAT 从站	所连接的 CJ 单元	用户定义的 POU 数	用户程序的存储大小 [MB]	恢复时间 [s]
NX701- □□□□	*1	—	113	7.13	约 50
NJ501- □□□□		*2	53	2.36	约 100
NJ301- □□□□	*3		20	0.53	约 70
NJ101- □□□□	*4		15	0.38	约 70

*1. AC 伺服驱动器 R88D-KNA-ECT:32 台、数字 I/O 终端 GX-ID1611:32 台、数字 I/O 终端 GX-OD1611:32 台的构成。
*2. 串行通讯单元 CJ1W-SCU22:4 台、 EtherNet/IP 单元 CJ1W-EIP21:1 台的构成。
*3. AC 伺服驱动器 R88D-KNA-ECT:8 台、数字 I/O 终端 GX-ID1611:8 台、数字 I/O 终端 GX-OD1611:8 台的构成。
*4. AC 伺服驱动器 R88D-KNA-ECT:2 台、数字 I/O 终端 GX-ID1611:2 台、数字 I/O 终端 GX-OD1611:2 台的构成。

9-2-3 核对 （控制器和 SD 存储卡之间）

对控制器的数据和 CPU 单元中安装的 SD 存储卡的备份文件数据进行核对。

处理内容

- 对控制器的数据和 SD 存储卡的指定目录下保存的备份文件数据进行核对。
- 核对对象数据为恢复指令文件 （RestoreCo m mand.ini）中指定为 " 恢复 " 的数据组。
- 变量的当前值、CJ 单元用存储器的当前值、绝对编码器原点位置偏置可能在执行核对的过程中发生替换，因此不核对。
- 执行核对后，将在指定目录下创建 " 核对结果文件 （VerifyResult.log） "，并保存核对结果。执行核对前，指定目录下已经存核对结果文件时，将被覆盖。但是， SD 存储卡有写保护时，不创建核对结果文件。
- SD 存储卡的指定目录下不存在恢复指令文件时，将核对指定目录下存在的所有可核对的备份文件数据。
- 备份文件的单元和从站构成与核对对象的实际单元和从站构成不同时，将发生核对错误。
- 核对完成后， SD 存储卡将保持安装状态。

操作方法

● 通过 CPU 单元正面开关操作的方法

处理阶段	操作步骤
开始指示	将拨动开关 1 到 4[*1] 分别设定为 1:OFF,2:OFF,3:OFF,4:OFF，并长按 SD 存储卡停止供电按钮 3 秒，将开始执行。
执行中	刚开始执行后[*2] SD PWR 将以亮灯➞熄灭 0.5s ➞亮灯的模式闪烁 1 次。 核对执行中 SD PWR 每亮起 3s 后熄灭 0.5s，以此频率闪烁。 SD BUSY 闪烁 （闪烁模式不确定）。 系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。
执行结果	正常结束，核对一致时 SD PWR 变为亮灯。 正常结束，核对不一致时 SD PWR 每亮起 0.5s 后熄灭 0.5s，以此频率闪烁。按下 SD 存储卡停止供电按钮则亮灯。 异常结束时 SD PWR 每亮起 0.5s 后熄灭 0.5s，以此频率闪烁。按下 SD 存储卡停止供电按钮则亮灯。

*1. NX 系列 CPU 单元时，请将拨动开关 5 到 8 全部设为 OFF。
*2. 未安装 SD 存储卡时， SD PWR 保持熄灯状态。

● 系统定义变量 _Card1BkupCmd （SD 存储卡备份指示）

处理阶段	操作步骤
开始指示	在系统定义变量 _Card1BkupCmd.DirName （目录名称）中保存保存位置的目录名称。 （例） 指定目录 dirA 下的子目录 dirB："dirA/dirB" 系统定义变量 _Card1BkupCmd.ExecVefy （核对启动标志）从 FALSE 变为 TRUE，则开始执行。
取消指示	取消核对执行指示。 将系统定义变量 _Card1BkupCmd.CancelVefy(核对取消标志）设为 TRUE，核对处理将异常结束。
执行中	系统定义变量 _Card1VefySta.Active （执行中标志）变为 TRUE。 系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。
执行结果	正常结束，核对一致时 系统定义变量 _Card1BkupSta.Done （完成标志）和 _Card1BkupSta.VefyRslt （核对结果标志）变为 TRUE。 正常结束，核对不一致时 系统定义变量 _Card1BkupSta.Done （完成标志）变为 TRUE， _Card1BkupSta.VefyRslt （核对结果标志）变为 FALSE。 异常结束时 系统定义变量 _Card1BkupSta.Err （错误标志）变为 TRUE。

（注） 无法通过用户程序访问该系统定义变量。

● 通过 Sysmac Studio 的 SD 存储卡画面操作的方法

处理阶段	处理步骤
开始指示	按下 Sysmac Studio SD 存储卡画面的 ［SD 存储卡核对］按钮，指定核对目标保存位置的目录名称，发出执行核对的指示。
执行中	显示核对进度状况画面。系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。 SD PWR 每亮起 3s 后熄灭 0.5s，以此频率闪烁。 SD BUSY 闪烁 （闪烁模式不确定）。
执行结果	显示核对结果画面。

操作的详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

相关系统定义变量

利用系统定义变量进行核对时，相关的系统定义变量如下。系统定义变量的详情请参考📖「A-5 系统定义变量的个别规格」（P.A-79）。

变量名称 成员名称	名称	功能	数据类型	R/W
_Card1BkupCmd*1	SD 存储卡 备份指示		_sBKUP_ CMD	RW
ExecVefy*1	核对启动标志	核对控制器和 SD 存储卡中的备份文件时，变为 TRUE。	BOOL	RW
CancelVefy*1	核对取消标志	停止核对控制器和 SD 存储卡中的备份文件时，变为 TRUE。	BOOL	RW
DirName*1	目录名	指定备份对象的 SD 存储卡目录名称。	STRING (64)	RW
_Card1VefySta*1	SD 存储卡 核对状态		_sVEFY_ STA	R
Done*1	完成标志	核对执行完成时为 TRUE。	BOOL	R
Active*1	执行中标志	正在执行核对时为 TRUE。	BOOL	R
VefyRslt*1	核对结果标志	核对一致时为 TRUE，核对不一致时为 FALSE。	BOOL	R
Err*1	错误标志	核对处理异常结束时为 TRUE。	BOOL	R
_BackupBusy	备份相关功能执行中标志	正在执行备份、恢复、核对时为 TRUE。	BOOL	R

*1. 无法通过用户程序访问该系统定义变量。

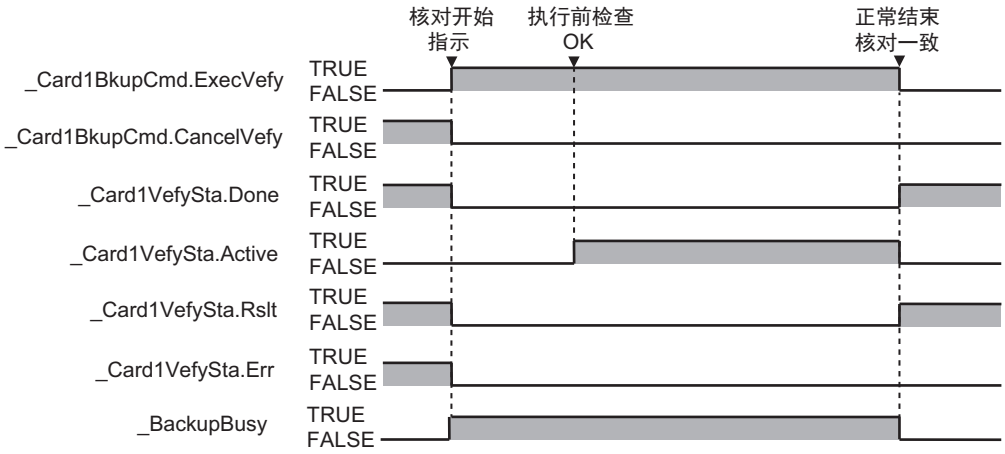
参考

- 连接NJ/NX系列控制器和NA系列显示器使用时，变量映射请参考📖《可编程终端 NA系列 用户手册 软件篇 (SBSA-CN5-546)》。
- 连接NJ/NX系列控制器和NS系列显示器使用时，将这些系统定义变量登录到CX-Designer变量表的方法请参考📖「A-9 CX-Designer 的变量表登录方法」（P.A-153）。

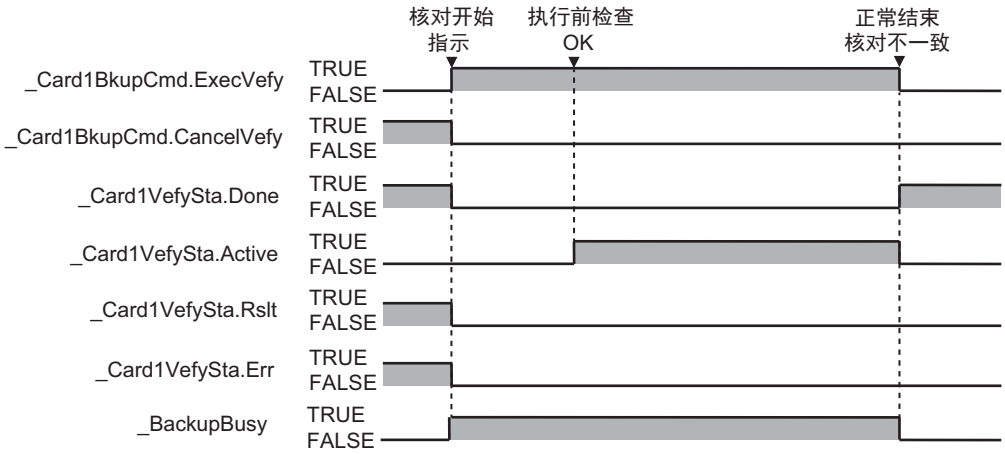
时序图

利用系统定义变量进行核对时，系统定义变量的动作如下。
图中的 " 执行前检查 " 是指在开始核对前，检查是否存在 SD 存储卡。_Card1VefySta.Active（执行中标志）的值变为 TRUE，则表示执行前检查的结果为 OK，已实际开始核对。

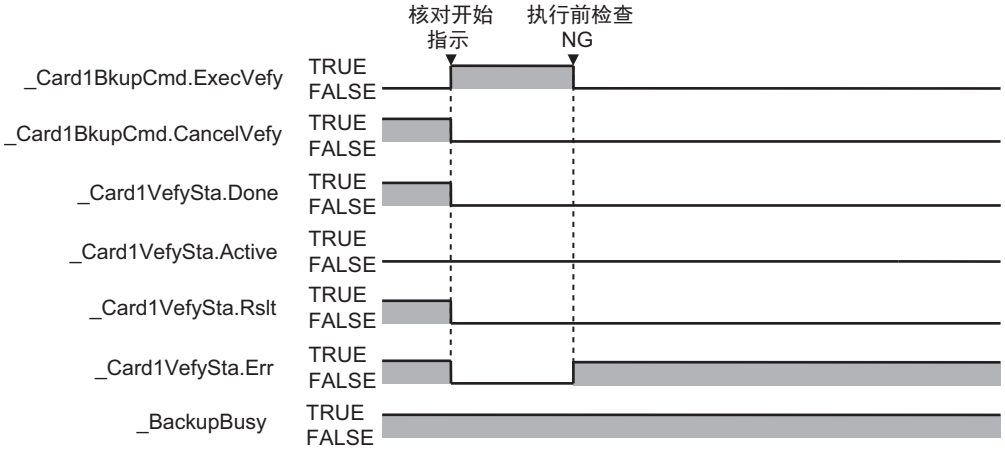
● 正常动作，核对一致时



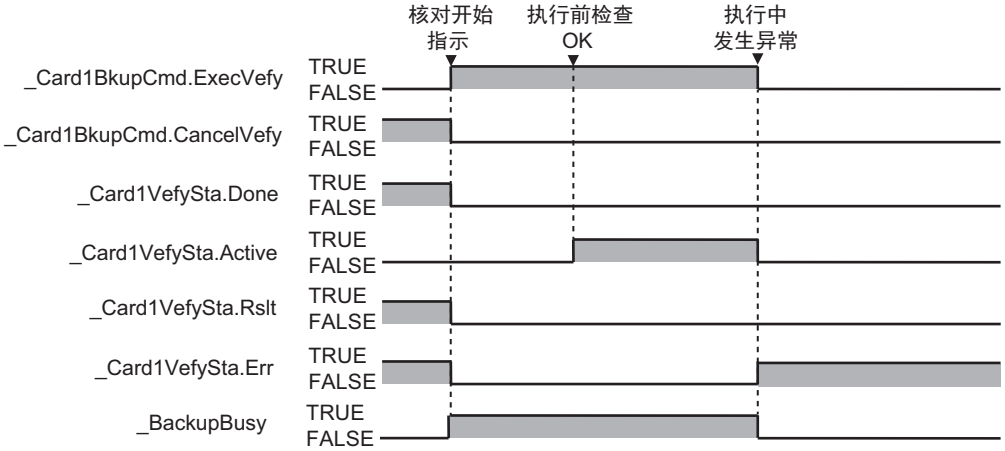
● 正常动作，核对不一致时



● **因正在执行其他备份功能，无法开始核对时**

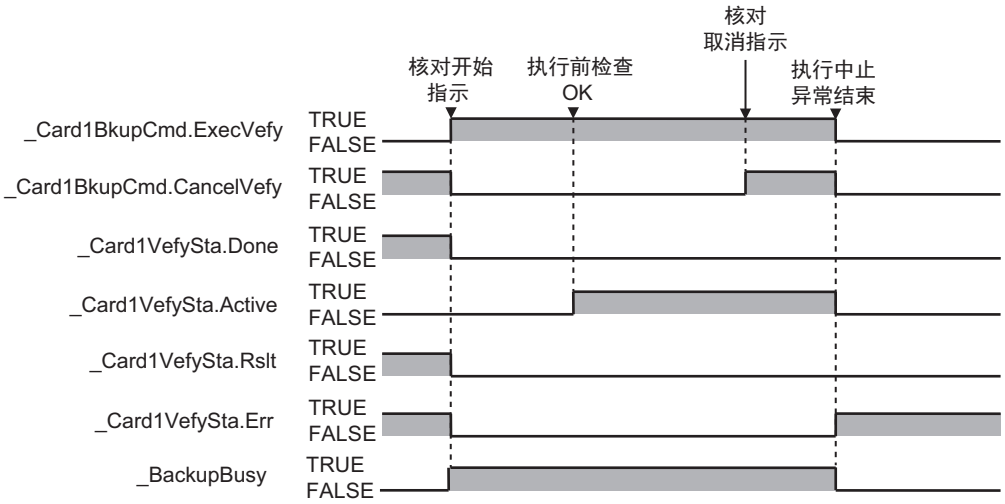


● **开始正常核对后，因某些原因导致核对失败时**



● **核对执行中发出取消指示时**

从发出取消指示到停止核对为止的时间因核对处理的进度状况不同而异。



9-3 SD 存储卡备份禁止功能

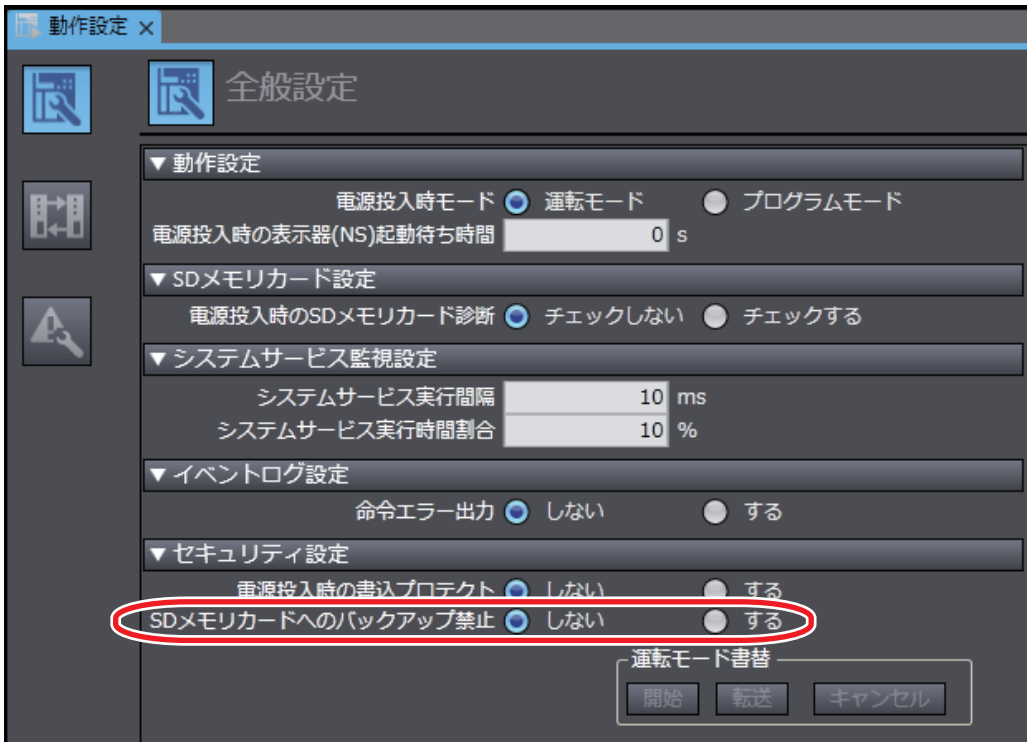
禁止向 SD 存储卡备份的功能。使用本功能，可保护用户资产信息。

SD 存储卡备份禁止功能的对象为以下三个。

- 通过 CPU 单元正面开关进行备份
- 通过系统定义变量进行备份
- 通过 Sysmac Studio 的 SD 存储卡画面进行备份

通过 BackupToMemoryCard 指令的备份不在备份禁止功能的对象范围内。即，即使设定为 SD 存储卡备份禁止，也可通过 BackupToMemoryCard 指令执行备份。

SD 存储卡备份禁止的操作方法如下。
在 Sysmac Studio 的 “ 配置和设定 ” | “ 控制器设定 ” | [动作设定] 的整体设定选项卡中，将 SD 存储卡的备份禁止设定为 [是]。

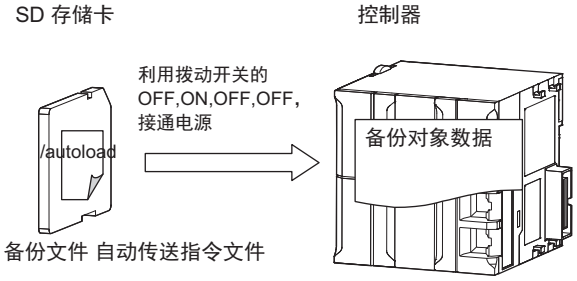


版本相关信息

BackupToMemoryCard 指令可在 CPU 单元版本 1.08 以上且 Sysmac Studio Ver.1.09 以上中使用。

9-4 SD 存储卡自动传送功能

接通电源时，将 CPU 单元上安装的 SD 存储卡的 “/autoload” 目录下保存的备份文件数据自动传送到控制器的功能。
要以 SD 存储卡中的备份文件数据运行 CPU 单元时使用。
该功能的操作只能通过 CPU 单元正面开关操作。



使用通过备份功能创建的 “ 备份文件 ” 和 “ 自动传送指令文件 ”。事先将这两个文件保存到 “/autoload” 目录下。

文件	作用
备份文件	用备份相关功能中处理的控制器数据制作的文件。
自动传送指令文件	执行自动传送时，用于指定传送对象数据组的文件。在计算机的文本编辑器中编辑文件，可指定传送对象。

本功能的操作方法、对象目录及执行时序如下。

操作方法	目录	执行时序
CPU 单元正面开关	SD 存储卡的 /autoload 目录	电源接通时

处理内容

- 接通电源时，将 SD 存储卡的目录 “/autoload” 下保存的备份文件数据自动传送到控制器中。
- 自动传送对象数据为 “/autoload” 中保存的自动传送指令文件（AutoloadCo m mand.ini）中指定的数据组。自动传送指令文件的详情请参考「9-12-4 自动传送指令文件的规格」（P.9-56）。
- SD 存储卡的 “/autoload” 中不存在自动传送指令文件时，将传送 “/autoload” 中存在的所有可传送的备份文件数据。
- 自动传送完成后的动作模式遵照 [控制器设定]-[接通电源时的模式] 的设定。
- 正在执行自动传送时， CPU 单元的状态为 “ 启动中 ”。
- 在自动传送处理的开始前检查中发生错误时，控制器中的数据将保持传送前的数据。
- 正在执行自动传送处理时如果控制器发生断电，将发生 “用户程序/控制器配置和设定传送错误（全部停止故障等级的控制器异常） ”。此时，控制器中的数据将变得不确定，请按以下方法排除异常。
  - 重新执行自动传送。
  - 通过 Sysmac Studio 执行存储器完全清除、项目的下载。
- 自动传送对象以外的数据保持当前值。
- 有保持属性的变量当前值不是传送对象时，该变量的当前值保持传送前的值。但是，不满足维持变量本身的条件时，变量的值将被初始化。维持变量本身的条件如下所示。
  - 传送前的变量和要传送的变量的变量名称、数据类型名称、数据类型大小应全部相同。
- 自动传送完成后， SD 存储卡将保持安装状态。
- 自动传送完成后， CPU 单元的写入保护状态遵照 “ 电源接通时的写入保护 ” 设定。

操作方法

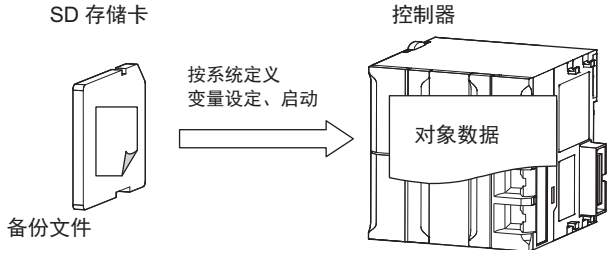
● 通过 CPU 单元正面开关操作的方法

处理阶段	操作步骤
开始指示	将拨动开关 1 到 4*1 设定为 1:OFF、 2:ON、 3:OFF、 4:OFF，接通控制器的电源。
执行中	传送执行中 SD PWR 每亮起 3s 后熄灭 0.5s， RUN 每亮起 0.5s 后熄灭 0.5s，分别以此频率闪烁。 SD BUSY 闪烁。闪烁模式不确定。
执行结果	正常结束时 SD PWR 亮灯。 完成后的动作模式遵照 [控制器设定]-[接通电源时的模式] 的设定。 异常结束时 RUN 熄灭、 ERR 亮灯，并发生全部停止故障等级的控制器异常。 SD PWR 亮灯。*2

*1. NX 系列 CPU 单元时，请将拨动开关 5 到 8 全部设为 OFF。
*2. 未安装 SD 存储卡时， SD PWR 保持熄灯状态。

9-5 SD 存储卡程序传送功能

根据系统定义变量 _Card1PrgTransferCmd （SD 存储卡程序传送指示），将 CPU 单元中安装的 SD 存储卡中保存的程序传送到控制器的功能。
在控制器设定中，设定是否使用该功能，并设定密码。
要通过显示器的操作等，以 SD 存储卡中的备份文件程序运行 CPU 单元时使用。



使用通过备份功能创建的 “ 备份文件 ”。事先将该文件保存到 SD 存储卡的任意目录下。
在系统定义变量中指定保存了 “ 备份文件 ” 的目录。

文件	作用
备份文件	用备份相关功能中处理的控制器数据制作的文件。

本功能的操作方法、对象目录及可执行的动作模式如下。

操作方法	目录*1	可执行的动作模式
系统定义变量	系统定义变量中指定的目录	运行模式、程序模式

*1. 仅可在 SD 存储卡内指定。

使用注意事项

- 若通过本功能传送接通电源时的动作模式为运行模式的程序，无论运行前的状态或设定如何，传送完成后都将切换为运行模式。确认装置开始运行且没有问题后，再执行本功能。
- 为了防止传送计划外的程序，请设计为每次都需要输入密码。
- 执行本功能后，将自动重置控制器。控制器重置过程中的输出遵照从站或单元的规格。同时，控制器重置过程中，无法从控制器外部访问控制器中的变量。
- 使用 EtherCAT 从站或执行运动控制时，程序传送完成后，可能发生 EtherCAT 主机功能模块、运动控制功能模块异常。发生异常时，请在程序传送完成后解除异常。通过用户程序解除异常时，请参考📖《NJ/NX 系列 指令基准手册 基本篇（SBCA-CN5-360）》的 ResetMCError 指令的示例程序。
- 因程序传送而删除了传送前参照的控制器上的变量时，在某些显示器上，可能发生无法解决所删除变量的异常，无法访问系统定义变量。请事先确认在所用的显示器上不会发生异常。例如，NS 系列显示器时，如果标签核对的结果为不一致，将显示标签核对结果一览，无法跳转到其他画面。
- 执行本功能的过程中发生断电时，下次接通电源时，可能发生 “ 用户程序 / 控制器构成・设定传送错误 ” （事件代码 10200000Hex）等异常。

- 传送了控制器设定的［通过系统定义变量进行程序传送］设定为［不使用］的备份文件时，之后将无法执行程序传送。要重复执行时，请传送设定为 [使用] 的备份文件。
- 程序传送过程中，无法执行其他备份功能。

处理内容

- 根据系统定义变量 _Card1PrgTransferCmd（SD 存储卡程序传送指示）的操作，将 CPU 单元中安装的 SD 存储卡中保存的程序传送到控制器中。
- 程序传送的对象备份文件为系统定义变量中指定目录下保存的文件。对象备份文件需要事先保存到 SD 存储卡的任意目录下。
- 程序传送的对象数据为系统定义变量中指定的数据组。相关系统定义变量的详情请参考📖「相关系统定义变量」（P.9-31）。
- 控制器设定的［SD 存储卡程序传送设定］-［密码］中设定的密码和系统定义变量 _Card1PrgTransferCmd.Password 中设定的密码不一致时，将发生 “SD 存储卡程序传送开始失败 ”。
- 开始程序传送后，系统定义变量 _Card1PrgTransferCmd.Password 中设定的密码将初始化。
- 执行程序传送的过程中，将自动重置控制器。控制器重置过程中的输出遵照从站或单元的规格。
- 重置控制器后，CPU 单元的状态为“启动中”。完成程序传送后，CPU 单元的状态变为“正常运行中”，按照通过程序传送功能传送的［控制器设定］的［接通电源时的模式］中设定的动作模式动作。
  但是， CPU 单元的正面开关为安全模式时，以程序模式动作。
- 在程序传送处理的开始前检查或执行前检查中发生错误时，控制器中的数据将保持传送前的数据。
- 正在执行程序传送处理时如果控制器发生断电，将发生“用户程序/控制器配置和设定传送错误（全部停止故障等级的控制器异常）”。此时，控制器中的数据将变得不确定，请按以下方法排除异常。
  - 重新执行 SD 存储卡程序传送。
  - 执行 SD 存储卡自动传送功能或 SD 存储卡备份功能的恢复。
  - 通过 Sysmac Studio 执行存储器完全清除、项目的下载。
- 程序传送对象以外的数据保持当前值。
- 有保持属性的变量当前值不是传送对象时，该变量的当前值保持传送前的值。但是，不满足维持变量本身的条件时，变量的值将被初始化。维持变量本身的条件如下所示。
  - 传送前的变量和要传送的变量的变量名称、数据类型名称、数据类型大小应全部相同。
- 正在执行程序传送时，即使按下 SD 存储卡停止供电按钮，也无效。
- 程序传送完成后， SD 存储卡将保持安装状态。
- 程序传送完成后， CPU 单元的写入保护状态遵照 “ 电源接通时的写入保护 ” 设定。

操作方法

● 系统定义变量 _Card1PrgTransferCmd （SD 存储卡程序传送指示）

处理阶段	操作步骤
开始前的准备	使用 SD 存储卡程序传送功能时，应将控制器设定的 ［通过系统定义变量进行程序传送］设定为 ［使用］。*1
开始指示	在系统定义变量 _Card1PrgTransferCmd.DirName （目录名称）中保存备份文件的保存目录名称。 （例） 指定目录 dirA 下的子目录 dirB："dirA/dirB" 在系统定义变量 _Card1PrgTransferCmd.Password （密码）中保存密码。*2 传送对象为内置 EtherNet/IP 端口设定时，将系统定义变量 _Card1PrgTransferCmd.TargetIPAdr （IP 地址的传送标志）设为 TRUE。*3 传送对象为变量的当前值 （仅保持属性）时，将系统定义变量 _Card1PrgTransferCmd.TargetVariable （变量当前值 （仅保持属性）的传送标志）设为 TRUE。 传送对象为 CJ 单元用存储器的当前值 （仅保持存储器）时，将系统定义变量 _Card1PrgTransferCmd.TargetMemory （CJ 单元用存储器当前值 （仅保持存储器）的传送标志）设为 TRUE。 系统定义变量 _Card1PrgTransferCmd.Exec （程序传送启动标志）从 FALSE 变为 TRUE，则开始执行。
执行中	系统定义变量 _BackupBusy （备份相关功能执行中标志）变为 TRUE。 系统定义变量 _Card1PrgTransferSta.Active （执行中标志）变为 TRUE。 程序传送执行中 SD PWR 每亮起 3s 后熄灭 0.5s，以此频率闪烁。 SD BUSY 闪烁 （闪烁模式不确定）。 RUN 每亮起 0.5s 后熄灭 0.5s，以此频率闪烁。 执行程序传送的过程中，将自动重置控制器。
执行结果	正常结束时 SD PWR 亮灯。 完成后的动作模式遵照 [控制器设定]-[接通电源时的模式] 的设定。 系统定义变量 _Card1PrgTransferSta.Done （完成标志）变为 TRUE。 开始前检查中异常结束时 SD PWR 亮灯。*4 RUN 和 ERR 为开始指示前的状态。 系统定义变量 _Card1PrgTransferSta.Err （错误标志）变为 TRUE。 开始前检查中异常结束时，不发生控制器重置。 执行前检查或执行过程中异常结束时 SD PWR 亮灯。 *1 RUN 熄灭、 ERR 亮灯，并发生全部停止故障等级的控制器异常。 系统定义变量 _Card1PrgTransferSta.Err （错误标志）变为 TRUE。

*1. 传送了控制器设定的 ［通过系统定义变量进行程序传送］设定为 ［不使用］的备份文件时，之后将无法执行程序传送。要重复执行时，请传送设定为 [使用] 的备份文件。
*2. 开始 SD 存储卡程序传送后，密码将初始化。请在每次开始时保存密码。
控制器设定的 ［SD 存储卡程序传送设定］ -［密码］中未设定密码时，如果系统定义变量 _Card1PrgTransfer-Cmd.Password 的值为初始值，将开始程序传送。如果不是初始值，不开始传送。
*3. IP 地址中包括设定种类、 IP 地址、子网掩码、默认网关。
*4. 未安装 SD 存储卡时， SD PWR 保持熄灯状态。

相关系统定义变量

相关的系统定义变量如下。系统定义变量的详情请参考📖「A-5 系统定义变量的个别规格」（P.A-79）。

变量名称 / 成员名称	名称	功能	数据类型	R/W
_Card1PrgTransferCmd	SD 存储卡程序传送指示		_sPRGTRANSFER_CMD	RW
Exec	程序传送启动标志	利用 SD 存储卡程序传送功能，向控制器传送 SD 存储卡中的备份文件数据时，变为 TRUE。	BOOL	RW
DirName	目录名	指定备份对象的 SD 存储卡目录名称。	STRING(64)	RW
Password	密码	指定开始传送程序时要核对的密码。 每次开始 SD 存储卡程序传送时，密码将初始化。	STRING(33)	RW
TargetUserProgram	用户程序、设定的传送标志*1	程序传送时的传送对象为用户程序或设定时，变为 TRUE。 在 SD 存储卡程序传送功能中，请务必设定为 TRUE。	BOOL	RW
TargetIPAdr	IP 地址的传送标志	程序传送时的传送对象为内置 EtherNet/IP 端口的 IP 地址时，变为 TRUE。 IP 地址中包括设定种类、 IP 地址、子网掩码、默认网关。	BOOL	RW
TargetVariable	变量当前值 （仅保持属性）的传送标志	程序传送时的传送对象为变量的当前值 （仅保持属性）时，变为 TRUE。	BOOL	RW
TargetMemory	CJ 单元用存储器当前值 （仅保持存储器）的传送标志	程序传送时的传送对象为 CJ 单元用存储器的当前值 （仅保持存储器）时，变为 TRUE。	BOOL	RW
_Card1PrgTransferSta	SD 存储卡程序传送状态		_sPRGTRANSFER_STA	R
Done	完成标志	程序传送执行完成时为 TRUE。	BOOL	R
Active	执行中标志	正在执行程序传送时为 TRUE。	BOOL	R
Err	错误标志	程序传送异常结束时为 TRUE。	BOOL	R

*1. 用户程序、设定数据组的数据为设定对象。但是，内置 EtherNet/IP 端口的 IP 地址用 IP 地址的传送标志设定。

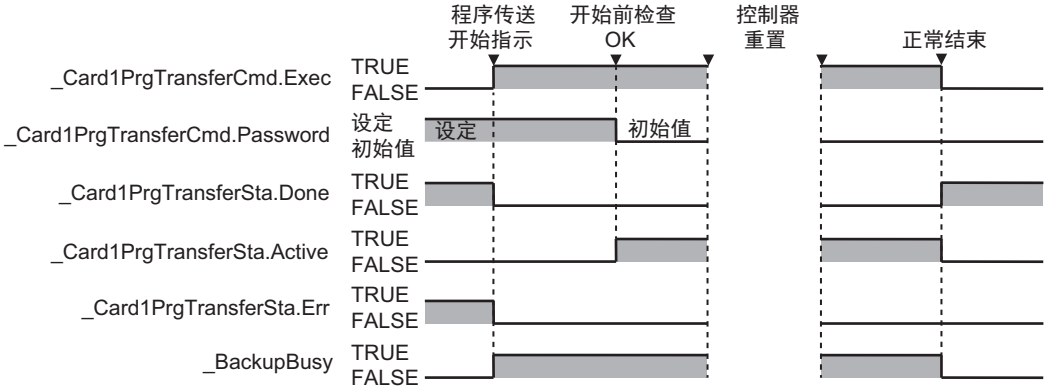
参考

- 连接NJ/NX系列控制器和NA系列显示器使用时，变量映射请参考📖《可编程终端 NA系列 用户手册 软件篇 (SBSA-CN5-546)》。
- 连接NJ/NX系列控制器和NS系列显示器使用时，将这些系统定义变量登录到CX-Designer变量表的方法请参考📖「A-9 CX-Designer 的变量表登录方法」（P.A-153）。

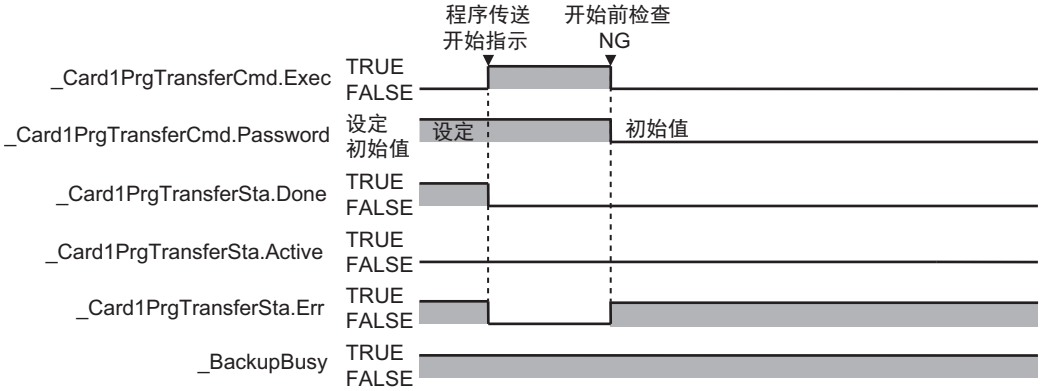
时序图

利用系统定义变量进行 SD 存储卡传送时，系统定义变量的动作如下。
图中的 " 开始前检查 " 是指在开始程序传送前，检查密码是否一致。 _Card1PrgTransferSta.Active （执行中标志）的值变为 TRUE，则表示开始前检查的结果为 OK。

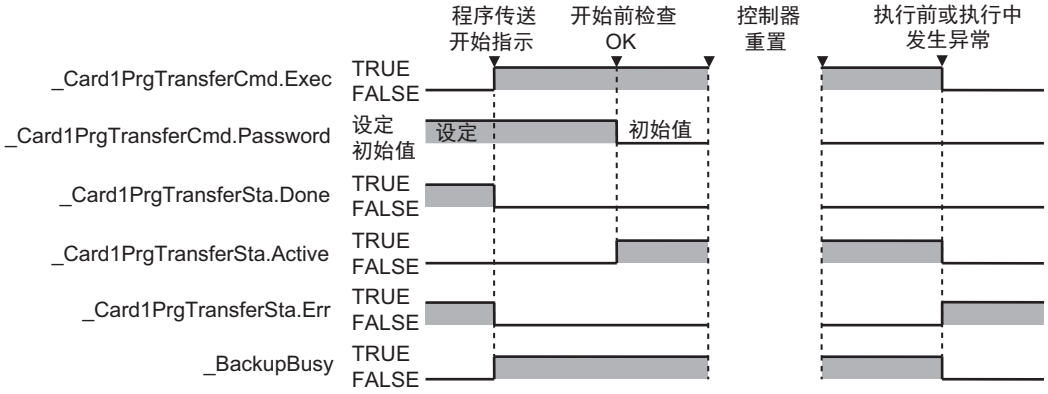
● 正常动作



● 因正在执行其他备份功能，无法开始程序传送时

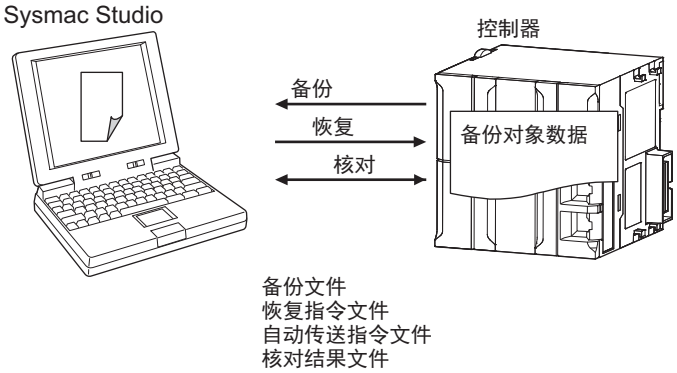


● **正常开始程序传送后，因某些原因导致程序传送失败时**



9-6 Sysmac Studio 控制器备份功能

使用 Sysmac Studio，通过计算机进行控制器数据的备份、恢复、核对的功能。



执行备份后，将在计算机的指定目录下创建“备份文件”、“恢复指令文件”、“自动传送指令文件”。备份相关文件的作用如下。

文件	作用			
	内容	备份	恢复	核对
备份文件	用备份相关功能中处理的控制器数据制作的文件。	创建	参考	参考
恢复指令文件	执行恢复时，用于指定传送对象数据组的文件。在计算机的文本编辑器中编辑文件，可指定传送对象。	创建	参考	参考
自动传送指令文件	执行 SD 存储卡自动传送时，用于指定传送对象数据组的文件。在计算机的文本编辑器中编辑文件，可指定传送对象。	创建	无操作	无操作

本功能的可执行动作模式如下。

处理	可执行的动作模式
备份	运行模式、程序模式
恢复	程序模式
核对	运行模式、程序模式

参考

正在执行备份或核对时，可变更 CPU 单元的动作模式。
但是，如 CPU 单元的存储器发生故障，无法正常执行备份或核对时，将发生错误。

9-6-1 备份 （从控制器到计算机）

将控制器的数据保存到计算机的指定目录中。

处理内容

- 备份数据中的单元和从站设定只能选择所有连接的 EtherCAT 从站及所有连接的 CJ 单元。
- 执行备份后，将在计算机的指定目录下创建 “ 备份文件 ”、“ 恢复指令文件 ”、“ 自动传送指令文件 ”。
- 指定目录下已经存在备份相关文件时，将被覆盖。
- 正在向指定目录写入备份相关文件时如果发生错误，已有的备份相关文件将被删除。然后，也不会创建新的备份相关文件。
- 创建新的备份相关文件前如果发生错误，将保持已有的文件，不创建新文件。
- 执行过程中，系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。

（注）可选择 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

操作方法

1 **执行 Sysmac Studio 的 [工具] ｜ [备份] ｜ [控制器的备份]。**

2 **指定创建备份文件、恢复指令文件、自动传送指令文件的目录。**

3 **在备份确认画面中按下 [执行] 按钮。**
将执行备份，并创建备份文件、恢复指令文件、自动传送指令文件。

9-6-2 恢复 （从计算机到控制器）

将计算机的指定目录下的备份文件数据恢复到控制器中。
仅可在程序模式下执行。

处理内容

- 将计算机的指定目录下的备份文件数据恢复到控制器中。
- 恢复对象数据组可在 Sysmac Studio 上选择。恢复所需的条件如下表所示。

数据组	恢复条件
用户程序设定	选择 “CPU 单元 ” 时
变量的当前值	选择 “ 保持属性的变量当前值 ” 时
CJ 单元用存储器的当前值	选择 “ 保持属性的 CJ 单元用存储器的当前值 ” 时*1
单元和从站设定	选择 “CJ 单元 ” 时 *1、选择 “EtherCAT 从站 ” 时
绝对编码器原点位置偏置	选择 “ 绝对编码器原点位置偏置 ” 时

*1. 可选择 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

- 在恢复处理的开始前检查中发生错误时，控制器中的数据将保持恢复前的数据。
- 正在执行恢复处理时如果控制器发生断电，将发生“用户程序/控制器配置和设定传送错误（全部停止故障等级的控制器异常） ”。此时，控制器中的数据将变得不确定，请按以下方法排除异常。
  - 重新执行恢复。
  - 通过 Sysmac Studio 执行存储器完全清除、项目的下载。
- 有保持属性的变量当前值不是恢复对象时，该变量的当前值保持恢复前的值。但是，不满足维持变量本身的条件时，变量的值将被初始化。维持变量本身的条件如下所示。
  - 恢复前的变量和要恢复的变量的变量名称、数据类型名称、数据类型大小应全部相同。
- 恢复执行完成后，请重新接通所有 EtherCAT 从站的电源。

操作方法

1 **执行 Sysmac Studio 的 [工具] ｜ [备份] ｜ [控制器的恢复]。**

2 **指定保存有备份文件、恢复指令文件的目录。**

3 **在恢复确认画面中按下 [执行] 按钮。**
执行恢复。

9-6-3 核对（控制器和计算机之间）

对控制器的数据和计算机中指定目录下的备份文件数据进行核对。

处理内容

- 对控制器的数据和计算机中指定目录下的备份文件数据进行核对。核对对象数据组可在Sysmac Studio上选择。核对所需的条件如下表所示。如果批量指定，将对指定的所有数据进行核对。

数据组	核对条件
用户程序设定	选择 “CPU 单元 ” 时
单元和从站设定	选择 “CJ 单元 ” 时*1、选择 “EtherCAT 从站 ” 时

*1. 可选择 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

- 核对结果将显示在 Sysmac Studio 的画面中。
- 执行过程中，系统定义变量 _BackupBusy （备份相关功能执行中标志）的值变为 TRUE。

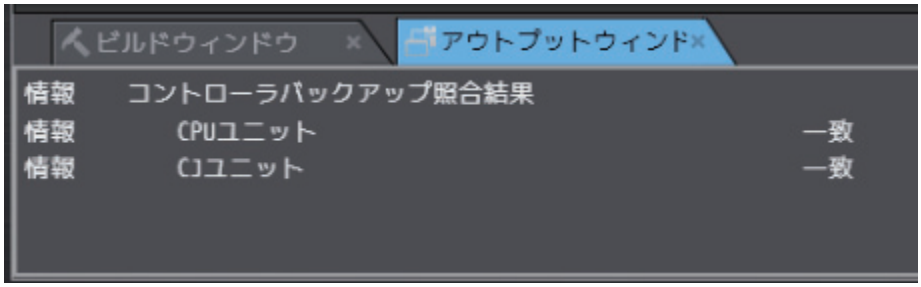
操作方法

1 **执行 Sysmac Studio 的 [工具] | [备份] | [备份文件的核对]。**

2 **指定保存有备份文件的目录。**

3 **在核对确认画面中按下 [执行] 按钮。**

将指定核对，并将核对结果文件保存到保存有备份文件的目录中。或者在输出窗口中显示核对结果。



9-7 Sysmac Studio 备份文件的导入 / 导出功能

不使用控制器，通过 Sysmac Studio 的项目，在计算机的指定目录下创建和读取备份文件的功能。
对象数据如下所示。

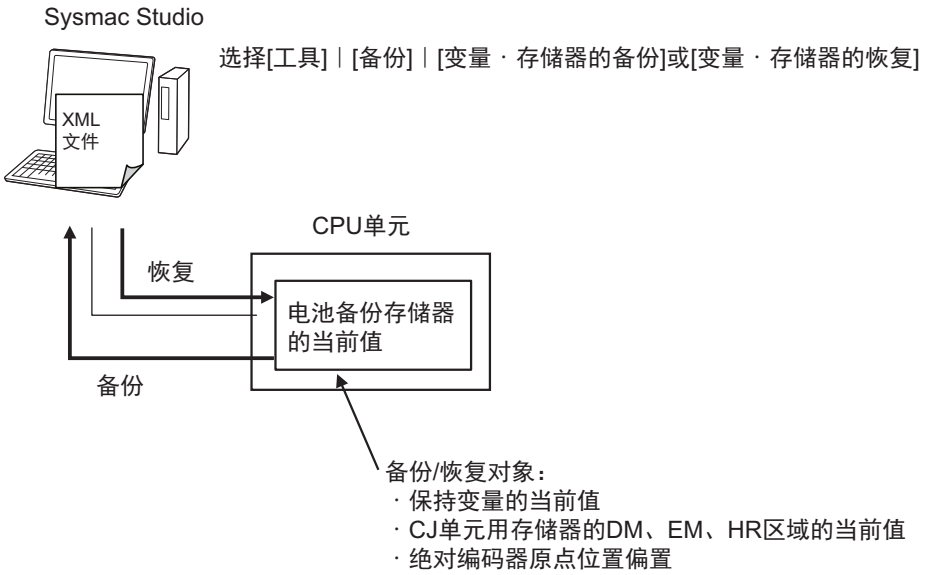
功能		数据组				
		用户程序设定*1	变量的当前值	CJ 单元用存储器的当前值*2	单元和从站设定	绝对编码器原点位置偏置
Sysmac Studio 备份文件的导入 / 导出功能	备份文件的导出	○	×	×	×	×
	备份文件的导入	○	×	×	○	×

*1. 以下数据不在处理对象范围内。控制器名称中的内置 EtherNet/IP 端口名称、控制器设定中的内置 EtherNet/IP 用标签数据链接设定、单元构成和单元设定中的 CPU 总线单元区域、操作权限认证、数据追踪设定。
*2. 可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

本功能的详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362D 之后）》。

9-8 Sysmac Studio 变量 / 存储器的备份功能

可将 CPU 单元中的电池备份存储器当前值以文件 （XML 格式）的形式备份 （读取）到计算机中，或者从计算机恢复 （写入）到 CPU 单元。
下面介绍 Sysmac Studio 变量 / 存储器备份功能的对象数据、操作方法及 CPU 单元型号之间的兼容性。



9-8-1 Sysmac Studio 变量 / 存储器备份功能的对象数据

Sysmac Studio 变量 / 存储器备份功能的对象为以下数据。

- 保持属性的变量当前值
- CJ 单元用存储器的 DM、 EM、 HR 区域的当前值
- 绝对编码器原点位置偏置

（注）可使用 CJ 单元用存储器的 CPU 单元仅限 NJ 系列 CPU 单元。

版本相关信息

CPU 单元版本 1.04 以上和 Sysmac Studio Ver.1.05 以上的组合时，可以只选择特定的变量，备份 / 恢复有保持属性的变量当前值。

9-8-2 Sysmac Studio 变量 / 存储器备份功能的操作方法

Sysmac Studio 上的操作方法如下所示。
在在线连接状态下，从 [工具] 菜单中选择 [备份] ｜ [变量・存储器的备份] 或 [变量・存储器的恢复]。
详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

9-8-3 CPU 单元型号间的兼容性

关于 Sysmac Studio 变量 / 存储器备份功能，即使备份源和恢复目标的 CPU 单元型号不同，也可恢复。

参考

与数据库连接 CPU 单元、搭载 SECS/GEM 的 CPU 单元及 NJ Robotics CPU 单元不兼容。
关于这些 CPU 单元的兼容性，请参考各 CPU 单元的手册。

版本相关信息

备份源和恢复目标的 CPU 单元型号不同也可恢复的版本为 CPU 单元版本 1.04 以上和 Sysmac Studio Ver.1.05 以上的组合。其他版本下的兼容性如下所示。

（○：有兼容性、×：无兼容性）

备份源的 CPU 单元型号	备份目标的 CPU 单元型号	
	NJ501-1500 NJ501-1400 NJ501-1300	NJ301-1200 NJ301-1100
NJ501-1500、 NJ501-1400、 NJ501-1300	○	×
NJ301-1200、 NJ301-1100	×	○

但是，根据备份 / 恢复的对象数据，需要注意以下事项。

保持属性的变量当前值

保持属性的变量当前值没有需要特别注意的事项。与备份源和恢复目标的 CPU 单元型号无关，均可恢复。

CJ 单元用存储器的 DM、 EM、 HR 区域的当前值

对于 CJ 单元用存储器的 DM、EM、HR 区域当前值，当备份源和恢复目标的 CPU 单元型号不同时，应注意以下事项。

备份源的 CPU 单元型号	备份目标的 CPU 单元型号	注意事项
NJ501-1300、 NJ501-1400、 NJ501-1500	NJ301-1200、 NJ301-1100、 NJ101-1000、 NJ101-9000	扩展数据存储器 (EM) 仅恢复备份文件中的 E0 ～ E3。备份文件中的 E4 ～ E18 的值将被忽略。
NJ301-1200、 NJ301-1100、 NJ101-1000、 NJ101-9000	NJ501-1300、 NJ501-1400、 NJ501-1500	扩展数据存储器 (EM) 仅恢复备份文件中的 E0 ～ E3。 CJ 单元的 E4 ～ E18 保持恢复前的值。

绝对编码器原点位置偏置

对于绝对编码器原点位置偏置，当备份源和恢复目标的 CPU 单元型号不同时，应注意以下事项。

备份源的 CPU 单元型号	备份目标的 CPU 单元型号	注意事项
NJ501-1300、 NJ501-1400、 NJ501-1500	NJ301-1200、 NJ301-1100、 NJ101-1000	与备份源的有效轴数无关，将从备份文件中的起始轴开始，按恢复目标的有效轴数进行恢复。备份文件中多余的数据将被忽略。
NJ301-1200、 NJ301-1100、 NJ101-1000	NJ501-1300、 NJ501-1400、 NJ501-1500	与备份源的有效轴数无关，将从备份文件中的起始轴开始，按恢复目标的有效轴数进行恢复。恢复目标的有效轴数大于备份文件中的数据时，多余的轴数据将保持恢复前的值。

9-9 连接 EtherCAT 从站时的备份功能

EtherCAT 从站为 SD 存储卡备份功能、Sysmac Studio 控制器备份功能及 Sysmac Studio 备份导入功能的对象。
下面以连接 EtherCAT 从站时为例，介绍备份对象数据、控制器各状态的备份功能执行可否、恢复执行条件、各型号 EtherCAT 从站的注意事项。

参考

备份功能的对象为 EtherCAT 从站终端时，请参考「9-10 连接 EtherCAT 从站终端时的备份功能」（P.9-48）。

9-9-1 EtherCAT 从站的备份对象数据

EtherCAT 从站的备份对象数据如下所示。

设定	备份对象数据
EtherCAT 主机设定	型号名称、产品名称、从站数、 PDO 通信周期、低效运行设定、 所有从站加入等待时间、 PDO 通信连续超时检测次数、 修订版本检查基准、序列号检查基准
EtherCAT 从站设定	设备名称、型号名称、产品名称、修订版本、节点地址 有效 / 无效设定、序列号、 PDO 映射设定、 DC 有效、 参考时钟功能、初始化参数设定

9-9-2 控制器在各种状态下的备份功能执行可否

控制器处于各种状态时，可否执行 EtherCAT 从站的备份、恢复、核对如下所示。

控制器的状态		执行可否		
		备份	恢复	核对
链接断开中		不可*1	不可*2	可*3
主机状态不正确*4		不可 *1	不可 *2	可 *3
网络信息与网络配置不一致*5		不可 *1	不可 *2	可 *3
与备份时的网络配置不一致		可	不可 *2	可 *3
网络配置中存在无效设定的从站	实际配置中存在从站	可*6	可 *6	可 *6
	实际配置中不存在从站	可	可	可
网络配置中存在根据 " 脱离 " 指示变为脱离状态的从站	实际配置中存在从站	不可 *1	可 脱离状态的从站也在对象范围内	可 脱离状态的从站也在对象范围内
	实际配置中不存在从站	不可 *1	不可 *2	可 *3
从站初始化异常		不可 *1	不可 *2	可 *3

*1. 事件日志中将记录备份执行失败 （EtherCAT 从站）。
*2. 事件日志中将记录恢复执行失败 （EtherCAT 从站）。
*3. 核对结果为不一致。
*4. " 从站节点地址重复 "、" 网络配置信息异常 "、" 网络配置异常 "、" 从站初始化异常 "、" 低效运行设定为 " 停止设定 " 时的网络配置核对异常 "、" 链接断开异常 "。
*5. " 备份时和网络配置不完全一致（分支从站上的连接端口不同将视为不一致）"、" 网络配置信息和网络配置不一致 （分支从站上的连接端口不同将视为不一致 "。
*6. 在 CPU 单元版本 1.04 以上和 Sysmac Studio Ver.1.05 以上的组合中，无效设定的从站也在对象范围内。其他版本中，无效设定的从站不在对象范围内。

9-9-3 EtherCAT 从站的恢复执行条件

为了恢复 EtherCAT 从站的备份数据，必须满足以下条件。

- 备份文件中存在 EtherCAT 从站的数据。
- 网络配置信息和恢复目标的实际网络配置一致。
- EtherCAT从站中事先设定的修订版本值一致。但是，一致条件遵照备份文件中保存的“修订版本检查基准 ”。此外，即使 “ 修订版本检查基准 ” 设定为 “ 不检查 ”， “ 设定修订版本 ” ＞ “ 实际机器修订版本 ” 时，也会变为不可恢复。此外，用户无法变更修订版本值。
- 备份文件中保存的 “ 序列号核对的有无 ” 设定为 “ 核对 ” 时，序列号一致。
- 通过硬件开关设定节点地址时，节点地址一致。

使用注意事项

- 恢复执行完成后，请重新接通所有 EtherCAT 从站的电源。
- 恢复执行完成后，所有从站变为脱离状态。要解除对象从站的脱离状态，需要再次加入对象从站的指示。
- 备份文件中保存的“序列号核对的有无”设定为“核对”时，若更换EtherCAT从站的硬件，将无法恢复。此时，请通过 Sysmac Studio 变更网络配置并下载，然后传送从站参数，恢复原样。此外，用硬件开关设定节点地址时，请设定为与备份时相同。

9-9-4 备份功能的对象 EtherCAT 从站

备份功能的对象 EtherCAT 从站如下所示。使用时请了解注意事项。

EtherCAT 从站的机型	注意事项
NX 系列 EtherCAT 耦合器单元 NX-ECC	EtherCAT 从站终端上的安全控制单元不在备份、恢复、核对的对象范围内。 安全控制单元的设定数据导入 / 导出功能请参考📖《NX 系列 安全控制单元 用户手册 （SGFM-CN5-710）》。
AC 伺服驱动器 （旋转型） R88D-KN □□□ -ECT	*1*2
AC 伺服驱动器 （直线型） R88D-KN □□□ -ECT-L	*1*2
变频器 3G3AX-MX2-ECT 3G3AX-RX-ECT	• Ver.1.11 以上的 CPU 单元时，变频器的参数不在恢复对象范围内。变频器不会发生 “ 恢复执行失败 （EtherCAT 从站） ” （事件代码 10300000Hex) 的事件。 • 请按照后述的📖「 变频器 3G3AX-MX2-ECT、3G3AX-RX-ECT 的参数写入方法 」（P.9-47），将来自 Sysmac Studio 的参数写入变频器中。 若不写入参数直接执行核对，变频器的 EtherCAT 从站核对结果文件中，可能出现 “Result=Not matched”。 • Ver.1.10 以下的 CPU 单元时，因变频器的限制，可能无法进行恢复。 执行恢复时，发生 “ 恢复执行失败 （EtherCAT 从站） ” （事件代码 10300000Hex) 的事件时，请按照后述的📖「 变频器 3G3AX-MX2-ECT、3G3AX-RX-ECT 的参数写入方法 」（P.9-47），将 Sysmac Studio 的参数写入变频器中。 此外，即使变频器恢复失败，变频器以外的单元和从站设定、用户程序和设定、变量的当前值等，所有对象数据均已恢复完成。
视觉传感器 FH-3 □□□□ FH-1 □□□□	传感器的设定数据 （场景数据、系统数据等）不在备份、恢复、核对的对象范围内。 要将传感器的设定数据传送到外部文件或传送到传感器时，请从 Sysmac Studio 配置和设定编辑画面，执行 [工具] ｜ [传感器数据] ｜ [保存到文件] 或 [从文件中读取]。 详情请参考📖《图像处理系统 FH/FZ5 系列 用户手册 （SDNB-CN5-712）》。
视觉传感器 FQ-M □□□□ -ECT FQ-M □□□□ -M-ECT	传感器的设定数据 （场景数据、系统数据等）不在备份、恢复、核对的对象范围内。 要将传感器的设定数据传送到外部文件或传送到传感器时，请从 Sysmac Studio 配置和设定编辑画面，执行 [工具] ｜ [传感器数据] ｜ [保存到文件] 或 [从文件中读取]。 详情请参考📖《视觉传感器 FQ-M 系列 定位专用视觉传感器 用户手册 （SDNB-CN5-706）》。

EtherCAT 从站的机型	注意事项
视觉传感器 FZM1- □□□ -ECT	传感器的设定数据 （场景数据、系统数据等）不在备份、恢复、核对的对象范围内。 要将传感器的设定数据保存到 USB 存储器或写入到控制器时，请使用专用工具。 详情请参考📖《视觉传感器 FZ3 系列 用户手册 （SCHB-762）》。
数字 I/O 终端 GX- □ D16 □□ GX- □ D32 □ 8 GX-OC1601	*1
模拟 I/O 终端 GX-AD0471 GX-DA0271	*1
编码器输入终端 GX-EC0211 GX-EC0241	—
EtherCAT 分支从站 GX-JC0 □	没有内置需要备份的数据。
光纤同轴位移传感器 ZW-CE1 □	所有设定数据不在备份、恢复、核对的对象范围内。 设定数据的保存、写入到控制器的方法请参考📖《位移传感器 ZW 系列 光纤同轴位移传感器 用户手册 （SDNE-CN5-703）》。
数字型传感器 E3NW-ECT E3X-ECT	传感器的参数不在备份、恢复、核对的对象范围内。
其他公司从站	• 只有在 ESI 中正确定义的数据是备份、恢复、核对的对象。要备份、恢复、核对未在 ESI 中定义的数据时，请使用专用工具。 • 备份、恢复、核对失败时，请向从站的制造商咨询处理方法。

*1. 恢复执行完成后，请重新接通从站的电源。恢复后进行核对时，请在重新接通从站的电源后再执行。若在重新接通电源前执行核对，将发生核对不一致。

*2. 符合以下任一条件时，从备份到核对期间，或从恢复到核对期间，请勿设为伺服 ON 状态。从备份到核对期间，或从恢复到核对期间，如果设为伺服 ON 状态，可能在核对前变更参数，发生核对不一致。

实时自动调谐模式设定为有效 （对象 3002h 的值为 1 ～ 4 或 6）。

适用过滤器设定为有效 （对象 3200h 的值为 1 或 2）。

变频器 3G3AX-MX2-ECT、 3G3AX-RX-ECT 的参数写入方法

Ver.1.11 以上的 CPU 单元时，变频器的参数不在恢复对象范围内。

Ver.1.10 以下的 CPU 单元时，若连接变频器 3G3AX-MX2-ECT、 3G3AX-RX-ECT 执行恢复，将发生 “恢复执行失败 （EtherCAT 从站） ”（事件代码 10300000Hex）的事件。

要将备份的参数写入到变频器时，请对 Sysmac Studio 进行以下操作。详情请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362）》。

1 **使用 Sysmac Studio 备份文件的导入功能，读取备份文件中的变频器参数。**
Sysmac Studio 的配置和设定的变频器参数选项卡中将显示变频器的参数。

2 **确认所读取参数的变频器型号和实际连接的变频器型号一致。**

3 **使用配置和设定的变频器参数选项卡的传送 [计算机➜驱动器] 功能，将参数写入到变频器中。**

使用注意事项

使用变频器 3G3AX-MX2-ECT 的变频器模式选择 （参数 No.b171）功能，写入参数前，请将所连接变频器的模式切换为备份时的模式。此外，要使模式切换变为有效，需要在模式切换操作后，进行变频器初始化。

参考

Ver.1.10 以下的 CPU 单元中，即使变频器恢复失败，变频器以外的单元和从站设定、用户程序和设定、变量的当前值等，所有对象数据均已恢复完成。

9-10 连接 EtherCAT 从站终端时的备份功能

EtherCAT 从站终端为 SD 存储卡备份功能、Sysmac Studio 控制器备份功能及 Sysmac Studio 备份导入功能的对象。
下面以连接 EtherCAT 从站终端时为例，介绍备份对象数据、控制器各状态的备份功能执行可否、恢复执行条件。

使用注意事项

EtherCAT 从站终端上的安全控制单元不在备份、恢复、核对的对象范围内。安全控制单元的设定数据导入 / 导出功能请参考📖《NX 系列 安全控制单元 用户手册 （SGFM-CN5-710）》。

版本相关信息

EtherCAT 从站终端可在 CPU 单元版本 1.05 以上和 Sysmac Studio Ver.1.06 以上的组合中连接。

9-10-1 EtherCAT 从站终端的备份对象数据

EtherCAT 从站终端的备份对象数据因 EtherCAT 耦合器单元和 NX 单元不同而异。各备份对象数据如下所示。

（○：对象　×：对象外）

单元	数据	备份	恢复	核对
EtherCAT 耦合器单元	构成信息*1	○	○	○
	单元动作设定	○	○	○
NX 单元	构成信息 *1	○	○	○
	单元动作设定	○	○	○
	单元应用数据*2	○	○	○

*1. 单元构成信息及 I/O 分配信息的总称。
*2. 各 NX 单元固有的数据。有的 NX 单元没有单元应用数据。

使用注意事项

将备份数据恢复到与备份源相同单元构成的 EtherCAT 从站终端时，请将硬件开关设定为与备份源相同的设定。硬件开关设定与备份数据不同时，无法恢复。将发生监视信息事件 “ 恢复开始失败（EtherCAT 从站） ”。

9-10-2 EtherCAT 从站终端在各状态下的备份功能执行可否

EtherCAT 从站终端处于各种状态时，可否执行备份、恢复、核对如下所示。

EtherCAT 从站终端的状态	执行可否		
	备份	恢复	核对
单元构成信息自动生成时	可*1	可*2	可
等待 NX 单元加入	不可*3	不可*4	可*5
EtherCAT 耦合器单元或 NX 单元发生看门狗定时器异常	不可 *3	不可 *4	可 *5
发生总线控制器异常	不可 *3	不可 *4	可 *5
发生单元构成信息异常	不可 *3	可	可 *5
发生单元构成核对异常	不可 *3	可	可 *5
与备份时的单元构成信息不一致	—	不可 *4	可 *5

*1. 将备份 " 单元构成信息不存在 " 这一信息。
*2. 恢复后，也变为单元构成自动生成的状态。
*3. 事件日志中将记录备份执行失败。
*4. 事件日志中将记录恢复执行失败。
*5. 核对结果为不一致。

9-10-3 EtherCAT 从站终端的恢复执行条件

为了恢复 EtherCAT 从站终端的备份数据，必须满足以下条件。

- 备份文件中存在 EtherCAT 耦合器单元及 NX 单元的数据。
- 备份源的单元构成信息和恢复目标的实际单元构成一致。
- 备份源的EtherCAT耦合器单元的序列号和恢复目标的EtherCAT耦合器单元的序列号一致。但是，仅限备份文件中的[通信耦合器单元]｜[单元动作设定]｜[序列号检查基准]的设定为"设定值＝实际机器"时。
- 备份源的NX单元的序列号和恢复目标的NX单元的序列号一致。但是，仅限备份文件中的 [通信耦合器单元] ｜ [单元动作设定] ｜ [序列号检查基准] 的设定为 " 设定值＝实际机器 " 时。
- 备份源的 EtherCAT 耦合器单元的硬件开关信息和恢复目标的 EtherCAT 耦合器单元的硬件开关设定完全一致。
- 备份源的 EtherCAT 耦合器单元的单元版本设定值和恢复目标的 EtherCAT 耦合器单元的实际单元版本一致或较旧。
- 备份源的 NX 单元的单元版本设定值和恢复目标的 NX 单元的实际单元版本一致或较旧。

9-11 连接 CJ 单元时的备份功能

CJ 单元为 SD 存储卡备份功能、 Sysmac Studio 控制器备份功能的对象。
下面以连接 CJ 单元时为例，介绍备份对象数据、控制器各状态的备份功能执行可否、恢复执行条件。

使用注意事项

可连接 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

9-11-1 CJ 单元的备份对象数据

备份对象数据为 CJ 单元用存储器的当前值、 CJ 单元内的参数。上述数据包括 CJ 单元内的数据和 CPU 单元内的数据。
但是，它们会自动汇总后进行备份、恢复和核对，因此用户无需在意数据的存在。

9-11-2 控制器在各种状态下的备份功能执行可否

控制器处于各种状态时，可否执行 CJ 单元的备份、恢复、核对如下所示。

控制器的状态	执行可否		
	备份	恢复	核对
发生 “I/O 总线检查错误 ” 发生 “ 未连接端盖 ” 发生 “ 单元 / 扩展装置连接不正确 ” 发生 “ 单元编号重复 ” 发生 “I/O 点数超限 ” 发生 “I/O 设定检查错误 ”	不可 *1	不可 *2	可 *3
CJ 单元重启中	不可 *1	可	可 *3

*1. 事件日志中将记录备份执行失败 （CJ 单元）。
*2. 事件日志中将记录恢复执行失败 （CJ 单元）。
*3. 核对结果为不一致。

9-11-3 CJ 单元的恢复执行条件

为了恢复 CJ 单元的备份数据，必须满足以下条件。

- 备份文件中存在 CJ 单元的数据。
- 备份文件中的单元构成和恢复目标的实际单元构成一致。
- 满足该 CJ 单元固有的条件 （该 CJ 单元的固有条件请参考各 CJ 单元的手册）。

使用注意事项

在连接了 CJ 单元的状态下，如果执行了 SD 存储卡备份功能的恢复或 Sysmac Studio 控制器备份功能的恢复，可能会发生 “CPU 单元服务监视异常 ”。这表示从 CPU 单元到 CJ 单元的服务处理未在一定时间内完成。但是，这是在恢复花费太长时间时发生的，并非异常。此时的现象如下所示。

- 通信类 CJ 单元时， MS LED 闪烁红灯。
- 有 7 段 LED 的 CJ 单元时， 7 段 LED 的显示变为 "HE"。
- 有 ERH LED 的 CJ 单元时， ERH LED 亮灯。
- 控制器事件日志中将登录事件代码 =00000002Hex、 CPU 单元服务监视异常或刷新超时的事件。
- 连接了 CJ 系列高速计数器单元 CJ1W-CT021 时，控制器事件日志中将登录事件代码 =68010000Hex、单元错误、附属信息 1=0002Hex 的事件。

9-12 备份相关文件

下面介绍备份相关文件的规格。
备份相关文件是指在 “Sysmac Studio 变量 / 存储器的备份功能 ” 以外的备份功能中使用的文件。

9-12-1 备份相关文件的种类

备份相关文件包括备份文件、恢复指令文件、自动传送指令文件、核对结果文件 4 种。

● 备份文件

用备份相关功能中处理的控制器数据制作的文件。执行备份时创建。

● 恢复指令文件

从 SD 存储卡执行恢复时，用于指定传送对象数据组的文件。在计算机的文本编辑器中编辑文件，可指定传送对象。执行备份时创建。

● 自动传送指令文件

执行从 SD 存储卡自动传送时，用于指定传送对象数据组的文件。在计算机的文本编辑器中编辑文件，可指定传送对象。执行备份时创建。

● 核对结果文件

该文件用于保存对控制器的数据和 CPU 单元中安装的 SD 存储卡的备份文件数据进行核对的核对结果。
核对结果文件分为以下 4 种。均在执行 SD 存储卡备份功能中的核对时创建。

核对结果文件	内容
控制器核对结果文件	记述了恢复指令文件中指定的所有备份数据核对结果的文件。
EtherCAT 从站核对结果文件	记述了各 EtherCAT 从站核对结果的文件。恢复指令对象文件中的恢复对象为单元和从站设定，且备份文件中含有 EtherCAT 从站设定时创建。
EtherCAT 从站终端核对结果文件	记述了 EtherCAT 耦合器单元及各 NX 单元核对结果的文件。同时满足以下 3 个条件时创建。 • 恢复指令对象文件中的恢复对象为单元和从站设定。 • 备份文件中包含 EtherCAT 从站设定。 • 连接了 EtherCAT 从站终端。 创建 EtherCAT 从站终端核对结果文件时，必定会同时创建 EtherCAT 从站核对结果文件。
CJ 单元核对结果文件*1	记述了各 CJ 单元核对结果的文件。恢复指令对象文件中的恢复对象为单元和从站设定，且备份文件中含有 CJ 单元设定时创建。

*1. 可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

9-12-2 备份文件的规格

备份文件的文件名、创建时间、创建目录如下。

文件名

备份文件的文件名因 CPU 单元的系列不同而异。

文件	对象 CPU 单元	文件名
备份文件	NX 系列 CPU 单元	NXBackup.dat
	NJ 系列 CPU 单元	NJBackup.dat

文件创建时间和创建目录

功能	操作方法	创建时间	创建目录
SD 存储卡的备份功能	CPU 单元正面开关	执行备份时	SD 存储卡中的根目录
	系统定义变量	执行备份时	SD 存储卡中，以系统定义变量指定的目录
	Sysmac Studio SD 存储卡画面	执行备份时	SD 存储卡中，由用户在画面中指定的目录
	专用指令*1	执行备份时	SD 存储卡中，由用户以 BackupToMemoryCard 指令的输入变量指定的目录
SysmacStudio 控制器备份功能	Sysmac Studio 控制器备份画面	执行备份时	计算机中，由用户在画面中指定的目录
SysmacStudio 备份文件的导入 / 导出功能	SysmacStudio 备份文件的导出画面	执行导出时	计算机中，由用户在画面中指定的目录

*1. 可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。

9-12-3 恢复指令文件的规格

恢复指令文件的文件名、创建时间、创建目录、数据组的指定方法如下。

文件名

文件	文件名
恢复指令文件	RestoreCommand.ini

文件创建时间和创建目录

功能	操作方法	创建时间	创建目录
SD 存储卡的备份功能	CPU 单元正面开关	执行备份时	与备份文件相同的目录
	系统定义变量	执行备份时	与备份文件相同的目录
	Sysmac Studio SD 存储卡画面	执行备份时	与备份文件相同的目录
SysmacStudio 控制器备份功能	Sysmac Studio 控制器备份画面	执行备份时	与备份文件相同的目录
SysmacStudio 备份文件的导入 / 导出功能	SysmacStudio 备份文件的导出画面	执行导出时	与备份文件相同的目录

要备份的数据组的指定方法

恢复指令文件是指按数据组进行恢复指定所需的文件。
在计算机的文本编辑器中，对按数据组进行指定的部分进行编辑，可以变更指定。
例如，将下表中 “ 文件内容 ” 第 8 行的 “Variable=yes” 变更为 “Variable=no” 后，“ 变量的当前值 ” 将排除在恢复对象之外。

文件内容 （创建文件时的默认内容）	说明
[Restore] ; --- User Program and Configuration. --- ; Always select "yes". UserProgram=yes	用户程序设定 yes 固定：必须恢复
; --- Present values of variables (Retained variables only). --- ; "yes":will be restored, "no":will not be restored Variable=yes	变量的当前值 （仅保持属性） yes/no：恢复 / 不恢复
; --- Present values of memory used for CJ-series Units (Holding, EM, and DM Area data). --- ; "yes":will be restored, "no":will not be restored Memory=yes	CJ 单元用存储器的当前值 （仅保持存储器） yes/no：恢复 / 不恢复
:---Unit/Slave Parameters.--- ; "yes";will be restored."no";will not be restored UnitConfig=yes	单元和从站设定 yes/no：恢复 / 不恢复
; --- Absolute encoder home offset. --- ; "yes":will be restored, "no":will not be restored AbsEncoder=yes	绝对编码器原点位置偏置 yes/no：恢复 / 不恢复

（注）1 本表的文件内容为创建恢复指令文件时的内容。记述的所有数据组均为恢复对象。
2 记述在创建备份文件时已在备份文件中，且指定了可恢复数据组的文件。
3 全部为半角英文和数字。不区分大小写。

使用注意事项

编辑恢复指令文件时，请只记述可选择的数据组的 “yes/no”。若编辑其他记述后执行恢复，控制器可能发生意外动作。

9-12-4 自动传送指令文件的规格

自动传送指令文件的文件名、创建时间、创建目录、数据组的指定方法如下。

文件名

文件	文件名
自动传送指令文件	AutoloadCommand.ini

文件创建时间和创建目录

功能	操作方法	创建时间	创建目录
SD 存储卡的备份功能	CPU 单元正面开关	执行备份时	与备份文件相同的目录
	系统定义变量	执行备份时	与备份文件相同的目录
	Sysmac Studio SD 存储卡画面	执行备份时	与备份文件相同的目录
SysmacStudio 控制器备份功能	Sysmac Studio 控制器备份画面	执行备份时	与备份文件相同的目录
SysmacStudio 备份文件的导入 / 导出功能	SysmacStudio 备份文件的导出画面	执行导出时	与备份文件相同的目录

要自动传送的数据组的指定方法

指定传送指令文件是指按数据组进行自动传送指定所需的文件。
在计算机的文本编辑器中，对按数据组进行指定的部分进行编辑，可以变更指定。
例如，将下表中 “ 文件内容 ” 第 8 行的 “Variable=yes” 变更为 “Variable=no” 后，“ 变量的当前值 ” 将排除在自动传送对象之外。

文件内容 （创建文件时的默认内容）	说明
[Autoload] ; --- User Program and Configuration. --- ; Always select "yes". UserProgram=yes	用户程序设定 yes 固定：必须传送
; --- Present values of variables (Retained variables only). --- ; "yes":will be transferred, "no":will not be transferred Variable=yes	变量的当前值 （仅保持属性） yes/no：传送 / 不传送
; --- Present values of memory used for CJ-series Units (Holding, EM, and DM Area data). --- ; "yes":will be transferred, "no":will not be transferred Memory=yes	CJ 单元用存储器的当前值 （仅保持存储器） yes/no：传送 / 不传送

（注）1 本表的文件内容为创建自动传送指令文件时的内容。记述的所有数据组均为自动传送对象。
2 记述在创建备份文件时已在备份文件中，且指定了可自动传送数据组的文件。
3 全部为半角英文和数字。不区分大小写。

使用注意事项

编辑自动传送指令文件时，请只记述可选择的数据组的 “yes/no”。若编辑其他记述后执行自动传送，控制器可能发生意外动作。

9-12-5 控制器核对结果文件的规格

控制器核对结果文件的文件名、创建时间、创建目录、核对结果的确认方法如下。

文件名

文件	文件名
控制器核对结果文件	VerifyResult.log

文件创建时间和创建目录

功能	操作方法	创建时间	创建目录
SD 存储卡的备份功能	Sysmac Studio SD 存储卡画面	执行核对时	与备份文件相同的目录
	系统定义变量	执行核对时	与备份文件相同的目录
	CPU 单元正面开关	执行核对时	与备份文件相同的目录

（注）但是， SD 存储卡有写保护时不创建。

核对结果的确认方法

核对结果文件用于保存按数据组对控制器的数据和 CPU 单元中安装的 SD 存储卡的备份文件数据进行核对的核对结果。
在保存了各数据组确认结果的部分，可确认核对结果。
若结果为 “Result=Matched”，则表示核对一致；若结果为 “Result=Not matched”，则表示核对不一致。
文件内容为以下内容时，用户程序设定为核对一致，单元 / 从站设定为核对不一致。

文件内容	说明
[UserProgram] ; --- User Program and Configuration. --- Result=Matched	用户程序设定 Matched：核对一致、 Not matched：核对不一致
[UnitConfig] ; --- Unit/Slave Parameters. --- Result=Not matched	单元 / 从站设定 Matched：核对一致、 Not matched：核对不一致

（注）仅记述执行了核对的数据组的核对结果。

9-12-6 EtherCAT 核对结果文件的规格

EtherCAT 核对结果文件的文件名、创建时间、创建目录、核对结果的确认方法如下。

文件名

文件	文件名
EtherCAT 核对结果文件	VerifyResult_ECAT.log

文件创建时间和创建目录

功能	操作方法	创建时间	创建目录
SD 存储卡的备份功能	Sysmac Studio SD 存储卡画面	执行核对时	与备份文件相同的目录
	系统定义变量	执行核对时	与备份文件相同的目录
	CPU 单元正面开关	执行核对时	与备份文件相同的目录

（注）但是， SD 存储卡有写保护时不创建。

核对结果的确认方法

核对结果文件用于保存按数据组对控制器的数据和 CPU 单元中安装的 SD 存储卡的备份文件数据进行核对的核对结果。

在保存了各 EtherCAT 从站确认结果的部分，可确认核对结果。

若结果为 “Result=Matched”，则表示核对一致；若结果为 “Result=Not matched”，则表示核对不一致。

文件内容为以下内容时，核对结果如下。

- 核对一致 ：EtherCAT 从站 Master、 EtherCAT 从站终端 E002
- 核对不一致 ：EtherCAT 从站 E001

文件内容	说明
[Verification Results] ; --- EtherCAT Parameters. --- : --- See the VerifyResult_ECAT_NX.log about detail result if NX mark is included in square brackets. [Master] Result=Matched [E001] Result=Not matched Factor=Verification error [E002:NX] Result=Matched	从站以用户设定的设备名称表示。 EtherCAT 从站终端时，设备名称后面带 ":NX"。*1 核对结果用以下表述表示。 Result=Matched 核对一致 Result=Not matched 核对不一致

*1. EtherCAT 从站终端为核对对象时，将创建 EtherCAT 从站终端核对结果文件。 EtherCAT 从站终端的详细核对结果记载于 EtherCAT 从站终端核对结果文件中。

（注）仅记述执行了核对的 EtherCAT 从站的核对结果。

9-12-7 EtherCAT 从站终端核对结果文件的规格

EtherCAT 从站终端核对结果文件的文件名、创建时间、创建目录、核对结果的确认方法如下。

文件名

文件	文件名
EtherCAT 从站终端核对结果文件	VerifyResult_ECAT_NX.log

文件创建时间和创建目录

功能	操作方法	创建时间	创建目录
SD 存储卡的备份功能	Sysmac Studio SD 存储卡画面	执行核对时	与备份文件相同的目录
	系统定义变量	执行核对时	与备份文件相同的目录
	CPU 单元正面开关	执行核对时	与备份文件相同的目录

（注）但是， SD 存储卡有写保护时不创建。

核对结果的确认方法

核对结果文件用于保存按数据组对控制器的数据和 CPU 单元中安装的 SD 存储卡的备份文件数据进行核对的核对结果。

在保存了 EtherCAT 耦合器单元及各 NX 单元确认结果的部分，可确认核对结果。

若结果为 “Result=Matched”，则表示核对一致；若结果为 “Result=Not matched”，则表示核对不一致。

文件内容为以下内容时，核对结果如下。

- 核对一致 ：EtherCAT 耦合器单元 E002、 NX 单元 N1、 NX 单元 N2
- 核对不一致 ：EtherCAT 耦合器单元 E005、 NX 单元 N3

文件内容	说明
[Verification Results] ; --- NX Parameters. --- [E002:UnitNo.0 NX-ECC201] Result=Matched [N1:UnitNo.1 NX-AD2203] Result=Matched [N2:UnitNo.2 NX-DA2203] Result=Matched [N3:UnitNo.3 NX-TS3201] Result=Not matched Factor=Verification error [E005:UnitNo.0 NX-ECC201] Result=Not matched Factor=Verification error	单元按以下规格表述。 { 设备名称 }:UnitNo.{ 单元编号 }[空格]{ 单元型号 } { 设备名称 } 用户设定的设备名称。 { 单元编号 } 10 进制数的字符串。值的范围为 0-125。 { 单元型号 } 表示单元型号的字符串。 从型号后面开始的连续空格将删除。 核对结果用以下表述表示。 Result=Matched 核对一致 Result=Not matched 核对不一致

9-12-8 CJ 单元核对结果文件的规格

CJ 单元核对结果文件的文件名、创建时间、创建目录、核对结果的确认方法如下。

使用注意事项

可使用 CJ 单元的 CPU 单元仅限 NJ 系列 CPU 单元。

文件名

文件	文件名
CJ 单元核对结果文件	VerifyResult_CJUnit.log

文件创建时间和创建目录

功能	操作方法	创建时间	创建目录
SD 存储卡的备份功能	Sysmac Studio SD 存储卡画面	执行核对时	与备份文件相同的目录
	系统定义变量	执行核对时	与备份文件相同的目录
	CPU 单元正面开关	执行核对时	与备份文件相同的目录

（注）但是， SD 存储卡有写保护时不创建。

核对结果的确认方法

核对结果文件用于保存按数据组对控制器的数据和 CPU 单元中安装的 SD 存储卡的备份文件数据进行核对的核对结果。

在保存了各 CJ 单元确认结果的部分，可确认核对结果。

若结果为 “Result=Matched”，则表示核对一致；若结果为 “Result=Not matched”，则表示核对不一致。

文件内容为以下内容时， CJ1W-CRM21 （MODE0）和 CJ1W-EIP21 为核对一致， CJ1W-DRM21 和 CJ1W-PRM21-DPV1 为核对不一致。

文件内容	说明
[Verification Results] ; --- CJ Unit Parameters. --- [Rack0 Slot0: CJ1W-CRM21(MODE0) UnitNo.10] Result=Matched [Rack0 Slot9: CJ1W-DRM21 UnitNo.0] Result=Not matched [Rack1 Slot0: CJ1W-EIP21 UnitNo.10]] Result=Matched [Rack1 Slot1: CJ1W-PRM21-DPV1 UnitNo.1] Result=Not matched	单元按以下规格表述。 Rack{ 装置 No.}[空格]Slot{ 插槽 No.}:[空格]{ 单元型号 }[空格]UnitNo.{ 单元号机 No.} { 装置 No.} 10 进制数的字符串。值的范围为 0-3。 { 插槽 No.} 10 进制数的字符串。值的范围为 0-9。 { 单元型号 } 表示单元型号的字符串。 单元型号使用从循环初始数据中获取的数据。从型号后面开始的连续空格将删除。 { 单元号机 No.} 10 进制数的字符串。消零 CPU 总线单元时，值的范围为 0-15。高功能 I/O 单元时为 0-95。 核对一致 / 核对不一致用以下表述表示。 Result=Matched 核对一致 Result=Not matched 核对不一致

（注）仅记述执行了核对的 EtherCAT 从站的核对结果。

9-13 备份功能的文件间兼容性

若执行备份时和执行恢复时的条件不同，需要考虑文件间的兼容性。
以下 3 种情况下需要考虑兼容性。

- 备份的功能和恢复功能不同时
- 备份源和恢复目标的 CPU 单元型号不同时
- 备份源和恢复目标的 CPU 单元或单元、从站的单元版本不同时

此外，这里我们将各备份功能的 “ 恢复 ”、“ 自动传送 ”、“ 程序传送 ”、“ 读取 ” 统称为 “ 恢复 ”。

9-13-1 备份功能间的兼容性

备份的功能和恢复功能不同时，文件的兼容性如下。

（○：有兼容性、×：无兼容性）

备份的功能	恢复功能				
	SD 存储卡备份功能 （恢复） 从 SD 存储卡到控制器	自动传送功能、程序传送功能	Sysmac Studio 控制器备份功能 （恢复） 从计算机到控制器	Sysmac Studio 变量 / 存储器的备份功能 （恢复） 从计算机到控制器	Sysmac Studio 备份文件的导入 从计算机到项目
SD 存储卡的备份功能 （备份） 从控制器到 SD 存储卡	○	○	○	×	○*1
Sysmac Studio 控制器备份功能 （备份） 从控制器到计算机	○	○	○	×	○*1
Sysmac Studio 变量 / 存储卡的备份功能 （备份） 从控制器到计算机	×	×	×	○	×
Sysmac Studio 备份文件的导出 从项目到计算机	○*1	○*1	○*1	×	○

*1. 但是，以下数据不在对象范围内。

- 控制器设定的内置 EtherNet/IP 端口名称、内置 EtherNet/IP 用标签数据链接设定
- 单元构成的 CPU 总线单元区域
- 操作权限认证
- 数据追踪设定
- 时区设定
- 变量的当前值
- CJ 单元用存储器的当前值
- 绝对编码器原点位置偏置

参考

Sysmac Studio变量 / 存储器的备份功能中处理的文件与其他备份功能中处理的文件不同，因此不兼容。
本功能的详情请参考「9-8 Sysmac Studio 变量 / 存储器的备份功能」（P.9-39）。

9-13-2 CPU 单元型号间的兼容性

备份源和备份目标的 CPU 单元型号组不同时，文件的兼容性如下。

（○：有兼容性、×：无兼容性）

备份源的 CPU 单元型号	备份目标的 CPU 单元型号				
	NX701-1700 NX701-1600	NJ501-1500 NJ501-1400 NJ501-1300	NJ301-1200 NJ301-1100	NJ101-1000	NJ101-9000
NX701-1700 NX701-1600	○	×*1	×*1	×*1	×*1
NJ501-1500 NJ501-1400 NJ501-1300	×*1	○	×*1	×*1	×*1
NJ301-1200 NJ301-1100	×*1	×*1	○	×*1	×*1
NJ101-1000	×*1	×*1	×*1	○	×
NJ101-9000	×*1	×*1	×*1	×	○

*1. Sysmac Studio 变量 / 存储器的备份功能有兼容性。但是，仅限在 CPU 单元版本 1.04 以上和 Sysmac Studio Ver.1.05 以上的组合中。Sysmac Studio 变量 / 存储器的备份功能中，CPU 单元间各型号的兼容性请参考「9-8-3　CPU 单元型号间的兼容性」（P.9-40）。

参考

与数据库连接 CPU 单元、搭载 SECS/GEM 的 CPU 单元及 NJ Robotics CPU 单元不兼容。
关于这些 CPU 单元的兼容性，请参考各 CPU 单元的手册。

即使 CPU 单元的型号兼容时，各 CPU 单元型号之间仍可能有所限制。
因 CPU 单元型号间兼容性的有无，恢复功能的动作如下所示。

（○：执行恢复、×：不执行恢复）

CPU 单元型号间的兼容性	恢复功能				
	SD 存储卡备份功能（恢复）从 SD 存储卡到控制器	自动传送功能、程序传送功能	Sysmac Studio 控制器备份功能（恢复）从计算机到控制器*1	Sysmac Studio 变量 / 存储器的备份功能（恢复）从计算机到控制器	Sysmac Studio 备份文件的导入功能 从计算机到项目
有兼容性	○*2	○*2	○	○	○
无兼容性	×*3	×*4	×	×	×

*1. 仅通过本功能备份的文件可以恢复。
*2. 备份文件的内容超出恢复目标的规格范围时，将无法正常动作，并发生全部停止故障等级的控制器异常或部分停止故障等级的控制器异常。例如，使用的控制轴数超出规格时，动作如下。
*3. 将发生监视信息 " 恢复开始失败 "。
*4. 自动传送功能中，将发生全部停止故障等级的控制器异常 " 自动传送开始错误 "，程序传送功能中，将发生监视信息等级的控制器异常 "SD 存储卡程序传送开始失败 "。

9-13-3 CPU 单元的单元版本兼容性

备份源和恢复目标的 CPU 单元版本不同时，备份文件的兼容性如下。
此外，备份、恢复前后单元版本相同时，可无限制执行恢复。

（○：执行恢复、×：不执行恢复）

CPU 单元的单元版本	恢复功能			
	SD 存储卡备份功能（恢复）从 SD 存储卡到控制器	自动传送功能、程序传送功能	Sysmac Studio 控制器备份功能（恢复）从计算机到控制器	Sysmac Studio 变量 / 存储器的备份功能（恢复）从计算机到控制器
恢复目标较新	○	○	○	○
恢复目标较旧	×	×*1	×	○

*1. 自动传送功能中，将发生全部停止故障等级的控制器异常 " 自动传送开始错误 "，程序传送功能中，将发生监视信息等级的控制器异常 "SD 存储卡程序传送开始失败 "。

9-14 无法与备份功能同时执行的功能

以下功能无法与备份功能同时执行。CPU 单元正在执行这些功能时，请勿执行备份功能。此外，正在执行备份功能时，请勿执行这些功能。

- 正在执行备份功能
- 同步传送 （计算机➞控制器）
- 执行在线编辑
- 执行存储器全部清除
- 执行时区变更
- 执行凸轮表保存指令 （MC_SaveCamTable）
- 执行 CPU 单元名称写入
- 执行从站终端的参数设定传送

10

通信设定

本章介绍 CPU 单元的在线连接以及与其他设备的连接。

10-1 通信系统的概要 10-2
10-1-1 概要 10-3
10-2 Sysmac Studio 的连接 10-5
10-2-1 可在线连接的构成 10-5
10-2-2 无法在线连接的构成 10-6
10-3 与其他控制器或从站连接 10-8
10-3-1 控制器与控制器之间的连接构成 10-8
10-3-2 控制器与从站的连接构成 10-11
10-4 与显示器和串行通信设备连接 10-12
10-4-1 与显示器的连接 10-12
10-4-2 与串行通信设备连接 10-13

10-1 通信系统的概要

下面介绍 NJ/NX 系列的通信系统概要。

在整体步骤中，与以下阴影标示的位置对应。

STEP 1. 软件设计

- STEP1-1 设计 I/O 和处理
- STEP1-2 设计任务
- STEP1-3 设计程序

↓

STEP 2. 软件设定和编程

- STEP2-1 从站 / 单元构成
- STEP2-2 控制器设定
- STEP2-3 编程
- STEP2-4 离线调试

↓

STEP 3. 安装和硬件的设定

↓

STEP 4. 配线

↓

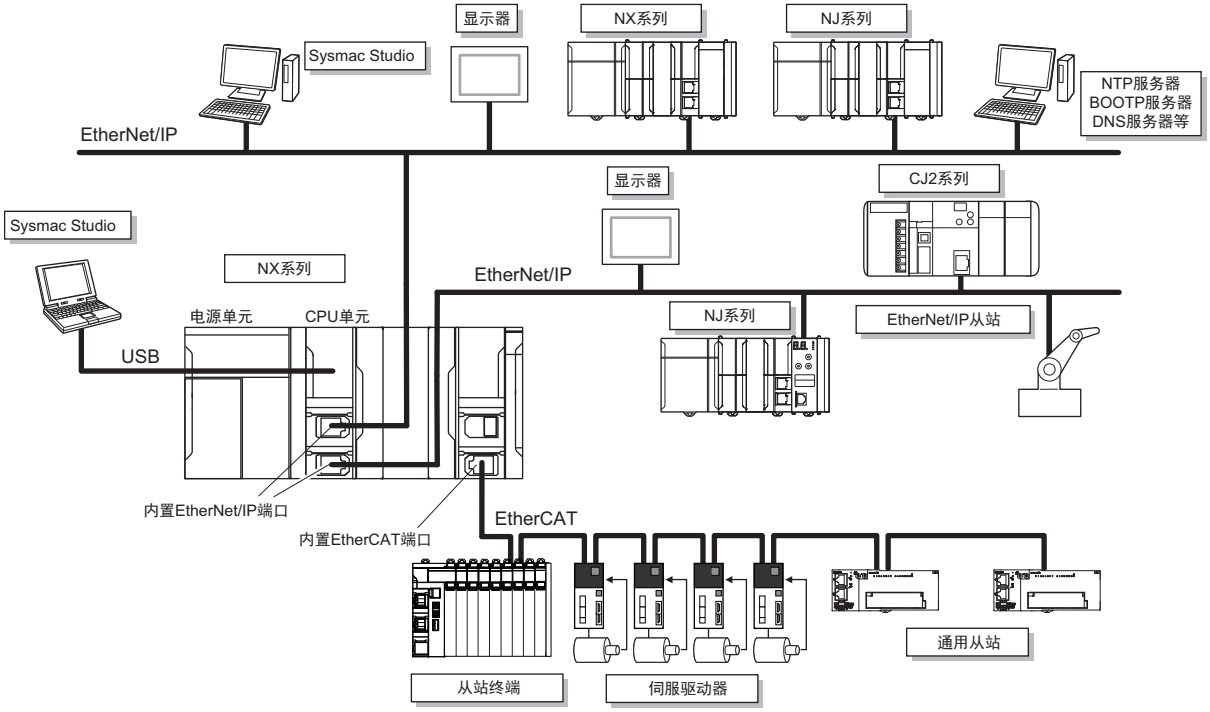
STEP 5. 上机确认动作和运行

详情请参考📖「1-3 NJ/NX 系列的整体使用步骤」（P.1-12)。

10-1-1 概要

● NX 系列系统

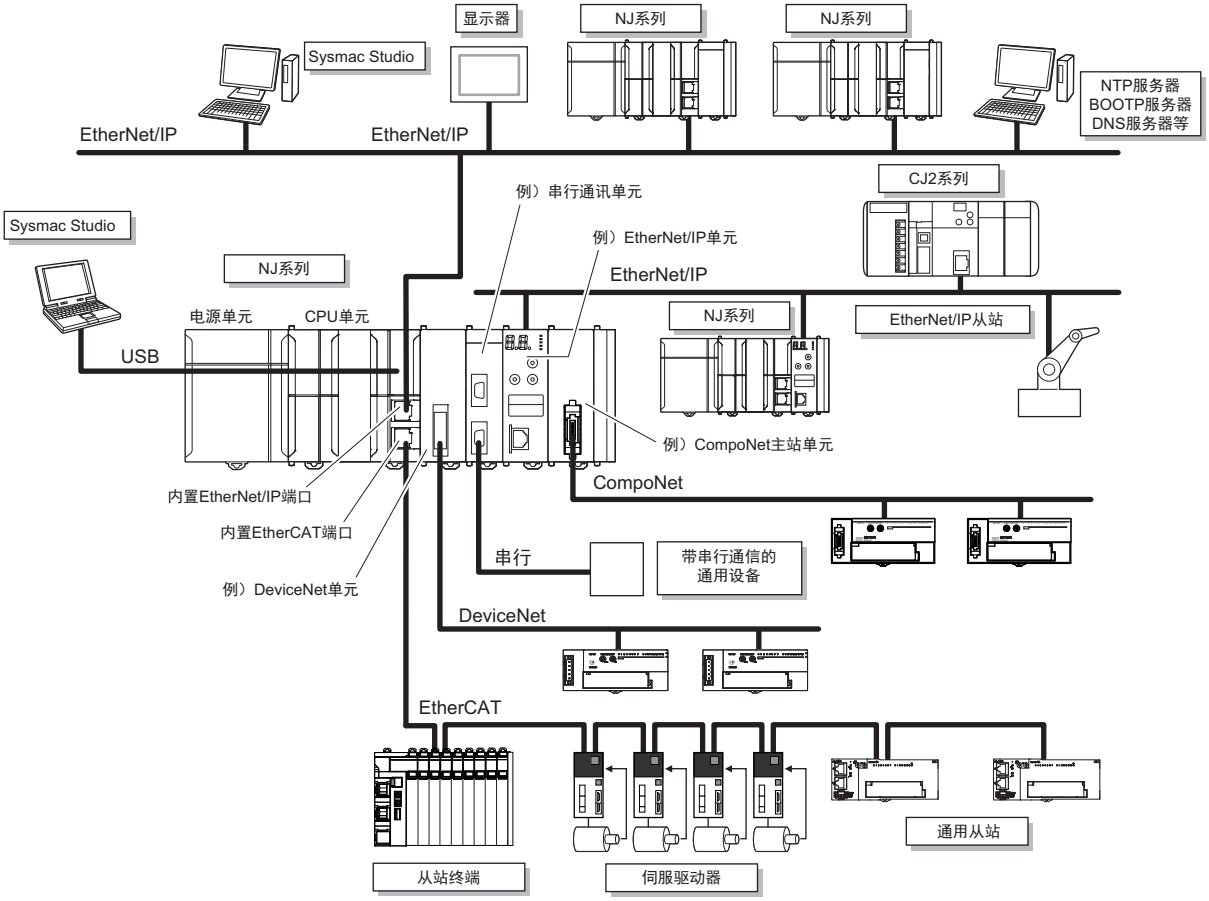
NX 系列系统可构建以下通信系统。



连接对象		连接方法
Sysmac Studio 的连接		使用 USB 或内置 EtherNet/IP 端口
控制器之间的连接	与 NJ/NX 系列控制器或 CJ2 系列 PLC 连接	使用内置 EtherNet/IP 端口
控制器与从站的连接	连接伺服驱动器、通用从站	使用内置 EtherCAT 端口
显示器的连接		使用内置 EtherNet/IP 端口
与各种服务器的连接	连接 FTP 服务器、 BOOTP 服务器、 DNS 服务器、 NTP 服务器	使用内置 EtherNet/IP 端口

● NJ 系列系统

NJ 系列系统可构建以下通信系统。



连接对象		连接方法
Sysmac Studio 的连接		使用 USB 或内置 EtherNet/IP 端口
控制器之间的连接	与 NJ 系列控制器或 CJ2 系列 PLC 连接	使用内置 EtherNet/IP 端口或 EtherNet/IP 单元 *1 的端口
控制器与从站的连接	连接伺服驱动器、通用从站	使用内置 EtherCAT 端口
	I/O 控制	安装 DeviceNet 单元，使用 DeviceNet，或安装 CompoNet 主站单元，使用 CompoNet
显示器的连接		使用内置 EtherNet/IP 端口或 EtherNet/IP 单元 *1、*2 的端口
通过串行通信连接		安装串行通讯单元
与各种服务器的连接	连接 FTP 服务器*3、 BOOTP 服务器、DNS 服务器、 NTP 服务器	使用内置 EtherNet/IP 端口或 EtherNet/IP 单元 *1 的端口

*1. 请使用单元版本 2.1 以上的 EtherNet/IP 单元。
同时， CPU 单元请使用 1.01 及以上版本， Sysmac Studio 请使用 1.02 及以上版本。
升级相关的信息请参考 「A-14 NJ 系列的版本相关信息」（P.A-180）。

*2. 请通过 CPU 单元的内置 EtherNet/IP 端口连接 NA 系列显示器。使用 NS 系列显示器的故障排除时，请通过 CPU 单元的内置 EtherNet/IP 端口连接 NS 系列显示器。

*3. FTP 服务器可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。

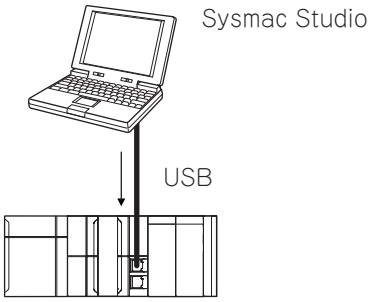
10-2 Sysmac Studio 的连接

下面介绍 Sysmac Studio 与 NJ/NX 系列的连接构成。

10-2-1 可在线连接的构成

Sysmac Studio 可通过 NJ/NX 系列 CPU 单元外设 USB 端口或内置 EtherNet/IP 端口，进行在线连接。

● 用 USB 连接

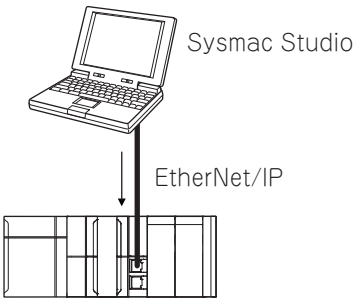


通过 Sysmac Studio 直接连接。无需指定连接设备。

（注）计算机和 CPU 单元之间请使用 USB2.0 标准的电缆，不可经过 USB 集线器连接。

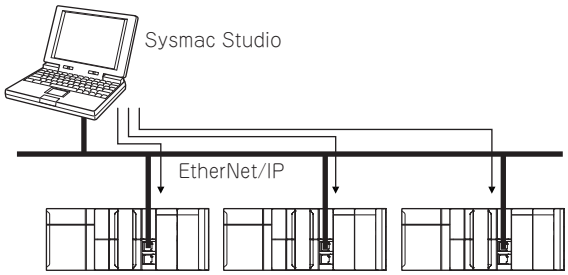
● 用 EtherNet/IP 连接

1：1 的连接：



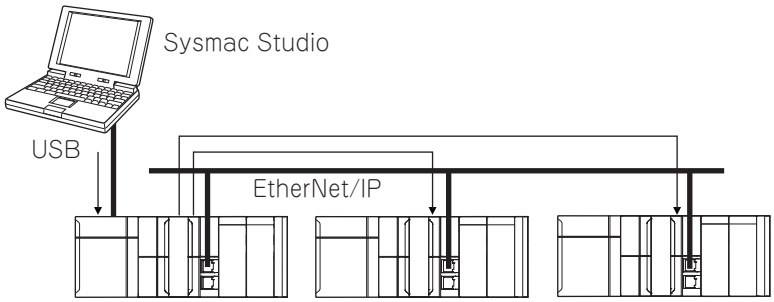
- 通过 Sysmac Studio 直接连接。无需指定 IP 地址和连接设备。
- 无论是否有交换式集线器，均可连接。
- 可使用交叉电缆、直通电缆中的任意一种。
- NX 系列 CPU 单元时，可进行 1: 1 连接的仅限端口 1。

1：N 的连接：



- 通过 Sysmac Studio 指定连接对象的 IP 地址。
- 可使用交叉电缆、直通电缆中的任意一种。

● 通过 USB 用 EtherNet/IP 连接



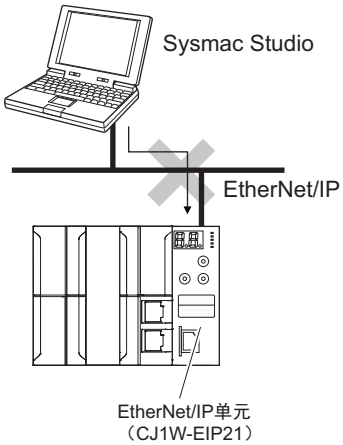
- 通过 Sysmac Studio 指定连接对象的 IP 地址。
- 可使用交叉电缆、直通电缆中的任意一种。

（注）计算机和 CPU 单元之间请使用 USB2.0 标准的电缆，不可经过 USB 集线器连接。

10-2-2 无法在线连接的构成

● 通过 EtherNet/IP 单元连接

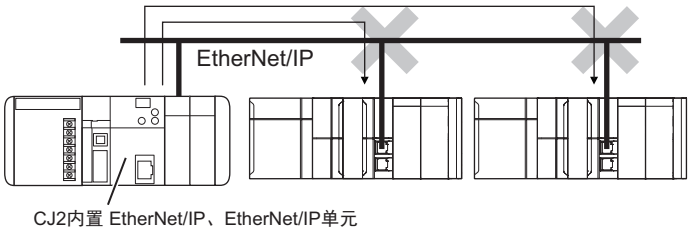
不可通过 NJ 系列 CPU 单元上连接的 EtherNet/IP 单元 （CJ1W-EIP21）连接。



（注）NX 系列 CPU 单元中，不可使用 EtherNet/IP （CJ1W-EIP21）单元。

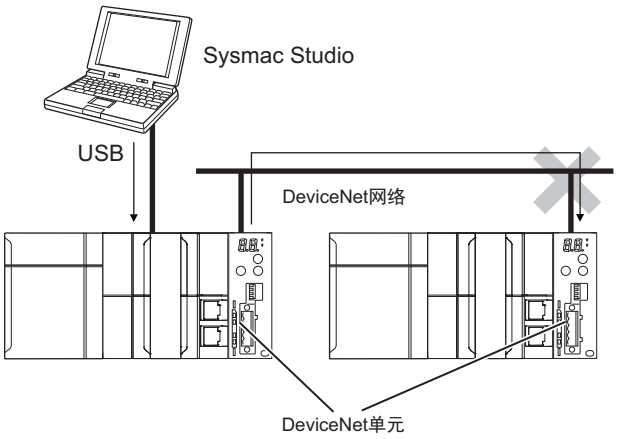
● 对 CS/CJ 系列的 EtherNet/IP 单元 / 端口进行路由

不可对 CS/CJ 系列的 EtherNet/IP 单元 / 端口（CS1W-EIP21、CJ1W-EIP21、CJ2 内置 EtherNet/IP 端口、 CJ2M 内置 EtherNet/IP 端口）进行路由后连接到 NJ/NX 系列 CPU 单元。



● DeviceNet 等 EtherNet/IP 以外的路由

不可在 EtherNet/IP 网络以外的网络上进行路由。（Controller Link 网络、 DeviceNet 网络等）



（注）NX 系列 CPU 单元中，不可连接 DeviceNet 单元。

10-3 与其他控制器或从站连接

下面介绍控制器与控制器、控制器与从站之间的连接构成。

10-3-1 控制器与控制器之间的连接构成

EtherNet/IP

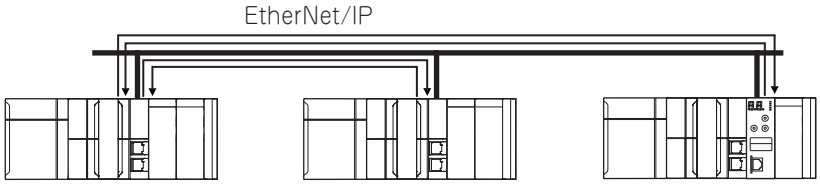
使用内置 EtherNet/IP 端口或 EtherNet/IP 单元 （CJ1W-EIP21）的端口。

内置 EtherNet/IP 端口的配线请参考📖《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册（SBCD-CN5-359）》， EtherNet/IP 单元 （CJ1W-EIP21）请参考📖《CJ 系列 EtherNet/IP 单元 用户手册 NJ 系列连接篇 （SBCD-CN5-355）》。

（注）NX 系列 CPU 单元中，不可使用 EtherNet/IP （CJ1W-EIP21）单元。

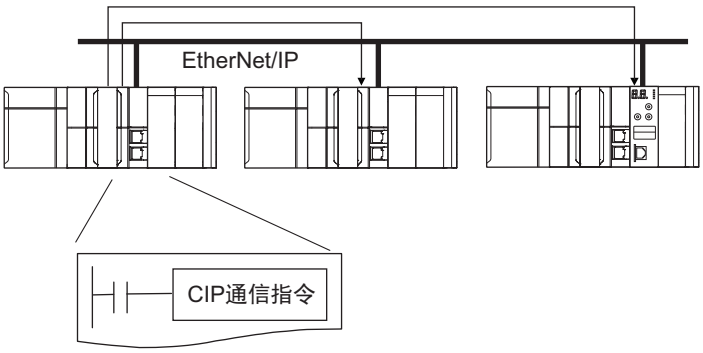
● **标签数据链接**

可与 EtherNet/IP 上的 NJ/NX 系列 CPU 单元进行标签数据链接。



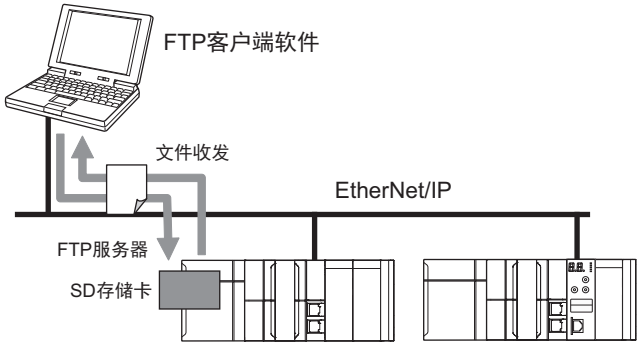
● **信息通信**

可通过程序发行 CIP 信息。



● 文件收发

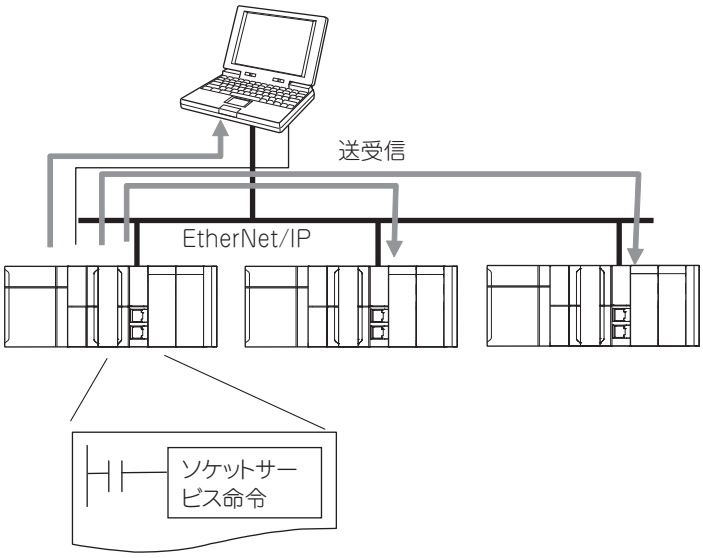
可通过 FTP 客户端软件，对 NJ/NX 系列 CPU 单元上安装的 SD 存储卡中的文件进行收发。



● Socket 服务

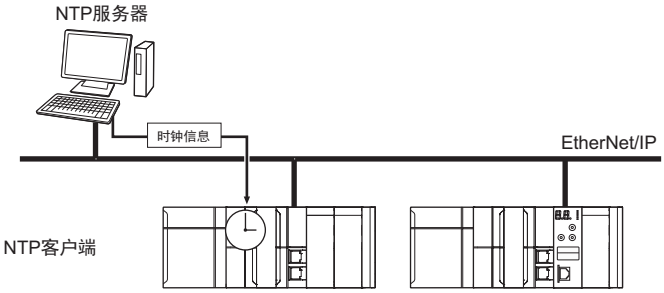
直接通过用户程序使用 TCP 或 UDP 的功能，可在主机 PC － PLC 之间或控制器之间与对象节点进行任意数据的收发。

本功能只能在内置 EtherNet/IP 端口上使用。



● 时钟信息的更新

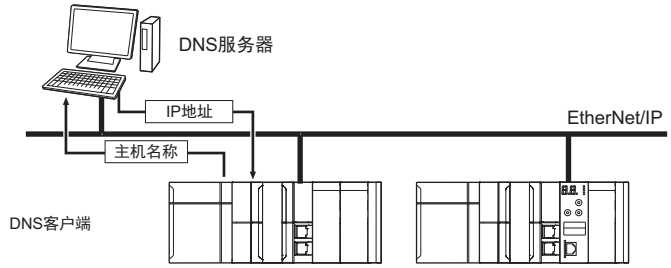
某条件成立时，可从 NTP 服务器获取时刻信息，更新内置时钟时刻。



● 以主机名称指定

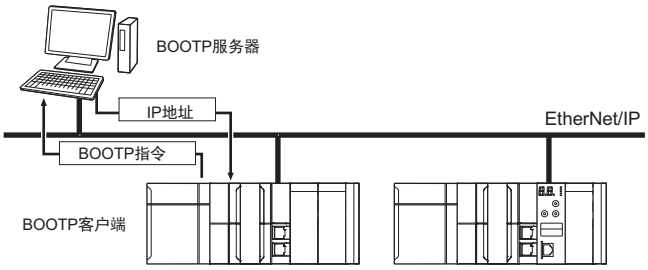
对于 NTP 服务器或 SNMP 管理器的 IP 地址或 Socket 指令、 CIP 通信指令的接收方，可用主机名称指定，而不是 IP 地址 （DNS 客户端功能或 Hosts 设定）。

例：在 DNS 服务器上设定主机名称



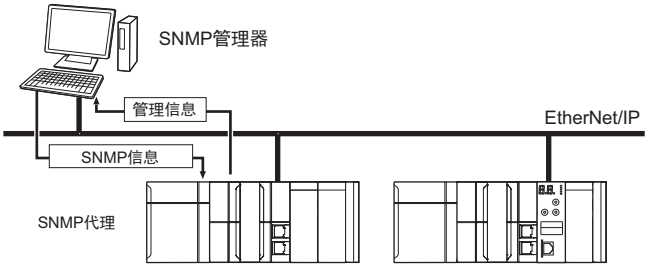
● 电源接通时 IP 地址的获取

电源接通时，可从 BOOTP 服务器获取内置 EtherNet/IP 端口的 IP 地址。



● SNMP 代理指定

向使用 SNMP 管理器的网络管理软件提供内置 EtherNet/IP 端口内部状态信息。

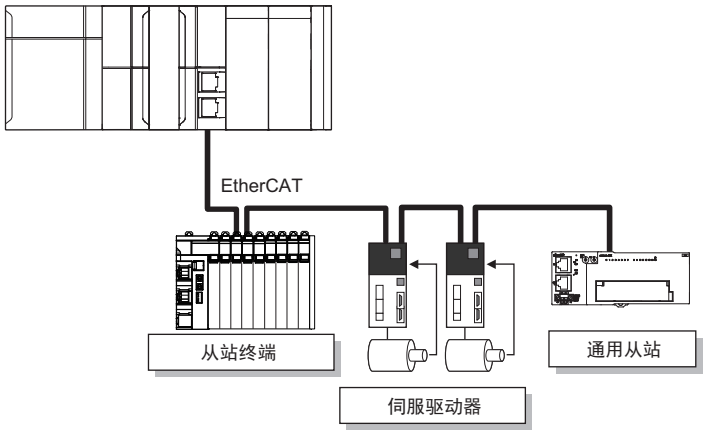


10-3-2 控制器与从站的连接构成

EtherCAT

可与伺服驱动器、通用从站进行高速、高精度通信。

详情请参考 《NJ/NX 系列 CPU 单元内置 EtherCAT 端口 用户手册 （SBCD-CN5-358）》。



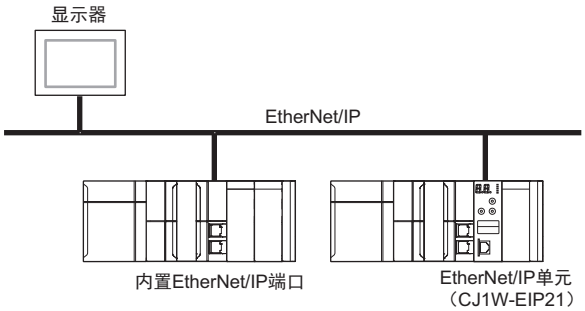
10-4 与显示器和串行通信设备连接

下面介绍显示器、串行通信设备与 NJ/NX 系列控制器的连接构成。

10-4-1 与显示器的连接

● EtherNet/IP

使用内置 EtherNet/IP 端口或 EtherNet/IP 单元 （CJ1W-EIP21）。



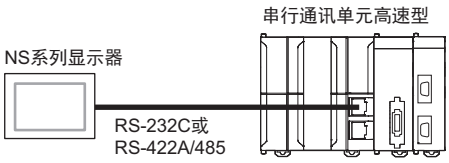
（注）NX 系列 CPU 单元中，不可使用 EtherNet/IP （CJ1W-EIP21）单元。

请通过 CPU 单元的内置 EtherNet/IP 端口连接 NA 系列显示器。
使用 NS 系列显示器的故障排除时，请通过 CPU 单元的内置 EtherNet/IP 端口连接 NS 系列显示器。

内置 EtherNet/IP 端口请参考 📖《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册（SBCD-CN5-359）》， EtherNet/IP 单元 （CJ1W-EIP21）请参考 📖《CJ 系列 EtherNet/IP 单元 用户手册 NJ 系列连接篇 （SBCD-CN5-355）》。

● 串行通信

使用串行通讯单元 （NS 系列显示器）。



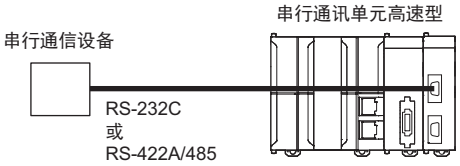
（注）NX 系列 CPU 单元中，不可使用串行通讯单元。

串行通讯单元请参考 📖《CJ 系列 串行通讯单元 用户手册 NJ 系列连接篇 （SBCD-CN5-354）》。

10-4-2 与串行通信设备连接

使用串行通讯单元。

详情请参考 《CJ 系列 串行通讯单元 用户手册 NJ 系列连接篇 （SBCD-CN5-354）》。



（注）NX 系列 CPU 单元中，不可使用串行通讯单元。

11

实际使用步骤示例

本章介绍实际使用 NJ 系列控制器时的使用步骤示例。

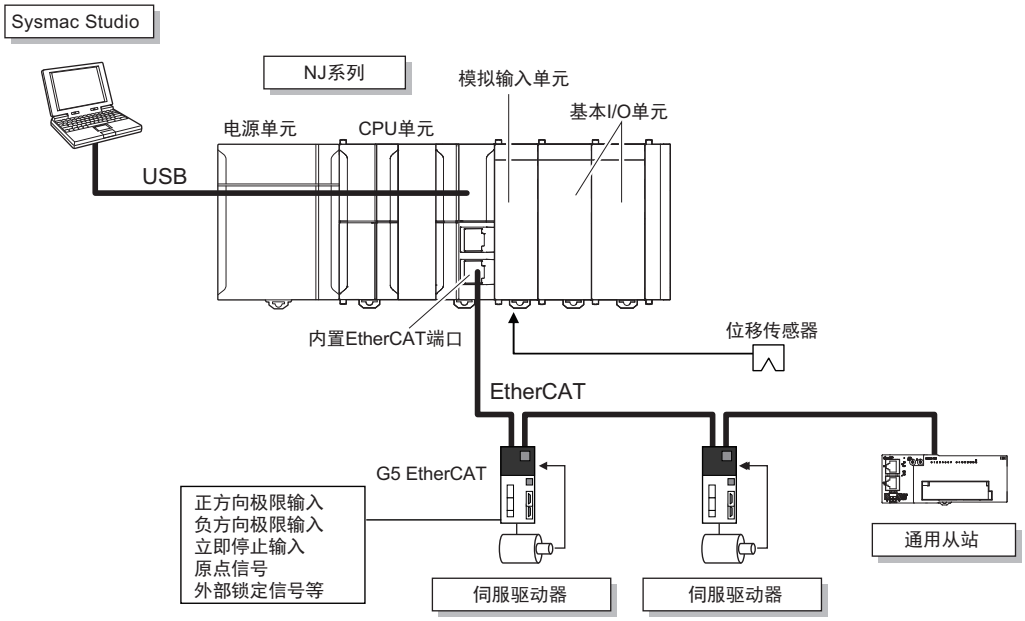
11-1 实际使用示例 11-2
11-1-1 系统构成 11-2
11-1-2 动作 11-2
11-2 操作顺序示例的概要 11-3
11-2-1 配线和设定 11-3
11-2-2 软件设计 11-3
11-2-3 Sysmac Studio 上的软件设定 11-4
11-2-4 Sysmac Studio 上的编程 11-8
11-2-5 Sysmac Studio 上的模拟 11-9
11-2-6 上机确认动作和运行 11-9

11-1 实际使用示例

下面介绍 NJ 系列的实际使用示例。

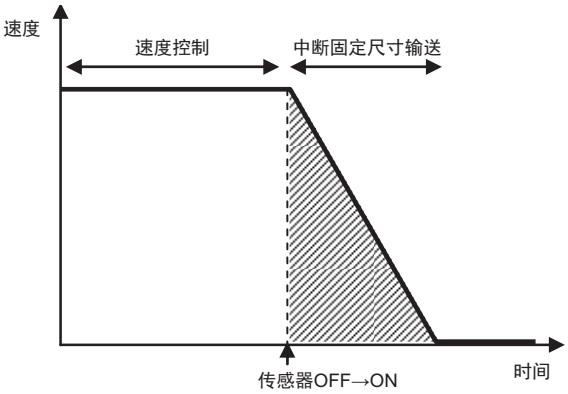
11-1-1 系统构成

单元名称		台数	连接设备
电源单元		1 台	—
CPU 单元		1 台	—
CJ 基本 I/O 单元		2 台	—
CJ 模拟输入单元		1 台	位移传感器
EtherCAT 从站	伺服驱动器 G5EtherCAT 型	2 台	—
	I/O 终端	1 台	—



11-1-2 动作

速度控制过程中，传感器信号从 OFF 变为 ON，进行中断固定尺寸输送。



根据变位传感器的模拟输入，变更纵向的位置。

11-2 操作顺序示例的概要

下面介绍 NJ 系列的实际操作步骤示例。

11-2-1 配线和设定

进行配线及硬件的设定。

11-2-2 软件设计

进行 I/O 的设计、任务设计、 POU 设计及变量设计。

I/O 的设计

- 设计外部 I/O 和单元的构成关系。
- 设定以多少间隔刷新哪个外部 I/O。

任务及 POU 的设计

研究以下内容。

- 如何设计任务构成
- 将哪个程序分配到哪种周期的任务中
- 将哪个单元分配到哪个任务中
- 将哪个处理编入程序，将哪个处理设计为功能块、函数

设计变量

研究以下内容。

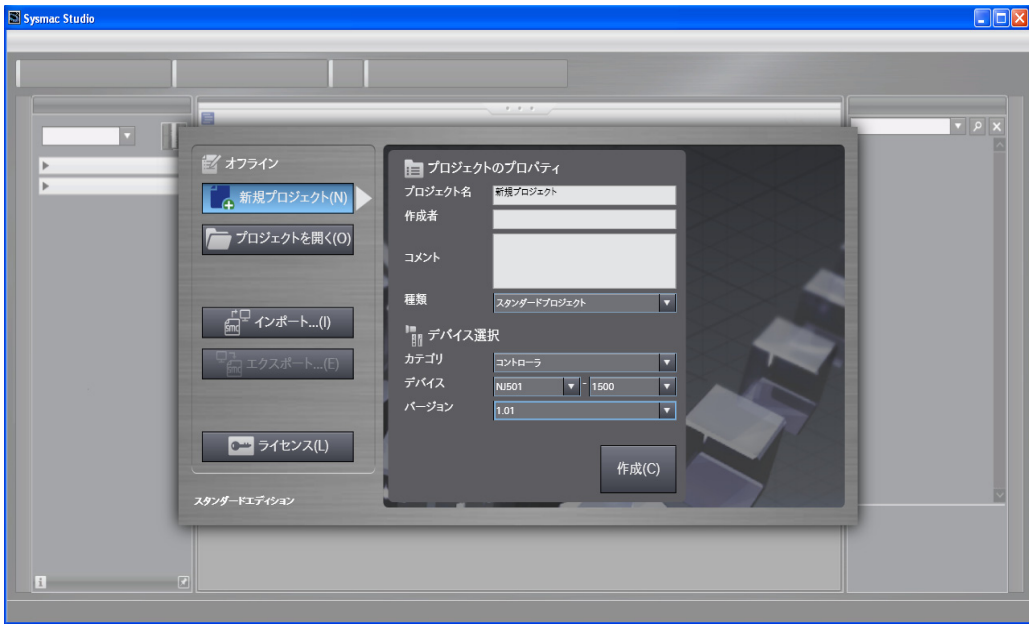
- 区分 POU 共同使用的变量 （全局变量）和各 POU 单独使用的变量 （本地变量）
- 设计用于访问从站 / 单元的设备变量之变量名称
- 设计变量的名称、保持等属性
- 设计变量的数据类型

11-2-3 Sysmac Studio 上的软件设定

在 Sysmac Studio 上，进行单元 / 从站构成、全局变量 / 设备变量的登录、轴 （轴变量）的创建及控制器 / 高功能单元设定。

启动 Sysmac Studio

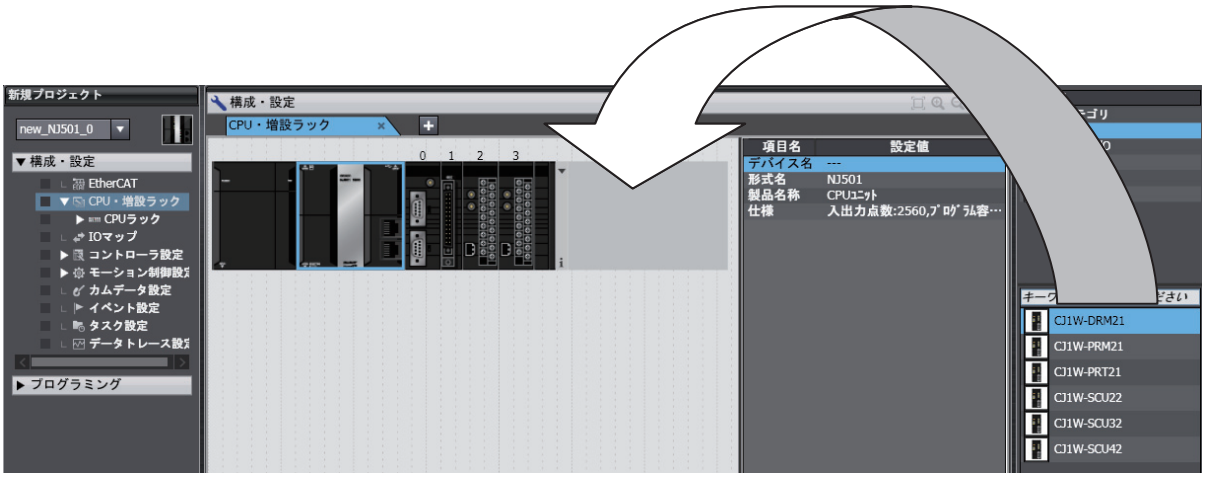
新建项目。



构成单元

1 双击［配置和设定］下的［CPU・扩展装置］。

2 拖放单元，创建单元构成。

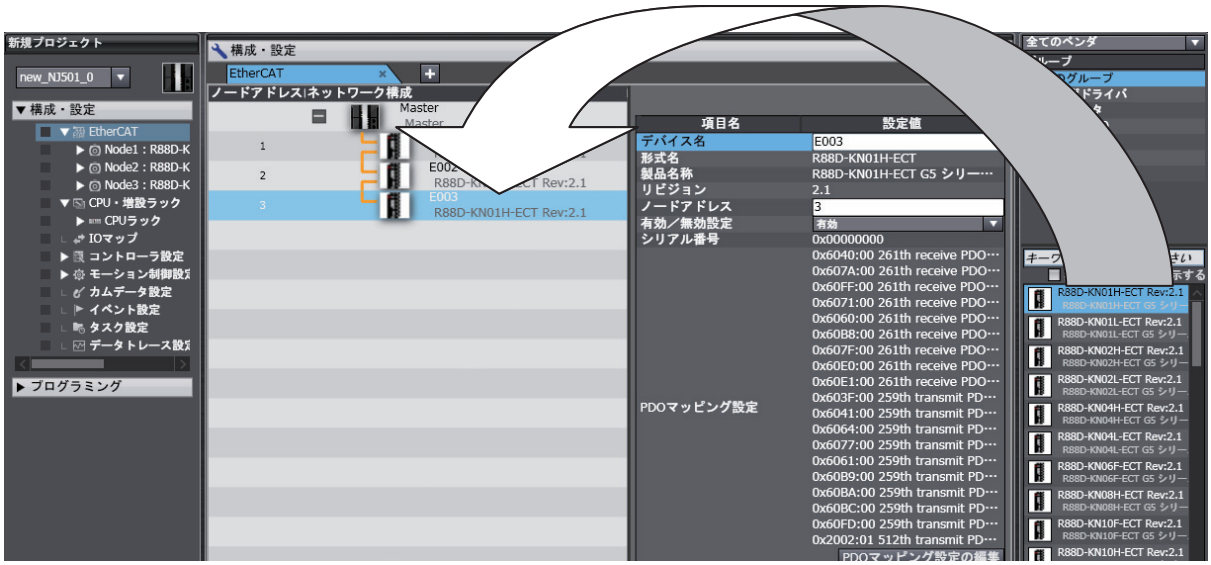


3 选择各单元，进行单元设定。

构成 EtherCAT 从站

1 双击［配置和设定］下的［EtherCAT］。

2 拖放从站，创建从站构成。



3 选择主机，设定主机参数。

4 选择各从站，设定从站的参数。

参考

此时，可通过 I/O 映射进行强制值刷新，以进行配线确认。

登录全局变量 / 设备变量。

● 全局变量的登录

1 双击 [编程]-[数据] 下的 [全局变量]。

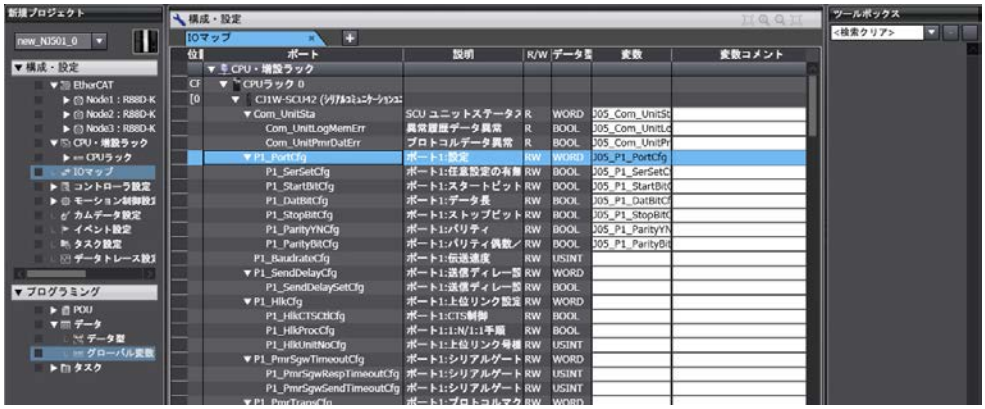
2 将全局变量登录到全局变量表中。

● 设备变量的登录

1 双击 ［配置和设定］下的 ［IO 映射］。

2 利用 I/O 映射，对 （根据单元 / 从站构成自动创建的） I/O 端口分配变量。

此外，还可通过 Sysmac Studio 自动生成设备变量名称。此时，选择 I/O 端口并右击，选择 [生成设备变量]。



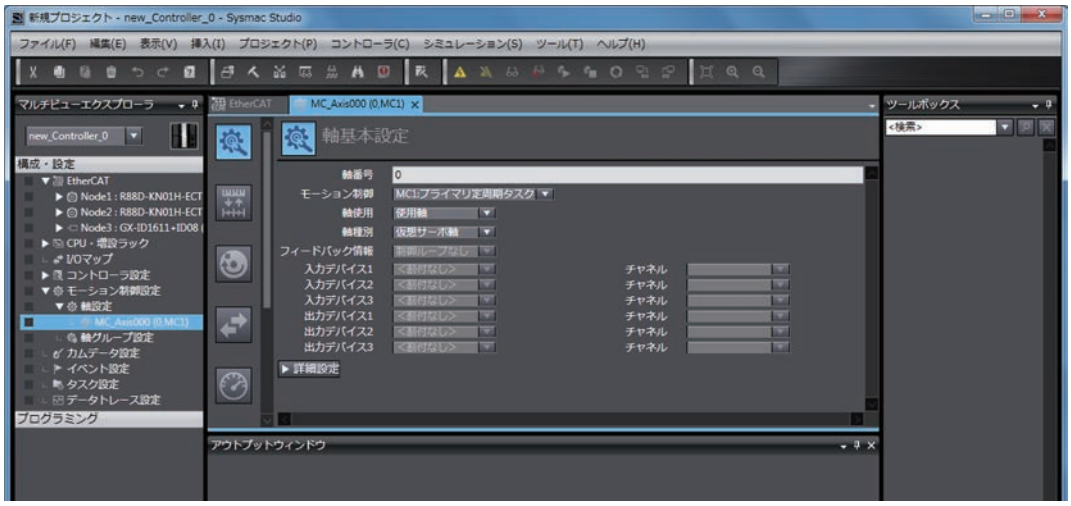
默认情况下，生成的设备变量将登录到全局变量表中。

根据需要，将 [变量种类] 从 “ 全局变量 ” 变更为各 POU 的本地变量 （内部变量）。

创建 “ 轴 ” （轴变量）

1 **选择 [配置和设定] 下的 [运动控制设定] ー [轴设定] 并右击，选择 [添加] ー [轴设定]。**

2 **对创建的轴 （轴变量）分配 EtherCAT 从站上的伺服驱动器。**



- 将 “ 轴使用 ” 选择为 “ 使用轴 ”。
- NX 系列 CPU 单元时， “ 运动控制 ” 选择为 “ 主固定周期任务 ”。
- “ 轴种类 ” 选择为 “ 伺服轴 ”。
- “ 输入设备 ”、 “ 输出设备 ” 选择为从站构成中登录的 EtherCAT 从站。

此外，设定单位转换、动作设定等轴参数。

进行控制器设定及高功能单元设定

● PLC 模块相关的初始设定：

控制器设定：电源接通时模式等

● 高功能单元的初始设定：

单元构成和设定：模拟输入单元的初始设定

11-2-4 Sysmac Studio 上的编程

通过 Sysmac Studio 进行程序创建、任务设定和链接。

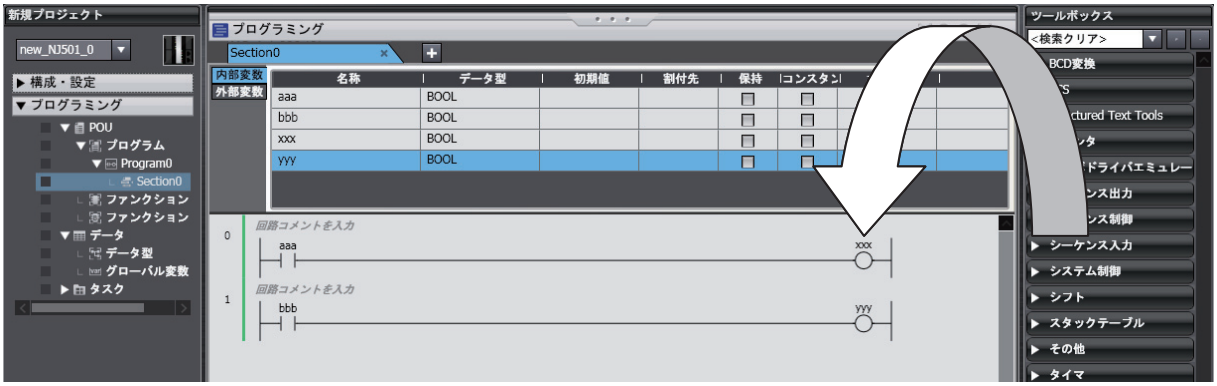
编程

1 **选择 [编程]-[POU] 下的 [程序] 并右击，选择 ［添加 ］ | [梯形图] 或 [ST]。**

2 **双击所创建 “ 程序 ” 下的 “Section □ ”。**

3 **登录各程序的本地变量。**

4 **创建程序。**



创建记述有以下指令的程序。

- 定原点：MC_Home 指令
- 速度控制：MC_MoveVelocity 指令
- 中断固定尺寸输送：MC_MoveFeed 指令
- 定位：MC_Move 指令

5 **根据需要，选择 [编程]-[POU] 下的 [函数] 或 [功能块]，选择 ［添加 ］ | [梯形图] 或 [ST]。**

双击创建的 “Function □ ” 或 “FunctionBlock □ ”。

登录各函数或功能块的本地变量。

创建各算法。

（注） 梯形图时，按 [R] 键，创建以下回路。

任务设置

双击 ［配置和设定］下的 ［任务设置］。

- 在 [任务设置] 中，从下拉菜单中选择主固定周期任务的 [周期 / 执行条件] 的任务周期。
- 在 [控制 I/O 的任务设置] 中，选择分配各单元 / 从站的任务名称。
- 在 [程序的分配设定] 中，将程序分配到主固定周期任务或固定周期任务 （执行优先度 16）中。

链接

从 [项目] 菜单中选择 [链接]。

11-2-5 Sysmac Studio 上的模拟

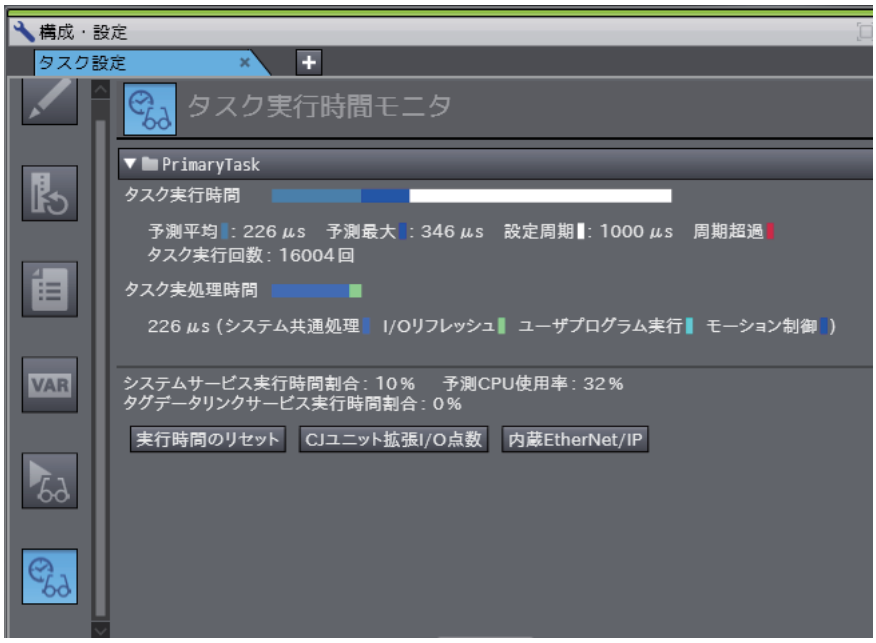
进行上机调试。
确认任务执行时间和任务实际处理时间的明细，根据需要修改任务设计。

模拟器启动和连接

从 [模拟] 菜单中选择 [以执行时间预测模式执行]。
在显示硬件修订版本的单元中，选择相应的硬件修订版本。
启动模拟器 （虚拟控制器），并自动在线连接。

通过模拟器确认任务执行时间

双击 ［配置和设定］下的 ［任务设置］。
通过 [任务执行时间监视]，确认任务执行时间是否超过任务周期。



根据需要，修改任务构成、程序分配、任务周期。

项目的保存

从 [文件] 菜单中选择 [另存为]。

11-2-6 上机确认动作和运行

在线连接后，进行项目的下载、配线确认和调试，再进行正式运行。

在线连接

1 **接通 NJ 系列 CPU 单元的电源。**

2 **用 USB 电缆连接计算机和 CPU 单元。**

3 **从 [控制器] 菜单中选择 [通信设置]。从 ［连接方法］中选择与连接形态对应的连接方法。**

4 **从 [控制器] 菜单中选择 [在线]。**

项目的下载 （利用 “ 同步 ” 功能）

从 [控制器] 菜单中选择 [同步]，将项目下载到控制器中。

（注） “ 下载 ” 和 “ 上传 ” 时使用 Sysmac Studio 的 “ 同步 ” 功能。

配线确认

通过 I/O 映射或梯形图编辑器对实际 I/O 进行强制值刷新，确认配线。

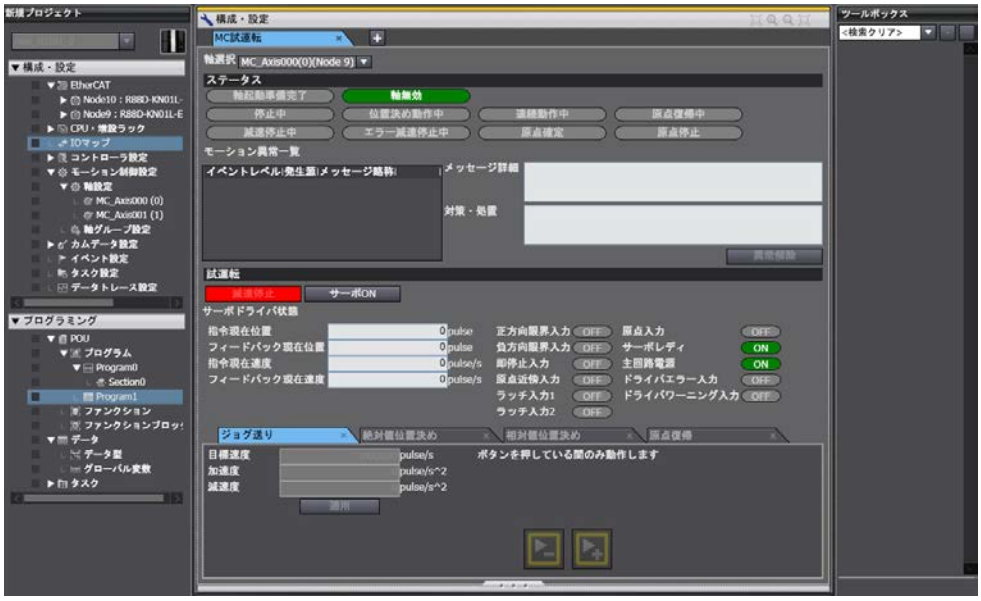
MC 调试

1 **打开 MC 调试画面。**

2 **将 CPU 单元设定为程序模式。**

3 **通过画面监视输入信号，检查配线。**

4 **通过画面进行点动进给。**



手动运行

将 CPU 单元设定为运行模式。

- 伺服 ON/OFF：执行运动控制指令的 MC_Power
- 点动进给：执行运动控制指令的 MC_MoveJog

定原点

定原点：执行 MC_Home 指令

正式运行

从 [控制器] 菜单中选择 [动作模式]-[运行模式]。

发生异常时，调查原因，修正用户程序。

12

异常的确认和处理

本章介绍发生异常时记录的事件代码、异常确认及其处理方法。

12-1 异常状态的动作 12-2
12-1-1 NJ/NX 系列的状态概要 12-2
12-1-2 CPU 单元无法动作的异常 12-4
12-1-3 CPU 单元可动作的异常 12-6

12-2 故障排除 12-13
12-2-1 确认 CPU 单元是否正在动作 12-13
12-2-2 CPU 单元可动作的异常时的处理流程 12-15
12-2-3 异常一览 12-16
12-2-4 异常内容 12-30
12-2-5 CPU 单元以外的异常及其处理 12-75

12-1 异常状态的动作

介绍 NJ/NX 系列的异常状态和各异常状态下的动作。

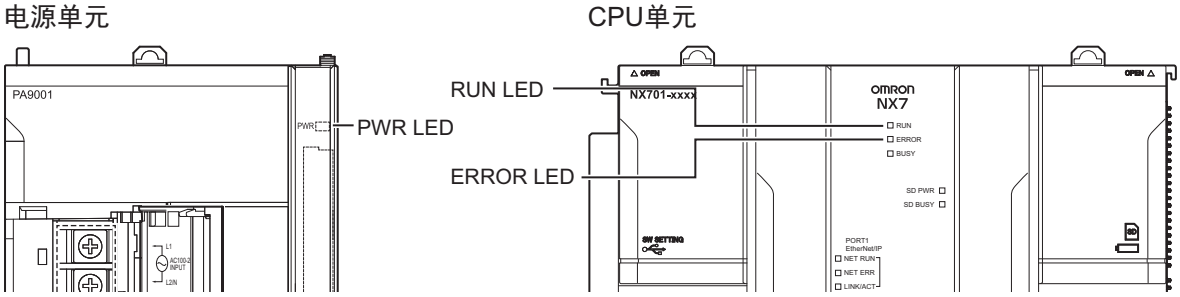
关于发生异常时的具体处理方法，请参考📖「12-2 故障排除」（P.12-13）。

此外， NJ/NX 系列整体的异常请参考📖《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。

12-1-1 NJ/NX 系列的状态概要

可通过电源单元及 CPU 单元正面上部的 LED(PWR LED, RUN LED, ERROR LED) 确认 CPU 单元的动作状态。

● NX 系列 CPU 单元



启动中、正常时、发生各异常时的正面 LED、用户程序的执行状态、与 Sysmac Studio 的在线连接及显示器的通信连接可否如下所示。

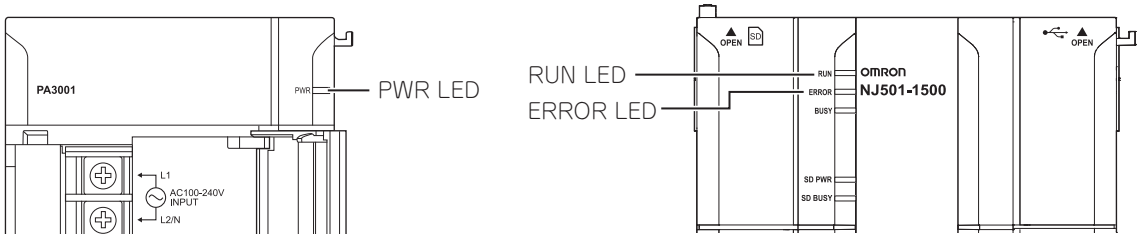
（○：亮灯 / ●：熄灯 / ◎：闪烁）

CPU 单元的动作状态		电源单元	CPU 单元		用户程序执行状态	与 Sysmac Studio 的在线连接、与显示器的通信连接
		PWR（绿色）	RUN（绿色）	ERROR（红色）		
启动中		○	◎（以 2s 为周期后以 0.5s 为周期）	●	停止	不可
正常运行中	“ 运行模式 ” 中	○	○	●	继续	可以
	“ 程序模式 ” 中	○	●	●	停止	
CPU 单元无法动作的异常	电源部异常 （*1）	●	●	●	停止	不可
	CPU 复位 （*1）	○	●	●	停止	
	CPU 异常 （*1）	○	●或◎ （以 2s 为周期 / 以 0.5s 为周期）	○	停止	
	系统初始化异常 （*1）	○	◎（以 2s 为周期）持续 30 秒以上	●	停止	
CPU 单元可动作的异常	全部停止故障 （*2）	○	●	○	停止	可以（EtherNet/IP 的功能正常运行时，显示器可进行通信连接）
	部分停止故障 （*2）	○	○	◎（周期为 1s）	继续（*3）	
	轻度故障 （*2）	○	○	◎（周期为 1s）	继续	
	监视信息 （*2）	○	○	●	继续	

*1 关于各异常，请参考📖「12-1-2 CPU 单元无法动作的异常」（P.12-4）。
*2 关于各异常，请参考📖「12-1-3 CPU 单元可动作的异常」（P.12-6）。
*3 发生异常的功能模块将停止动作。

● NJ 系列 CPU 单元

电源单元　　CPU单元



启动中、正常时、发生各异常时的正面 LED、用户程序的执行状态、与 Sysmac Studio 的在线连接及显示器的通信连接可否如下所示。

（○：亮灯 / ●：熄灯 / ◎：闪烁）

CPU 单元的动作状态		电源单元	CPU 单元		用户程序执行状态	与 Sysmac Studio 的在线连接、与显示器的通信连接
		PWR（绿色）	RUN（绿色）	ERROR（红色）		
启动中		○	◎（周期为 1s）	●	停止	不可
正常运行中	“运行模式”中	○	○	●	继续	可以
	“程序模式”中	○	●	●	停止	
CPU 单元无法动作的异常	电源部异常（*1）	●	●	●	停止	不可
	CPU 复位（*1）	○	●	●	停止	
	电源连接不正确（*1）	○	◎（周期为 3s）	○	停止	
	CPU 异常（WDT 异常）（*1）	○	●	○	停止	
CPU 单元可动作的异常	全部停止故障（*2）	○	●	○	停止	可以（EtherNet/IP 的功能正常运行时，显示器可进行通信连接）
	部分停止故障（*2）	○	○	◎（周期为 1s）	继续（*3）	
	轻度故障（*2）	○	○	◎（周期为 1s）	继续	
	监视信息（*2）	○	○	●	继续	

*1 关于各异常，请参考📖「12-1-2 CPU 单元无法动作的异常」（P.12-4）。
*2 关于各异常，请参考📖「12-1-3 CPU 单元可动作的异常」（P.12-6）。
*3 发生异常的功能模块将停止动作。

12-1-2 CPU 单元无法动作的异常

CPU 单元无法动作的异常种类

发生某些异常时， CPU 单元可能无法动作。 CPU 单元无法动作的异常如下。
发生 CPU 单元无法动作的异常时，无法与 Sysmac Studio 进行在线连接或与显示器进行通信连接。

● 电源部异常

电源未接通、超出容许电压范围或电源单元故障。

● CPU 复位

因硬件错误， CPU 单元处于停止状态。除了硬件故障之外，以下情况下也会变为此状态。

- 扩展装置的电源 OFF
- I/O 连接电缆的安装不正确
  - 连接器的 IN-OUT 接反
  - 咬合不充分
- I/O 控制单元在 CPU 装置上超过 2 台或连接到了扩展装置上

● 电源连接不正确

NJ 系列 CPU 单元上连接了 CJ 系列电源单元，因此控制器处于停止状态。

● CPU 异常 （WDT 异常）

NJ 系列 CPU 单元本体异常。因硬件故障或暂时的数据损坏引起 CPU 崩溃，导致 WDT （看门狗定时器）超时而发生异常。

● CPU 异常

NX 系列 CPU 单元本体异常。因硬件故障或暂时的数据损坏引起 CPU 崩溃而发生异常。

● 系统初始化异常

NX 系列 CPU 单元本体异常。因硬件故障导致的异常。
CPU 单元正在启动时，RUN LED 以 2 秒为周期闪烁，但这样的状态持续超过 30 秒时，将变为本异常。

CPU 单元无法动作的异常确认方法

如下所示，通过电源单元的 PWR LED、CPU 单元的 RUN、ERROR LED 显示以及可否与 Sysmac Studio 在线连接，可确认是否发生 CPU 单元无法动作的异常。

● NX 系列 CPU 单元

（○：亮灯 / ●：熄灯 / ◎：闪烁）

LED			与 Sysmac Studio 的在线连接	CPU 单元的动作状态
PWR （绿色）	RUN （绿色）	ERROR （红色）		
●	●	●	不可 （*）	电源部异常
○	●	●		CPU 复位
○	●或 ◎ （以 2s 为周期 / 以 0.5s 为周期）	○		CPU 异常
○	◎ （以 2s 为周期） 持续 30 秒以上	●		系统初始化异常

* 若要识别 “CPU 复位 ”、 “CPU 异常 ” 和 CPU 单元可动作的异常，需要确认 Sysmac Studio 的在线连接。 “ 电源部异常 ”、 “ 系统初始化异常 ” 只需 LED 即可识别，因此无需确认与 Sysmac Studio 的在线连接。

● NJ 系列 CPU 单元

（○：亮灯 / ●：熄灯 / ◎：闪烁）

LED			与 Sysmac Studio 的在线连接	CPU 单元的动作状态
PWR （绿色）	RUN （绿色）	ERROR （红色）		
●	●	●	不可 （*）	电源部异常
○	●	●		CPU 复位
○	◎ （周期为 3s）	○		电源连接不正确
○	●	○		CPU 异常 （WDT 异常）

* 若要识别 “CPU 复位 ”、 “CPU 异常 （WDT 异常） ” 和 CPU 单元可动作的异常，需要确认 Sysmac Studio 的在线连接。 “ 电源部异常 ”、 “ 电源连接不正确 ” 只需 LED 即可识别，因此无需确认与 Sysmac Studio 的在线连接。

12-1-3 CPU 单元可动作的异常

异常的重要程度

若为 CPU 单元可动作的异常，在 NJ/NX 系列中，将异常作为控制器事件管理。在控制器事件中，将根据对控制的影响大小来决定重要程度。
重要程度会在发生异常时显示在 Sysmac Studio 或显示器中。
“ 控制器事件 ” 请参考 📖《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。

- 全部停止故障 (Major fault) 等级
  控制器整体无法控制的异常。
  检测到全部停止故障时，将立即停止执行用户程序，断开包括远程 I/O 在内所有从站的负载。
  此外， EtherCAT 从站、部分 NX 单元、部分 CJ 系列高功能单元是关闭还是保持输出，可在从站 / 单元的设定中选择。
  全部停止故障等级的异常不可通过用户程序及 Sysmac Studio/ 显示器解除。排除异常原因后重新接通电源，或通过 Sysmac Studio 执行控制器重置，可以恢复。

- 部分停止故障 (Partial fault) 等级
  控制器的某些功能模块整体无法控制的异常。
  发生部分停止故障等级的异常后， NJ/NX 系列 CPU 单元将继续执行用户程序。可在用户程序中编入异常处理程序，使装置安全停止。
  排除异常原因后，可执行以下任一操作，以恢复为正常状态。
  - 通过用户程序、 Sysmac Studio 或显示器解除异常
  - 电源重新接通
  - 通过 Sysmac Studio 进行控制器重置

- 轻度故障 (Minor fault) 等级
  控制器的某些功能模块的部分功能无法控制的异常。
  发生轻度故障等级的异常后，可执行的处理与发生部分停止故障等级的异常时一样。

- 监视信息 (Observation)
  对控制器的控制没有影响的异常。
  发生监视信息时不会对控制造成影响，但仍会通知用户，避免发展成轻度故障等级以上的异常。

- 一般信息 (Information)
  除异常外向用户通知的信息。

● 各重要程度的动作一览

根据控制器事件的重要程度不同，发生异常时的动作也不同。

异常的重要程度 / 项目		控制器异常				控制器信息
		全部停止故障等级	部分停止故障等级	轻度故障等级	监视信息	一般信息
定义		控制器整体无法控制的重大异常	除 PLC 功能模块以外，某些功能模块整体无法控制的异常	某些功能模块的部分功能无法控制的异常	对控制没有影响的异常	并非异常，但仍记录到事件日志中，并向用户通知的信息
异常示例（记载有部分异常。所有异常的一览请参考📖《NJ/NX 系列故障排除手册 (SBCA-CN5-361)》		• 非易失性存储器数据损坏（PLC 功能）	• 运动控制周期超限（运动控制功能） • 通信控制器异常（EtherCAT 主机功能）	• 正方向极限输入检测（运动控制功能） • 电池电压低（PLC 功能）	• 接收缓存满，数据包丢弃（EtherNet/IP 功能）	• 接通电源 • 断开电源 • 存储器全部清除
正面 LED（*1）	PWR LED（绿色）	点亮	点亮	点亮	点亮	点亮
	RUN LED（绿色）	熄灭	点亮	点亮	点亮	点亮
	ERROR LED（红色）	点亮	闪烁（周期为 1s）	闪烁（周期为 1s）	熄灭	熄灭
NJ/NX 系列 CPU 单元的动作	电源单元的 RUN 接点	OFF	ON	ON	ON	ON
	用户程序执行状态	停止	继续（*2）	继续	继续	继续
	断开负载	有	无	无	无	无
	解除异常	不可	可或不可（取决于异常内容）	可或不可（取决于异常内容）	—	—
	事件日志	记录（有不可记录的异常）	记录	记录	记录	记录
EtherCAT 从站、基本输出单元的输出		📖参考「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）	• 在 EtherCAT 主机功能模块中发生时：遵照从站侧的设定 • 在其他功能模块中发生时：遵照程序	遵照程序	遵照程序	遵照程序
Sysmac Studio 的显示（在线连接中）		在控制器状态窗口中自动显示是否有发生中的异常。 通过用户操作在故障排除对话框中显示详细信息			发生中异常画面中不显示	

*1 发生多个控制器异常时，LED 中将显示重要程度最高的异常。

*2 发生异常的功能模块（运动控制、EtherCAT 主机、EtherNet/IP）将停止动作。

● **为各重要程度的事件（异常）发生源时，各功能模块的动作**

异常的重要程度 / 功能模块	全部停止故障等级	部分停止故障等级	轻度故障等级	监视信息
PLC 功能模块	停止用户程序的执行	—	继续动作	
运动控制功能模块	停止所有轴（停止方法因异常内容不同而异）	停止所有轴（停止方法因异常内容不同而异）	• 停止该轴 / 轴组（停止方法因设定内容不同而异） • 放弃运动指令的启动（轴动作相关的指令）	• 继续轴动作 • 放弃运动指令的启动（与轴动作无关的指令）
EtherCAT 主机功能模块	通过 EtherCAT 通信停止 I/O 刷新（从站的动作遵照从站的设定）	EtherCAT 通信停止（从站的动作遵照从站的设定）	按照主机侧的 “ 低效运行设定 ”，继续通过 EtherCAT 通信停止 I/O 刷新、或通过 EtherCAT 通信执行 I/O 刷新（I/O 刷新停止时从站的动作遵照从站侧的设定）	通过 EtherCAT 通信继续 I/O 刷新
EtherNet/IP 功能模块	EtherNet/IP 通信部分停止（可与 Sysmac Studio 进行在线连接，可与显示器进行通信连接，标签数据链接的输出（发送）标签遵照标签集的设定）	EtherNet/IP 通信停止（不可与 Sysmac Studio 进行在线连接、与显示器的进行通信连接）	EtherNet/IP 通信部分停止（原因不在 Sysmac Studio 或显示器的通信连接部分时，可进行在线连接 / 通信连接）	EtherNet/IP 通信继续

● 发生全部停止故障等级的控制器异常的输入输出动作

发生以下异常时， CPU 单元侧的动作和输入输出设备侧的动作如表所示。

- 连接不支持的单元
- I/O 总线检查错误
- 未连接端盖
- 单元 / 扩展装置连接不正确
- 单元编号重复
- I/O 点数超限
- I/O 设定检查错误

输入输出设备	CPU 单元的动作	输入输出设备的动作
NX 系列从站终端	将 NX 系列从站终端设为可安全操作的状态	遵照 NX 单元的设定
EtherCAT 从站*1	将从站设为可安全操作的状态	遵照从站的设定*2
“ 轴 ” 中分配的伺服驱动器、NX 单元	停止指令值的更新	所有轴立即停止
CJ 系列基本 I/O 单元	停止刷新	• 所有输出变为 OFF • 所有输入变为 OFF
CJ 系列高功能单元	停止刷新	遵照各单元的动作规格（ERH LED 亮灯）
EtherNet/IP 连接设备	• 为标签数据链接的始发端时，不更新输入 （接收）标签的变量或 I/O 存储器 • 为标签数据连接的目标端时，遵照输出（发送）标签的标签集设定*3	遵照连接设备的规格

*1. “ 轴 ” 中分配的伺服驱动器除外。
*2. 设定方法和设定内容取决于从站。请参考所用从站的手册。为伺服驱动器时，遵照对象 605EHex（故障指示选项代码）的设定。
*3. 可设定为清除输出数据或继续发送发生异常前的数据。详情请参考 📖《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册 (SBCD-CN5-359)》。

发生上述以外的异常时， CPU 单元侧的动作和输入输出设备侧的动作如下所示。

输入输出设备	CPU 单元的动作	输入输出设备的动作
NX 系列从站终端	将 NX 系列从站终端设为可安全操作的状态	遵照 NX 单元的设定
EtherCAT 从站*1	将从站设为可安全操作的状态	遵照从站的设定*2
“ 轴 ” 中分配的伺服驱动器、NX 单元	停止指令值的更新	所有轴立即停止
CJ 系列基本 I/O 单元	• 所有输出值清零 • 继续进行输入刷新	• 所有输出变为 OFF • 反映外部输入
CJ 系列高功能单元	继续刷新	遵照各单元的动作规格
EtherNet/IP 连接设备	• 为标签数据链接的始发端时，不更新输入 （接收）标签的变量或 I/O 存储器 • 为标签数据连接的目标端时，遵照输出（发送）标签的标签集设定*3	遵照连接设备的规格

*1. “ 轴 ” 中分配的伺服驱动器除外。
*2. 设定方法和设定内容取决于从站。请参考所用从站的手册。为伺服驱动器时，遵照对象 605EHex（故障指示选项代码）的设定。
*3. 可设定为清除输出数据或继续发送发生异常前的数据。详情请参考 《NJ/NX 系列 CPU 单元内置 EtherNet/IP 端口 用户手册 (SBCD-CN5-359)》。

CPU 单元可动作的异常确认方法

可按照以下方法确认 CPU 单元可动作的异常的发生状态。

确认方法	可确认内容
通过 LED 确认	控制器异常的重要程度、 EtherCAT 主机功能模块、 EtherNet/IP 功能模块的异常状态
通过 Sysmac Studio 的故障排除功能确认	发生中的控制器异常、过去发生的控制器异常记录、异常的发生源、异常的发生原因和处理对策、 CJ 系列高功能单元的异常记录 *1
通过显示器的故障排除确认*2	发生中的控制器异常、过去发生的控制器异常记录、异常的发生源、异常的发生原因和处理对策
通过各高功能单元的异常状态获取指令确认	发生中的控制器异常中，最重要的状态和最重要的事件代码
通过系统定义变量确认	各功能模块中发生的控制器异常的状态

*1. 不会显示异常的发生原因、处理对策等详细信息。
*2. 使用显示器的故障排除时，请通过 CPU 单元的内置 EtherNet/IP 端口连接显示器。关于显示器的故障排除应对范围，请参考 《NJ/NX 系列 故障排除手册 (SBCA-CN5-361)》的附录。

下面介绍各种确认方法。

通过 LED 确认

● 确认控制器异常的重要程度

根据电源单元的 PWR LED 和 CPU 单元的 RUN、 ERROR LED 的显示，可确认异常的重要程度。LED 的显示和异常的重要程度关系如下。

（○：亮灯 / ●：熄灯 / ◎：闪烁）

LED			异常的重要程度
PWR （绿色）	RUN （绿色）	ERROR （红色）	
○	●	○	全部停止故障等级
○	○	◎ （周期为 1s）	部分停止故障等级
			轻度故障等级
○	○	●	监视信息

● 确认 EtherCAT 端口、 EtherNet/IP 端口的状态

使用 EtherCAT 端口及 EtherNet/IP 端口时，通过 EtherCAT 及 EtherNet/IP 的各 NET ERR LED 显示，可确认是否发生会影响过程数据的异常或轻度故障等级以上的异常。
可根据 LED 的显示来确认的状态如下。

LED	可确认的状态
EtherCAT NET ERR	EtherCAT 端口的状态 ・亮灯： 发生通过用户处理无法恢复为正常状态的异常 （需要更换 CPU 单元或联系本公司经销商 / 代理店的异常） ・闪烁： 发生通过用户处理可恢复为正常状态的异常 ・熄灯： 未发生会影响过程数据的异常
EtherNet/IP NET ERR	EtherNet/IP 端口的状态 ・亮灯： 发生通过用户处理无法恢复为正常状态的异常 （需要更换 CPU 单元或联系本公司经销商 / 代理店的异常） ・闪烁： 发生通过用户处理可恢复为正常状态的异常 ・熄灯： 未发生轻度故障等级以上的异常

通过 Sysmac Studio 的故障排除功能确认

发生异常时，将 Sysmac Studio 在线连接到控制器上，可确认发生中的异常、过去发生的异常记录。

● 发生中的异常

在 Sysmac Studio 的 “ 控制器异常 ” 标签中，可确认发生中的异常的 “ 重要程度 ”、“ 发生源 ”、“ 发生源详情 ”、 “ 事件名称 ”、 “ 事件代码 ”、 “ 详细信息 ”、 “ 附属信息 1 ～ 4”、 “ 处理・对策 ”。
“ 监视信息 ” 的异常不显示。

● 过去发生的异常记录

在 Sysmac Studio 的 “ 控制器异常 ” 标签中，可确认过去发生的异常的 “ 日期时间 ”、 “ 重要程度 ”、 “ 发生源 ”、“ 发生源详情 ”、“ 事件名称 ”、“ 事件代码 ”、“ 详细信息 ”、“ 附属信息 1 ～ 4”、“ 处理･对策 ”。

CJ 系列高功能单元中的异常记录也将显示于 “ 控制器事件日志 ” 标签中。此时，不会显示详细信息。若要确认详细信息，请根据显示的事件代码，参考各单元的手册中记载的错误代码的说明。错误代码和事件代码的关系记载于 📖《NJ/NX 系列 故障排除手册 (SBCA-CN5-361)》中。

关于 Sysmac Studio 的故障排除功能，请参考 📖《Sysmac Studio Version 1 操作手册（SBCA-CN5-362）》。

通过显示器的故障排除确认

发生异常时，若显示器和控制器建立通信，可确认发生中的异常、过去发生的异常记录。
使用显示器的故障排除时，请通过 CPU 单元的内置 EtherNet/IP 端口连接显示器。

使用注意事项

关于显示器的故障排除应对范围，请参考📖《NJ/NX 系列 故障排除手册 (SBCA-CN5-361)》的附录。

● 发生中的异常

可确认发生中的异常的 " 事件名称 "、" 事件代码 "、" 重要程度 "、" 发生源 "、" 发生源详情 "、" 详细信息 "、" 附属信息 1 ～ 4"。
" 监视信息 " 的异常不显示。

● 过去发生的异常记录

可确认过去发生的异常的 " 发生日期时间 "、" 重要程度 "、" 发生源 "、" 事件名称 "、" 事件代码 "、" 详细信息 "、" 附属信息 1 ～ 4"。

关于显示器的故障排除请参考各显示器的手册。

通过各高功能单元的异常状态获取指令确认

获取用户程序上的异常状态时，使用各功能模块上预备的异常状态获取指令。
可获取发生中的最重要的状态和最重要的事件代码。

对象功能模块	指令名称	指令
PLC 功能模块	获取 PLC 异常状态	GetPLCError
	获取 I/O 总线异常状态	GetCJBError
运动控制功能模块	获取运动控制异常状态	GetMCError
EtherCAT 功能模块	获取 EtherCAT 异常状态	GetECError
EtherNet/IP 功能模块	获取 EtherNet/IP 异常状态	GetEIPError

关于异常状态获取指令，请参考 📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》。

通过系统定义变量确认

参考系统定义变量中的 " 异常状态变量 "，可确认控制器中发生的异常的状态。
异常状态变量可介由通信从外部参考状态。
关于系统定义变量，请参考📖「A-4 系统定义变量一览」（P.A-38）。

12-2 故障排除

下面介绍 NJ/NX 系列发生异常时基本的确认和处理流程，以及 PLC 功能模块中发生的软件异常及处理。

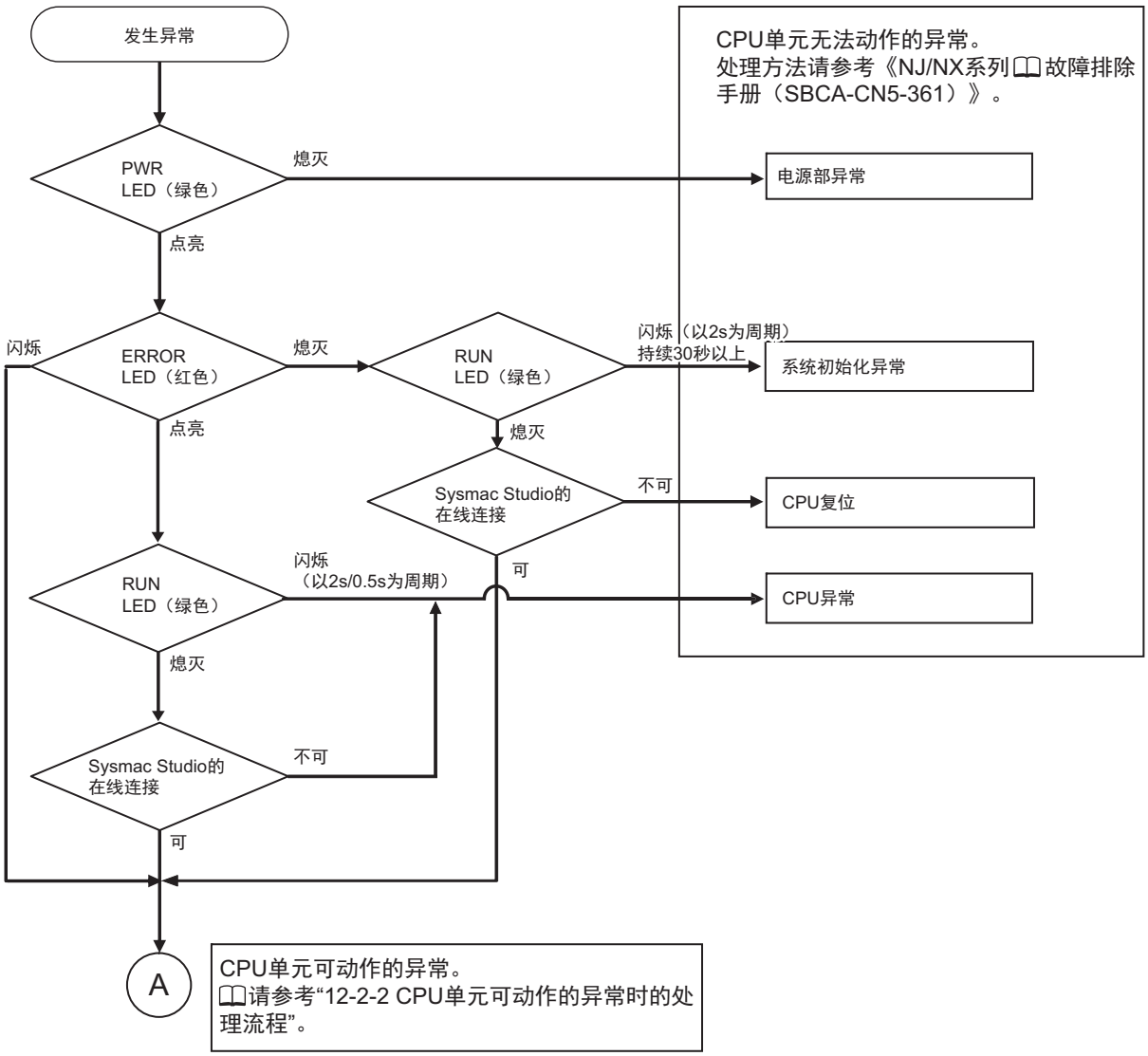
12-2-1 确认 CPU 单元是否正在动作

NJ/NX 系列发生异常时，请按照以下流程，先确认是 “CPU 单元可动作的异常 ” 还是 “CPU 单元无法动作的异常 ”。

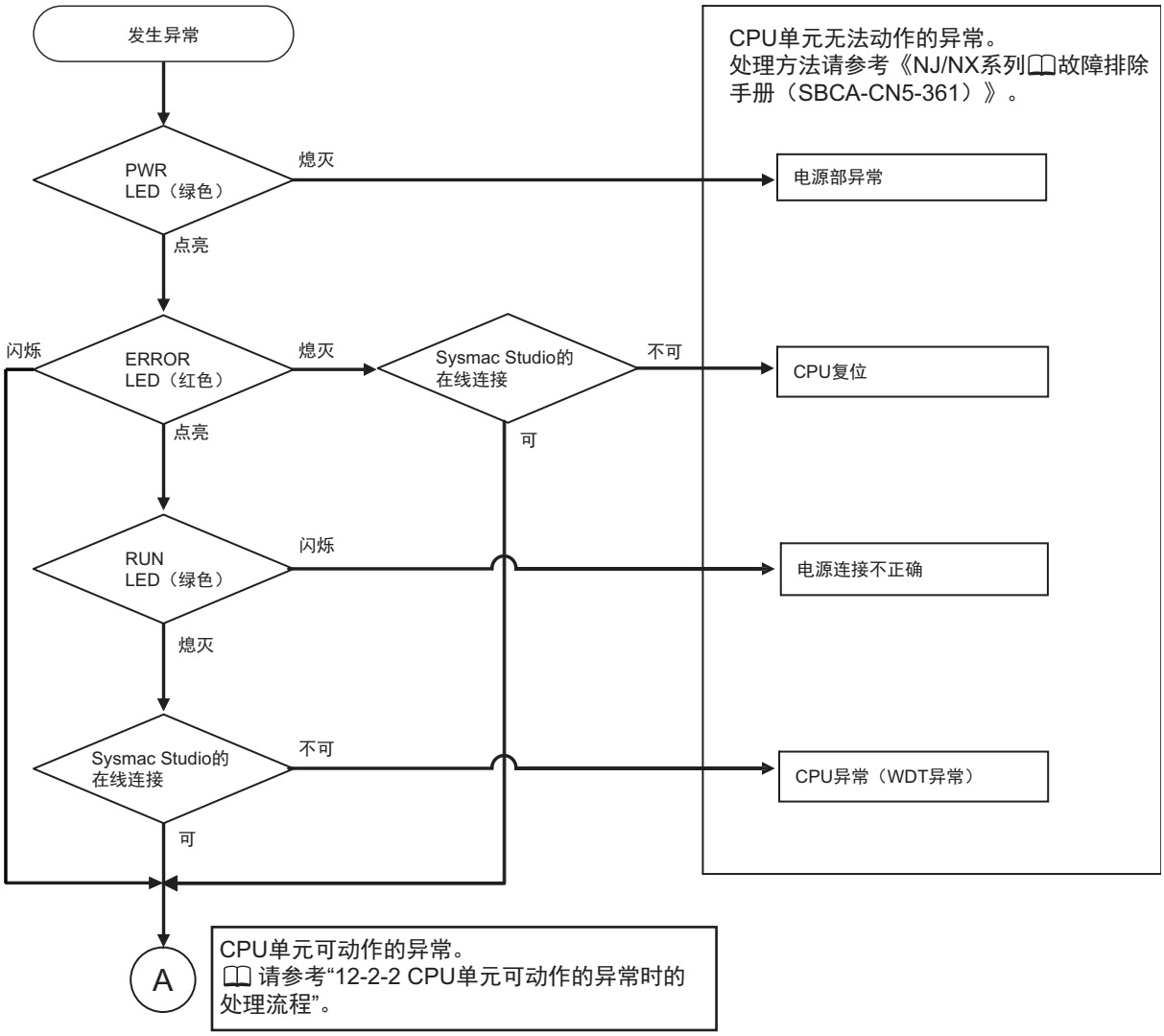
流程中的 “Sysmac Studio 的在线连接 ” 请尽量采用 “USB- 直接连接 ”。使用 Ethernet 时，有很多因素可能造成 Sysmac Studio 无法在线连接，判断是否是 CPU 单元可动作的异常需要一定的时间。

Sysmac Studio 无法在线连接时，在判断为 CPU 单元无法动作的异常前，请确认📖《NJ/NX 系列　故障排除手册 (SBCA-CN5-361)》的 “2-4 Sysmac Studio 无法在线连接时的处理方法 ” 中记载的内容。

● NX 系列 CPU 单元时



● NJ 系列 CPU 单元时

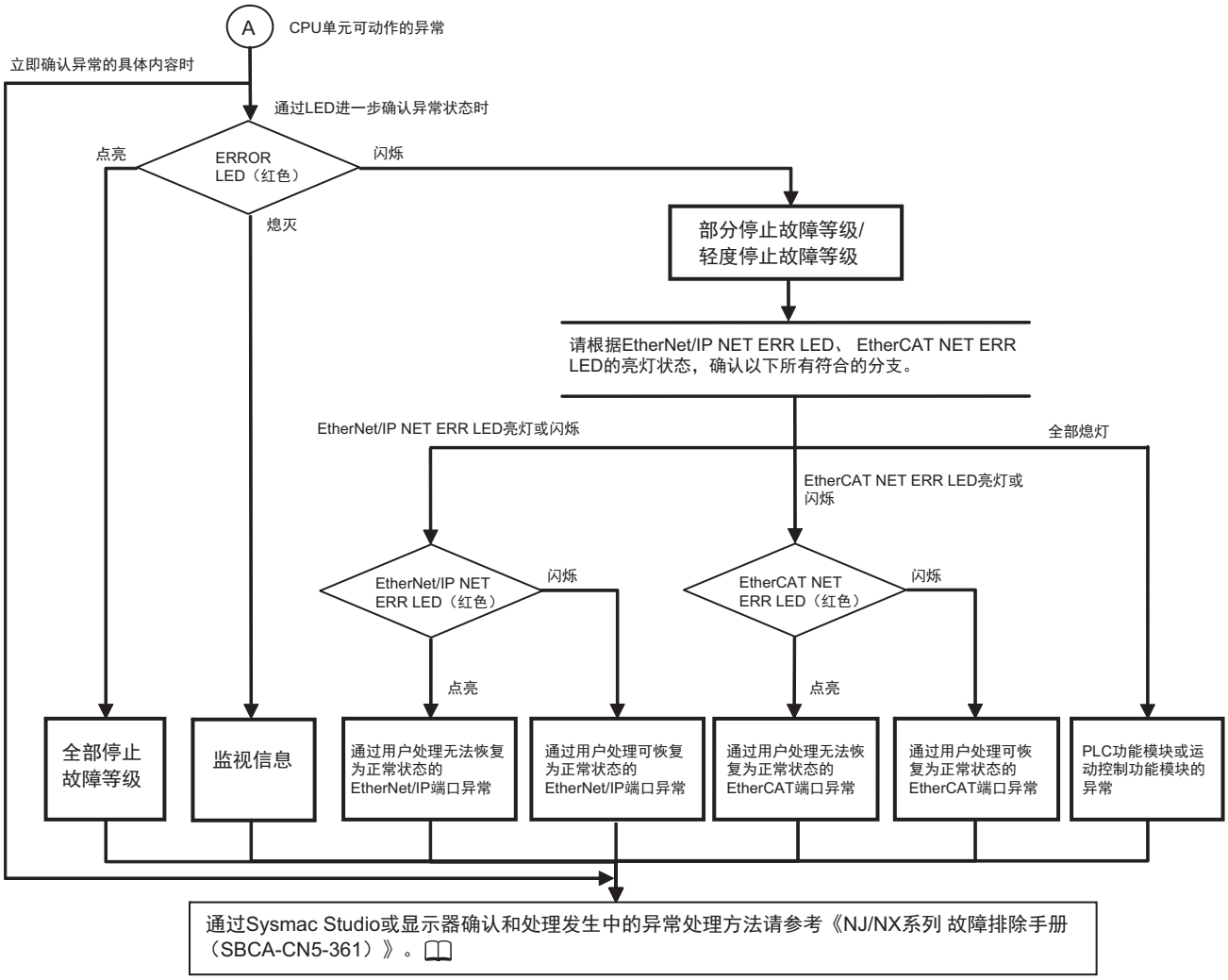

发生异常
PWR
LED（绿色）
熄灭
点亮
电源部异常
ERROR
LED（红色）
闪烁
熄灭
点亮
Sysmac Studio的
在线连接
不可
可
CPU复位
RUN
LED（绿色）
闪烁
熄灭
电源连接不正确
Sysmac Studio的
在线连接
不可
可
CPU异常（WDT异常）
CPU单元无法动作的异常。
处理方法请参考《NJ/NX系列故障排除手册（SBCA-CN5-361）》。
A
CPU单元可动作的异常。
请参考“12-2-2 CPU单元可动作的异常时的处理流程”。

12-2-2 CPU 单元可动作的异常时的处理流程

若为 “CPU 单元可动作的异常 ”，如以下流程所示，请通过 Sysmac Studio 或显示器确认发生的异常，并进行处理。

可通过 LED 确认以下内容。

- 重要程度
- EtherNet/IP 端口或 EtherCAT 端口是否发生异常
- 发生源为 EtherNet/IP 端口或 EtherCAT 端口时，是否是可以通过用户处理恢复为正常状态的异常



使用注意事项

关于显示器的故障排除应对范围，请参考《NJ/NX 系列 故障排除手册 (SBCA-CN5-361)》的附录。

12-2-3 异常一览

PLC 功能模块中发生的软件异常 （事件）及包括软件异常在内的控制器操作异常 （事件）一览如下所示。表中的重要程度含义如下。

全：全部停止故障等级
部：部分停止故障等级
轻：轻度故障等级
监：监视信息
般：一般信息

事件代码栏的 （ ）中表示限定发生该事件的 CPU 单元时， CPU 单元的版本。

关于 NJ/NX 系列的所有事件代码，请参考《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。

任务相关

事件代码	事件名称	内容	发生原因 （推测原因）	重要程度					参照处
				全	部	轻	监	般	
60020000Hex	任务执行超时	任务执行超出超时检测时间	• 超时检测时间的设定过短 • 任务周期的设定过短 • 用户程序的规模较大 • 处理的重复次数为大于预期的值 • 任务优先度错误 • 事件任务的执行频率较高	○					P.12-31
60030000Hex	I/O 刷新周期超限	连续发生在主固定周期任务、固定周期任务的任务周期内无法完成 I/O 刷新的状态	• 任务周期的设定过短 • 任务周期、事件任务的任务优先度错误 • 在任务周期内进行 I/O 刷新的单元 / 从站数较多 • 事件任务的执行频率较高	○					P.12-32
60040000Hex	系统服务时间不足	无法确保设定的系统服务执行时间	• 没有任务执行、标签数据链接服务的空闲时间 • 系统服务监视设定中，系统服务执行间隔过短，或系统服务执行时间比例过长	○					P.12-33

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
60010000Hex	任务周期超限	主固定周期任务、固定周期任务中，在设定周期内，任务未执行完成	• 任务周期的设定过短 • 用户程序的规模较大 • 处理的重复次数为大于预期的值 • 任务周期、事件任务的任务优先度错误 • 事件任务的执行频率较高			○			P.12-34
60050000Hex	任务周期超限	主固定周期任务、固定周期任务中，在设定周期内，任务未执行完成	• 任务周期的设定过短 • 用户程序的规模较大 • 处理的重复次数为大于预期的值 • 任务周期、事件任务的任务优先度错误 • 事件任务的执行频率较高				○		P.12-35

控制器操作相关

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
10200000Hex	用户程序 / 控制器配置和设定传送错误	无法正确传送用户程序或控制器配置和设定	• 下载用户程序或控制器配置和设定的过程中发生控制器断电，导致程序或控制器配置和设定不正确 • 在线编辑的操作过程中发生控制器断电，导致程序或控制器配置和设定不正确 • 存储器全部清除的过程中发生控制器断电，导致程序或控制器配置和设定不正确 • 恢复过程中发生控制器断电，导致程序或控制器配置和设定不正确 • 非易失性存储器故障	○					P.12-36
10210000Hex	用户程序执行用 ID 不正确	用户程序和 CPU 单元中设定的用户程序执行用 ID 不一致	• 用户程序及CPU单元中设定的用户程序执行用 ID 不一致 • CPU 单元中设定了用户程序执行用 ID，但用户程序中未设定	○					P.12-37
10240000Hex	不正确的用户程序	用户程序不正确	• 功能块/函数的嵌套层级超出限制	○					P.12-37

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
10250000Hex	用户程序 / 控制器配置和设定不正确	使用存储器超出上限，用户程序或控制器配置和设定数据损坏	• 数据量超出上限值 • 主存储器容量超限 • 非易失性存储器老化、故障	○					📖 P.12-38
10270000Hex（Ver.1.03 以上）	开始自动传送错误	自动传送的执行前检查时检测到异常	• 未安装 SD 存储卡 • SD 存储卡种类不正确 • SD 存储卡的格式不正确 • SD 存储卡中不存在 “autoload” 文件夹 • SD 存储卡的 “autoload” 文件夹中不存在备份文件 • SD 存储卡的 “autoload” 文件夹中的备份文件损坏或必须要传送的数据在 SD 存储卡的备份文件中不存在 • 传送对象的CPU单元的单元版本比 SD 存储卡的备份文件的单元版本旧 • 传送对象的 CPU 单元型号与 SD 存储卡的备份文件的 CPU 单元型号不一致 • 执行了 SD 存储卡的修复 • CPU 单元的写入保护功能有效 • 自动传送指令文件（AutoloadCommand.ini）的记述内容不正确 • SD 存储器故障或格式不正确，读取自动传送对象数据失败 • SD 存储卡发生故障 • 传送对象的 CPU 单元 DB 连接服务的版本比 SD 存储卡的备份文件 DB 连接服务的版本旧 • 传送对象的 CPU 单元的 Robot 版本比 SD 存储卡的备份文件的 Robot 版本旧	○					📖 P.12-39

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
10280000Hex （Ver.1.03 以上）	自动传送执行错误	自动传送异常结束	• 读取自动传送对象数据失败 • 自动传送过程中拔出了 SD 存储卡 • SD 存储卡的“autoload”文件夹中不存在备份文件 • SD 存储卡的“autoload”文件夹中的备份文件损坏 • SD 存储卡发生故障	○					P.12-41

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
10330000Hex（Ver.1.11 以上）	SD 存储卡程序传送执行前错误	SD 存储卡程序传送的执行前检查时检测到异常	• 未安装 SD 存储卡 • SD 存储卡种类不正确 • SD 存储卡的格式不正确 • SD 存储卡中不存在系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹 • SD 存储卡中，系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中不存在备份文件 • SD 存储卡的系统定义变量 _Card1PrgTransferCmd.DirName 中指定文件夹的备份文件损坏或必须要传送的数据在 SD 存储卡的备份文件中不存在 • 传送对象的CPU单元的单元版本比 SD 存储卡的备份文件的单元版本旧 • 传送对象的 CPU 单元型号与 SD 存储卡的备份文件的 CPU 单元型号不一致 • CPU 单元的写入保护功能有效 • 系统定义变量 _Card1PrgTransferCmd 中，必须传送的文件未在传送对象内 • SD 存储器故障或格式不正确，读取自动传送对象数据失败 • SD 存储卡发生故障 • 传送对象的 CPU 单元 DB 连接服务的版本比 SD 存储卡的备份文件 DB 连接服务的版本旧 • 传送对象的 CPU 单元的 Robot 版本比 SD 存储卡的备份文件的 Robot 版本旧	○					📖 P.12-42

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
10340000Hex（Ver.1.11 以上）	SD 存储卡程序传送执行错误	SD 存储卡程序传送异常结束	• 读取 SD 存储卡程序传送对象数据失败 • 传送 SD 存储卡程序的过程中拔出了 SD 存储卡 • SD 存储卡中，系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中不存在备份文件 • SD 存储卡中，系统定义变量 _Card1PrgTransferCmd.DirName 中指定文件夹中的备份文件损坏 • SD 存储卡发生故障	○					📖 P.12-45
40110000Hex	PLC 功能处理异常	PLC 功能模块中检测到致命异常	• 软件发生错误	○					📖 P.12-46
40160000Hex（Ver.1.02 以下）	安全模式	以安全模式启动了控制器	• 将CPU单元的拨动开关设定为安全模式并打开了控制器的电源	○					📖 P.12-46
44420000Hex（Ver.1.05 以上）	PLC 功能处理异常	PLC 功能模块中检测到致命异常	• 软件发生错误	○					📖 P.12-47
40120000Hex	PLC 功能处理异常	PLC 功能模块中检测到致命异常	• 软件发生错误		○				📖 P.12-47
40130000Hex	PLC 功能处理异常	PLC 功能模块的一部分检测到致命异常	• 软件发生错误			○			📖 P.12-48
10230000Hex	事件日志保存失败	保存事件日志失败	• 因电池电压低导致断电保持失败				○		📖 P.12-48
10260000Hex	追踪设定传送失败	传送追踪设定的过程中发生断电	• 传送追踪设定的过程中断电				○		📖 P.12-48

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
10290000Hex（Ver.1.03 以上）	备份开始失败	备份的执行前检查时检测到异常	• 未安装 SD 存储卡 • SD 存储卡种类不正确 • SD 存储卡的格式不正确 • SD 存储卡有写保护 • “禁止向SD存储卡备份”设定为“禁止”时 • 其他备份功能正在运行 • 正在执行同步、在线编辑或存储器全部清除 • 通过用户操作中止了备份 • 与 Sysmac Studio 的在线连接断开 • SD 存储卡发生故障				○		📖 P.12-49
102A0000Hex（Ver.1.03 以上）	备份执行失败	备份异常结束	• SD 存储卡容量不足 • 保存备份对象数据失败 • 备份过程中拔出了 SD 存储卡 • 单元 / 从站备份失败 • 通过用户操作中止了备份 • 正在执行凸轮表保存指令或 CPU 单元名称更新 • 与 Sysmac Studio 的在线连接断开 • 向计算机保存备份对象数据失败 • SD 存储卡发生故障				○		📖 P.12-51

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
102B0000Hex（Ver.1.03 以上）	恢复开始失败	恢复的执行前检查时检测到异常	• 未安装 SD 存储卡 • SD 存储卡种类不正确 • SD 存储卡的格式不正确 • SD 存储卡中不存在备份文件 • SD 存储卡的备份文件损坏或必须要恢复的数据在 SD 存储卡的备份文件中不存在 • 恢复对象的CPU单元的单元版本比 SD 存储卡的备份文件的单元版本旧 • 恢复对象的 CPU 单元型号与 SD 存储卡的备份文件的 CPU 单元型号不一致 • 执行了 SD 存储卡的修复 • CPU 单元的写入保护功能有效 • 恢复指令文件（RestoreCommand.ini）的记述内容不正确 • 其他备份功能正在运行 • 正在执行同步、在线编辑或存储器全部清除 • 与 Sysmac Studio 的在线连接断开 • SD 存储器故障或格式不正确，读取恢复对象数据失败 • SD 存储卡发生故障 • 恢复对象的 CPU 单元 DB 连接服务的版本比 SD 存储卡的备份文件 DB 连接服务的版本旧 • 恢复对象的 CPU 单元的 Robot 版本比 SD 存储卡的备份文件的 Robot 版本旧				○		P.12-53

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
102C0000Hex（Ver.1.03 以上）	恢复执行失败	恢复异常结束	• 读取恢复对象数据失败 • 恢复过程中拔出了 SD 存储卡 • 单元 / 从站恢复失败 • SD 存储卡发生故障				○		📖 P.12-55
10320000Hex（Ver.1.11 以上）	SD 存储卡程序传送开始失败	SD 存储卡程序传送的开始前检查时检测到异常	• “SD存储卡程序传送的有效/无效设定 ” 为无效 • SysmacStudio 中设定的 “ 执行 SD 存储卡程序传送时密码 ” 和系统定义变量 _Card1PrgTransferCmd.Password 不一致 • CPU 单元的拨动开关设定不是可开始 SD 存储卡程序传送的设定 • 未安装 SD 存储卡 • SD 存储卡种类不正确 • SD 存储卡的格式不正确 • SD 存储卡中不存在系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹 • SD 存储卡中，系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中不存在备份文件 • SD 存储卡的系统定义变量 _Card1PrgTransferCmd.DirName 中指定文件夹的备份文件损坏或必须要传送的数据在 SD 存储卡的备份文件中不存在 • 传送对象的CPU单元的单元版本比 SD 存储卡的备份文件的单元版本旧 • 传送对象的 CPU 单元型号与 SD 存储卡的备份文件的 CPU 单元型号不一致 • CPU 单元的写入保护功能有效				○		📖 P.12-56

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
			• 其他备份功能正在运行 • 正在执行同步、在线编辑或存储器全部清除 • 系统定义变量 _Card1PrgTransferCmd 中，必须传送的文件未在传送对象内 • SD 存储器故障或格式不正确，读取 SD 存储卡程序传送对象数据失败 • SD 存储卡发生故障 • 传送对象的 CPU 单元 DB 连接服务的版本比 SD 存储卡的备份文件 DB 连接服务的版本旧 • 传送对象的 CPU 单元的 Robot 版本比 SD 存储卡的备份文件的 Robot 版本旧						
40140000Hex	PLC 系统信息	PLC 功能模块的内部信息	• PLC 功能模块的内部信息。记录为其他事件的追加信息。				○		P.12-59
40170000Hex（Ver.1.03 以上）	安全模式	以安全模式启动了控制器	• 将CPU单元的拨动开关设定为安全模式并打开了控制器的电源				○		P.12-59
80230000Hex（Ver.1.05 以上）	NX 信息通信异常	信息通信中发生异常	• 通信电缆断线 • 通信电缆的连接器松脱 • NX 信息通信的负载过高				○		P.12-60
40150000Hex	PLC 系统信息	PLC 功能模块的内部信息	• PLC 功能模块的内部信息。记录为其他事件的追加信息。					○	P.12-60
44430000Hex（Ver.1.05 以上）	PLC 系统信息	PLC 功能模块的内部信息	• PLC 功能模块的内部信息。记录为其他事件的追加信息。					○	P.12-61
90010000Hex	时刻的变更	变更了时刻	• 变更了时刻					○	P.12-61
90020000Hex	时区的变更	变更了时区	• 变更了时区					○	P.12-61
90050000Hex（Ver.1.10 以上）	用户程序 / 控制器配置和设定的下载	下载了用户程序 / 控制器配置和设定	• 下载了用户程序/控制器配置和设定					○	P.12-62

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
90070000Hex（Ver.1.10 以上）	在线编辑传送	在线编辑了用户程序	• 在线编辑用户程序并反映到了控制器中					○	P.12-62
90080000Hex	强制值刷新的设置	指定了强制值刷新的设置	• 用户指定了强制值刷新的设置					○	P.12-63
90090000Hex	强制值刷新的重置	指定了强制值刷新的重置	• 用户指定了强制值刷新的重置					○	P.12-63
900A0000Hex	强制值刷新的全部解除	指定了强制值刷新的全部解除	• 用户指定了强制值刷新的全部解除					○	P.12-63
900B0000Hex	存储器全部清除	执行了存储器全部清除	• 拥有管理员权限的用户执行了存储器全部清除					○	P.12-64
900C0000Hex	事件日志的清除	清除了事件日志	• 用户清除了事件日志					○	P.12-64
900F0000Hex（Ver.1.03 以上）	自动传送完成	自动传送已完成	• 自动传送已完成					○	P.12-64
90110000Hex	接通电源	电源已接通	• 电源已接通					○	P.12-65
90120000Hex	断开电源	电源已断开	• 电源已断开					○	P.12-65
90130000Hex	运行开始	已开始运行	• 受理了运行开始					○	P.12-65
90140000Hex	运行停止	已停止运行	• 受理了运行停止					○	P.12-65
90150000Hex	执行复位	执行了复位	• 受理了复位指示					○	P.12-66
90160000Hex	写入用户程序执行用 ID	CPU 单元中设定或变更了用户程序执行用 ID	• 拥有管理员权限的用户变更了 CPU 单元的用户程序执行用 ID					○	P.12-66
90180000Hex	发生中控制器异常的全部解除	发生中的异常已全部解除	• 用户解除了所有发生中的异常					○	P.12-66
90190000Hex	强制值刷新的解除	指定了强制值刷新的解除	• 用户指定了强制值刷新的解除					○	P.12-67
901A0000Hex（Ver.1.03 以上）	备份开始	已开始备份	• 已开始备份					○	P.12-67
901B0000Hex（Ver.1.03 以上）	备份结束	备份已正常结束	• 备份已正常结束					○	P.12-67
901C0000Hex（Ver.1.03 以上）	恢复开始	已开始恢复	• 已开始恢复					○	P.12-68
901D0000Hex（Ver.1.03 以上）	恢复完成	恢复已正常结束	• 恢复已正常结束					○	P.12-68
90200000Hex（Ver.1.11 以上）	SD 存储卡程序传送开始	已开始 SD 存储卡程序传送	• 已开始 SD 存储卡程序传送					○	P.12-68

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
90210000Hex（Ver.1.11 以上）	SD 存储卡程序传送完成	已完成 SD 存储卡程序传送	• 已完成 SD 存储卡程序传送					○	P.12-69

FINS 相关

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
14010000Hex	CPU 总线单元系统设定区域错误	CPU 总线单元系统设定区域的存储器检查中检测到错误	• 下载CPU总线单元系统设置的过程中发生控制器断电 / 与 Sysmac Studio 的通信切断			○			📖 P.12-70
34100000Hex	IP 地址表设定错误	IP 地址表设定错误	• IP地址转换方式设定为“并用方式 ”、 “IP 地址表方式 ”，但 IP 地址表的设定错误			○			📖 P.12-70
34130000Hex	FINS/TCP 连接表设定错误	FINS/TCP 连接表错误	• 下载 FINS/TCP 连接表的过程中发生控制器断电 / 与 Sysmac Studio 的通信切断			○			📖 P.12-71
34110000Hex	接收方节点不详	接收方节点不详	• 传送 FINS 信息时，未找到接收方节点				○		📖 P.12-71
80100000Hex	信息包废弃	废弃了信息包	• 接收了发送至 CPU 单元的 FINS 响应 • FINS 响应接收单元不存在				○		📖 P.12-71
80110000Hex	信息包废弃	废弃了信息包	• 欲发送超过 2002byte 的 FINS 响应 • 路由了超过 2002byte 的 FINS 响应 • 接收了路由错误 “ 无该号机 ” 的信息包 • 接收了路由错误 “ 路由异常 ” 的信息包 • 接收了路由错误 “ 路由表未登录 ” 的信息包 • 接收了路由错误 “ 事件区域大小超限 ” 的信息包 • 内部缓存的可用空间变大 • 受高负载影响，FINS 信息的路由失败				○		📖 P.12-72

事件代码	事件名称	内容	发生原因（推测原因）	重要程度					参照处
				全	部	轻	监	般	
80120000Hex	信息包废弃	废弃了信息包	• 接收了接收方网络地址 (DNA)= 本网络、接收方节点地址 (DA1) ≠本节点的 FINS 响应 • 接收了接收方网络地址 (DNA) ≧ 80Hex 的集线器网络地址指定的 FINS 指令或 FINS 响应 • 内部缓存的可用空间变大 • 接收了未满最小指令长度的 FINS 指令 • 接收了超过最大指令长度的 FINS 指令 • 信息包发送失败 • 受高负载影响，FINS 信息的路由失败或者接收了发送源网络地址 (SNA)=0、发往内置 EtherNet/IP 端口的指令 • 接收了发往内置EtherNet/IP端口的 FINS 响应 • 未登录路由表时，接收了 FINS 响应或无需响应的指令 • 发生路由表异常时，接收了 FINS 响应或无需响应的指令 • 接收了超出中继次数的FINS响应或无需响应的指令 • 路由表中未设定发送地址，无法发送 • [内置EtherNet/IP端口设定]的 [FINS 节点地址设定] 为 0 或 255，无法路由				○		📖 P.12-73

12-2-4 异常内容

下面介绍各异常的内容。

控制器异常说明的解释

［ ］内表示表中各项目的含义，用于解释各种异常。

<table>
<tr><td>事件名称</td><td colspan="3">［异常 （事件）的名称］</td><td>事件代码</td><td colspan="2">［异常 （事件）的代码］</td></tr>
<tr><td>内容</td><td colspan="6">［异常 （事件）的内容］</td></tr>
<tr><td>发生位置</td><td colspan="2">［发生异常 （事件）的位置］</td><td>发生位置详情</td><td>［发生位置的详细信息］</td><td>检测时序</td><td>［检测异常的时序］</td></tr>
<tr><td>异常属性</td><td>重要程度</td><td>［对控制的影响程度］（*1）</td><td>恢复方法</td><td>［恢复方法］（*2）</td><td>日志类别</td><td>［所保存日志的种类］（*3）</td></tr>
<tr><td>产生的影响</td><td>用户程序</td><td>［用户程序的执行状态］（*4）</td><td>动作</td><td colspan="3">［与发生异常 （事件）时的动作相关的特殊事项］</td></tr>
<tr><td>LED</td><td colspan="6">内置 EtherNet/IP 端口用 LED、内置 EtherCAT 端口用 LED 的显示状态。仅当发生位置为 EtherCAT 主机功能模块、 EtherNet/IP 功能模块时记载］</td></tr>
<tr><td rowspan="2">系统
定义变量</td><td>变量名称</td><td colspan="2">数据类型</td><td colspan="3">名称</td></tr>
<tr><td colspan="6">［检测异常的系统定义变量、受异常影响的系统定义变量、引起异常的系统定义变量的变量名称和数据类型、名称］</td></tr>
<tr><td rowspan="2">发生原因和
处理措施</td><td>发生原因 （推测原因）</td><td colspan="2">处理对策</td><td colspan="3">防止再发</td></tr>
<tr><td colspan="6">［异常 （时间）的发生原因、处理对策和防止再发方法］</td></tr>
<tr><td>附属信息</td><td colspan="6">［Sysmac Studio ／显示器上显示的附属信息内容］ (*5)</td></tr>
<tr><td>注意事项 / 备注</td><td colspan="6">［其他注意事项、限制事项、补充说明等。有用户可选择的重要程序时，对可选择的重要程度、恢复方法、动作等相关事项进行记载］</td></tr>
</table>

*1 以下任一
全部停止故障：全部停止故障等级
部分停止故障：部分停止故障等级
轻度故障：轻度故障等级
监视信息
一般信息

*2 以下任一
自动恢复：排除原因后，自动恢复正常
异常解除：排除原因并执行异常解除后恢复正常
电源再接通：排除原因并重新接通控制器的电源后恢复正常
控制器重置：排除原因并重置控制器后恢复正常
取决于发生原因：根据发生原因而定

*3 以下任一
系统：系统事件日志
访问：访问事件日志

*4 以下任一
继续：继续执行用户程序
停止：停止执行用户程序
开始：开始执行用户程序

*5 关于显示器的故障排除应对范围，请参考📖《NJ/NX 系列 故障排除手册 (SBCA-CN5-361)》的附录。

任务相关

事件名称	任务执行超时			事件代码	60020000Hex	
内容	任务执行超出超时检测时间					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*1		
系统定义变量	变量名称		数据类型		名称	
	_< 任务名称 >_Exceeded		BOOL		任务周期超出标志	
	_< 任务名称 >_ExceedCount		UDINT		任务周期超出次数	
	_< 任务名称 >_LastExecTime		TIME		上次任务执行时间	
	_< 任务名称 >_MaxExecTime		TIME		任务执行时间最大值	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	超时检测时间的设定过短		请将超时时间变更为较大的值		请预先考虑处理对策的内容，再设计任务。	
	任务周期的设定过短		请将任务周期变更为较大的值			
	用户程序的规模较大		请分割任务内的处理，如将不需要以短周期执行的处理移动到优先度较低的周期任务中			
	处理的重复次数为大于预期的值		如果程序中发现重复次数异常大的位置，请修正程序，使重复次数变正确。要确认重复次数，应在用户程序中设置捕捉器，监视执行次数			
	任务优先度错误		请将该任务设定为较高的任务优先度。或者将其他任务设定为较低的任务优先度			
	事件任务的执行频率较高		请减小事件任务的执行频率。或者将事件任务设定为较低的任务优先度			
附属信息	附属信息 1：发生异常的任务名称					
注意事项 / 备注	无					

*1. 详情请参考「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	I/O 刷新周期超限			事件代码	60030000Hex	
内容	连续发生在主固定周期任务、固定周期任务的任务周期内无法完成 I/O 刷新的状态					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*1		
系统定义变量	变量名称		数据类型		名称	
	_< 任务名称 >_Exceeded		BOOL		任务周期超出标志	
	_< 任务名称 >_ExceedCount		UDINT		任务周期超出次数	
	_< 任务名称 >_LastExecTime		TIME		上次任务执行时间	
	_< 任务名称 >_MaxExecTime		TIME		任务执行时间最大值	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	任务周期的设定过短		请确认任务执行时间，将任务周期变更为与之相符的值		请预先考虑处理对策的内容，再设计任务。	
	任务周期、事件任务的任务优先度错误		请将周期任务设定为较高的任务优先度。或者将事件任务的任务优先度设定为低于周期任务			
	在任务周期内进行 I/O 刷新的单元 / 从站数较多		请分割任务内的 I/O 刷新处理，如将任务内的 I/O 刷新处理移动到其他任务中			
	事件任务的执行频率较高		请减小事件任务的执行频率。或者将事件任务设定为较低的任务优先度			
附属信息	附属信息 1：发生异常的任务名称					
注意事项 / 备注	无					

*1. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	系统服务时间不足			事件代码	60040000Hex	
内容	无法确保设定的系统服务执行时间					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*1		
系统定义变量	变量名称		数据类型		名称	
	_< 任务名称 >_Exceeded		BOOL		任务周期超出标志	
	_< 任务名称 >_ExceedCount		UDINT		任务周期超出次数	
	_< 任务名称 >_LastExecTime		TIME		上次任务执行时间	
	_< 任务名称 >_MaxExecTime		TIME		任务执行时间最大值	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	没有任务执行、标签数据链接服务的空闲时间		确认任务执行的空闲时间和标签和数据链接设定，为确保任务执行和标签数据链接服务的时间，将任务周期或标签数据链接设定的数据包间隔 （RPI）变更为较大的值		请预先考虑处理对策的内容，再进行系统服务监视设定。	
	系统服务监视设定中，系统服务执行间隔过短，或系统服务执行时间比例过长		通过实际动作确认对系统服务中所执行处理的影响，再将系统服务执行间隔变更为较大的值或将系统服务执行时间比例变更为较小的值。			
附属信息	无					
注意事项 / 备注	无					

*1. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	任务周期超限			事件代码	60010000Hex	
内容	主固定周期任务、固定周期任务中，在设定周期内，任务未执行完成					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	轻度故障	恢复方法	解除异常	日志类别	系统
产生的影响	用户程序	继续	动作	任务的执行未在设定周期内完成时， I/O 刷新动作如下 • CJ 单元：不执行 I/O 刷新。任务执行完成后，在下一个周期执行 I/O 刷新 • EtherCAT 从站：刷出刷新将输出与上次输出刷新相同的值。 任务的执行未在设定周期内结束时，可能无法进行装置整体的控制，敬请注意			

系统定义变量	变量名称	数据类型	名称
	_< 任务名称 >_Exceeded	BOOL	任务周期超出标志
	_< 任务名称 >_ExceedCount	UDINT	任务周期超出次数
	_< 任务名称 >_LastExecTime	TIME	上次任务执行时间
	_< 任务名称 >_MaxExecTime	TIME	任务执行时间最大值

发生原因和处理措施	发生原因 （推测原因）	处理对策	防止再发
	任务周期的设定过短	请确认任务执行时间，将任务周期变更为与之相符的值	请预先考虑处理对策的内容，再设计任务。
	用户程序的规模较大	请分割任务内的处理，如将不需要以短周期执行的处理移动到优先度较低的周期任务中	
	处理的重复次数为大于预期的值	如果程序中发现重复次数异常大的位置，请修正程序，使重复次数变正确。要确认重复次数，应在用户程序中设置捕捉器，监视执行次数	
	任务周期、事件任务的任务优先度错误	请将周期任务设定为较高的任务优先度。或者将事件任务的任务优先度设定为低于周期任务	
	事件任务的执行频率较高	请减小事件任务的执行频率。或者将事件任务设定为较低的任务优先度	
注意事项 / 备注	通过任务设定，可将异常的重要程度变更为 “ 监视信息 ”		

事件名称	任务周期超限			事件代码	60050000Hex	
内容	主固定周期任务、固定周期任务中，在设定周期内，任务未执行完成					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	任务的执行未在设定周期内完成时， I/O 刷新动作如下 • CJ 单元：不执行 I/O 刷新。任务执行完成后，在下一个周期执行 I/O 刷新 • EtherCAT 从站：刷出刷新将输出与上次输出刷新相同的值。 任务的执行未在设定周期内结束时，可能无法进行装置整体的控制，敬请注意			

系统定义变量	变量名称	数据类型	名称
	_< 任务名称 >_Exceeded	BOOL	任务周期超出标志
	_< 任务名称 >_ExceedCount	UDINT	任务周期超出次数
	_< 任务名称 >_LastExecTime	TIME	上次任务执行时间
	_< 任务名称 >_MaxExecTime	TIME	任务执行时间最大值

发生原因和处理措施	发生原因 （推测原因）	处理对策	防止再发
	任务周期的设定过短	请确认任务执行时间，将任务周期变更为与之相符的值	请预先考虑处理对策的内容，再设计任务。
	用户程序的规模较大	请分割任务内的处理，如将不需要以短周期执行的处理移动到优先度较低的周期任务中	
	处理的重复次数为大于预期的值	如果程序中发现重复次数异常大的位置，请修正程序，使重复次数变正确。要确认重复次数，应在用户程序中设置捕捉器，监视执行次数	
	任务周期、事件任务的任务优先度错误	请将周期任务设定为较高的任务优先度。或者将事件任务的任务优先度设定为低于周期任务	
	事件任务的执行频率较高	请减小事件任务的执行频率。或者将事件任务设定为较低的任务优先度	

附属信息	附属信息 1：发生异常的任务名称
注意事项 / 备注	通过任务设定，将异常的重要程度变更为 “ 监视信息 ” 时的异常

控制器操作相关

事件名称	用户程序 / 控制器配置和设定传送错误			**事件代码**	10200000Hex	
内容	无法正确传送用户程序或控制器配置和设定					
发生位置	PLC 功能模块		**发生位置详情**	无 / I/O 总线主机	**检测时序**	电源接通时 / 控制器 复位时
异常属性	**重要程度**	全部停止故障	**恢复方法**	电源重新接通 / 控制器 复位	**日志类别**	系统
产生的影响	**用户程序**	停止	**动作**	停止*1		
系统定义变量	**变量名称**		**数据类型**		**名称**	
	无		—		—	
发生原因和处理措施	**发生原因 （推测原因）**		**处理对策**		**防止再发**	
	下载用户程序或控制器配置和设定的过程中发生控制器断电，导致程序或控制器配置和设定不正确		请通过 Sysmac Studio 将存储器完全清除后，再下载项目。 登录有附属信息时，请在重新接通控制器的电源后，采取上述对策。 无法通过 Sysmac Studio 完全清除存储器时，请从 SD 存储卡恢复，再将项目传送到控制器中		下载用户程序或控制器配置和设定的过程，请勿断开控制器的电源	
	在线编辑的操作过程中发生控制器断电，导致程序或控制器配置和设定不正确				请勿在在线编辑操作过程中断开控制器的电源	
	存储器全部清除的过程中发生控制器断电，导致程序或控制器配置和设定不正确				请勿在存储器全部清除过程中断开控制器的电源	
	恢复过程中发生控制器断电，导致程序或控制器配置和设定不正确				请勿在恢复过程中断开控制器的电源	
	非易失性存储器故障		若采取上述措施后仍频繁发生本异常，请更换 CPU 单元		无	
附属信息	附属信息 1：发生原因详情 无 ：下载过程中断电、在线编辑过程中断电或恢复过程中断电 Downloading/preDownloading ：非上述原因时的异常发生时间（传送中 / 传送准备中）					
注意事项 / 备注	无					

*1. 详情请参考「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	用户程序执行用 ID 不正确			事件代码	10210000Hex	
内容	用户程序和 CPU 单元中设定的用户程序执行用 ID 不一致					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	下载用户程序时 / 电源接通时 / 控制器 复位时
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器 复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*1		
系统 定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和 处理措施	发生原因 （推测原因）		处理对策		防止再发	
	用户程序及 CPU 单元中设定的用户程序执行用 ID 不一致		请修改设定，使用户程序和 CPU 单元的用户程序执行用 ID 一致		请将用户程序和 CPU 单元的用户程序执行用 ID 设定为相同的值 用户程序及 CPU 单元中设定的用户程序执行用 ID 不会显示，因此请作好记录	
	CPU 单元中设定了用户程序执行用 ID，但用户程序中未设定		用户程序中不设定用户程序执行用 ID 时，请对 CPU 单元执行存储器全部清除，清除 CPU 单元中设定的用户程序执行用 ID			
附属信息	无					
注意事项 / 备注	无					

*1. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	不正确的用户程序			事件代码	10240000Hex	
内容	用户程序不正确					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	下载时 / 电源接通时 / 控制器 复位时
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器 复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*1		
系统 定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和 处理措施	发生原因 （推测原因）		处理对策		防止再发	
	功能块 / 函数的嵌套层级超出限制		找出功能块 / 函数的嵌套层级超出限制的位置，并修改用户程序，使其在限制范围内。然后，重新下载用户程序		请创建用户程序，使功能块 / 函数的嵌套层级在限制范围内。 可在 Sysmac Studio 中执行程序检查，确认嵌套层级在限制范围内	
附属信息	无					
注意事项 / 备注	无					

*1. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	用户程序 / 控制器配置和设定不正确			事件代码	10250000Hex	
内容	使用存储器超出上限，用户程序或控制器配置和设定数据损坏					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	下载时 / 电源接通时 / 控制器 复位时
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器 复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*1		
系统 定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和 处理措施	发生原因 （推测原因）		处理对策		防止再发	
	数据量超出上限值		与本异常同时发生个数限制相关的事件时，请按照该事件的处理对策，修改程序和设定，使个数在使用个数的范围内，然后再下载		无	
	主存储器容量超限		发生本异常时，若未同时发生个数限制相关的事件，请清除全部存储器后重新接通电源，确认本异常是否解除。若解除，则执行项目程序的共通化，减小容量后再下载			
	非易失性存储器老化、故障		若采取第二项措施后仍发生本异常，请更换 CPU 单元			
附属信息	无					
注意事项 / 备注	无					

*1. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	开始自动传送错误			事件代码	10270000Hex *1	
内容	自动传送的执行前检查时检测到异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*2		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	未安装 SD 存储卡		请安装 SD 存储卡		请安装 SD 存储卡	
	SD 存储卡种类不正确		请更换为 SD 或 SDHC 的 SD 存储卡		请使用 SD 或 SDHC 的 SD 存储卡	
	SD 存储卡的格式不正确		请用 Sysmac Studio 对 SD 存储卡进行格式化，然后创建 “autoload” 文件夹，保存备份文件		请用格式化后的 SD 存储卡，创建 “autoload” 文件夹，然后将备份文件保存到 “autoload” 文件夹中。 此外， SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源	
	SD 存储卡中不存在 “autoload” 文件夹		请在 SD 存储卡中创建 “autoload” 文件夹，保存备份文件			
	SD 存储卡的 “autoload” 文件夹中不存在备份文件		请将备份文件保存到 “autoload” 文件夹中			
	SD 存储卡的 “autoload” 文件夹中的备份文件损坏或必须要传送的数据在 SD 存储卡的备份文件中不存在		请将备份文件重新保存到 “autoload” 文件夹中。采取本措施后若仍然发生本异常，请重新执行备份，并保存到 “autoload” 文件夹中			
	传送对象的 CPU 单元的单元版本比 SD 存储卡的备份文件的单元版本旧		请更换为与执行备份的 CPU 单元相同版本的 CPU 单元或新 CPU 单元。此外，请将与所用 CPU 单元版本匹配的备份文件保存到 “autoload” 文件夹中		请统一 CPU 单元的单元版本和备份文件的单元版本	
	传送对象的 CPU 单元型号与 SD 存储卡的备份文件的 CPU 单元型号不一致		请更换为与执行备份的 CPU 单元相同型号的 CPU 单元。此外，请将与所用 CPU 单元型号一致的备份文件保存到 “autoload” 文件夹中		请统一 CPU 单元型号与备份文件的 CPU 单元型号	
	执行了 SD 存储卡的修复		“autoload” 文件夹中不存在备份文件、自动传送指令文件时，请重新保存文件		无	
	CPU 单元的写入保护功能有效		使用自动传送功能时，请将 CPU 单元的 “ 接通电源时的写保护 ” 设定为 “ 无 ”		使用自动传送功能时，请将 CPU 单元的 “ 接通电源时的写保护 ” 设定为 “ 无 ”	

发生原因和处理措施	发生原因 （推测原因）	处理对策	防止再发
	自动传送指令文件（AutoloadCommand.ini）的记述内容不正确	请在自动传送指令文件的记述内容中，将必须传送文件的传送设定为 “yes”	请在自动传送指令文件的记述内容中，将必须传送文件的传送设定为 “yes”
	SD 存储器故障或格式不正确，读取自动传送对象数据失败	请采取与 “SD 存储卡的格式不正确 ”、 “SD 存储卡故障 ” 时相同的处理措施	请采取与 “SD 存储卡的格式不正确 ”、 “SD 存储卡故障 ” 时相同的防再发措施
	SD 存储卡发生故障	若非上述原因，请更换 SD 存储卡	SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡
	传送对象的 CPU 单元 DB 连接服务的版本比 SD 存储卡的备份文件 DB 连接服务的版本旧	请更换为与执行备份的 CPU 单元的 DB 连接服务相同版本的 CPU 单元或新 CPU 单元。此外，请将与所用 CPU 单元的 DB 连接服务版本匹配的备份文件保存到 “autoload” 文件夹中	请统一 CPU 单元的 DB 连接服务版本和备份文件的 DB 连接服务版本
	传送对象的 CPU 单元的 Robot 版本比 SD 存储卡的备份文件的 Robot 版本旧	请更换为与执行备份的 CPU 单元的 Rotbot 版本相同的 CPU 单元或新 CPU 单元。此外，请将与所用 CPU 单元的 Robot 版本匹配的备份文件保存到 “autoload” 文件夹中	请统一 CPU 单元的 Robot 版本和备份文件的 Robot 版本
附属信息	附属信息 1：异常的详情 0001Hex：未安装 SD 存储卡 0002Hex：SD 存储卡故障、格式不正确、种类不正确 0004Hex：执行 SD 存储卡的修复 0101Hex：“autoload” 文件夹不存在 0102Hex：“autoload” 文件夹中不存在备份文件 0103Hex：备份文件损坏 0104Hex：自动传送指令文件的记述内容不正确 0105Hex：备份文件中不存在必须要传送的数据 0201Hex：CPU 单元的单元版本过低 0202Hex：CPU 单元的型号不一致 0203Hex：CPU 单元的写入保护功能为有效 0211Hex：CPU 单元的 DB 连接服务的版本或 Robot 版本过低 0301Hex：读取自动传送对象数据失败、 SD 存储卡故障		
注意事项 / 备注	无		

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。
*2. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	自动传送执行错误			事件代码	10280000Hex *1	
内容	自动传送异常结束					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*2		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	读取自动传送对象数据失败		请用 Sysmac Studio 对 SD 存储卡进行格式化，然后创建 “autoload” 文件夹，将备份文件保存到 “autoload” 文件夹中		SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡	
	自动传送过程中拔出了 SD 存储卡		请将备份文件保存到 “autoload” 文件夹中，然后安装 SD 存储卡，重新接通电源，执行自动传送		自动传送过程中请勿拔出 SD 存储卡	
	SD 存储卡的 “autoload” 文件夹中不存在备份文件		请在 SD 存储卡中创建 “autoload” 文件夹，保存备份文件		请用格式化后的 SD 存储卡，创建 “autoload” 文件夹，然后将备份文件保存到 “autoload” 文件夹中。此外， SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源	
	SD 存储卡的 “autoload” 文件夹中的备份文件损坏		请将备份文件重新保存到 “autoload” 文件夹中。采取本措施后若仍然发生本异常，请重新执行备份，并保存到 “autoload” 文件夹中			
	SD 存储卡发生故障		若非上述原因，请更换 SD 存储卡		SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡	
附属信息	附属信息 1：异常的详情 0001Hex：SD 存储卡拔出 0102Hex：“autoload” 文件夹中不存在备份文件 0103Hex：备份文件损坏 0301Hex：读取自动传送对象数据失败、 SD 存储卡故障					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。
*2. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	SD 存储卡程序传送执行前错误			事件代码	10330000Hex *1	
内容	SD 存储卡程序传送的执行前检查时检测到异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	控制器重置后，SD 存储卡程序传送执行前
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*2		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	未安装 SD 存储卡		请安装 SD 存储卡		请安装 SD 存储卡	
	SD 存储卡种类不正确		请更换为 SD 或 SDHC 的 SD 存储卡		请使用 SD 或 SDHC 的 SD 存储卡	
	SD 存储卡的格式不正确		请用 SysmacStudio 对 SD 存储卡进行格式化，然后创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，保存备份文件		请用格式化后的 SD 存储卡，创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，然后将备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中。 此外， SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源	
	SD 存储卡中不存在系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹		请在 SD 存储卡中创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，保存备份文件			
	SD 存储卡中，系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中不存在备份文件					
	SD 存储卡的系统定义变量 _Card1PrgTransferCmd.DirName 中指定文件夹的备份文件损坏或必须要传送的数据在 SD 存储卡的备份文件中不存在		请将备份文件重新保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中 采取本措施后若仍然发生本异常，请重新执行备份，并保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中			
	传送对象的 CPU 单元的单元版本比 SD 存储卡的备份文件的单元版本旧		请更换为与执行备份的 CPU 单元相同版本的 CPU 单元或新 CPU 单元。 此外，请将与所用 CPU 单元版本匹配的备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中		请统一 CPU 单元的单元版本和备份文件的单元版本	

发生原因和处理措施	发生原因 （推测原因）	处理对策	防止再发
	传送对象的 CPU 单元型号与 SD 存储卡的备份文件的 CPU 单元型号不一致	请更换为与执行备份的 CPU 单元相同型号的 CPU 单元。 此外，请将与所用 CPU 单元型号一致的备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中	请统一 CPU 单元型号与备份文件的 CPU 单元型号
	CPU 单元的写入保护功能有效	使用 SD 存储卡程序传送功能时，请将 CPU 单元的 “ 接通电源时的写保护 ” 设定为 “ 无 ”	使用 SD 存储卡程序传送功能时，请将 CPU 单元的 “ 接通电源时的写保护 ” 设定为 “ 无 ”
	系统定义变量 _Card1PrgTransferCmd 中，必须传送的文件未在传送对象内	请在系统定义变量 _Card1PrgTransferCmd 中，将必须传送文件的传送设定为 “TRUE”	请在系统定义变量 _Card1PrgTransferCmd 中，将必须传送文件的传送设定为 “TRUE”
	SD 存储器故障或格式不正确，读取自动传送对象数据失败	请采取与 “SD 存储卡的格式不正确 ”、 “SD 存储卡故障 ” 时相同的处理措施	请采取与 “SD 存储卡的格式不正确 ”、 “SD 存储卡故障 ” 时相同的防再发措施
	SD 存储卡发生故障	若非上述原因，请更换 SD 存储卡	SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡
	传送对象的 CPU 单元 DB 连接服务的版本比 SD 存储卡的备份文件 DB 连接服务的版本旧	请更换为与执行备份的 CPU 单元的 DB 连接服务相同版本的 CPU 单元或新 CPU 单元。此外，请将与所用 CPU 单元的 DB 连接服务版本匹配的备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中	请统一 CPU 单元的 DB 连接服务版本和备份文件的 DB 连接服务版本
	传送对象的 CPU 单元的 Robot 版本比 SD 存储卡的备份文件的 Robot 版本旧	请更换为与执行备份的 CPU 单元的 Rotbot 版本相同的 CPU 单元或新 CPU 单元。此外，请将与所用 CPU 单元的 Robot 版本匹配的备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中	请统一 CPU 单元的 Robot 版本和备份文件的 Robot 版本

附属信息	附属信息 1：异常的详情 0001Hex：未安装 SD 存储卡 0002Hex：SD 存储卡故障、格式不正确、种类不正确 0004Hex：执行 SD 存储卡的修复 0101Hex：系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹 0102Hex：系统定义变量 _Card1PrgTransferCmd.DirName 中指定的 SD 存储卡的文件夹中，不存在备份文件 0103Hex：备份文件损坏 0104Hex：系统定义变量 _Card1PrgTransferCmd 中，必须传送的文件不在传送对象内 0105Hex：备份文件中不存在必须要传送的数据 0201Hex：CPU 单元的单元版本过低 0202Hex：CPU 单元的型号不一致 0203Hex：CPU 单元的写入保护功能为有效 0211Hex：CPU 单元的 DB 连接服务的版本或 Robot 版本过低 0212Hex：“SD 存储卡程序传送的有效 / 无效设定 ” 为无效 0213Hex：SysmacStudio 中设定的 “SD 存储卡程序传送执行时密码 ” 和系统定义变量 _Card1PrgTransferCmd.Password 不一致 0214Hex：CPU 单元的拨动开关设定不是可开始 SD 存储卡程序传送的设定不是可开始的设定 0301Hex：读取程序传送对象数据失败、 SD 存储卡故障
注意事项 / 备注	无

*1. CPU 单元版本为 1.11 以上的环境下发生的事件。
*2. 详情请参考「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	SD 存储卡程序传送执行错误			事件代码	10340000Hex *1	
内容	SD 存储卡程序传送异常结束					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	SD 存储卡程序传送执行中
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*2		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	读取 SD 存储卡程序传送对象数据失败		请用 Sysmac Studio 对 SD 存储卡进行格式化，创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，然后将备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中		SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡	
	传送 SD 存储卡程序的过程中拔出了 SD 存储卡		请将备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中，然后安装 SD 存储卡，重新执行 SD 存储卡程序传送		传送 SD 存储卡程序的过程中请勿拔出 SD 存储卡	
	SD 存储卡中，系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中不存在备份文件		请在 SD 存储卡中创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，保存备份文件		请用格式化后的 SD 存储卡，创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，然后将备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中。 此外， SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源	
	SD 存储卡中，系统定义变量 _Card1PrgTransferCmd.DirName 中指定文件夹中的备份文件损坏		请将备份文件重新保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中采取本措施后若仍然发生本异常，请重新执行备份，并保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中			
	SD 存储卡发生故障		若非上述原因，请更换 SD 存储卡		SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡	
附属信息	附属信息 1：异常的详情 0001Hex：SD 存储卡拔出 0102Hex：系统定义变量 _Card1PrgTransferCmd.DirName 中指定的 SD 存储卡的文件夹中，不存在备份文件 0103Hex：备份文件损坏 0301Hex：读取 SD 存储卡程序传送对象数据失败、 SD 存储卡故障					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.11 以上的环境下发生的事件。
*2. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	PLC 功能处理异常			事件代码	40110000Hex	
内容	PLC 功能模块中检测到致命异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通	日志类别	系统
产生的影响	用户程序	停止	动作	停止*1		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	软件发生错误		请与本公司营业或经销商联系		无	
附属信息	附属信息 1：系统信息 附属信息 2：系统信息 附属信息 3：系统信息 附属信息 4：系统信息					
注意事项 / 备注	无					

*1. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	安全模式			事件代码	40160000Hex *1	
内容	以安全模式启动了控制器					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时 / 控制器复位时
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器复位	日志类别	系统
产生的影响	用户程序	停止	动作	停止*2		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	将 CPU 单元的拨动开关设定为安全模式并打开了控制器的电源		—		—	
附属信息	无					
注意事项 / 备注	以安全模式启动了控制器时，即使控制器的电源接通时的动作模式为 “ 运行模式 ”，也不会执行用户程序					

*1. CPU 单元版本为 1.02 以下的环境下发生的事件。
*2. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	PLC 功能处理异常			事件代码	44420000Hex *1	
内容	PLC 功能模块中检测到致命异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	全部停止故障	恢复方法	电源重新接通 / 控制器重置	日志类别	系统
产生的影响	用户程序	停止	动作	停止*2		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	软件发生错误		请与本公司营业或经销商联系		无	
附属信息	附属信息 1：系统信息 附属信息 2：系统信息 附属信息 3：系统信息 附属信息 4：系统信息					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.05 以上的环境下发生的事件。
*2. 详情请参考📖「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	PLC 功能处理异常			事件代码	40120000Hex	
内容	PLC 功能模块中检测到致命异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	部分停止故障	恢复方法	电源重新接通	日志类别	系统
产生的影响	用户程序	停止	动作	停止*1		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	软件发生错误		请与本公司营业或经销商联系		无	
附属信息	附属信息 1：系统信息 附属信息 2：系统信息 附属信息 3：系统信息 附属信息 4：系统信息					
注意事项 / 备注	无					

*1. 与全部停止故障等级的异常相同的动作详情📖请参考「发生全部停止故障等级的控制器异常的输入输出动作」（P.12-9）。

事件名称	PLC 功能处理异常			事件代码	40130000Hex	
内容	PLC 功能模块的一部分检测到致命异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	轻度故障	恢复方法	电源重新接通 / 控制器重置	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	软件发生错误		请与本公司营业或经销商联系		无	
附属信息	附属信息 1：系统信息 附属信息 2：系统信息 附属信息 3：系统信息 附属信息 4：系统信息					
注意事项 / 备注	无					

事件名称	事件日志保存失败			事件代码	10230000Hex	
内容	保存事件日志失败					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时 / 控制器复位时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	开始	动作	无影响。但是，部分或全部过去事件日志将无法读取		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	因电池电压低导致断电保持失败		请更换电池		定期更换电池	
附属信息	附属信息 1： 异常的详情 0：所有类别的日志保存失败、 1：系统事件日志的保存失败、 2：访问事件日志保存失败、 100：用户事件日志的保存失败					
注意事项 / 备注	无					

事件名称	追踪设定传送失败			事件代码	10260000Hex	
内容	传送追踪设定的过程中发生断电					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时 / 控制器复位时
异常属性	重要程度	监视信息	恢复方法	电源重新接通 / 控制器重置	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	传送追踪设定的过程中断电		请重新传送追踪设定		传送追踪设定的过程中请勿断电	
附属信息	无					
注意事项 / 备注	发生本异常时，将初始化所有的追踪设定					

事件名称	备份开始失败			事件代码	10290000Hex *1	
内容	备份的执行前检查时检测到异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出备份指示时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	未安装 SD 存储卡		请安装 SD 存储卡		请安装 SD 存储卡	
	SD 存储卡种类不正确		请更换为 SD 或 SDHC 的 SD 存储卡		请使用 SD 或 SDHC 的 SD 存储卡	
	SD 存储卡的格式不正确		请用 Sysmac Studio 对 SD 存储卡进行格式化		请使用格式化后的 SD 存储卡。 此外， SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源	
	SD 存储卡有写保护		请解除 SD 存储卡的写保护		请在 SD 存储卡的写保护处于解除的状态下使用	
	“ 禁止向 SD 存储卡备份 ” 设定为 “ 禁止 ” 时		请解除 “ 禁止向 SD 存储卡备份 ” 设定		请勿使用 “ 禁止向 SD 存储卡备份 ” 设定	
	其他备份功能正在运行		请待其他备份功能执行完成后，再重新执行备份		正在执行其他备份功能时，请勿执行备份	
	正在执行同步、在线编辑或存储器全部清除		请待同步、在线编辑、存储器全部清除完成后，再重新执行备份		正在执行同步、在线编辑、存储器全部清除时，请勿执行备份	
	通过用户操作中止了备份		无		无	
	与 Sysmac Studio 的在线连接断开		请确认电缆的连接。请先离线，再重新在线连接，然后执行备份		请确认电缆没有断线。请确认电缆已切实连接	
	SD 存储卡发生故障		若非上述原因，请更换 SD 存储卡		SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡	

附属信息	附属信息 1：操作种类 0101Hex：控制器→ SD 存储卡 （通过 CPU 单元正面开关的操作进行指示） 0102Hex：控制器→ SD 存储卡 （通过系统变量进行指示） 0103Hex：控制器→ SD 存储卡 （通过 Sysmac Studio 操作进行指示） 0104Hex：控制器→ SD 存储卡 （通过指令进行指示） 0201Hex：控制器→计算机 附属信息 2：异常的详情 0001Hex：未安装 SD 存储卡 0002Hex：SD 存储卡故障、格式不正确、种类不正确 0003Hex：SD 存储卡有写保护 0204Hex：“ 禁止向 SD 存储卡备份 ” 设定为 “ 禁止 ” 0205Hex：正在执行其他备份功能 0206Hex：正在执行同步、在线编辑或存储器全部清除 0207Hex：系统定义变量中指定的目录名称中使用了禁用字符 0401Hex：通过用户操作中止了备份 0501hex：与 Sysmac Studio 的在线连接断开
注意事项 / 备注	无

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	备份执行失败			事件代码	102A0000Hex *1	
内容	备份异常结束					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	备份执行中
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	SD 存储卡容量不足		请更换为有足够剩余空间的 SD 存储卡		请使用有足够剩余空间的 SD 存储卡	
	保存备份对象数据失败		请在未向 CPU 单元写入的状态下重新备份		备份过程中，请勿进行 CPU 单元的写入操作	
	备份过程中拔出了 SD 存储卡		请安装 SD 存储卡		请安装 SD 存储卡	
	单元 / 从站备份失败		请参考 “ 备份执行失败 （CJ 单元） ”(102D0000Hex) 或 “ 备份执行失败 （EtherCAT 从站） ”(102F0000Hex) 的处理对策		请参考 “ 备份执行失败 （CJ 单元） ”(102D0000Hex) 或 “ 备份执行失败 （EtherCAT 从站） ”(102F0000Hex) 的防止再发	
	通过用户操作中止了备份		无		无	
	正在执行凸轮表保存指令或 CPU 单元名称变更		请在凸轮表保存指令或 CPU 单元名称变更完成后操作		正在执行凸轮表保存指令或 CPU 单元名称更新时，请勿备份	
	与 Sysmac Studio 的在线连接断开		请确认电缆的连接。请先离线，再重新在线连接，然后执行备份		请确认电缆没有断线。请确认电缆已切实连接	
	向计算机保存备份对象数据失败		请增加计算机硬盘的可用空间		请确认硬盘有足够的可用空间后再备份	
	SD 存储卡发生故障		若非上述原因，请更换 SD 存储卡		SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。请根据写入次数限制，定期更换 SD 存储卡	

附属信息	附属信息 1：操作种类 0101Hex：控制器→ SD 存储卡 （通过 CPU 单元正面开关的操作进行指示） 0102Hex：控制器→ SD 存储卡 （通过系统变量进行指示） 0103Hex：控制器→ SD 存储卡 （通过 Sysmac Studio 操作进行指示） 0104Hex：控制器→ SD 存储卡 （通过指令进行指示） 0201Hex：控制器→计算机 附属信息 2：异常的详情 0001Hex：SD 存储卡拔出 0005Hex：SD 存储卡容量不足 0006Hex：超出文件或目录的最大数 0206Hex：正在执行凸轮表保存指令或 CPU 单元名称更新 0210Hex：存在与指定目录同名的文件 0302Hex：保存备份对象数据失败、 SD 存储卡故障 0304Hex：单元 / 从站的备份失败 0401Hex：通过用户操作中止了备份 0501Hex：与 Sysmac Studio 的在线连接断开 0502Hex：向计算机保存备份对象数据失败
注意事项 / 备注	无

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	恢复开始失败			事件代码	102B0000Hex *1	
内容	恢复的执行前检查时检测到异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出恢复指示时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	—	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	未安装 SD 存储卡		请安装 SD 存储卡		请安装 SD 存储卡	
	SD 存储卡种类不正确		请更换为 SD 或 SDHC 的 SD 存储卡		请使用 SD 或 SDHC 的 SD 存储卡	
	SD 存储卡的格式不正确		请用 Sysmac Studio 对 SD 存储卡进行格式化，并保存备份文件		请用格式化后的 SD 存储卡，保存备份文件 此外， SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源	
	SD 存储卡中不存在备份文件		请将备份文件保存到 SD 存储卡的指定文件夹中			
	SD 存储卡的备份文件损坏或必须要恢复的数据在 SD 存储卡的备份文件中不存在		请重新进行备份			
	恢复对象的 CPU 单元的单元版本比 SD 存储卡的备份文件的单元版本旧		请更换为与执行备份的 CPU 单元相同版本的 CPU 单元或新 CPU 单元。此外，请指定与所用 CPU 单元版本匹配的备份文件		请统一 CPU 单元的单元版本和备份文件的单元版本	
	恢复对象的 CPU 单元型号与 SD 存储卡的备份文件的 CPU 单元型号不一致		请更换为与执行备份的 CPU 单元相同型号的 CPU 单元。此外，请指定与所用 CPU 单元型号一致的备份文件		请统一 CPU 单元型号与备份文件的 CPU 单元型号	
	执行了 SD 存储卡的修复		SD 存储卡的指定文件夹中不存在备份文件、恢复指令文件时，请重新保存文件		无	
	CPU 单元的写入保护功能有效		使用恢复功能时，请将 CPU 单元的 “ 接通电源时的写保护 ” 设定为 “ 无 ”		使用恢复功能时，请将 CPU 单元的 “ 接通电源时的写保护 ” 设定为 “ 无 ”	
	恢复指令文件（RestoreCommand.ini）的记述内容不正确		请在恢复指令文件的记述内容中，将必须恢复文件的传送设定为 “yes”		请在恢复指令文件的记述内容中，将必须恢复文件的传送设定为 “yes”	
	其他备份功能正在运行		请待其他备份功能执行完成后，再重新执行恢复		正在执行其他备份功能时，请勿执行恢复	
	正在执行同步、在线编辑或存储器全部清除		请待同步、在线编辑、存储器全部清除完成后，再重新执行恢复		正在执行同步、在线编辑、存储器全部清除时，请勿执行恢复	

发生原因和处理措施	发生原因 （推测原因）	处理对策	防止再发
	与 Sysmac Studio 的在线连接断开	请确认电缆的连接。请先离线，再重新在线连接，然后执行备份	请确认电缆没有断线。请确认电缆已切实连接
	SD 存储器故障或格式不正确，读取恢复对象数据失败	请采取与 “SD 存储卡的格式不正确 ”、 “SD 存储卡故障 ” 时相同的处理措施	请采取与 “SD 存储卡的格式不正确 ”、 “SD 存储卡故障 ” 时相同的防再发措施
	SD 存储卡发生故障	若非上述原因，请更换 SD 存储卡	SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡
	恢复对象的 CPU 单元 DB 连接服务的版本比 SD 存储卡的备份文件 DB 连接服务的版本旧	请更换为与执行备份的 CPU 单元的 DB 连接服务相同版本的 CPU 单元或新 CPU 单元。此外，请指定与所用 CPU 单元的 DB 连接服务版本匹配的备份文件	请统一 CPU 单元的 DB 连接服务版本和备份文件的 DB 连接服务版本
	恢复对象的 CPU 单元的 Robot 版本比 SD 存储卡的备份文件的 Robot 版本旧	请更换为与执行备份的 CPU 单元的 Rotbot 版本相同的 CPU 单元或新 CPU 单元。此外，请指定与所用 CPU 单元的 Robot 版本匹配的备份文件	请统一 CPU 单元的 Robot 版本和备份文件的 Robot 版本
附属信息	附属信息 1：操作种类 0101Hex：SD 存储卡➞控制器 （通过 CPU 单元正面开关的操作进行指示） 0201Hex：计算机➞控制器 附属信息 2：异常的详情 0001Hex：未安装 SD 存储卡 0002Hex：SD 存储卡故障、格式不正确、种类不正确 0004Hex：执行 SD 存储卡的修复 0102Hex：不存在备份文件 0103Hex：备份文件损坏 0104Hex：恢复指令文件的记述内容不正确 0105Hex：备份文件中不存在必须要传送的数据 0201Hex：CPU 单元的单元版本过低 0202Hex：CPU 单元的型号不一致 0203Hex：CPU 单元的写入保护功能为有效 0205Hex：正在执行其他备份功能 0206Hex：正在执行同步、在线编辑或存储器全部清除 0211Hex：CPU 单元的 DB 连接服务的版本或 Robot 版本过低 0301Hex：读取恢复对象数据失败、 SD 存储卡故障 0501Hex：与 Sysmac Studio 的在线连接断开		
注意事项 / 备注	无		

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	恢复执行失败			事件代码	102C0000Hex *1	
内容	恢复异常结束					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	恢复执行中
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	—	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	读取恢复对象数据失败		请用 Sysmac Studio 对 SD 存储卡进行格式化，并保存备份文件		SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡	
	恢复过程中拔出了 SD 存储卡		请安装保存有备份文件的 SD 卡，重新执行恢复		恢复过程中请勿拔出 SD 存储卡	
	单元 / 从站恢复失败		请参考 “ 恢复执行失败 （CJ 单元） ”(102E0000Hex) 或 “ 恢复执行失败 （EtherCAT 从站） ”(10300000Hex) 的处理对策		请参考 “ 恢复执行失败 （CJ 单元） ”(102E0000Hex) 或 “ 恢复执行失败 （EtherCAT 从站） ”(10300000Hex) 的防止再发	
	SD 存储卡发生故障		若非上述原因，请更换 SD 存储卡		SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡	
附属信息	附属信息 1：操作种类 0101Hex：SD 存储卡➞控制器 （通过 CPU 单元正面开关的操作进行指示） 0201Hex：计算机➞控制器 附属信息 2：异常的详情 0001Hex：SD 存储卡拔出 0102Hex：不存在备份文件 0103Hex：备份文件损坏 0301Hex：读取恢复对象数据失败、 SD 存储卡故障 0303Hex：单元 / 从站恢复失败					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	SD 存储卡程序传送开始失败			事件代码	10320000Hex *1	
内容	SD 存储卡程序传送的开始前检查时检测到异常					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	SD 存储卡程序传送开始时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	—	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	“SD 存储卡程序传送的有效 / 无效设定 ” 为无效		请将 “SD 存储卡程序传送的有效 / 无效设定 ” 设定为有效		请将 “SD 存储卡程序传送的有效 / 无效设定 ” 设定为有效	
	SysmacStudio 中设定的 “ 执行 SD 存储卡程序传送时密码 ” 和系统定义变量 _Card1PrgTransferCmd.Password 不一致		请将系统定义变量 _Card1PrgTransferCmd.Password 设定为 SysmacStudio 中设定的 “ 执行 SD 存储卡程序传送时密码 ”		请将系统定义变量 _Card1PrgTransferCmd.Password 设定为 SysmacStudio 中设定的 “ 执行 SD 存储卡程序传送时密码 ”	
	CPU 单元的拨动开关设定不是可开始 SD 存储卡程序传送的设定		请将 CPU 单元的拨动开关全部关闭后，开始 SD 存储卡程序传送		请将 CPU 单元的拨动开关全部关闭后，开始 SD 存储卡程序传送	
	未安装 SD 存储卡		请安装 SD 存储卡		请安装 SD 存储卡	
	SD 存储卡种类不正确		请更换为 SD 或 SDHC 的 SD 存储卡		请使用 SD 或 SDHC 的 SD 存储卡	
	SD 存储卡的格式不正确		请用 Sysmac Studio 对 SD 存储卡进行格式化，然后创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，保存备份文件		请用格式化后的 SD 存储卡，创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，然后将备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中。 此外， SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源	
	SD 存储卡中不存在系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹		请在 SD 存储卡中创建系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹，保存备份文件			
	SD 存储卡中，系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中不存在备份文件					
	SD 存储卡的系统定义变量 _Card1PrgTransferCmd.DirName 中指定文件夹的备份文件损坏或必须要传送的数据在 SD 存储卡的备份文件中不存在		请将备份文件重新保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中 采取本措施后若仍然发生本异常，请重新执行备份，并保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中			

发生原因和处理措施	发生原因 （推测原因）	处理对策	防止再发
	传送对象的 CPU 单元的单元版本比 SD 存储卡的备份文件的单元版本旧	请更换为与执行备份的 CPU 单元相同版本的 CPU 单元或新 CPU 单元。 此外，请将与所用 CPU 单元版本匹配的备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中	请统一 CPU 单元的单元版本和备份文件的单元版本
	传送对象的 CPU 单元型号与 SD 存储卡的备份文件的 CPU 单元型号不一致	请更换为与执行备份的 CPU 单元相同型号的 CPU 单元。 此外，请将与所用 CPU 单元型号一致的备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中	请统一 CPU 单元型号与备份文件的 CPU 单元型号
	CPU 单元的写入保护功能有效	使用 SD 存储卡程序传送功能时，请将 CPU 单元的 “ 接通电源时的写保护 ” 设定为 “ 无 ”	使用 SD 存储卡程序传送功能时，请将 CPU 单元的 “ 接通电源时的写保护 ” 设定为 “ 无 ”
	其他备份功能正在运行	请待其他备份功能执行完成后，再重新执行备份	正在执行其他备份功能时，请勿执行备份
	正在执行同步、在线编辑或存储器全部清除	请待同步、在线编辑、存储器全部清除完成后，再重新执行备份	正在执行同步、在线编辑、存储器全部清除时，请勿执行备份
	系统定义变量 _Card1PrgTransferCmd 中，必须传送的文件未在传送对象内	请在系统定义变量 _Card1PrgTransferCmd 中，将必须传送文件的传送设定为 “TRUE”	请在系统定义变量 _Card1PrgTransferCmd 中，将必须传送文件的传送设定为 “TRUE”
	SD 存储器故障或格式不正确，读取 SD 存储卡程序传送对象数据失败	请采取与 “SD 存储卡的格式不正确 ”、 “SD 存储卡故障 ” 时相同的处理措施	请采取与 “SD 存储卡的格式不正确 ”、 “SD 存储卡故障 ” 时相同的防再发措施
	SD 存储卡发生故障	若非上述原因，请更换 SD 存储卡	SD BUSY LED 亮灯时，请勿拔出 SD 存储卡或断开电源。此外，请根据写入次数限制，定期更换 SD 存储卡
	传送对象的 CPU 单元 DB 连接服务的版本比 SD 存储卡的备份文件 DB 连接服务的版本旧	请更换为与执行备份的 CPU 单元的 DB 连接服务相同版本的 CPU 单元或新 CPU 单元。 此外，请将与所用 CPU 单元的 DB 连接服务版本匹配的备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中	请统一 CPU 单元的 DB 连接服务版本和备份文件的 DB 连接服务版本

发生原因和处理措施	传送对象的 CPU 单元的 Robot 版本比 SD 存储卡的备份文件的 Robot 版本旧	请更换为与执行备份的 CPU 单元的 Rotbot 版本相同的 CPU 单元或新 CPU 单元。 此外，请将与所用 CPU 单元的 Robot 版本匹配的备份文件保存到系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹中	请统一 CPU 单元的 Robot 版本和备份文件的 Robot 版本
附属信息	附属信息 1：异常的详情 0001Hex：未安装 SD 存储卡 0002Hex：SD 存储卡故障、格式不正确、种类不正确 0004Hex：执行 SD 存储卡的修复 0101Hex：系统定义变量 _Card1PrgTransferCmd.DirName 中指定的文件夹 0102Hex：系统定义变量 _Card1PrgTransferCmd.DirName 中指定的 SD 存储卡的文件夹中，不存在备份文件 0103Hex：备份文件损坏 0104Hex：系统定义变量 _Card1PrgTransferCmd 中，必须传送的文件不在传送对象内 0105Hex：备份文件中不存在必须要传送的数据 0201Hex：CPU 单元的单元版本过低 0202Hex：CPU 单元的型号不一致 0203Hex：CPU 单元的写入保护功能为有效 0205Hex：正在执行其他备份功能 0206Hex：正在执行同步、在线编辑或存储器全部清除 0211Hex：CPU 单元的 DB 连接服务的版本或 Robot 版本过低 0212Hex：“SD 存储卡程序传送的有效 / 无效设定 ” 为无效 0213Hex：SysmacStudio 中设定的 “SD 存储卡程序传送执行时密码 ” 和系统定义变量 _Card1PrgTransferCmd.Password 不一致 0214Hex：CPU 单元的拨动开关设定不是可开始 SD 存储卡程序传送的设定不是可开始的设定 0301Hex：读取程序传送对象数据失败、 SD 存储卡故障		
注意事项 / 备注	无		

*1. CPU 单元版本为 1.11 以上的环境下发生的事件。

事件名称	PLC 系统信息			事件代码	40140000Hex	
内容	PLC 功能模块的内部信息					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	PLC 功能模块的内部信息。记录为其他事件的追加信息。		—		—	
附属信息	附属信息 1：系统信息 附属信息 2：系统信息 附属信息 3：系统信息 附属信息 4：系统信息					
注意事项 / 备注	无					

事件名称	安全模式			事件代码	40170000Hex*1	
内容	以安全模式启动了控制器					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时 / 控制器复位时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	停止	动作	—		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	将 CPU 单元的拨动开关设定为安全模式并打开了控制器的电源		—		—	
附属信息	无					
注意事项 / 备注	以安全模式启动了控制器时，即使控制器的电源接通时的动作模式为 “ 运行模式 ”，也将以 “ 程序模式 ” 动作					

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	NX 信息通信异常			事件代码	80230000Hex*1	
内容	信息通信中发生异常					
发生位置	PLC 功能模块 EtherCAT 主机功能模块 EtherNet/IP 功能模块		发生位置详情	无	检测时序	NX 信息收发时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	—		
系统 定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和 处理措施	发生原因 （推测原因）		处理对策		防止再发	
	通信电缆断线		请确认通信电缆是否断线，若断线，请更换电缆		请确认所用通信电缆没有断线	
	通信电缆的连接器松脱		请确认连接器是否正确接合，然后重新连接		请确认通信电缆已切实连接	
	NX 信息通信的负载过高		请减少 NX 信息发送指令的使用频率。 或者增大指令输入变量 “TimeOut” 的值。 连接有对个 Sysmac Studio 时，请减少同时操作的频率		请减少 NX 信息发送指令的使用频率。 或者增大指令输入变量 “TimeOut” 的值。 连接有对个 Sysmac Studio 时，请减少同时操作的频率	
附属信息	附属信息 1：系统信息 附属信息 2：通信种类 0：NX 总线 1：EtherCAT 65535：单元内部 （路由处理）					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.05 以上的环境下发生的事件。

事件名称	PLC 系统信息			事件代码	40150000Hex	
内容	PLC 功能模块的内部信息					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统 定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和 处理措施	发生原因 （推测原因）		处理对策		防止再发	
	PLC 功能模块的内部信息。记录为其他事件的追加信息。		—		—	
附属信息	附属信息 1：系统信息 附属信息 2：系统信息 附属信息 3：系统信息 附属信息 4：系统信息					
注意事项 / 备注	无					

事件名称	PLC 系统信息			事件代码	44430000Hex *1	
内容	PLC 功能模块的内部信息					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	始终
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	PLC 功能模块的内部信息。记录为其他事件的追加信息。		—		—	
附属信息	附属信息 1：系统信息 附属信息 2：系统信息 附属信息 3：系统信息 附属信息 4：系统信息					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.05 以上的环境下发生的事件。

事件名称	时刻的变更			事件代码	90010000Hex	
内容	变更了时刻					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	_CurrentTime		DATE_AND_TIME		系统时刻	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	变更了时刻		—		—	
附属信息	附属信息 1：变更前时刻					
注意事项 / 备注	通过 NTP 时钟修正指令 (SetTime) 变更时刻时，不会记录到事件日志中 本事件的时间戳为变更时刻后的时刻					

事件名称	时区的变更			事件代码	90020000Hex	
内容	变更了时区					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	下载时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	_CurrentTime		DATE_AND_TIME		系统时刻	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	变更了时区		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	用户程序 / 控制器配置和设定的下载			事件代码	90050000Hex *1	
内容	下载了用户程序 / 控制器配置和设定					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户程序 / 控制器配置和设定的下载时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	根据下载的用户程序 / 控制器设定开始动作		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	下载了用户程序 / 控制器配置和设定		—		—	
附属信息	附属信息 1：连接方法 1：USB 直接连接 2：Ethernet 直接连接 3：USB 远程连接或 Ethernet HUB 连接 附属信息 2：连接源的 IP 地址。通过 Proxy 时为 Proxy 的 IP 地址。 （附属信息 1 为 2 或 3 时）					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.10 以上的环境下发生的事件。

事件名称	在线编辑传送			事件代码	90070000Hex *1	
内容	在线编辑了用户程序					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	在线编辑的传送开始时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	根据变更后的用户程序动作		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	在线编辑用户程序并反映到了控制器中		—		—	
附属信息	附属信息 1：连接方法 1：USB 直接连接 2：Ethernet 直接连接 3：USB 远程连接或 Ethernet HUB 连接 附属信息 2：连接源的 IP 地址。通过 Proxy 时为 Proxy 的 IP 地址。 （附属信息 1 为 2 或 3 时）					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.10 以上的环境下发生的事件。

事件名称	强制值刷新的设置			事件代码	90080000Hex	
内容	指定了强制值刷新的设置					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	根据强制值刷新动作		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	用户指定了强制值刷新的设置		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	强制值刷新的重置			事件代码	90090000Hex	
内容	指定了强制值刷新的重置					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	根据强制值刷新动作		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	用户指定了强制值刷新的重置		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	强制值刷新的全部解除			事件代码	900A0000Hex	
内容	指定了强制值刷新的全部解除					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	解除所有强制值刷新，按照原来的用户程序动作		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	用户指定了强制值刷新的全部解除		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	存储器全部清除			事件代码	900B0000Hex	
内容	执行了存储器全部清除					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	—	动作	恢复为出厂状态		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	拥有管理员权限的用户执行了存储器全部清除		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	事件日志的清除			事件代码	900C0000Hex	
内容	清除了事件日志					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	用户清除了事件日志		—		—	
附属信息	附属信息 1：清除对象 0： 清除所有类别的日志 1： 清除系统事件日志 2： 清除访问事件日志 100：清除用户事件日志					
注意事项 / 备注	无					

事件名称	自动传送完成			事件代码	900F0000Hex *1	
内容	自动传送已完成					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	根据自动传送的用户控制器配置和设定开始动作		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	自动传送已完成		—		—	
附属信息	无					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	接通电源			事件代码	90110000Hex	
内容	电源已接通					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	—	动作	动作开始		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	电源已接通		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	断开电源			事件代码	90120000Hex	
内容	电源已断开					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	断开电源时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	停止	动作	全部停止		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	电源已断开		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	运行开始			事件代码	90130000Hex	
内容	已开始运行					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	转移到运行模式时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	开始	动作	用户程序执行开始		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	受理了运行开始		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	运行停止			事件代码	90140000Hex	
内容	已停止运行					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	转移到程序模式时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	停止	动作	用户程序停止		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	受理了运行停止		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	执行复位			事件代码	90150000Hex	
内容	执行了复位					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	—	动作	复位后、动作开始		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	受理了复位指示		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	写入用户程序执行用 ID			事件代码	90160000Hex	
内容	CPU 单元中设定或变更了用户程序执行用 ID					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	下载时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	拥有管理员权限的用户变更了 CPU 单元的用户程序执行用 ID		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	发生中控制器异常的全部解除			事件代码	90180000Hex	
内容	发生中的异常已全部解除					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	解除已排除异常原因的异常		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	用户解除了所有发生中的异常		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	强制值刷新的解除			事件代码	90190000Hex	
内容	指定了强制值刷新的解除					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	用户发出指示时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	访问
产生的影响	用户程序	继续	动作	解除强制值刷新，按照原来的用户程序动作		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	用户指定了强制值刷新的解除		—		—	
附属信息	无					
注意事项 / 备注	无					

事件名称	备份开始			事件代码	901A0000Hex *1	
内容	已开始备份					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	备份开始时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	已开始备份		—		—	
附属信息	附属信息 1：操作种类 0101Hex：控制器→ SD 存储卡 （通过 CPU 单元正面开关的操作进行指示） 0102Hex：控制器→ SD 存储卡 （通过系统变量进行指示） 0103Hex：控制器→ SD 存储卡 （通过 Sysmac Studio 操作进行指示） 0104Hex：控制器→ SD 存储卡 （通过指令进行指示） 0201Hex：控制器→计算机					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	备份结束			事件代码	901B0000Hex *1	
内容	备份已正常结束					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	备份处理正常结束时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	备份已正常结束		—		—	
附属信息	附属信息 1：操作种类 0101Hex：控制器→ SD 存储卡 （通过 CPU 单元正面开关的操作进行指示） 0102Hex：控制器→ SD 存储卡 （通过系统变量进行指示） 0103Hex：控制器→ SD 存储卡 （通过 Sysmac Studio 操作进行指示） 0104Hex：控制器→ SD 存储卡 （通过指令进行指示） 0201Hex：控制器→计算机					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	恢复开始			事件代码	901C0000Hex *1	
内容	已开始恢复					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	恢复开始时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	—	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	已开始恢复		—		—	
附属信息	附属信息 1：操作种类 0101Hex：SD 存储卡→控制器 （通过 CPU 单元正面开关的操作进行指示） 0201Hex：计算机→控制器					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	恢复完成			事件代码	901D0000Hex *1	
内容	恢复已正常结束					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	恢复处理正常结束时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	—	动作	恢复完成后，无法开始运行。关闭控制器的电源，将 CPU 单元的拨动开关全部关闭，在重新接通电源，将按照恢复的用户程序 / 设定开始动作		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	恢复已正常结束		—		—	
附属信息	附属信息 1：操作种类 0101Hex：SD 存储卡→控制器 （通过 CPU 单元正面开关的操作进行指示） 0201Hex：计算机→控制器					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.03 以上的环境下发生的事件。

事件名称	SD 存储卡程序传送开始			事件代码	90200000Hex *1	
内容	已开始 SD 存储卡程序传送					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	SD 存储卡程序传送开始时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	—	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	已开始 SD 存储卡程序传送		—		—	
附属信息	无					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.11 以上的环境下发生的事件。

事件名称	SD 存储卡程序传送完成			事件代码	90210000Hex *1	
内容	已完成 SD 存储卡程序传送					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	SD 存储卡程序传送完成时
异常属性	重要程度	一般信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	—	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	已完成 SD 存储卡程序传送		—		—	
附属信息	无					
注意事项 / 备注	无					

*1. CPU 单元版本为 1.11 以上的环境下发生的事件。

FINS 相关

事件名称	CPU 总线单元系统设定区域错误			事件代码	14010000Hex	
内容	CPU 总线单元系统设定区域的存储器检查中检测到错误					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时 / 控制器重置时 /CPU 总线单元系统设定区域的写入时
异常属性	重要程度	轻度故障	恢复方法	解除异常 / 电源重新接通	日志类别	系统
产生的影响	用户程序	继续	动作	CPU 总线单元可能会停止		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	下载 CPU 总线单元系统设置的过程中发生控制器断电 / 与 Sysmac Studio 的通信切断		请进行存储器全部清除或重新下载 CPU 总线单元系统设置。如果本异常再次发生，请更换 CPU 单元。		下载 CPU 总线单元系统设置时，请勿断开控制器的电源 / 切断与 Sysmac Studio 的通信	
附属信息	无					
注意事项 / 备注	无					

事件名称	IP 地址表设定错误			事件代码	34100000Hex	
内容	IP 地址表设定错误					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时 / 控制器复位时 / 重启内置 EtherNet/IP 端口时
异常属性	重要程度	轻度故障	恢复方法	解除异常	日志类别	系统
产生的影响	用户程序	继续	动作	不可使用 FINS/UDP 通信功能		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	IP 地址转换方式设定为 “ 并用方式 ”、“IP 地址表方式 ”，但 IP 地址表的设定错误		请修改 IP 地址表的设定		请正确设定 IP 地址表	
附属信息	无					
注意事项 / 备注	无					

事件名称	FINS/TCP 连接表设定错误			事件代码	34130000Hex	
内容	FINS/TCP 连接表错误					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	电源接通时 / 控制器复位时 / 重启内置 EtherNet/IP 端口时
异常属性	重要程度	轻度故障	恢复方法	解除异常	日志类别	系统
产生的影响	用户程序	继续	动作	不可使用 FINS/TCP 通信功能		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	下载 FINS/TCP 连接表的过程中发生控制器断电 / 与 Sysmac Studio 的通信切断		请重新下载 FINS/TCP 连接表		下载 FINS/TCP 连接表的过程中，请勿断开控制器的电源 / 切断与 Sysmac Studio 的通信	
附属信息	无					
注意事项 / 备注	无					

事件名称	接收方节点不详			事件代码	34110000Hex	
内容	接收方节点不详					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	发送 FINS 消息时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响。信息包将废弃		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	传送 FINS 信息时，未找到接收方节点		请修改 FINS/UDP 通信方式的接收方节点设定。或者请确认发送源的 FINS 消息，并修改接收方节点地址		请正确设定 FINS/UDP 通信方式的接收方节点。或者请确认发送源 FINS 消息的接收方节点地址是否正确	
附属信息	无					
注意事项 / 备注	无					

事件名称	信息包废弃			事件代码	80100000Hex	
内容	废弃了信息包					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	接收 FINS 消息时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	接收了发送至 CPU 单元的 FINS 响应		请修改发送源的 FINS 消息内容		请正确设定 FINS 消息	
	FINS 响应接收单元不存在					
附属信息	附属信息 1：信息包废弃原因 02Hex：发送响应失败			01Hex：接收发送至 CPU 单元的 FINS 响应		
注意事项 / 备注	无					

事件名称	信息包废弃			事件代码	80110000Hex	
内容	废弃了信息包					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	接收 FINS 消息时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	1：欲发送超过 2002byte 的 FINS 响应		请勿发送超过 2002byte 的 FINS 响应		请正确设定发送源的 FINS 消息	
	2：路由了超过 2002byte 的 FINS 响应					
	3：接收了路由错误 “ 无该号机 ” 的信息包		请确认发送源的 FINS 消息，将响应帧或无需响应的指令帧所指定的号机编号修改为存在的号机编号			
	4：接收了路由错误 “ 路由异常 ” 的信息包		请确认发送源的 FINS 消息，将响应帧或无需响应的指令帧所指定的号机编号修改为路由表中存在的号机编号			
	5：接收了路由错误 “ 路由表未登录 ” 的信息包		请确认发送源的 FINS 消息，修改为将接收方节点登录到路由表中			
	6：接收了路由错误 “ 事件区域大小超限 ” 的信息包		请确认发送源的 FINS 消息，将响应帧或无需响应的指令帧修改为小于事件区域的大小			
	7：内部缓存的可用空间变大		请减少发送源的 FINS 消息发行频率		请尽量减少 FINS 消息的发行频率	
	8：受高负载影响， FINS 信息的路由失败					
附属信息	附属信息 1：信息包废弃原因	1：01 Hex、 2：02 Hex、 3：03 Hex、 4：04 Hex、 5：05 Hex、 6：06 Hex、 7：07 Hex、 8：08 Hex 各编号与上述发生原因的编号对应				
注意事项 / 备注	无					

事件名称	信息包废弃			事件代码	80120000Hex	
内容	废弃了信息包					
发生位置	PLC 功能模块		发生位置详情	无	检测时序	接收 FINS 消息时
异常属性	重要程度	监视信息	恢复方法	—	日志类别	系统
产生的影响	用户程序	继续	动作	无影响		
系统定义变量	变量名称		数据类型		名称	
	无		—		—	
发生原因和处理措施	发生原因 （推测原因）		处理对策		防止再发	
	1：接收了接收方网络地址 (DNA)= 本网络、接收方节点地址 (DA1) ≠本节点的 FINS 响应		请修改 IP 地址表的设定		请确认 IP 地址表的设定是否正确	
	2：接收了接收方网络地址 (DNA) ≧ 80Hex 的集线器网络地址指定的 FINS 指令或 FINS 响应		请修改为发送源的 FINS 消息不会指定为接收方网络地址 (DNA) ≧ 80Hex 的集线器网络地址		请确认发送源的 FINS 消息不会指定为接收方网络地址 (DNA) ≧ 80Hex 的集线器网络地址	
	3：内部缓存的可用空间变大		请减少发送源的 FINS 消息发行频率		请尽量减少 FINS 消息的发行频率	
	4：接收了未满最小指令长度的 FINS 指令		请修改为发送源的 FINS 指令大于最小指令长度		请确保发送源的 FINS 指令大于最小指令长度	
	5：接收了超过最大指令长度的 FINS 指令		请修改为发送源的 FINS 指令小于最大指令长度		请确保发送源的 FINS 指令小于最大指令长度	
	6：信息包发送失败		接收方节点未加入网络时，请加入网络		请确认接收方节点已加入网络	
	7：受高负载影响， FINS 信息的路由失败。或者接收了发送源网络地址 (SNA)=0、发往内置 EtherNet/IP 端口的指令		请减少发送源的 FINS 消息发行频率。或者请修改发送源 FINS 消息的接收方节点地址 (SNA)		请尽量减少 FINS 消息的发行频率。或者请正确设定发送源 FINS 消息的接收方节点地址 (SNA)	
	8：接收了发往内置 EtherNet/IP 端口的 FINS 响应		请修改发送源的 FINS 消息内容		请正确设定 FINS 消息	
	9：未登录路由表时，接收了 FINS 响应或无需响应的指令		请登录路由表		请登录路由表	
	A：发生路由表异常时，接收了 FINS 响应或无需响应的指令		请重新登录路由表。路由表中存在异常时，将发生 “ 用户程序 / 控制器配置和设定不正确 ”(10250000Hex)		无	
	B：接收了超出中继次数的 FINS 响应或无需响应的指令		请增大发送源路由表的网络计数器设定值		请根据系统配置设定发送源路由表的网络计数器	
	C：路由表中未设定接收地址，无法发送		请在路由表中登录接收地址。		请在路由表中登录接收地址。	
	D：[内置 EtherNet/IP 端口设定] 的 [FINS 节点地址设定] 为 0 或 255，无法路由		请通过 Sysmac Studio 将 [内置 EtherNet/IP 端口设定] 的 [FINS 节点地址设定] 设为 0 或 255 以外的值		在 [内置 EtherNet/IP 端口设定] 的 [TCP/IP 设定] 中将 IP 地址的后 8 位设为 0 或 255 时，请在 [IP 地址⇔ FINS 节点地址转换方式] 中选择 [自动生成方式] 以外的选项	

附属信息	附属信息 1：信息包废弃原因	1：01 Hex、 2：02 Hex、 3：03 Hex、 4：04 Hex、 5：05 Hex、 6：06 Hex、 7：07 Hex、 8：08 Hex、 9：09 Hex、 A：0A Hex、 B：0B Hex、C：0C Hex、D：0D Hex 各编号与上述发生原因的编号对应
注意事项 / 备注	无	

12-2-5 CPU 单元以外的异常及其处理

安全相关

编号	故障	处理
1	忘记管理员密码	无法确认设定的管理员密码。请在备忘录中记录管理员密码，防止忘记
2	无法通过 Sysmac Studio 解除操作锁定	操作锁定的解除请以在线连接时的认证权限以上的权限操作
3	在 Sysmac Studio 上进行操作权限认证时操作锁定	操作权限认证时，若连续 5 次输入密码错误，操作将锁定 10 分钟。请等待操作锁定解除
4	已通过操作所需的操作权限认证，并在线连接，但特定操作时需要操作权限认证	以下功能中，为了防止因误操作而危害装置或人员，每次执行功能时都要进行操作权限认证 • 由 “ 操作者 ” 进行 I/O 监视 （写入） • 由 “ 维修者 ” 进行动作模式变更 • 由 “ 维修者 ” 开始在线编辑
5	因操作者离席不在，无法通过 Sysmac Studio 解除操作锁定	通过操作者以上的操作权限认证，可解除操作锁定。此时的操作权限为操作者的操作权限 （连接 Sysmac Studio 时认证的操作权限）
6	部分 （以下）操作时，无法读取用户程序数据 • 监视器 变量 • 动作指示 设置 / 复位、强制值刷新操作、在线编辑、数据追踪操作、 MC 调试、用户程序 （将执行用 ID 设置到 CPU 单元） • 同步 / 上传 / 核对 / 备份 POU 算法	下载用户程序时，正在进行无复原信息传送。进行正常的用户程序传送后可读取
7	部分 （以下）操作时，无法向 CPU 单元写入 • 名称 CPU 单元名称 • 动作指示 在线编辑、存储器全部清除、事件日志清除、将用户程序执行用 ID 设置到 CPU 单元 • 同步 / 下载 用户程序、 CPU・扩展装置的构成和设定、EtherCAT 设定、控制器设定、轴设定、凸轮表设定、数据追踪设定、用户事件设定、恢复	将 CPU 单元的写入保护设为 “ 可写入 ”
8	不知道用户程序的用户程序执行用 ID 变更方法	一旦设定用户程序执行用 ID，将无法变更 / 删除
9	忘记用户程序的用户程序执行用 ID	无法确认设定的用户程序执行用 ID。请在备忘录中记录用户程序执行用 ID，防止忘记
10	忘记 CPU 单元的用户程序执行用 ID	无法确认设定的用户程序执行用 ID。将重新设定用户程序执行用 ID。或者在存储器全部清除后，清除 CPU 单元中的用户程序执行用 ID

A

附录

本章介绍 CPU 单元的详细规格、任务执行处理时间、系统定义变量的个别规格等文本的补充信息。

A

A-1 规格 A-3
A-1-1 一般规格 A-3
A-1-2 性能规格 A-4
A-1-3 功能规格 A-7

A-2 NX 系列中任务实际处理时间的计算标准 A-13
A-2-1 任务实际处理时间平均值的计算方法 A-14
A-2-2 任务实际处理时间平均值的计算和任务周期的设定示例 A-21

A-3 NJ 系列中任务实际处理时间的计算标准 A-24
A-3-1 任务实际处理时间平均值的计算方法 A-25
A-3-2 任务实际处理时间平均值的计算和任务周期的设定示例 A-35

A-4 系统定义变量一览 A-38
A-4-1 NJ/NX 系列控制器的系统整体 （类别名称：无） A-38
A-4-2 PLC 功能模块 （类别名称：_PLC） A-46
A-4-3 PLC 功能模块 （类别名称：_CJB） A-47
A-4-4 运动控制功能模块 （类别名称：_MC） A-49
A-4-5 EtherCAT 主机功能模块 （类别名称：_EC） A-52
A-4-6 EtherNet/IP 功能模块 （类别名称：_EIP） A-56
A-4-7 异常状态中各个位的含义 A-78

A-5 系统定义变量的个别规格 A-79
A-5-1 NJ/NX 系列控制器的系统整体 （类别名称：无） A-80
A-5-2 PLC 功能模块 （类别名称：_PLC） A-93
A-5-3 PLC 功能模块 （类别名称：_CJB） A-95
A-5-4 运动控制功能模块 （类别名称：_MC） A-99
A-5-5 EtherCAT 主机功能模块 （类别名称：_EC） A-102
A-5-6 EtherNet/IP 功能模块 （类别名称：_EIP） A-110

A-6 CPU 单元内数据的属性 A-132

A-7 CJ 单元用存储器的详情 A-135
A-7-1 CIO 区域 A-135
A-7-2 内部辅助继电器 A-138
A-7-3 保持继电器 A-138
A-7-4 数据存储器 A-138
A-7-5 扩展数据存储器 A-139

A-8 变量的存储器确保方法 A-140
A-8-1 变量的存储器确保规则 A-140
A-8-2 应注意的案例 A-149

A-9 CX-Designer 的变量表登录方法 A-153

A-10 EtherCAT 从站或轴的有效 / 无效切换 A-156
A-10-1 使用 EtherCAT 从站和轴时的项目设定 A-156
A-10-2 执行 EtherCAT 从站或轴的有效 / 无效切换的指令 A-156
A-10-3 表示 EtherCAT 从站或轴状态的系统定义变量 A-157
A-10-4 程序的执行 / 停止指令 A-158
A-10-5 确认程序的执行 / 停止 A-158
A-10-6 Sysmac Studio 上的设定 A-159
A-10-7 EtherCAT 从站或轴的有效 / 无效切换应用示例 A-160

A-11 用户程序的大小相关的限制 A-163
A-11-1 用户程序的各部件限制 A-163
A-11-2 用户程序的各部件数量 A-167

A-12 Ver.1.02 以下的 CPU 单元更换方法 A-169
A-12-1 CPU 单元内数据的上传 A-169
A-12-2 连接新的 CPU 单元 A-172
A-12-3 将数据下载到 CPU 单元 A-173

A-13 NX 系列的版本相关信息 A-177
A-13-1 单元版本和 Sysmac Studio 版本之间的关系 A-177
A-13-2 通过更新单元版本追加和变更支持功能 A-179

A-14 NJ 系列的版本相关信息 A-180
A-14-1 单元版本和 Sysmac Studio 版本之间的关系 A-180
A-14-2 硬件修订版本和 Sysmac Studio 版本之间的关系 A-181
A-14-3 通过更新单元版本追加和变更支持功能 A-182
A-14-4 通过单元版本更新改善性能 A-185

A-1 规格

下面介绍 NJ/NX 系列的规格。

A-1-1 一般规格

项目		NX701- □□□□	NJ501- □□□□ NJ301- □□□□ NJ101- □□□□
结构		控制柜内置型	
接地方法		D 种接地 （第 3 种接地）	
外形 高度 × 厚度 × 宽度		100mm×100mm×132mm	90mm×90mm×90mm
质量		880g （含端盖）	550g （含端盖）
消耗电力或消耗电流		40W （含存储卡、端盖）	DC5V 1.90A （含存储卡、端盖）
使用环境	使用环境温度	0 ～ 55 ℃	
	使用环境湿度	10 ～ 95%RH （不凝露）	10 ～ 90%RH （不凝露）
	大气环境	无腐蚀性气体	
	保存环境温度	-25 ～ 70 ℃ （电池、风扇单元除外）	-20 ～ 75 ℃ （不包括电池）
	使用海拔	2000 m以下	
	污染等级	污染等级 2 以下：符合 JIS B 3502、 IEC 61131-2 标准	
	抗干扰性能	符合 IEC 61000-4-4　2kV （电源线）	
	过电压种类	类别 II ：符合 JIS B 3502、 IEC 61131-2 标准	
	EMC 抗扰度等级	区域 B	
	耐振动	符合 IEC 60068-2-6 5 ～ 8.4Hz　振幅 3.5mm、 8.4 ～ 150Hz、加速度 9.8m/s^2 X、 Y、 Z 各方向 100 分钟 （扫描时间 10 分钟 × 扫描次数 10 次 = 总计 100 分钟）	
	耐冲击	符合 IEC 60068-2-27 147m/s^2　X、 Y、 Z 各方向 3 次	符合 IEC 60068-2-27 147m/s^2　X、 Y、 Z 各方向 3 次 （继电器输出单元为 100m/s^2）
电池	寿命*1	2.5 年　25 ℃	5 年　25 ℃
	使用型号	CJ1W-BAT01	
适用标准*2		符合 cULus、 EU 指令、 KC	符合 cULus、 EU 指令、 NK、 LR、 KC*3

*1. 通电时间率 0% （无通电）时的值。
*2. 各型号最新的适用标准请通过本公司主页 (www.fa.omron.co.jp 或 www.ia.omron.com）或本公司销售负责人确认。
*3. 适用于 CPU 单元版本 1.01 以上。

A-1-2 性能规格

项目				NX701-		NJ501-			NJ301-		NJ101-	
				17 □□	16 □□	□5□□	□4□□	□3□□	12 □□	11 □□	10 □□	90 □□
处理时间	指令执行时间	LD 指令		0.37ns ～		1.2ns （～ 1.9ns）*1			2.0ns （～ 3.0ns）		3.3ns （～ 5.0ns）	
		算术指令 （双精度实数型）		3.2ns ～		26ns ～*2			42ns ～		70ns ～	
编程	程序容量*3	大小		80MB		20MB			5MB		3MB	
		数量	POU 定义数	6,000		3,000			750		450	
			POU 实例数	48,000		9,000(*)			3,000(*)		1,800	
	变量容量	有保存属性*4	大小	4MB		2MB			0.5MB			
			变量数量	40,000		10,000			5,000(*)			
		无保存属性*5	大小	256MB		4MB			2MB			
			变量数量	360,000		90,000			22,500			
	数据类型	数据类型的数量		8,000		2,000			1,000			
	CJ 单元用存储器 （可通过变量的 AT 指定进行指定）	通道 I/O （CIO）		—		6144 通道 （0 ～ 6143）						
		工件继电器 （WR）		—		512 通道 （W0 ～ W511）						
		保持继电器 （HR）		—		1536 通道 （H0 ～ H1535）						
		数据存储器 （DM）		—		32768 通道 （D0 ～ D32767）						
		扩展数据存储器 （EM）		—		32768 通道 ×25BANK （E0_00000 ～ E18_32767）			32768 通道 ×4BANK （E0_00000 ～ E3_32767）			
运动控制	控制轴数	最大控制轴数*6		256 轴	128 轴	64 轴	32 轴	16 轴	15 轴 (*)		6 轴	—
		最大使用实轴数*7		256 轴	128 轴	64 轴	32 轴	16 轴	8 轴	4 轴	2 轴	—
		最大单轴控制数		256 轴	128 轴	64 轴	32 轴	16 轴	15 轴 (*)		6 轴	—
		直线插补控制最大数		每 1 轴组 4 轴								—
		圆弧插补控制轴数		每 1 轴组 2 轴								—
	轴组最大数			64 组		32 组						—
	运动控制周期			与 EtherCAT 通信的过程数据通信周期相同								—
	凸轮	凸轮数据点数	每 1 凸轮表的最大点数	65,535 点								—
			所有凸轮表的最大点数	1,048,560 点					262,140 点			—
		凸轮表最大表数		640 表					160 表			—
	位置单位			脉冲、 mm、 μm、 nm、 degree、 inch								—
	超驰			0.00、 0.01 ～ 500.00%								—
外设（USB）端口	支持服务			Sysmac Studio 连接								
	物理层			USB2.0 标准 B 型连接器								
	传送距离			最大 5m								

项目				NX701-		NJ501-			NJ301-		NJ101-	
				17 □□	16 □□	□5□□	□4□□	□3□□	12 □□	11 □□	10 □□	90 □□
内置 EtherNet/IP 端口	端口数量			2		1						
	物理层			10BASE-T/100BASE-TX/1000BASE-T		10BASE-T/100BASE-TX						
	帧长度			最大 1,514 字节								
	介质存取方式			CSMA/CD								
	调制方式			基带								
	拓扑结构			星形								
	传送速度			1Gb/s（1000BASE-T）		100Mb/s （100BASE-TX）						
	传送介质			双绞线电缆 （带屏蔽：STP）：类别 5、 5e 以上								
	传输距离 （集线器和节点之间的距离）最大值			100m								
	串联连接最大数量			使用交换式集线器时无限制								
	CIP 服务：标签数据链接 （周期链接通信）	最大连接数		256 / 端口 总计 512		32						
		Packet 间隔*8		各连接可设定 0.5 ～ 10000ms （以 0.5ms 为单位）		各连接可设定 1 ～ 10000ms （以 1ms 为单位） (*)						
		单元允许通信带宽		40,000 pps*9 （含 Heartbeat）		3,000 pps*9 （含 Heartbeat） (*)						
		最大标签集数		256 / 端口 总计 512		32						
		标签种类		网络变量		网络变量、 CIO / WR / HR / DM / EM						
		1 连接 （=1 标签集）的标签数		8 （标签集中包含控制器状态时为 7）								
		标签最大数量		256 / 端口 总计 512		256						
		每 1 节点的最大链接数据容量 （所有标签的总大小）		369,664 字节		19,200 字节						
		每 1 连接的最大数据容量		1,444 字节		600 字节						
		最大可登录的标签集数		256 / 端口 总计 512 （1 连接 =1 标签集）		32 （1 连接 =1 标签集）						
		1 标签集的最大容量		1,444 字节 （标签集中包含控制器状态时使用 2 字节）		600 字节 （标签集中包含控制器状态时使用 2 字节）						
		多点传送 Packet 过滤功能*10		可								
	CIP 信息服务：Explicit 信息	Class3 （连接数）		128 / 端口 总计 256 （客户端＋服务器）		32 （客户端＋服务器）						
		UCMM （非连接型）	可同时通信的最大客户端数	32 / 端口 总计 64		32						
			可同时通信的最大服务器数	32 / 端口 总计 64		32						
	TCP Socket 数			30(*)								
内置 EtherCAT 端口	通信标准			IEC 61158 Type12								
	EtherCAT 主站规格			符合 Class B （符合 Feature Pack Motion Control）								
	物理层			100BASE-TX								
	调制方式			基带								

项目				NX701-		NJ501-			NJ301-		NJ101-	
				17 □□	16 □□	□5□□	□4□□	□3□□	12 □□	11 □□	10 □□	90 □□
内置 EtherCAT 端口	传送速度			100Mb/s （100BASE-TX）								
	Duplex 模式			Auto								
	拓扑结构			线、菊花链、分支布线								
	传送介质			类别 5 以上　双绞线电缆 （推荐铝带编织双重隔离屏蔽电缆、直线型）								
	节点间距离最大值			100m								
	最大从站数			512		192					64	
	可设定的节点地址范围			1 ～ 512		1 ～ 192						
	过程数据的最大容量			IN：11,472 字节 OUT：11,472 字节 但要在 8 帧 （最大过程数据帧数）范围内		IN：5,736 字节 OUT：5,736 字节 但要在 4 帧 （最大过程数据帧数）范围内						
	1 从站的最大容量			IN：1,434 字节 OUT：1,434 字节								
	通信周期			• 主固定周期任务 125μs 250μs ～ 8ms （可以 250μs 为单位设定） • 固定周期任务 （执行优先度 5） 125μs 250μs ～ 100ms （可以 250μs 为单位设定）		500/1,000/2,000/4,000μs(*)					1,000/2,000/4,000μs	
	同步速度偏差			1μs 以下								
单元构成	可安装的单元数	每 1 装置 （CPU 装置或扩展装置）中 CJ 单元的最大数		—		10						
		系统整体的 CJ 单元最大数		—		40						
		系统整体的 NX 单元最大数		4,000 （EtherCAT 从站终端上）							400 （EtherCAT 从站终端上）	
	最大扩展装置数			0		3						
	输入输出点数	CJ 单元最大输入输出点数		—		2,560						
	电源 （CPU 装置或扩展装置）	型号		NX-PA9001 NX-PD7001		NJ-P □ 3001						
		断电确定时间	AC 电源	30 ～ 45ms								
			DC 电源	5 ～ 20ms		22 ～ 25ms						
内置时钟				环境温度　55 ℃　：　月差　-3.5 分～ +0.5 分 环境温度　25 ℃　：　月差　-1.5 分～ +1.5 分 环境温度　0 ℃　：　月差　-3 分～ +1 分								

*1. 硬件修订版本为 “A” 的单元时，为 “1.1ns （～ 1.7ns） ”。
*2. 硬件修订版本为 “A” 的单元时，为 “24ns ～ ”。
*3. 执行对象、变量表 （变量名等）的容量。
*4. 不含 CJ 单元用存储器的 HR、 DM、 EM。
*5. 不含 CJ 单元用存储器的 CIO、 WR。
*6. 所有轴种类的合计。
*7. 轴种类为 ［伺服轴］或 ［编码器轴］，且轴使用设定为 ［使用轴］的轴合计。
*8. 与节点数无关，以设定周期更新线路中的数据。
*9. pps 代表 Packet Per Second，表示 1 秒内可处理的收发信息包数。
*10.EtherNet/IP 端口安装了 IGMP 客户端，使用了支持 IGMP Snooping 的交换式集线器，所以会进行不必要的多点传送数据包的过滤。

（注）表中带 （*）的项目为版本升级后性能得到改善的功能。升级相关的信息请参考📖「A-13　NX 系列的版本相关信息」（P.A-177）及 📖「A-14 NJ 系列的版本相关信息」（P.A-180）。

A-1-3 功能规格

项目				NX701- □□□□	NJ501- □□□□	NJ301- □□□□	NJ101- □□□□
任务功能	功能			以指定执行条件和执行优先度的 " 任务 " 单位，执行 I/O 刷新及用户程序的功能			
		以固定周期执行的任务	主固定周期任务最大数	1			
			固定周期任务最大数	4	3		
		条件成立时执行的任务 (*)	事件任务最大数	32			
			执行条件	事件任务启动指令执行时、变量的条件表达式一致时			
	设定	系统服务监视设定		—	监视系统服务 （CPU 单元执行的任务处理之外的处理）的执行间隔及相对于用户程序执行时间整体的比例的功能		
编程功能	POU（Program Organization Unit）	程序		任务中分配单位的 POU			
		功能块		编写带状态元件时使用的 POU			
		功能		运算处理等，编写输出与输入一对一的元件时使用的 POU			
	程序语言	种类		梯形图[*1] 结构化文本 （ST）			
	名称空间 (*)			按照名称进行 POU 定义分组的功能			
	变量	变量的外部参照功能	网络变量	许可来自显示器、高位电脑以及其它控制器等的访问的功能			
	数据类型	基本数据型	布尔型	BOOL			
			位列型	BYTE、 WORD、 DWORD、 LWORD			
			整数型	INT、 SINT、 DINT、 LINT、 UINT、 USINT、 UDINT、 ULINT			
			实型数据	REAL、 LREAL			
			持续时间型	TIME			
			日期型	DATE			
			时刻型	TIME_OF_DAY			
			日期时刻型	DATE_AND_TIME			
			字符串型	STRING			
		派生数据型		结构体型、联合体型、列举型			
		结构体型	功能	将数据型的多个不同数据汇总为 1 个进行处理的功能			
			成员最大数	2,048			
			嵌套最大数	8			
			成员的数据类型	基本数据型、结构体型、联合体型、枚举类型、排列变量			
			成员的偏置指定 （*）	将结构体成员配置到任意存储器位置的功能			
		联合体型	功能	可通过多个不同数据类型对同一数据进行访问的功能			
			成员最大数	4			
			成员的数据类型	BOOL、 BYTE、 WORD、 DWORD、 LWORD			
		列举型	功能	将变量值称为 " 枚举例 " 以标签 （字符串）表示的功能			
	数据类型的属性	排列指定	功能	汇总相同数据类型的要素，从开头起使用编号 （后缀）进行指定的功能			
			次元最大数	3			
			要素最大数	65,535			
			FB 实例的排列指定	可以			
		范围指定		明确表示只能提取预先设定的范围内数值的功能			
	库 （*）			可使用 (用户）库			

项目				NX701- □□□□	NJ501- □□□□	NJ301- □□□□	NJ101- □□□□
运动控制功能 *2	控制模式			位置控制、速度控制、转矩控制			
	轴种类			伺服轴、虚拟伺服轴、编码器轴、虚拟编码器轴			
	可管理位置			指令位置、反馈位置			
	单轴	单轴位置控制	绝对值定位	指定绝对坐标的目标位置，进行定位的功能			
			相对值定位	指定指令当前位置起的移动距离，进行定位的功能			
			中断固定尺寸定位	指定基于外部输入的中断输入发生位置起的移动距离，进行定位的功能			
			周期性同步绝对定位控制 (*)	在定位控制模式下，按控制周期输出指令位置的功能			
		单轴速度控制	速度控制	在定位控制模式下进行速度控制的功能			
			周期性同步速度控制	在速度控制模式下，按控制周期输出速度指令的功能			
		单轴转矩控制	转矩控制	执行电机转矩控制的功能			
		单轴同步控制	凸轮动作开始	使用指定的凸轮表开始凸轮动作的功能			
			凸轮动作解除	完成输入参数指定轴的凸轮动作的功能			
			齿轮动作开始	设定主轴与从轴间的齿轮比，进行齿轮动作的功能			
			位置指定齿轮动作	设定主轴与从轴间的齿轮比和要同步的位置，进行齿轮动作的功能			
			齿轮动作解除	中止执行中的齿轮动作、位置指定齿轮动作的功能			
			梯形模式凸轮	与指定的主轴同步进行定位的功能			
			主轴相对值相位补偿	执行同步控制中的主轴相位补偿的功能			
			加减法定位	将 2 个轴的指令位置相加或相减得到的数值，作为指令位置输出的功能			
		单轴手动操作	可运转	将伺服驱动器的状态切换为伺服 ON 状态，实现轴动作的功能			
			点动进给	依据指定的目标速度执行点动进给的功能			
		单轴控制辅助	轴错误复位	解除轴异常的功能			
			原点复位	驱动电机，使用极限信号、近原点信号、原点信号确定机械原点的功能			
			参数指定原点复位 (*)	指定参数驱动电机，使用极限信号、近原点信号、原点信号确定机械原点的功能			
			高速原点复位	将绝对坐标 “0” 作为目标位置，进行定位并返回原点的功能			
			强制停止	使轴减速停止的功能			
			立即停止	使轴立即停止的功能			
			超驰值设定	变更轴目标速度的功能			
			当前位置变更	将轴的指令当前位置和反馈当前位置变更为任意数值的功能			
			外部锁定有效	发生触发，记录轴位置的功能			
			外部锁定无效	使执行中的锁定无效的功能			
			区域监视	判断轴的指令位置或反馈当前位置是否存在于指定范围 （区域）内的功能			
			数字凸轮开关有效 (*)	根据轴的位置将数字输出设为 ON 或 OFF 的功能			
			轴间偏差监视	监视指定的 2 轴指令位置或反馈位置的差异量是否超出了容许值的功能			
			偏差计数器复位	将指令当前位置和反馈当前位置间的偏差归零的功能			
			转矩限制	通过切换伺服驱动器转矩限制功能的有效 / 无效和设定转矩限制值，限制输出转矩的功能			
			指令位置补偿	对动作中的轴进行位置补偿的功能			
			启动速度 (*)	设定轴动作开始时初速度的功能			

项目				NX701- □□□□	NJ501- □□□□	NJ301- □□□□	NJ101- □□□□
运动控制功能 *2	轴组	多轴协调控制	绝对值直线插补	指定绝对位置进行直线插补的功能			
			相对值直线插补	指定相对位置进行直线插补的功能			
			2 轴圆弧插补	进行 2 轴圆弧插补的功能			
			轴组周期性同步绝对位置控制 （*）	在定位控制模式下，按控制周期输出位置指令的功能			
		多轴协调控制辅助	轴组错误复位	解除轴组及轴异常的功能			
			轴组有效	将轴组动作设为有效的功能			
			轴组无效	将轴组动作设为无效的功能			
			轴组强制停止	使插补动作中的所有轴减速停止的功能			
			轴组即停	使插补动作中的所有轴即停的功能			
			轴组超驰值设定	变更插补动作中合成目标速度的功能			
			轴组位置获取 （*）	获取轴组指令当前位置和反馈当前位置的功能			
			轴组构成轴写入 （*）	暂时改写轴组参数的［构成轴］的功能			
	通用	凸轮	凸轮表属性更新	对输入参数指定的凸轮表的终点索引进行更新的功能			
			凸轮表保存	将输入参数指定的凸轮表保存至 CPU 单元内非易失性存储器中的功能			
			凸轮表生成 （*）	根据输入参数指定的凸轮属性和凸轮节点生成凸轮表的功能			
		参数	MC 设定写入	暂时改写部分轴参数及轴组参数的功能			
			轴参数的变更 （*）	通过用户程序查看、变更轴参数的功能			
	辅助功能	计数模式		可选择线性模式 （有限长）或旋转模式 （无限长）			
		单位转换		可根据机械设定各轴的显示单位			
		加减速控制	自动加减速控制	以跃度设定轴及轴组动作时加减速曲线的功能			
			变更加减速度	即使在加减速动作中仍可变更加减速度的功能			
		位置检查		旨在检查定位完成，设定位置宽度和位置检查时间的功能			
		停止方法选择		设定即停输入信号及极限输入信号有效时的停止方法的功能			
		运动控制指令的重启		变更执行中运动控制指令的输入变量并重启，在动作中变更目标值的功能			
		运动控制指令的多重启动（缓冲模式）		对动作中启动其它运动控制指令时的执行开始时间和动作间速度的连接方法进行指定的功能			
		轴组动作的连续动作（转换模式）		对基于轴组动作多重启动的连续动作方法进行指定的功能			
		监视功能	软件限位	监视轴动作范围的功能			
			位置偏差	监视轴的指令当前值和反馈当前值之间的位置偏差的功能			
			速度 / 加减速度 / 扭矩 / 插补速度 / 插补加减速度	按轴及轴组分别设定警告值并监视的功能			
		适用于绝对值编码器		使用欧姆龙制伺服驱动器 G5 系列的带绝对值编码器电机，无需接通电源时原点复位的功能			
		输入信号的逻辑反转 （*）		使立即停止输入信号、正方向极限输入信号、负方向极限输入信号、近原点输入信号的逻辑反转的功能			
	外部 I/F 信号			可使用伺服驱动器侧的下列输入信号 原点信号、近原点信号、正方向极限信号、负方向极限信号、即停信号、中断输入信号			
单元（输入输出）管理功能	EtherCAT 从站	最大从站数		512	192		64
	CJ 系列单元	单元最大数		—	40		

项目				NX701- □□□□	NJ501- □□□□	NJ301- □□□□	NJ101- □□□□
通信功能	外设 （USB）端口			与计算机上的各种支持软件进行通信的功能			
	Ethernet/IP 端口	通信协议		TCP/IP、 UDP/IP			
		CIP 通信服务	标签数据链接	无需程序即可与 EtherNet/IP 网络上的设备进行周期性的数据交换的功能。			
			信息通信	可与 EtherNet/IP 网络上的设备进行任意的 CIP 指令接收 / 发送的功能			
		TCP/IP 应用	Socket 服务	通过 UDP 或 TCP 协议与 Ethernet 上的任意节点之间接收 / 发送任意数据的功能通过 Socket 通信用指令执行的功能			
			FTP 客户端 (*)	通过 CPU 单元使用 FTP 向 Ethernet 上的其它计算机及控制器传输文件的功能。通过 FTP 客户端通信指令执行			
			FTP 服务器	通过 Ethernet 上的其它计算机读写控制器的 CPU 单元的 SD 存储卡内文件的功能			
			钟表自动调整	以指定的时刻或接通 CPU 单元电源后指定的一定时间间隔，从 NTP 服务器获取时间信息，更新 CPU 单元内部钟表信息的功能			
			SNMP 代理	向使用 SNMP 管理器的网络管理软件提供内置 EtherNet/IP 端口内部状态信息的功能			
	EtherCAT 端口	支持服务	过程数据通信	在通过 CoE 定义的 EtherCAT 主站和从站之间，将控制信息作为一定周期内周期性通信进行数据交换的通信方式			
			SDO 通信	在通过 CoE 定义的 EtherCAT 主站和从站之间，将控制信息作为非固定周期的事件通信进行数据交换的通信方式			
		网络扫描		读取连接的从站设备的信息，自动生成从站构成的功能			
		DC （Distributed Clock）		所有的 EtherCAT 设备 （包含主站）共享相同 “EtherCAT System Time”，进行时刻同步的功能			
		Packet 监视器		保存主站发送和接收的帧的功能。可通过 WireShark 等查看已保存的数据 (*)			
		设定从站有效 / 无效		将从站作为通信对象，设定为有效或无效的功能			
		从站脱离 / 再加入		出于更换从站等保养的目的，暂时将相应从站从 EtherCAT 网络中脱离 / 再加入的功能			
		支持应用协议	CoE	在 EtherCAT 中，向从站发送 CAN 应用信息 （SDO）的功能			
	通信用指令			FTP 客户端指令、CIP 通信指令、 Socket 通信用指令、 SDO 信息指令、Modbus RTU 协议指令 (*)	FTP 客户端指令 (*)、 CIP 通信指令、 Socket 通信用指令、SDO 信息指令、无协议通信指令、协议宏指令、 Modbus RTU 协议指令 (*)		
运转管理功能	运转中输出接点			运转模式时，电源单元的接点变为 ON 的功能			
系统管理功能	事件日志	功能		记录事件发生的功能			
		最多件数	系统事件日志	2,048	1,024	512	
			访问事件日志	1,024	1,024	512	
			用户事件日志	1,024	1,024	512	

项目				NX701- □□□□	NJ501- □□□□	NJ301- □□□□	NJ101- □□□□
调试功能	在线编辑			在线变更程序、功能块、功能、全局变量的功能 经由网络多个操作者可变更个别 POU			
	强制值刷新			以强制值刷新特定接点的功能			
		最大点数	EtherCAT 从站设备变量	64			
			CJ 单元设备变量、 AT （分配对象）指定的变量	—	64		
	MC 调试 *2			通过 Sysmac Studio 确认电机动作及接线的功能			
	同步			在线连接时将 Sysmac Studio 的项目文件和 CPU 单元的数据实现一致的功能			
	微分监控 (*)			监控接点的上升沿或下降沿的功能			
		最大点数		8			
	数据追踪	种类	触发跟踪	建立触发后，达到设定的采样数时自动停止跟踪的功能			
			连续跟踪	继续执行数据跟踪， Sysmac Studio 始终收集跟踪数据的功能			
		同时启动最大数		4		2	
		记录最大数		10,000			
		采样最大点数		192 变量		48 变量	
		采样时间		指定任务周期、指定时间、采样指令执行时			
		触发跟踪		通过设定触发条件，记录问题发生前后数据的功能			
			触发条件	• BOOL 型变量的上升沿 / 下降沿 • BOOL 型变量以外的常数值比较 比较方法：＝、＞、≧、＜、≦、≠			
			延时值	设定触发成立前 / 成立后采样数比例的功能			
	仿真功能			通过 Sysmac Studio 模拟 CPU 单元动作的功能			
高可靠性功能	自诊断	控制器异常	重要程度	全部停止故障、部分停止故障、轻度故障、监视信息、一般信息			
			信息语言最大数	9(Sysmac Studio) 2 （NS 系列显示器）			
		用户异常	功能	设计任意异常并预先登录，通过执行指令留下记录的功能			
			重要程度	8 级			
			信息语言最大数	9			
安全功能	顾客资产保护 / 防止误操作	CPU 单元名称功能及串行 ID 功能		通过 Sysmac Studio 进行在线连接时，确认项目中的 CPU 单元名称与连接对象 CPU 单元的 CPU 名称是否一致的功能			
		保护功能	无用户程序复原信息传送功能	进行设置使其不能通过 Sysmac Studio 读取 CPU 单元内数据的功能			
			CPU 单元的写入保护功能	进行设置使其不能通过 Sysmac Studio/SD 存储卡写入 CPU 单元内数据的功能			
			项目文件整体的保护功能	通过密码禁止在 Sysmac Studio 中打开 .smc 文件的功能			
			数据保护功能 （*）	通过密码在 Sysmac Studio 中对 POU 进行保护的功能			
		操作权限的认证功能		因操作失误可能对装置或人施加危害时，根据操作权限限制在线操作功能的功能			
			组数 （*）	5			
		用户程序执行用 ID 认证功能		对于特定的硬件 （CPU 单元），只要不从 Sysmac Studio 输入用户程序执行用 ID，就不能执行用户程序的功能			

项目				NX701- □□□□	NJ501- □□□□	NJ301- □□□□	NJ101- □□□□
SD 存储卡功能	存储器的种类			SD 卡、 SDHC 卡			
	应用程序	SD 存储卡自动传送功能 (*)		接通控制器电源时，将 SD 存储卡内 autoload 目录中保存的数据下载至控制器的功能			
		SD 存储卡程序传送功能 (*)		根据系统定义变量的指示，将 SD 存储卡中保存的用户程序传送到控制器的功能			
		SD 存储卡操作指令		通过用户程序中的指令访问 SD 存储卡的功能			
		通过 Sysmac Studio 的文件操作		用户不仅可以向 SD 存储卡保存及读取控制器用文件，还可以保存及读取通用文档文件等			
		SD 存储卡的寿命检测功能		通过系统定义变量或事件日志通知 SD 存储卡寿命的功能			
备份功能 (*)	SD 存储卡的备份功能	操作方法	CPU 单元正面开关	通过 CPU 单元的正面开关操作，进行备份、比对、恢复的功能			
			通过系统定义变量进行指示	通过系统定义变量的操作进行备份及比对的功能			
			SysmacStudio 存储卡画面	通过 Sysmac Studio 的 SD 存储卡操作画面进行备份及比对的功能			
			专用指令 (*)	通过专用指令进行备份的功能			
		保护功能	禁止向 SD 存储卡备份	禁止向 SD 存储卡备份的功能			
	Sysmac Studio 控制器备份功能			使用 Sysmac Studio 进行控制器备份、恢复、比对的功能			

*1. 可联机 ST （梯形图中描述 ST 语言的梯形图语言的要素）。
*2. NJ101-90 □□不适用。

（注）表中带（*）的项目为版本升级后追加的功能。升级相关的信息请参考「A-13 NX 系列的版本相关信息」（P.A-177）及「A-14 NJ 系列的版本相关信息」（P.A-180）。

A-2 NX 系列中任务实际处理时间的计算标准

下面介绍 NX 系列系统中任务实际处理时间平均值的参考计算方法。

任务实际处理时间和任务执行时间需要在实际设备的控制器上确认。详情请参考📖「5-10 任务设计方法及输入输出响应时间」（P.5-87）。

安全要点

在实际的控制器上，因用户程序中搭载的逻辑是否动作、是否发生通信指令、数据链接动作、是否使用数据追踪等各种功能不同，任务执行时间会发生变动。
请务必使用实际机器在各种条件下确认性能，确认不会发生任务周期超限，确保良好的通信响应性。

使用注意事项

计算固定周期任务 （执行优先度 5）的任务实际处理时间平均值时，在使用 Sysmac Studio 创建 EtherCAT 构成后，需要 EtherCAT 构成编辑画面中显示的［PDO 通信周期 2: 传送延迟时间］的值。

参考

正在执行优先度高于固定周期任务的任务时，固定周期任务会暂停。“ 任务实际处理时间 ” 不含暂停的时间。“ 任务执行时间 ” 中包含暂停时间，表示任务从开始到结束的时间。“ 任务实际处理时间 ” 和 “ 任务执行时间 ” 的详细区别请参考📖「 任务实际时间和任务实际处理时间的含义」（P.5-86）。

A-2-1 任务实际处理时间平均值的计算方法

任务实际处理时间的平均值为 I/O 刷新处理时间、用户程序执行时间、运动控制处理时间、共通处理时间的总和。

任务实际处理时间的平均值 =I/O 刷新处理时间 + 用户程序执行时间
+ 运动控制处理时间 + 共通处理时间

各处理内容如下所示。

○：有，—：无

处理		处理内容	主 固定周期任务	固定周期任务（执行优先度 5）	固定周期任务（执行优先度 16）	固定周期任务（执行优先度 17、18）
I/O 刷新处理		执行与 EtherCAT 从站的刷新。	○	○	—	—
用户程序执行		• 按分配顺序执行任务中分配的程序。	○	○	○	○
运动控制处理		• 执行用户程序发出的运动控制指令。 • 运动输出处理	○	○	—	—
共通处理时间	系统共通处理 1	• 变量的更新处理 （参照任务时） • 运动输入处理 • 数据追踪处理	○	○	○	○
	系统共通处理 2	• 变量的更新处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理	○	○	○	○
	系统额外时间	上述以外的系统共通处理	○	○	○	○

下面表示各处理时间的计算标准。

I/O 刷新处理时间

I/O 刷新处理时间按以下公式计算。

I/O 刷新处理时间 =EtherCAT 从站处理时间

EtherCAT 从站处理时间在主固定周期任务和固定周期任务 （执行优先度 5）中的计算公式不同。
下面表示各任务中 EtherCAT 从站处理时间的计算方法。

使用注意事项

计算固定周期任务 （执行优先度 5）时，首先需要确定主固定周期任务的任务周期。

参考

在 EtherCAT 中未分配从站的任务中， EtherCAT 从站处理时间为 “0”。

● 主固定周期任务中的 EtherCAT 从站处理时间

主固定周期任务中的 EtherCAT 从站处理时间按以下公式计算。

EtherCAT 从站处理时间 [μs] ＝ 0.0006×pDout + 0.0001×pDin + 0.082×pDinout
+ (1.24×Snum + 0.01×Clen)

pDout	:	分配到主固定周期任务的 EtherCAT 从站的输出过程数据总大小 [字节]
pDin	:	分配到主固定周期任务的 EtherCAT 从站的输入过程数据总大小 [字节]
pDinout	:	分配到主固定周期任务的各 EtherCAT 从站的输出过程数据和输入过程数据中总大小较大者 [字节]
Snum	:	内置 EtherCAT 端口上连接的 EtherCAT 从站总台数 [台]
Clen	:	内置 EtherCAT 端口上连接的电缆总长度 [m]

● 固定周期任务 （执行优先度 5）中的 EtherCAT 从站处理时间

固定周期任务 （执行优先度 5）中的 EtherCAT 从站处理时间按以下公式计算。

EtherCAT 从站处理时间 [μs] ＝ 0.0006×pDout + （以下 A 和 B 中的较大者）
+ sTsend + 0.0001×sDin + 16

A[μs] ＝ ((0.0623×sDout + 37)×pTcycle)÷(pTcycle － (0.0243×pDout ＋ 35))
B[μs] ＝ 0.082×pDinout

pDout	:	分配到主固定周期任务的 EtherCAT 从站的输出过程数据总大小 [字节]
sTsend*1	:	Sysmac Studio 上显示的 PDO 通信周期 2 的传送延迟时间 [μs]
sDin	:	分配到固定周期任务 （执行优先度 5）的 EtherCAT 从站的输入过程数据总大小 [字节]
sDout	:	分配到固定周期任务 （执行优先度 5）的 EtherCAT 从站的输出过程数据总大小 [字节]
pTcycle	:	主固定周期任务的任务周期 [μs]
pDinout	:	分配到主固定周期任务的各 EtherCAT 从站的输出过程数据和输入过程数据中总大小较大者 [字节]

*1. sTsend（传送延迟时间）无法用公式计算。使用 Sysmac Studio 创建 EtherCAT 构成后，在 EtherCAT 构成编辑画面中显示的 ［PDO 通信周期 2: 传送延迟时间] 的值中代入 sTsend （传送延迟时间）。
在 Sysmac Studio 上显示 ［PDO 通信周期 2: 传送延迟时间] 的方法请参考📖《Sysmac Studio Version 1 操作手册 （SBCA-CN5-362P 以后）》。

用户程序执行时间

用户程序执行时间取决于指令种类 × 指令数。

作为参考，按以下 3 大分类来检测和预计指令数。

- 标准指令
- 双精度实数 （LREAL）型的四则运算指令
- 双精度实数 （LREAL）型的三角函数指令

此外，梯形图程序和 ST 语言程序中使用的指令不同。各指令的构成请参考 「“ 梯形图语言的标准指令 ” 的指令种类构成」（P.A-17）及 「“ST 语言的标准指令 ” 的指令种类构成」（P.A-19）。

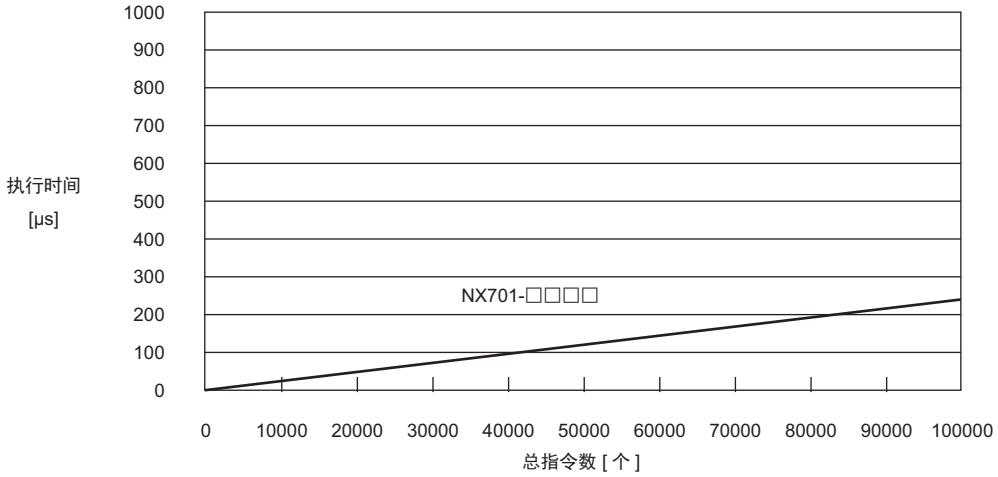
● 简单估算

根据各分类的指令数，按以下图表，读取以下各指令的执行时间并合计。

- 标准指令的执行时间
- 双精度实数 （LREAL）型的四则运算指令执行时间
- 双精度实数 （LREAL）型的三角函数指令执行时间

由此可估算出用户程序的执行时间。

梯形图语言的标准指令执行时间

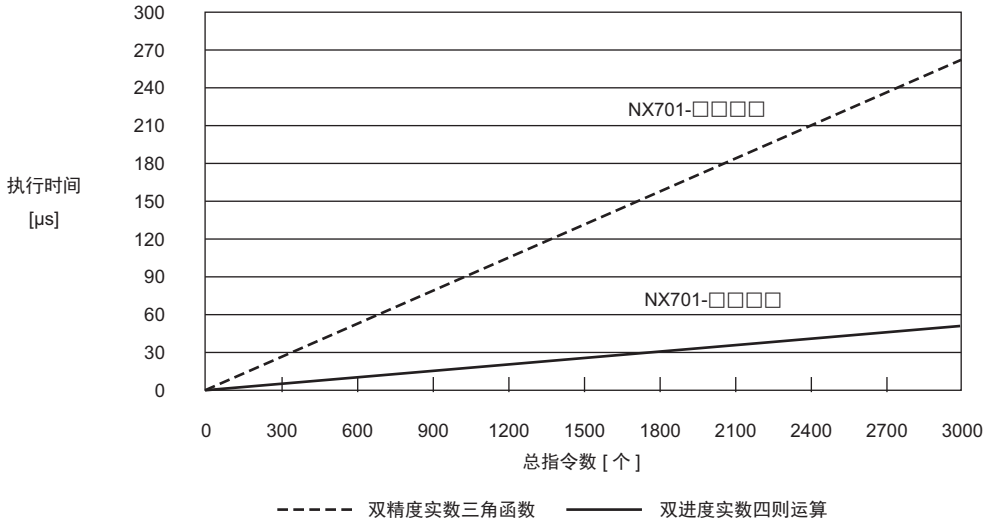


- “ 梯形图语言的标准指令 ” 的指令种类构成

本构成的指令执行率为 20%。

指令种类	指令名称	指令个数的比率 [%]	各指令种类的执行时间比率 [%]
梯形图指令	LD, AND, OUT, SET, RESET	81.0	40.2
比较指令	EQ, LT	4.1	8.3
计时指令、计数指令	Timer, TON/TOF, CTU/CTD	1.6	7.3
算术指令	+, -, *, /, ADD, SUB, MUL, DIV	2.4	6.5
BCD 转换指令、数据转换指令	INT_TO_DINT, WORD_BCD_TO_UINT	0.2	1.2
位列运算指令	AND, OR	6.2	13.0
数据传送指令	MOVE	4.6	23.5
总计		100.0	100.0

梯形图语言双精度实数 （LREAL）型的四则运算指令和三角函数指令执行时间



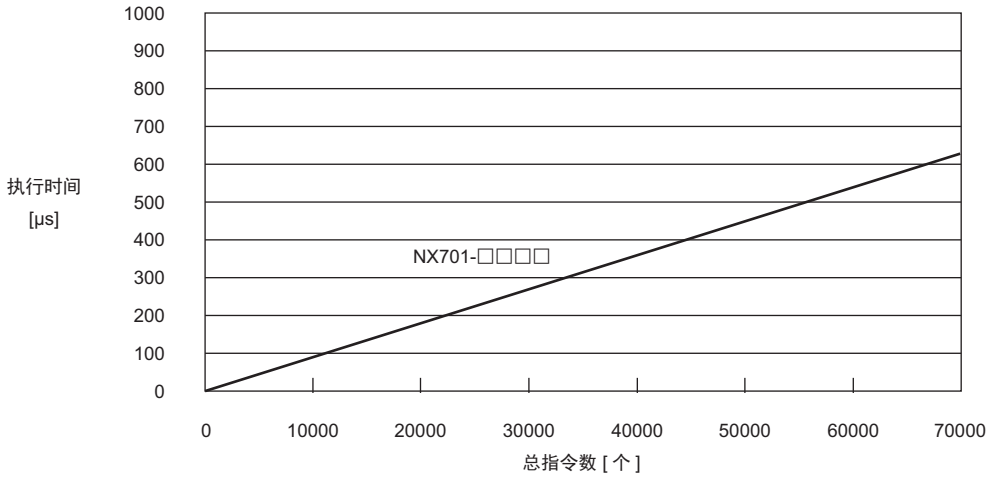
- “ 双进度实数的四则运算 ” 构成

指令名称	指令个数的比率 [%]
双精度实数的加法	20.0
双精度实数的减法	20.0
双精度实数的乘法	30.0
双精度实数的除法	30.0
总计	100.0

- “ 双精度实数的三角函数 ” 构成

指令名称	指令个数的比率 [%]
双精度实数的 Sin	16.7
双精度实数的 Cos	16.7
双精度实数的 Tan	16.7
双精度实数的 Sin^{-1}	16.7
双精度实数的 Cos^{-1}	16.7
双精度实数的 Tan^{-1}	16.7
总计	100.0

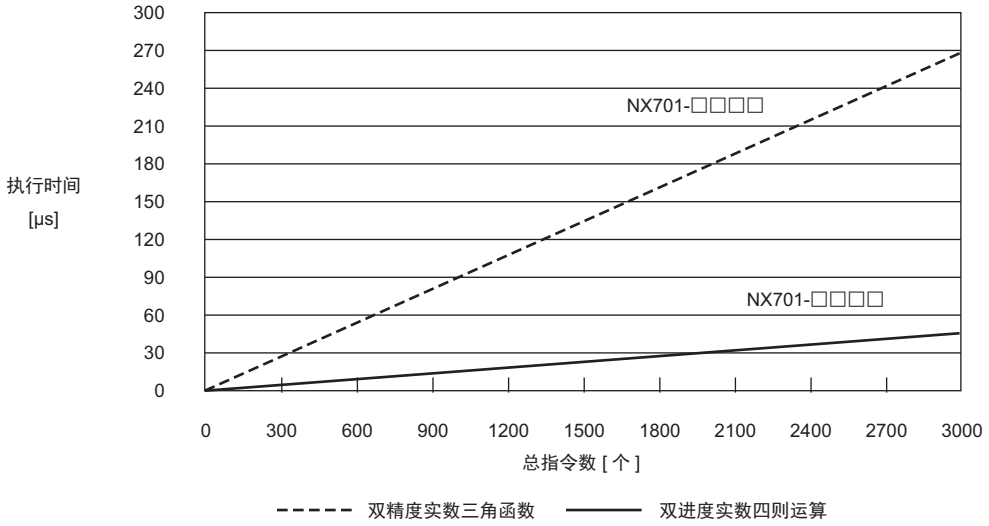
ST 语言的标准指令执行时间



- “ST 语言的标准指令 ” 的指令种类构成

指令种类	指令名称	指令个数的比率 [%]	各指令种类的执行时间比率 [%]
ST 语法指令	IF ELSIF END_IF	75.4	41.6
比较指令	EQ, LT	5.2	8.7
计时指令、计数指令	Timer, TON/TOF, CTU/CTD	2.1	18.8
算术指令	+, -, *, /	3.1	10.2
BCD 转换指令、数据转换指令	INT_TO_DINT, WORD_BCD_TO_UINT	0.2	1.6
位列运算指令	AND, OR	8.0	11.7
数据传送指令	:=	5.9	7.3
总计		100.0	100.0

ST 语言双精度实数 （LREAL）型的四则运算指令和三角函数指令执行时间



- “ 双进度实数的四则运算 ” 构成

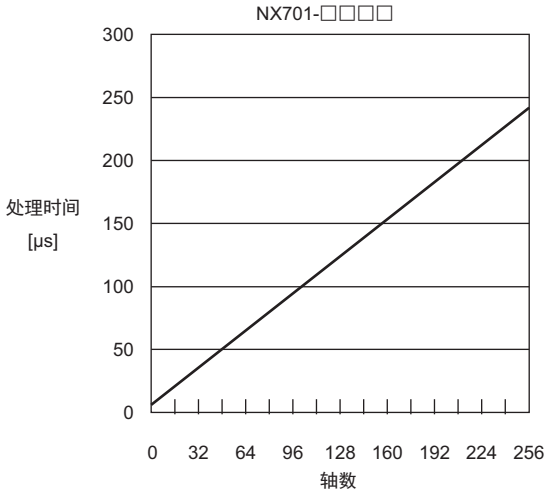
指令名称	指令个数的比率 [%]
双精度实数的加法	20.0
双精度实数的减法	20.0
双精度实数的乘法	30.0
双精度实数的除法	30.0
总计	100.0

- “ 双精度实数的三角函数 ” 构成

指令名称	指令个数的比率 [%]
双精度实数的 Sin	16.7
双精度实数的 Cos	16.7
双精度实数的 Tan	16.7
双精度实数的 Sin^{-1}	16.7
双精度实数的 Cos^{-1}	16.7
双精度实数的 Tan^{-1}	16.7
总计	100.0

运动控制处理时间

运动控制处理时间取决于所用伺服的轴数 （包括虚拟轴）。
根据使用的伺服轴数，按以下图表，读取运动控制处理时间。



共通处理时间

共通处理时间是指系统额外时间、系统共通处理 1 及系统共通处理 2 的总和，值如下。共通处理时间因任务类型不同而异。

任务类型	共通处理时间 [μs] （参考值）
	NX701- □□□□
主固定周期任务	45
固定周期任务 （执行优先度 5）	55
固定周期任务 （执行优先度 16、 17、 18）	10

A-2-2 任务实际处理时间平均值的计算和任务周期的设定示例

任务实际处理时间平均值的计算示例

计算使用单元版本为 1.10 的 NX701- □□□□、以下条件下的任务实际处理时间的平均值。
任务类型为主固定周期任务。

项目		条件	
使用的从站 / 单元	EtherCAT 从站	输入从站	GX-ID1611(Ver.1.1)：1 台
		输出从站	GX-OD1611(Ver.1.1)：1 台
		伺服电机	R88D- □□ (Ver.2.1)：4 台
		EtherCAT 从站终端：1 台	
		EtherCAT 耦合器单元	NX-ECC203：1 台
		DC 输入单元	NX-ID5342：1 台
		晶体管输出单元	NX-OD5121：1 台
		模拟输入单元	NX-AD3608：1 台
		模拟输出单元	NX-DA2605：1 台
		通讯接口单元	NX-CIF101：1 台 NX-CIF105：1 台
用户程序：	使用语言	梯形图语言	
	标准指令构成	指令数：5,000 个	
	双进度实数的四则运算	指令数：200 个	
	双精度实数的三角函数	指令数：100 个	
运动控制处理	轴数	4 轴	

（注）内置 EtherCAT 端口上连接的电缆总长度为 10[m]。

● **I/O 刷新时间**

EtherCAT 从站处理时间：

输入从站 GX-ID1611(Ver.1.1)、输出从站 GX-OD1611(Ver.1.1)、伺服电机 R88D- □□ (Ver.2.1) 及上述构成下 EtherCAT 从站终端的输出过程数据大小 pDout、输入过程数据大小 pDin、输出数据大小和输入数据大小中的较大者 pDinout 的值如下所示。

EtherCAT 从站	输出过程数据大小 pDout [字节]	输入过程数据大小 pDin [字节]	输出数据大小和输入数据大小的较大者 pDinout
GX-ID1611(Ver.1.1)	0	3	3
GX-OD1611(Ver.1.1)	2	1	2
R88D- □□ (Ver.2.1)	23	26	26
EtherCAT 从站终端合计	62	88	88
NX-ECC203	0	18	—
NX-ID5342	0	2	
NX-OD5121	2	0	
NX-AD3608	0	8	
NX-DA2605	4	0	
NX-CIF101	28	30	
NX-CIF105	28	30	

pDout、 pDin 及 pDinout 各自的总计值如下。

pDout=2 + 23×4 + 62=156[字节]
pDin=3 + 1 + 26×4 + 88=196[字节]
pDinout=3 + 2 + 26×4 + 88=197[字节]

根据上述值，按以下公式计算 I/O 刷新时间。

I/O 刷新处理时间

=EtherCAT 从站处理时间

=0.0006×pDout + 0.0001×pDin + 0.082×pDinout + (1.24×Snum + 0.01×Clen)

=0.0006×156 + 0.0001×196 + 0.082×197 + (1.24×7 + 0.01×10)

=0.0936 + 0.0196 + 16.154 + (8.68 + 0.1)

≒ 25[μs]

● 用户程序执行时间

从各图表中读取以下数值。

- 标准指令构成
  根据梯形图语言的标准指令执行时间图表，NX701-□□□□中指令数为 5000 个的用户程序执行时间为 12μs。
- 双进度实数的四则运算
  根据梯形图语言的双精度实数（LREAL）型的四则运算指令和三角函数指令的执行时间图表，NX701-□□□□中指令数为 200 个的用户程序执行时间为 4μs。
- 双精度实数的三角函数
  根据梯形图语言的双精度实数（LREAL）型的四则运算指令和三角函数指令的执行时间图表，NX701-□□□□中指令数为 100 个的用户程序执行时间为 9μs。

因此，用户程序执行时间为以上的合计值，结果如下。

用户程序执行时间 =12 + 4 + 9

=25[μs]

● 运动控制处理时间

根据运动控制处理的执行时间图表，NX701-□□□□的 4 轴运动控制处理的执行时间读取为 10μs。

● 共通处理时间

任务类型为主固定周期任务，因此 NX701-□□□□的共通处理时间为 45μs。

因此，任务实际处理时间的平均值按以下公式计算。

任务实际处理时间的平均值 =I/O 刷新时间 + 用户程序执行时间

+ 运动控制处理 + 共通处理时间

=25 + 25 + 10 + 45

=105[μs]

任务周期的设定示例

根据上面计算得到的任务实际处理时间的平均值，设定任务周期。任务类型为主固定周期任务。
任务周期的值应大于计算得到的任务实际处理时间的平均值。此时，请预留出余量，将任务周期的值设定为大于任务实际处理时间平均值的 1.1 倍。

任务周期≧任务实际处理时间的平均值 ×1.1

在实际的控制器上，因用户程序中搭载的逻辑是否动作、是否发生通信指令、数据链接动作、是否使用数据追踪等各种功能不同，任务执行时间会发生变动。此外，如果是固定周期任务，因是否有执行优先度更高的任务动作，任务执行时间会发生变动。
请务必使用实机的控制器，根据任务执行时间监视器，进行任务执行时间的验证。

A-3 NJ 系列中任务实际处理时间的计算标准

下面介绍 NJ 系列系统中任务实际处理时间平均值的参考计算方法。

任务实际处理时间和任务执行时间需要在实际设备的控制器上确认。详情请参考📖「5-10 任务设计方法及输入输出响应时间」（P.5-87）。

安全要点

- 在实际的控制器上，因用户程序中搭载的逻辑是否动作、是否发生通信指令、数据链接动作、是否使用数据追踪等各种功能不同，任务执行时间会发生变动。
  请务必使用实际机器在各种条件下确认性能，确认不会发生任务周期超限，确保良好的通信响应性。
- 若硬件修订版本不同，性能也会不同。向不同硬件修订版本的 CPU 单元传送用户程序及各种数据、设定值时，请进行充分的动作确认后再转移到正式运行。

使用注意事项

硬件修订版本不同时，即使用户程序及各种数据、设定值相同，事件任务、标签数据链接服务及系统服务的实际执行时间也可能不同。

参考

正在执行优先度高于固定周期任务的任务时，将暂停。“ 任务实际处理时间 ” 不含暂停的时间。“ 任务执行时间 ” 中包含暂停时间，表示任务从开始到结束的时间。 “ 任务实际处理时间 ” 和 “ 任务执行时间 ” 的详细区别请参考📖「 任务实际时间和任务实际处理时间的含义」（P.5-86）。

A-3-1 任务实际处理时间平均值的计算方法

任务实际处理时间的平均值为 I/O 刷新处理时间、用户程序执行时间、运动控制处理时间、共通处理时间的总和。

任务实际处理时间的平均值 =I/O 刷新处理时间 + 用户程序执行时间
+ 运动控制处理时间 + 共通处理时间

各处理内容如下所示。

○：有， —：无

处理		处理内容	主 固定周期任务	固定周期任务 （执行优先度 16）	固定周期任务 （执行优先度 17、 18）
I/O 刷新处理		执行与 CJ 单元 （基本 I/O 单元、高功能 I/O 单元、 CPU 总线单元）、 EtherCAT 从站的刷新。	○	○	—
用户程序执行		• 按分配顺序执行任务中分配的程序。	○	○	○
运动控制处理		• 执行用户程序发出的运动控制指令。 • 运动输出处理	○	—	—
共通处理时间	系统 共通处理 1	• 变量的更新处理 （参照任务时） • 运动输入处理 • 数据追踪处理	○	○	○
	系统 共通处理 2	• 变量的更新处理 （更新任务时） • 为确保与任务执行的同时性，从控制器外部进行的变量访问处理	○	○	○
	系统额外时间	上述以外的系统共通处理	○	○	○

下面表示各处理时间的计算标准。

I/O 刷新处理时间

I/O 刷新处理时间按以下公式计算。

I/O 刷新处理时间 =I/O 刷新额外时间
+(EtherCAT 从站处理时间和 CJ 单元处理时间的较大者）

下面介绍上述 I/O 刷新额外时间、 EtherCAT 从站处理时间、 CJ 单元处理时间的计算方法。

● I/O 刷新额外时间

I/O 刷新额外时间根据 EtherCAT 从站的有无、CJ 单元的有无及 CPU 单元的型号不同，以下表为准。

EtherCAT 从站	CJ 单元	I/O 刷新额外时间 [μs]		
		NJ501- □□□□	NJ301- □□□□	NJ101- □□□□
有	有	65*1	95	135
有	无	35*2	50	65
无	有	30	45	70

*1. CPU 单元版本低于 1.01 时为 “90”。
*2. CPU 单元版本低于 1.01 时为 “60”。

● EtherCAT 从站处理时间

EtherCAT 从站处理时间按以下公式计算。
请对所有 EtherCAT 从站进行计算。

EtherCAT 从站处理时间 [μs] ＝ Tout×Dout + Tin×Din + Tref×Dinout
+ (1.24×Snum + 0.01×Clen － Tec)

Tout : 每 1 字节的输出处理时间 [μs]
Dout : 输出过程数据大小总计 [字节]
Tin : 每 1 字节的输入处理时间 [μs]
Din : 输入过程数据大小总计 [字节]
Snum : EtherCAT 从站连接台数 [台]
Tref : 每 1 字节的刷新处理时间 [μs]
Dinout : 输出过程数据大小总计和输入过程数据大小总计的较大者 [字节]
Clen : 电缆总长 [m]
Tovh : I/O 刷新额外时间 [μs]
Tec : EtherCAT 通信调整时间 [μs]

上述表达式中的输出处理时间、输入处理时间、刷新处理时间及 EtherCAT 通信调整时间为固定值。根据 CPU 单元的型号不同，具体如下。

CPU 单元	每 1 字节的输出处理时间 Tout [μs]	每 1 字节的输入处理时间 Tin[μs]	每 1 字节的刷新处理时间 Tref[μs]	EtherCAT 通信调整时间 Tec[μs]
NJ501- □□□□	0.004	0.011	0.082	55*1
NJ301- □□□□	0.005	0.013	0.082	90
NJ101- □□□□	0.010	0.004	0.082	145

*1. CPU 单元版本低于 1.01 时为 “70”。

此外，上述表达式中的 “1.24×Snum + 0.01×Clen － Tec” 计算结果为负数时，视为 “0”。

● CJ 单元处理时间

CJ 单元处理时间按以下公式计算。

Σ （各 CJ 单元的 I/O 刷新时间 × 单元台数）－ Tcj [μs]

但是，上述表达式中的 Σ 表示对所有 CJ 单元的总计。
此外，上述表达式的计算结果为负数时， CJ 单元处理时间视为 0μs。
各 CJ 单元的 I/O 刷新时间的计算方法后面再介绍。
Tcj 因 CPU 单元的型号和单元版本不同而异。

型号	单元版本	Tcj [μs]
NJ501- □□□□	Ver.1.01 以下	230
	Ver.1.02 以上	110
NJ301- □□□□	Ver.1.01 以下	230
	Ver.1.02 以上	165
NJ101- □□□□	Ver.1.10 以上*1	225

*1. NJ101- □□□□无 1.09 以下的版本。

使用 1 台或更多以下 CJ 单元时，无论单元台数多少，请在上述表达式的计算结果中加上 +Tcj [μs]。

- 绝缘型全量程多输入单元 CJ1W-PH41U
- 绝缘型全量程多输入单元 CJ1W-AD04U
- 绝缘型直流输入单元　　CJ1W-PDC15
- ID 传感器单元　CJ1W-V680C11
- ID 传感器单元　CJ1W-V680C12
- CompoNet 主站单元　CJ1W-CRM21

CJ 单元的 I/O 刷新时间

各 CJ 单元的 I/O 刷新时间如下所示。

● 基本 I/O 单元

在 CPU 装置中使用基本 I/O 单元时的 I/O 刷新时间如下所示。在扩展装置中使用时的 I/O 刷新时间约为表中值的 1.5 倍。

单元名称	型号	每 1 个单元的 I/O 刷新时间 [μs]
DC 输入单元 （8/16 点）	CJ1W-ID201/211/212	1
DC 输入单元 （32 点）	CJ1W-ID231/232/233	2
DC 输入单元 （64 点）	CJ1W-ID261/262	4
AC 输入单元 （8 点 /16 点）	CJ1W-IA201/111	1
中断输入单元 （16 点）	CJ1W-INT01	1
快速响应输入单元 （16 点）	CJ1W-IDP01	1
继电器接点输出单元	CJ1W-OC201/OC211	1
三端双向输出单元	CJ1W-OA201	1
晶体管输出单元 （8 点 /16 点）	CJ1W-OD201/202/203/204/211/212/213	1
晶体管输出单元 （32 点）	CJ1W-OD231/232/233/234	2
晶体管输出单元 （64 点）	CJ1W-OD261/262/263	4
DC24V 输入 / 晶体管输出单元 （输入 16 点 / 输出 16 点）	CJ1W-MD231/232/233	1
DC24V 输入 / 晶体管输出单元 （输入 32 点 / 输出 32 点）	CJ1W-MD261/263	2
TTL 输入输出单元 （16 点输入 /16 点输出）	CJ1W-MD563	4
B7A 接口单元	CJ1W-B7A04	4
	CJ1W-B7A14	4
	CJ1W-B7A22	4

● 高功能 I/O 单元

在 CPU 装置中使用高功能 I/O 单元时的 I/O 刷新时间如下所示。在扩展装置中使用时的 I/O 刷新时间约为表中值的 1.5 倍。

单元名称	型号		每 1 个单元的 I/O 刷新时间 [μs]
绝缘型全量程多输入单元	CJ1W-AD04U		66
模拟输入单元	CJ1W-AD041-V1/081-V1/042		24
模拟输出单元	CJ1W-DA021/041/042V/08V/08C		24
模拟输入输出单元	CJ1W-MAD42		24
绝缘型全量程多输入单元	CJ1W-PH41U		80(180)*1
绝缘型直流输入单元	CJ1W-PDC15		60(100)*1
温度控制单元	CJ1W-TC □□□		114
ID 传感器单元	CJ1W-V680C11		76
	CJ1W-V680C12		86
高速计数器单元	CJ1W-CT021		54
CompoNet 主站单元	CJ1W-CRM21	通信模式 No.0	34
		通信模式 No.1	52
		通信模式 No.2	88
		通信模式 No.3	84
		通信模式 No.8	14+ （1.0× 占用 CH 数）*2

*1. 括号中的数值为使用扩展分配存储区域时的 I/O 刷新时间。
*2. 占用 CH 数为所有从站占用的 I/O 区域的 CH 数。

● CPU 总线单元

使用 CPU 总线单元时的 I/O 刷新时间如下所示。无论是在 CPU 装置中使用还是在扩展装置中使用，均为相同的值。

单元名称	型号	每 1 个单元的 I/O 刷新时间 [μs]
串行通讯单元	CJ1W-SCU42 CJ1W-SCU32 CJ1W-SCU22	系数*1 ×25*2
DeviceNet 单元	CJ1W-DRM21	系数 *1× （占用 CH 数*3 ＋ 25）
EtherNet/IP 单元	CJ1W-EIP21	7.0 ＋ 0.8× 标签数*4

*1. 根据 CPU 单元的型号，使用以下系数。
NJ501- □□□□ : 0.1
NJ301- □□□□ : 0.2
NJ101- □□□□ : 0.3
*2. 执行协议宏时，最大将加上以下值。
系数 × 刷新 CH 数 [μs]
*3. 占用 CH 数为所有从站占用的 I/O 区域的 CH 数。
*4. 使用单元版本 1.03 及以上的 CPU 单元时， EtherNet/IP 单元的 I/O 刷新时间不会相加。

用户程序执行时间

用户程序执行时间取决于指令种类 × 指令数。

作为参考，按以下 3 大分类来检测和预计指令数。

- 标准指令
- 双精度实数 （LREAL）型的四则运算指令
- 双精度实数 （LREAL）型的三角函数指令

此外，梯形图程序和 ST 语言程序中使用的指令不同。各指令的构成请参考 「“ 梯形图语言的标准指令 ” 的指令种类构成」（P.A-30）及 「“ST 语言的标准指令 ” 的指令种类构成」（P.A-32）。

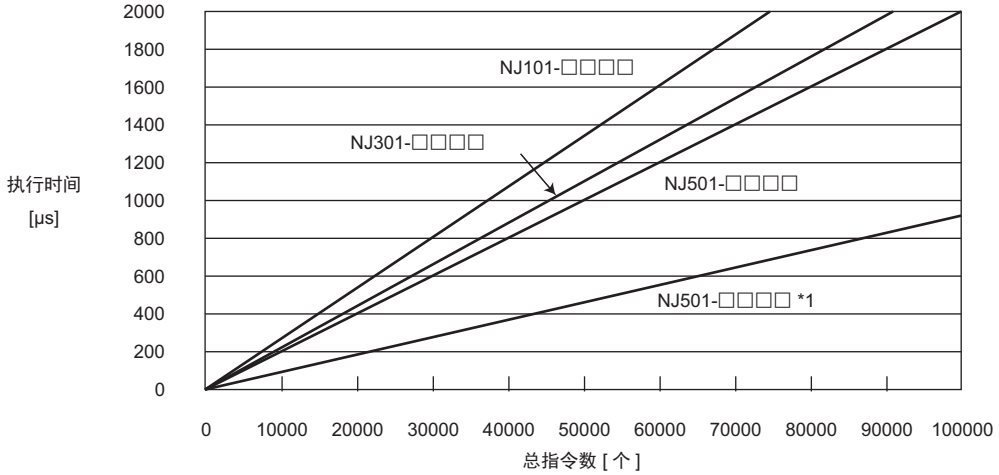
● 简单估算

根据各分类的指令数，按以下图表，读取以下各指令的执行时间并合计。

- 标准指令的执行时间
- 双精度实数 （LREAL）型的四则运算指令执行时间
- 双精度实数 （LREAL）型的三角函数指令执行时间

由此可估算出用户程序的执行时间。

梯形图语言的标准指令执行时间



*1 硬件修订版本为 “A” 的单元时。其他表示硬件修订版本为 “ 无 ” 的单元时。

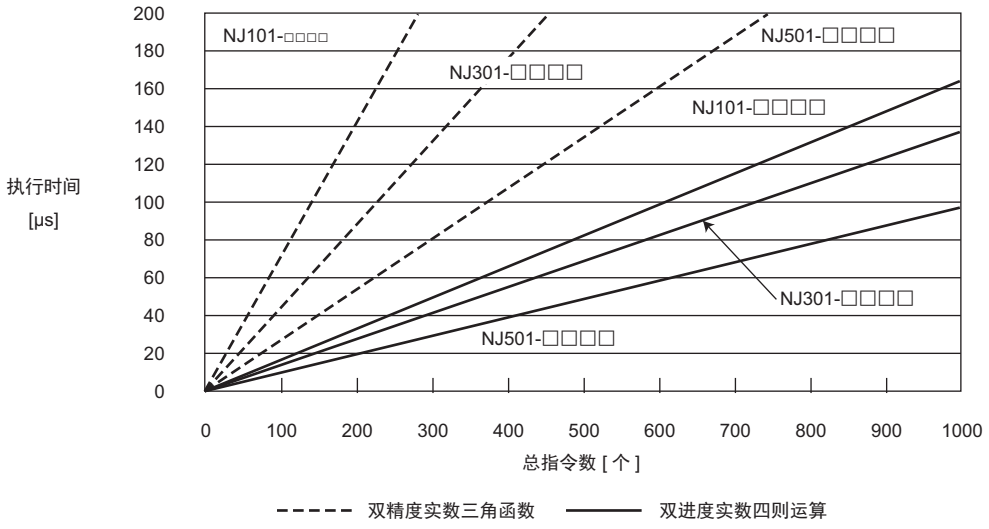
- “ 梯形图语言的标准指令 ” 的指令种类构成

本构成的指令执行率为 20%。

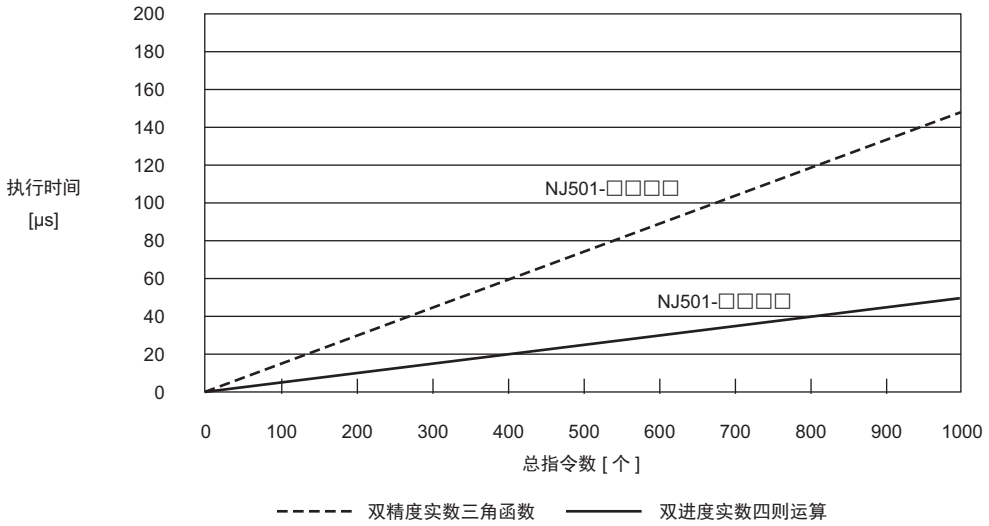
指令种类	指令名称	指令个数的比率 [%]	各指令种类的执行时间比率 [%]
梯形图指令	LD, AND, OUT, SET, RESET	81.0	40.2
比较指令	EQ, LT	4.1	8.3
计时指令、计数指令	Timer, TON/TOF, CTU/CTD	1.6	7.3
算术指令	+, -, *, /, ADD, SUB, MUL, DIV	2.4	6.5
BCD 转换指令、数据转换指令	INT_TO_DINT, WORD_BCD_TO_UINT	0.2	1.2
位列运算指令	AND, OR	6.2	13.0
数据传送指令	MOVE	4.6	23.5
总计		100.0	100.0

梯形图语言双精度实数 （LREAL）型的四则运算指令和三角函数指令执行时间

- 硬件修订版本为 “ 无 ” 的单元时



- 硬件修订版本为 “A” 的单元时



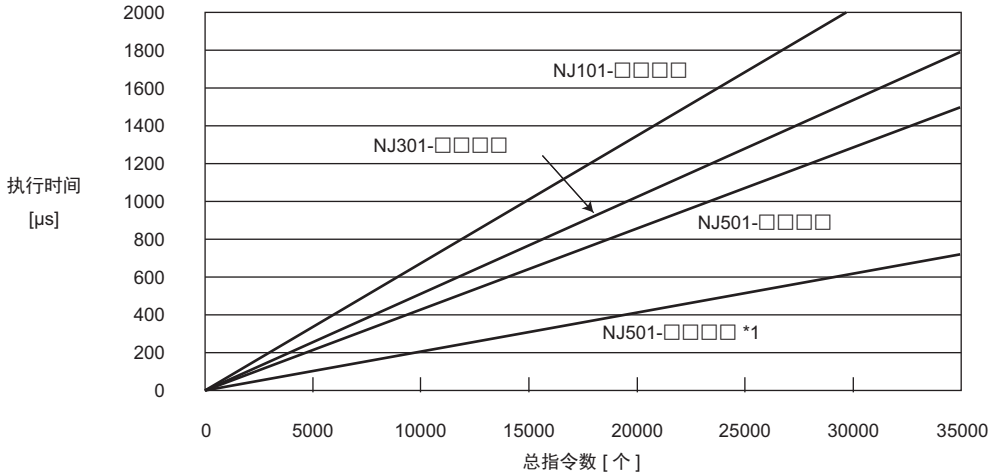
- “ 双进度实数的四则运算 ” 构成

指令名称	指令个数的比率 [%]
双精度实数的加法	20.0
双精度实数的减法	20.0
双精度实数的乘法	30.0
双精度实数的除法	30.0
总计	100.0

- “ 双精度实数的三角函数 ” 构成

指令名称	指令个数的比率 [%]
双精度实数的 Sin	16.7
双精度实数的 Cos	16.7
双精度实数的 Tan	16.7
双精度实数的 Sin^{-1}	16.7
双精度实数的 Cos^{-1}	16.7
双精度实数的 Tan^{-1}	16.7
总计	100.0

ST 语言的标准指令执行时间



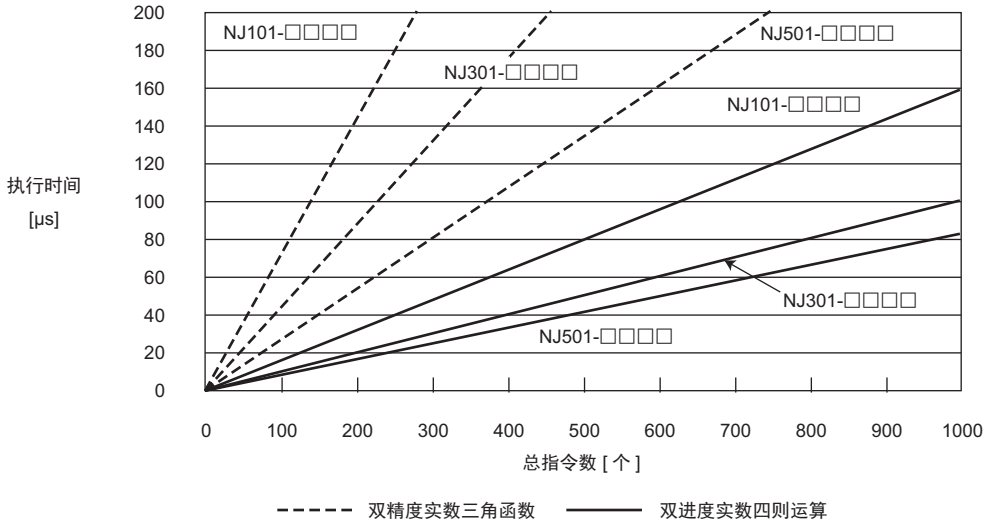
*1 硬件修订版本为 “A” 的单元时。其他表示硬件修订版本为 “ 无 ” 的单元时。

- “ST 语言的标准指令 ” 的指令种类构成

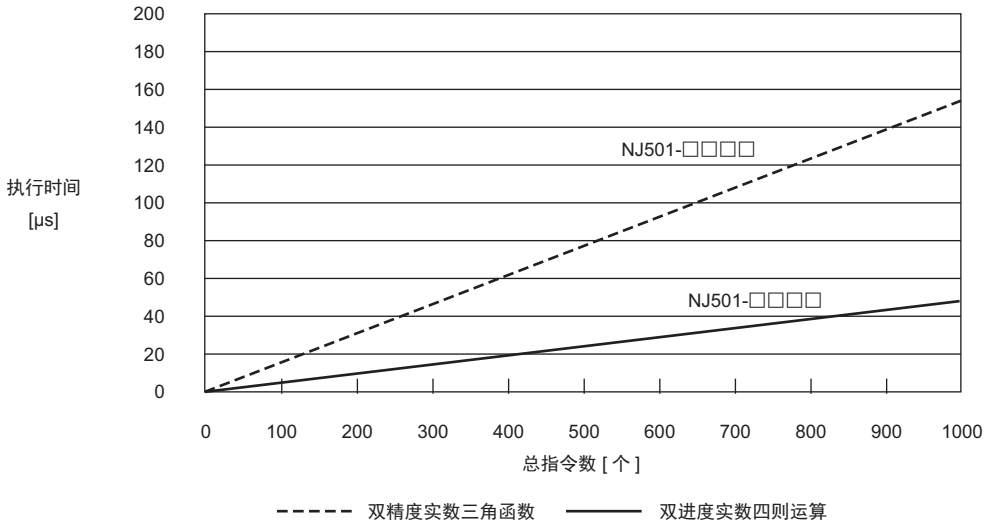
指令种类	指令名称	指令个数的比率 [%]	各指令种类的执行时间比率 [%]
ST 语法指令	IF ELSIF END_IF	75.4	41.6
比较指令	EQ, LT	5.2	8.7
计时指令、计数指令	Timer, TON/TOF, CTU/CTD	2.1	18.8
算术指令	+, -, *, /	3.1	10.2
BCD 转换指令、数据转换指令	INT_TO_DINT, WORD_BCD_TO_UINT	0.2	1.6
位列运算指令	AND, OR	8.0	11.7
数据传送指令	:=	5.9	7.3
总计		100.0	100.0

ST 语言双精度实数 （LREAL）型的四则运算指令和三角函数指令执行时间

- 硬件修订版本为 “ 无 ” 的单元时



- 硬件修订版本为 “A” 的单元时



- “ 双进度实数的四则运算 ” 构成

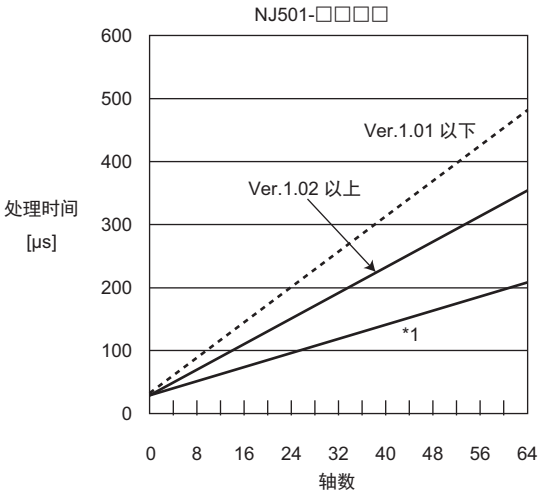
指令名称	指令个数的比率 [%]
双精度实数的加法	20.0
双精度实数的减法	20.0
双精度实数的乘法	30.0
双精度实数的除法	30.0
总计	100.0

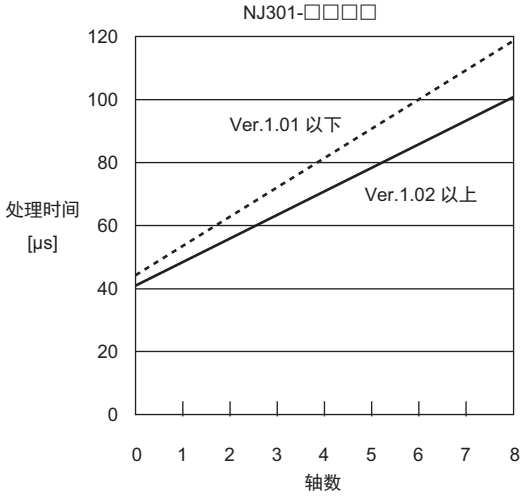
- “ 双精度实数的三角函数 ” 构成

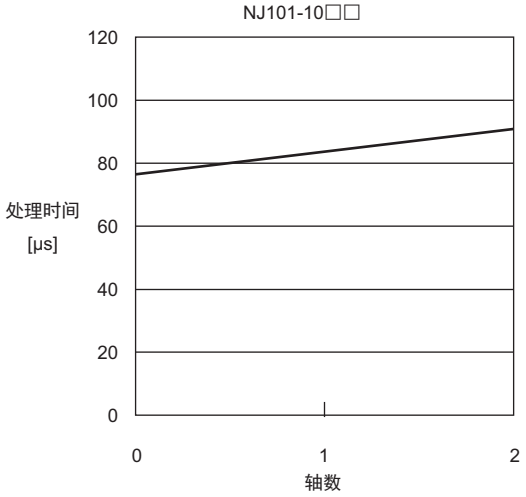
指令名称	指令个数的比率 [%]
双精度实数的 Sin	16.7
双精度实数的 Cos	16.7
双精度实数的 Tan	16.7
双精度实数的 Sin^{-1}	16.7
双精度实数的 Cos^{-1}	16.7
双精度实数的 Tan^{-1}	16.7
总计	100.0

运动控制处理时间

运动控制处理时间取决于所用伺服的轴数 （包括虚拟轴）。
根据使用的伺服轴数，按以下图表，读取运动控制处理时间。







*1 硬件修订版本为 “A” 的单元时。其他表示硬件修订版本为 “ 无 ” 的单元时。

（注）NJ101-90 □□中，不可使用运动控制功能。

共通处理时间

共通处理时间是指系统额外时间、系统共通处理 1 及系统共通处理 2 的总和，值如下。共通处理时间因任务类型不同而异。

任务类型	共通处理时间 [μs] （参考值）		
	NJ501- □□□□	NJ301- □□□□	NJ101- □□□□
主固定周期任务	150*1	240*2	300
固定周期任务	10	32	82

*1. 单元版本 1.01 以下的 CPU 单元中为 265μs。
*2. 单元版本 1.01 以下的 CPU 单元中为 360μs。

A-3-2 任务实际处理时间平均值的计算和任务周期的设定示例

任务实际处理时间平均值的计算示例

计算使用硬件修订版本为 " 无 "、单元版本为 1.02 的 NJ501- □□□□、以下条件下的任务实际处理时间的平均值。
任务类型为主固定周期任务。

项目		条件
使用的从站 / 单元	EtherCAT 从站	• 输入从站　GX-ID1611(Ver.1.1)：1 台 • 输出从站　GX-OD1611(Ver.1.1)：1 台 • 伺服电机　R88D- □□ (Ver.2.1)：4 台
	CJ 单元 （CPU 装置内）	• DC 输入单元　CJ1W-ID211：1 台 • 晶体管输出单元 CJ1W-OD211：1 台 • 模拟输入单元　CJ1W-AD042：1 台 • 模拟输出单元　CJ1W-DA021：1 台 • 串行通讯单元 CJ1W-SCU42：1 台 （不使用协议宏）
用户程序：	使用语言	梯形图语言
	标准指令构成	指令数：5,000 个
	双进度实数的四则运算	指令数：200 个
	双精度实数的三角函数	指令数：100 个
运动控制处理	轴数	4 轴

（注）内置 EtherCAT 端口上连接的电缆总长度为 10[m]。

● **I/O 刷新时间**

I/O 刷新额外时间：

由于 EtherCAT 从站与 CJ 单元连接，因此 I/O 刷新额外时间为 65μs。

EtherCAT 从站处理时间：

使用 CPU 单元 NJ501- □□□□时，每 1 字节的输出处理时间 Tout、每 1 字节的输入处理时间 Tin、每 1 字节的刷新处理时间 Tref 的值如下。

每 1 字节的 输出处理时间 Tout [μs]	每 1 字节的 输入处理时间 Tin[μs]	每 1 字节的 刷新处理时间 Tref[μs]	EtherCAT 通信调整时间 Tec[μs]
0.004	0.011	0.082	55

输入从站 GX-ID1611(Ver.1.1)、输出从站 GX-OD1611(Ver.1.1)、伺服电机　R88D- □□ (Ver.2.1) 的输出过程数据大小 Dout、输入过程数据大小 Din、输出数据大小和输入数据大小中的较大者 Dinout 的值如下所示。

EtherCAT 从站	输出过程数据 大小 Dout[字节]	输入过程数据 大小 Din[字节]	输出数据大小和 输入数据大小的 较大者 Dinout
GX-ID1611(Ver.1.1)	0	3	3
GX-OD1611(Ver.1.1)	2	1	2
R88D- □□ (Ver.2.1)	23	26	26

Dout、 Din 及 Dinout 各自的总计值如下。

Dout=2 + 23×4=94[字节]

Din=3 + 1 + 26×4=108[字节]

Dinout=3 + 2 + 26×4=109[字节]

根据上述值，按以下公式计算 I/O 刷新时间。

I/O 刷新处理时间

=EtherCAT 从站处理时间

=Tout×Dout + Tin×Din + Tref×Dinout + (1.24×Snum + 0.01×Clen － Tec)

=0.004×94 + 0.011×108 + 0.082×109 + (1.24×6 + 0.01×10) － 55

=0.376 + 1.188 + 8.938 + (7.44 + 0.1) － 55

=-36.958[μs]

由于上述表达式的计算结果为负数，因此 EtherCAT 从站处理时间视为 0μs。

CJ 单元处理时间：

CPU 装置中，每个 DC 输入单元 CJ1W-ID211、晶体管输出单元 CJ1W-OD211、模拟输入单元 CJ1W-AD042、模拟输出单元 CJ1W-DA021、串行通讯单元 CJ1W-SCU42 的 I/O 刷新时间如下所示。

型号	每个单元的 I/O 刷新时间 [μs]
CJ1W-ID211	1
CJ1W-OD211	1
CJ1W-AD042	24
CJ1W-DA021	24
CJ1W-SCU42	2.5

由于连接台数全部为 1，因此 CJ 单元处理时间按以下公式计算。

CJ 单元处理时间 =Σ （各 CJ 单元的 I/O 刷新时间 × 连接台数） - 110

=1×1 + 1×1 + 24×1 + 24×1 + 2.5×1 - 110

= -57.5[μs]

由于上述表达式的计算结果为负数，因此 CJ 单元处理时间视为 0μs。

至此，我们已经计算得到 I/O 刷新额外时间、EtherCAT 从站处理时间、CJ 单元处理时间的值，分别如下。

项目	值 [μs]
I/O 刷新额外时间	65
EtherCAT 从站处理时间	0
CJ 单元处理时间	0

根据上述值，按以下公式计算 I/O 刷新时间。

I/O 刷新时间 =I/O 刷新额外时间 +

(EtherCAT 从站处理时间和 CJ 单元处理时间的较大者）

=65 + 0

=65[μs]

● 用户程序执行时间

从各图表中读取以下数值。

- 标准指令构成
  根据梯形图语言的标准指令执行时间图表，NJ501- □□□□中指令数为 5000 个的用户程序执行时间为 100μs。
- 双进度实数的四则运算
  根据梯形图语言的双精度实数(LREAL)型的四则运算指令和三角函数指令的执行时间图表，NJ501-□□□□中指令数为 200 个的用户程序执行时间为 20μs。
- 双精度实数的三角函数
  根据梯形图语言的双精度实数(LREAL)型的四则运算指令和三角函数指令的执行时间图表，NJ501-□□□□中指令数为 100 个的用户程序执行时间为 27μs。

因此，用户程序执行时间为以上的合计值，结果如下。

用户程序执行时间 =100 + 20 + 27
=147[μs]

● 运动控制处理时间

根据运动控制处理的执行时间图表， NJ501- □□□□ Ver.1.02 的 4 轴运动控制处理的执行时间读取为 46μs。

● 共通处理时间

任务类型为主固定周期任务，因此 NJ501- □□□□的共通处理时间为 150μs。

因此，任务实际处理时间的平均值按以下公式计算。

任务实际处理时间的平均值 =I/O 刷新时间 + 用户程序执行时间
+ 运动控制处理 + 共通处理时间
=65 + 147 + 46 + 150
=408[μs]

任务周期的设定示例

根据上面计算得到的任务实际处理时间的平均值，设定任务周期。任务类型为主固定周期任务。
任务周期的值应大于计算得到的任务实际处理时间的平均值。此时，请预留出余量，将任务周期的值设定为大于任务实际处理时间平均值的 1.1 倍。

任务周期≧任务实际处理时间的平均值 ×1.1

由于上面计算得到的任务实际处理时间的平均值为 408μs，因为要设定为大于其 1.1 倍的值，所以任务周期为
500μs。

在实际的控制器上，因用户程序中搭载的逻辑是否动作、是否发生通信指令、数据链接动作、是否使用数据追踪等各种功能不同，任务执行时间会发生变动。此外，如果是固定周期任务，因是否有执行优先度更高的任务动作，任务执行时间会发生变动。
请务必使用实机的控制器，根据任务执行时间监视器，进行任务执行时间的验证。

A-4 系统定义变量一览

系统中分配了特定功能的变量。
已事先登录到全局变量表或各 POU 的本地变量表中。
变量名称固定。由以 “_(下划线)” 开始的部分和以 “P_” 开始的部分组成。

系统定义变量有只读和可读写两类。

可通过用户程序、外部通信或 Sysmac Studio、 NS/NA 系列显示器读写。

系统定义变量基本上按功能模块分类。
变量名称以以下类别名称开头。

功能模块	类别名称
NJ/NX 系列控制器的系统整体	_ （无）
PLC 功能模块	_PLC
	_CJB
运动控制功能模块	_MC、 _MC1、 _MC2
EtherCAT 主机功能模块	_EC
EtherNet/IP 功能模块	_EIP、 _EIP1、 _EIP2

一览表的阅读方法如下所示。

变量名称	名称	功能	数据类型	值的范围	参照页
[系统定义变量名称。开头带类别名称]	[名称]	[介绍功能]	[表示变量的数据类型]	[表示变量值的范围]	[表示个别规格表的页面]

变量名称栏的 () 中表示追加了该系统定义变量的 CPU 单元的单元版本。

使用注意事项

因控制器的系列不同，可能有不支持的或排列数等规格不同的系统定义变量。详情请参考「A-5 系统定义变量的个别规格」（P.A-79）中的个别规格。

A-4-1 NJ/NX 系列控制器的系统整体 （类别名称：无）

● 功能分类：时钟相关

变量名称	名称	功能	数据类型	值的范围	参照页
_CurrentTime	系统时刻	保存 CPU 单元的内部时钟数据。	DATE_AND_TIME	• NX 系列 CPU 单元时 DT#1970-01-01-00:00:00 ～ DT#2069-12-31-23:59:59 • NJ 系列 CPU 单元时 DT#1970-01-01-00:00:00 ～ DT#2106-02-06-23:59:59	A-80

● 功能分类：任务相关

变量名称	名称	功能	数据类型	值的范围	参照页
_< 任务名称 >_Active	任务执行中标志	正在执行任务时为 TRUE。 任务非执行中时为 FALSE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。	BOOL	TRUE、FALSE	A-80
_< 任务名称 >_LastExecTime	上次任务执行时间	表示上次执行任务的时间 （以 0.1μs 为单位）。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。	TIME	遵照数据类型	A-80
_< 任务名称 >_MaxExecTime	任务执行时间最大值	表示任务执行时间的最大值 （以 0.1μs 为单位）。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。	TIME	遵照数据类型	A-80
_< 任务名称 >_MinExecTime	任务执行时间最小值	表示任务执行时间的最小值 （以 0.1μs 为单位）。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。	TIME	遵照数据类型	A-80
_< 任务名称 >_ExecCount	任务执行次数	表示任务的执行次数。 若超出 4294967295 次，将恢复为 0，然后继续计数。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。	UDINT	遵照数据类型	A-81
_< 任务名称 >_Exceeded	任务周期超出标志	超出任务周期后，变为 TRUE。 若在任务周期内完成任务，则变为 FALSE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。	BOOL	TRUE、FALSE	A-81
_< 任务名称 >_ExceedCount	任务周期超出次数	保存超出周期的次数。 若当前值超出数据类型的最大值，当前值将恢复为 0，然后继续计数。 若超出 4294967295 次，将恢复为 0，然后继续计数。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。	UDINT	遵照数据类型	A-81

● 功能分类：异常相关

变量名称	名称	功能	数据类型	值的范围	参照页
_ErrSta	控制器异常 状态	正在发生控制器异常时为 TRUE。 未发生控制器异常时为 FALSE。 （注）出于实时性及与表示各功能模块异常原因的状态之间的同时性考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#C0F0	A-82
_AlarmFlag	用户 异常状态	发生用户异常时，与事件重要程度对应的位将变为 TRUE。 User fault Level 1 ～ 8 对应位 00 ～ 07。 未发生用户异常时，为 16 进制的 0000。	WORD	16#0000 ～ 16#00FF	A-82

● 功能分类：SD 存储卡相关

变量名称	名称	功能	数据类型	值的范围	参照页
_Card1Ready	SD 存储卡 可使用标志	识别到 SD 存储卡时，变为 TRUE。 未识别到时，为 FALSE。 TRUE：可使用 FALSE：不可使用	BOOL	TRUE、 FALSE	A-82
_Card1Protect	SD 存储卡 写保护 标志	SD 存储卡的 LOCK 开关为写保护时，变为 TRUE。 TRUE：有写保护 FALSE：无写保护	BOOL	TRUE、 FALSE	A-82
_Card1Err	SD 存储卡 错误标志	安装了不可使用的 SD 存储卡或格式异常时，变为 TRUE。 TRUE：有错误 FALSE：无错误	BOOL	TRUE、 FALSE	A-83
_Card1Access	SD 存储卡 访问中标志	正在访问 SD 存储卡时，变为 TRUE。 TRUE：访问中 FALSE：非访问中 本标志以 100ms 为周期由系统更新。因此， SD 存储卡的访问状态最多会延迟 100ms 反映到本标志中。所以，不建议在用户程序上使用。	BOOL	TRUE、 FALSE	A-83
_Card1Deteriorated	SD 存储卡 寿命警告标志	检测到 SD 存储卡达到使用寿命时，变为 TRUE。 TRUE：检测到使用寿命 FALSE：未检测到使用寿命	BOOL	TRUE、 FALSE	A-83
_Card1PowerFail	SD 存储卡 访问中断电 标志	正在访问 SD 存储卡的过程中，若 CPU 单元发生断电，则变为 TRUE。 TRUE：正在访问 SD 存储卡时发生断电 FALSE：正常	BOOL	TRUE、 FALSE	A-83

变量名称 成员名称	名称	功能	数据类型	值的范围	参照页
_Card1BkupCmd （Ver.1.03)	SD 存储卡 备份指示		_sBKUP_ CMD		A-83
ExecBkup	备份启动标志	向 SD 存储卡备份控制器的数据时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、 FALSE	A-83
CancelBkup	备份取消标志	停止向 SD 存储卡备份时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、 FALSE	A-84
ExecVefy	核对启动标志	核对控制器和 SD 存储卡中的备份文件时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、 FALSE	A-84
CancelVefy	核对取消标志	停止核对控制器和 SD 存储卡中的备份文件时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、 FALSE	A-84
DirName	目录名	指定备份对象的 SD 存储卡目录名称。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	STRING(64)	遵照数据类型	A-84
_Card1BkupSta (Ver.1.03)	SD 存储卡 备份 状态		_sBKUP_ STA		A-84
Done	完成标志	备份执行完成时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、 FALSE	A-84
Active	执行中标志	正在执行备份时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、 FALSE	A-85
Err	错误标志	备份处理异常结束时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、 FALSE	A-85

变量名称 成员名称	名称	功能	数据类型	值的范围	参照页
_Card1VefySta (Ver.1.03)	SD 存储卡 核对状态		_sVEFY_STA		A-85
Done	完成标志	核对执行完成时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、FALSE	A-85
Active	执行中标志	正在执行核对时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、FALSE	A-85
VefyRslt	核对结果标志	核对一致时为 TRUE，核对不一致时为 FALSE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、FALSE	A-85
Err	错误标志	核对处理异常结束时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。	BOOL	TRUE、FALSE	A-85
_Card1PrgTransferCmd(Ver.1.11)	SD 存储卡程序传送指示		_sPRGTRANSFER_CMD		A-86
Exec	程序传送启动标志	利用 SD 存储卡程序传送功能，向控制器传送 SD 存储卡中的备份文件数据时，变为 TRUE。	BOOL	TRUE、FALSE	A-86
DirName	目录名	指定保存有程序传送对象备份文件的 SD 存储卡目录名称。	STRING(64)	遵照数据类型	A-86
Password	密码	指定开始传送程序时要核对的密码。 每次开始 SD 存储卡程序传送时，密码将初始化。	STRING(33)	遵照数据类型	A-86
TargetUserProgram	用户程序、设定的传送标志	程序传送时的传送对象为用户程序或设定时，变为 TRUE。 在 SD 存储卡程序传送功能中，请务必设定为 TRUE。	BOOL	TRUE、FALSE	A-87
TargetIPAdr	IP 地址的传送标志	程序传送时的传送对象为内置 EtherNet/IP 端口的 IP 地址时，变为 TRUE。 IP 地址中包括设定种类、 IP 地址、子网掩码、默认网关。	BOOL	TRUE、FALSE	A-87
TargetVariable	变量当前值 （仅保持属性）的传送标志	程序传送时的传送对象为变量的当前值（仅保持属性）时，变为 TRUE。	BOOL	TRUE、FALSE	A-87
TargetMemory	CJ 单元用存储器当前值 （仅保持存储器）的传送标志	程序传送时的传送对象为 CJ 单元用存储器的当前值 （仅保持存储器）时，变为 TRUE。	BOOL	TRUE、FALSE	A-87
_Card1PrgTransferSta(Ver.1.11)	SD 存储卡程序传送状态		_sPRGTRANSFER_STA		A-88

变量名称 成员名称	名称	功能	数据类型	值的范围	参照页
Done	完成标志	程序传送执行完成时为 TRUE。	BOOL	TRUE、FALSE	A-88
Active	执行中标志	正在执行程序传送时为 TRUE。	BOOL	TRUE、FALSE	A-88
Err	错误标志	程序传送异常结束时为 TRUE。	BOOL	TRUE、FALSE	A-88

● 功能分类：备份相关

变量名称	名称	功能	数据类型	值的范围	参照页
_BackupBusy (Ver.1.03)	备份相关功能执行中标志	正在执行备份、恢复、核对时为 TRUE。	BOOL	TRUE、FALSE	A-88

● 功能分类：电源管理相关

变量名称	名称	功能	数据类型	值的范围	参照页
_PowerOnHour	通电时间	表示通电时间。 以 1 小时为单位保存 CPU 单元的通电时间。 重置时，请保存 0。 变为 4294967295 后，不会继续更新。 电源 ON 时不会初始化。	UDINT	0 ～ 4294967295	A-89
_PowerOnCount	断电发生次数	表示发生断电的次数。 从第一次打开 CPU 单元的电源后，每次发生断电时累加 （+1）。 重置时，请保存 0。 变为 4294967295 后，不会继续更新。 电源 ON 时不会初始化。	UDINT	0 ～ 4294967295	A-89
_RetainFail	断电保持失败标志	以下情况下，将变为 TRUE （断电保持失败）。 • 接通电源时，电池备份存储器检查错误时 以下情况下，将变为 FALSE （无断电保持失败）。 • 接通电源时，电池备份存储器检查 OK 时 • 下载了用户程序时 • 执行了存储器全部清除时 （注）无法保持绝对编码器原点位置偏置数据时，将反映到轴变量的异常状态中，而不是本标志。	BOOL	TRUE、FALSE	A-89

● 功能分类：编程相关

变量名称	名称	功能	数据类型	值的范围	参照页
P_On	始终 ON 标志	表示始终为 ON 的标志。	BOOL	TRUE	A-89
P_Off	始终 OFF 标志	表示始终为 OFF 的标志。	BOOL	FALSE	A-90
P_CY	进位标志	在部分指令中更新的标志。	BOOL	TRUE、FALSE	A-90
P_First_RunMode	运行开始时 1 个周期 ON 标志	控制器的动作模式从程序模式变更为运行模式时，若该程序正在执行，则在 1 个任务周期内为 TRUE。 若该程序不是正在执行，则保持 FALSE。 用于在 CPU 单元开始运行时进行初始处理等情况。 （注）本系统定义变量在函数中不可使用。	BOOL	TRUE、FALSE	A-90
P_First_Run (Ver.1.08)	程序启动时 1 个周期 ON 标志	该程序开始执行后，仅在 1 个任务周期内为 TRUE。 用于每次开始执行程序时需要进行初始处理的情况。 （注）本系统定义变量在函数中不可使用。	BOOL	TRUE、FALSE	A-90
P_PRGER	指令错误发生标志	相应的程序或通过该程序调用的 FB 或 FUN 发生指令错误时，持续变为 TRUE 的标志。一旦变为 TRUE，在通过用户程序变为 FALSE 前，将保持 TRUE。	BOOL	TRUE、FALSE	A-90

● 功能分类：通信相关

变量名称	名称	功能	数据类型	值的范围	参照页
_Port_numUsingPort	使用端口数量	表示当前正在使用的内部逻辑端口数量。 用于用户程序的调试等。	USINT	0 ～ 32	A-91
_Port_isAvailable	网络通信指令可执行标志	表示是否有可使用的内部逻辑端口。 有可使用的内部逻辑端口时为 TRUE，没有时为 FALSE。	BOOL	TRUE、FALSE	A-91
_FINSTCPConnSta	FINS/TCP 连接状态	表示 FINS/TCP 连接的状态。	WORD	16#0000 ～ 16#FFFF	A-91

● 功能分类：版本管理相关

变量名称	名称	功能	数据类型	值的范围	参照页
_UnitVersion (Ver.1.08)	单元版本	保存 CPU 单元的单元版本。 在要素编号 0 中保存单元版本的整数部分。 在要素编号 1 中保存单元版本的小数部分。 例 1) 单元版本为 1.08 时，要素编号 0 中保存 1、要素编号 1 中保存 8。 例 2) 单元版本为 1.10 时，要素编号 0 中保存 1、要素编号 1 中保存 10。	ARRAY[0..1] OF USINT	0 ～ 99	A-91
_HardwareRevision (Ver.1.11)	硬件修订版本	保存 CPU 单元的硬件修订版本。硬件修订版本为 “ 无 ” 时，保存 “'-'”，其他情况下保存 “'A' ～ 'Z'”。	STRING[2]	'-'、 'A' ～ 'Z'	A-91

● 功能分类：自诊断相关

变量名称	名称	功能	数据类型	值的范围	参照页
_SelfTest_HighTemperature (Ver.1.10)	CPU 单元内部温度超限标志	CPU 单元内部温度为高温时，变为 TRUE。	BOOL	TRUE、 FALSE	A-92
_SelfTest_LowBattery (Ver.1.10)	电池电压低标志	电池拆下或电池电压低时变为 TRUE。	BOOL	TRUE、 FALSE	A-92
_SelfTest_LowFanRevolution (Ver.1.10)	风扇转速降低标志	风扇拆下或风扇的转速下降时变为 TRUE。 （注）NJ 系列 CPU 单元时，始终为 FALSE。	BOOL	TRUE、 FALSE	A-92

A-4-2 PLC 功能模块 （类别名称：_PLC）

● 功能分类：调试相关

变量名称 成员名称	名称	功能	数据类型	值的范围	参照页
_PLC_TraceSta[0..3]			_sTRACE_STA		A-93
.IsStart	追踪执行中 ON 标志	开始追踪时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 监视数据追踪状态时使用。	BOOL	TRUE、FALSE	A-93
.IsComplete	追踪完成时 ON 标志	追踪执行完成时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 监视数据追踪状态时使用。	BOOL	TRUE、FALSE	A-93
.IsTrigger	追踪触发监视标志	追踪触发条件成立时，变为 TRUE。 下次开始追踪时，变为 FALSE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 监视数据追踪状态时使用。	BOOL	TRUE、FALSE	A-93
.ParamErr	追踪参数异常标志	开始追踪时，若追踪设定异常，则变为 TRUE。 正常时，为 FALSE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 监视数据追踪状态时使用。	BOOL	TRUE、FALSE	A-93

● 功能分类：异常相关

变量名称	名称	功能	数据类型	值的范围	参照页
_PLC_ErrSta	PLC 功能模块 异常状态	发生与 PLC 功能模块相关的控制器异常时，变为 TRUE。 未发生与 PLC 功能模块相关的控制器异常时，变为 FALSE。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#00F0	A-94

A-4-3 PLC 功能模块（类别名称：_CJB）

● 功能分类：I/O 总线的状态相关

变量名称	名称	功能	数据类型	值的范围	参照页
_CJB_MaxRackNo	最大装置编号	表示控制器识别到的扩展装置的最大装置编号。 （注）本系统定义变量仅可在NJ系列CPU单元中使用。	UINT	0 ～ 3 为 0 时，表示仅 CPU 装置	A-95
_CJB_MaxSlotNo	最大插槽编号	表示控制器识别到的 CJ 单元各装置的（最大插槽编号 +1）。 （注）本系统定义变量仅可在NJ系列CPU单元中使用。	ARRAY [0..3] OF UINT	0 ～ 10 为 0 时，表示未安装 CJ 单元	A-95

● 功能分类：I/O 总线的异常相关

变量名称	名称	功能	数据类型	值的范围	参照页
_CJB_ErrSta	I/O 总线异常状态	I/O 总线的异常状态。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。 （注）本系统定义变量仅可在NJ系列CPU单元中使用。	WORD	16#0000 ～ 16#C0F0	A-95
_CJB_MstrErrSta	I/O 总线主机异常状态	I/O 总线主机的异常状态。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。 （注）本系统定义变量仅可在NJ系列CPU单元中使用。	WORD	16#0000 ～ 16#00F0	A-95
_CJB_UnitErrSta	I/O 总线单元异常状态	I/O 总线单元的异常状态。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。 （注）本系统定义变量仅可在NJ系列CPU单元中使用。	ARRAY [0..3, 0..9] OF WORD	16#0000 ～ 16#80F0	A-96
_CJB_InRespTm	基本输入单元的输入时间常数	表示基本输入单元的输入时间常数。 （注）本系统定义变量仅可在NJ系列CPU单元中使用。	ARRAY [0..3, 0..9] OF UINT	0 ～ 320	A-96

● 功能分类：CJ 单元用存储器的特殊辅助继电器相关

变量名称	名称	功能	数据类型	值的范围	参照页
_CJB_IOUnitInfo	基本 I/O 单元信息区域	显示基本 I/O 单元 （带负载短路保护功能）的警报输出。 TRUE：有负载短路 FALSE：无负载短路 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。	ARRAY [0..3, 0..9, 0..7] OF BOOL	TRUE、FALSE	A-96
_CJB_CBU00InitSta ・・・ _CJB_CBU15InitSta	CPU 总线单元 初始化中标志	CPU 总线单元正在初始化时，变为 TRUE。 初始化完成时，变为 FALSE。 各个位与单元编号对应。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。	BOOL	TRUE、FALSE	A-96
_CJB_SIO00InitSta ・・・ _CJB_SIO95InitSta	高功能 I/O 单元 初始化中标志	高功能 I/O 单元正在初始化时，变为 TRUE。 初始化完成时，变为 FALSE。 各个位与号机 No. 对应。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。	BOOL	TRUE、FALSE	A-97
_CJB_CBU00Restart ・・・ _CJB_CBU15Restart	CPU 总线单元 重启标志	上升时，重启 CPU 总线单元 （重启处理后，在系统中变为 FALSE）。 各个位与单元编号对应。 执行指令时，若重启标志变为 TRUE，将在执行下一任务周期的刷新处理时，开始重启处理。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。	BOOL	TRUE、FALSE	A-97
_CJB_SIO00Restart ・・・ _CJB_SIO95Restart	高功能 I/O 单元 重启标志	上升时，重启高功能 I/O 单元 （重启处理后，在系统中变为 FALSE）。 各个位与号机 No. 对应。 执行指令时，若重启标志变为 TRUE，将在执行下一任务周期的刷新处理时，开始重启处理。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。	BOOL	TRUE、FALSE	A-97
_CJB_SCU00P1ChgSta _CJB_SCU00P2ChgSta ・・・	串行通讯单元 0 号机端口 1/2 用串行端口设定变更中标志	正在变更该端口的设定时，变为 TRUE。 设定变更完成时，变为 FALSE。 通过执行指令或用户操作将该位变为 TRUE 后，可通知串行端口设定的变更。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。	BOOL	TRUE、FALSE	A-98
・・・ _CJB_SCU15P1ChgSta _CJB_SCU15P2ChgSta	串行通讯单元 1-15 号机端口 1/2 用串行端口设定变更中标志		BOOL	TRUE、FALSE	A-98

A-4-4 运动控制功能模块（类别名称：_MC）

● 功能分类：运动控制功能相关

变量名称	名称	功能	数据类型	值的范围	参照页
_MC_ErrSta	运动控制功能模块的异常状态	表示运动控制功能模块中检测到的异常状态。 可在用户程序中使用（直接参照）。 各个位的含义请参考后面介绍的“异常状态中各个位的含义”。	WORD	16#0000～16#40F0	A-99
_MC_ComErrSta	共通异常状态	表示在运动控制的共通处理中检测到的异常状态。 可在用户程序中使用（直接参照）。各个位的含义请参考后面介绍的“异常状态中各个位的含义”。	WORD	16#0000～16#00F0	A-99
_MC_AX_ErrSta	轴异常状态	表示以轴为单位检测到的异常状态。最多可显示 256 个轴[*1]。 可在用户程序中使用（直接参照）。各个位的含义请参考后面介绍的“异常状态中各个位的含义”。	ARRAY [0..255] OF WORD[*1]	16#0000～16#00F0	A-99
_MC_GRP_ErrSta	轴组异常状态	表示以轴组为单位检测到的异常状态。最多可显示 64 个轴组[*2]。 可在用户程序中使用（直接参照）。各个位的含义请参考后面介绍的“异常状态中各个位的含义”。	ARRAY [0..63] OF WORD[*2]	16#0000～16#00F0	A-99
_MC_COM	通用变量	进行运动控制功能模块共通的状态监视。 结构体的成员详情请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。	_sCOMMON_REF	—	A-100
_MC_GRP	轴组变量	NX 系列 CPU 单元时，进行运动控制 1 中使用的运动控制指令的轴组指定、多轴协调控制的状态监视、多轴协调控制的设定监视。 NJ 系列 CPU 单元时，进行运动控制指令的轴组指定、多轴协调控制的状态监视、多轴协调控制的设定监视。 若用 Sysmac Studio 创建“轴组”，将生成用户自定义名称的轴组变量。 通常使用自定义名称的轴组变量。 结构体的成员详情请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。	NX 系列：ARRAY[0..63] OF _sGROUP_REF NJ 系列：ARRAY[0..31] OF _sGROUP_REF	—	A-100

变量名称	名称	功能	数据类型	值的范围	参照页
_MC1_GRP	轴组变量	进行运动控制 1 中使用的运动控制指令的轴组指定、多轴协调控制的状态监视、多轴协调控制的设定监视。 若用 Sysmac Studio 创建 “ 轴组 ”，将生成用户自定义名称的轴组变量。 通常使用自定义名称的轴组变量。 结构体的成员详情请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。 （注）本系统定义变量仅可在NX系列CPU单元中使用。 若 “_MC1_GRP” 和 “_MC_GRP” 的排列要素编号相同，则参照相同的值。	ARRAY[0..63] OF _sGROUP_REF	—	A-100
_MC2_GRP	轴组变量	进行运动控制 2 中使用的运动控制指令的轴组指定、多轴协调控制的状态监视、多轴协调控制的设定监视。 若用 Sysmac Studio 创建 “ 轴组 ”，将生成用户自定义名称的轴组变量。 通常使用自定义名称的轴组变量。 结构体的成员详情请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY[0..63] OF _sGROUP_REF	—	A-100
_MC_AX	轴变量	NX 系列 CPU 单元时，进行运动控制 1 中使用的运动控制指令的轴指定、单轴控制的状态监视、单轴控制的设定监视。 NJ 系列 CPU 单元时，进行运动控制指令的轴指定、单轴控制的状态监视、单轴控制的设定监视。 若用 Sysmac Studio 创建 “ 轴 ”，将生成用户自定义名称的轴变量。 通常使用自定义名称的轴变量。 结构体的成员详情请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。	NX 系列： ARRAY[0..255] OF _sAXIS_REF NJ 系列： ARRAY[0..63] OF _sAXIS_REF	—	A-101
_MC1_AX	轴变量	进行运动控制 1 中使用的运动控制指令的轴指定、单轴控制的状态监视、单轴控制的设定监视。 若用 Sysmac Studio 创建 “ 轴 ”，将生成用户自定义名称的轴变量。 通常使用自定义名称的轴变量。 结构体的成员详情请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。 （注）本系统定义变量仅可在NX系列CPU单元中使用。若 “_MC1_AX” 和 “_MC_AX” 的排列要素编号相同，则参照相同的值。	ARRAY[0..255] OF _sAXIS_REF	—	A-101

变量名称	名称	功能	数据类型	值的范围	参照页
_MC2_AX	轴变量	进行运动控制 2 中使用的运动控制指令的轴指定、单轴控制的状态监视、单轴控制的设定监视。 若用 Sysmac Studio 创建 “ 轴 ”，将生成用户自定义名称的轴变量。 通常使用自定义名称的轴变量。 结构体的成员详情请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY[0..255] OF _sAXIS_REF	—	A-101

*1. NJ 系列 CPU 单元时，最多 64 个轴，数据类型为 ARRAY [0..63] OF WORD。
*2. NJ 系列 CPU 单元时，最多 32 个轴组，数据类型为 ARRAY [0..31] OF WORD。

A-4-5 EtherCAT 主机功能模块 （类别名称：_EC）

● 功能分类：EtherCAT 通信的异常相关

变量名称	名称	功能	数据类型	值的范围	参照页
_EC_ErrSta	内置 EtherCAT 异常	集合了 EtherCAT 主机功能模块异常的系统定义变量。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#40F0	A-102
_EC_PortErr	通信端口异常	集合了 EtherCAT 主机用通信端口异常的系统定义变量。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#00F0	A-102
_EC_MstrErr	主机异常	集合了 EtherCAT 主机异常和 EtherCAT 主机检测到的从站异常的系统定义变量。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#00F0	A-102
_EC_SlavErr	从站异常	集合了 EtherCAT 从站整体异常状态的系统定义变量。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#00F0	A-102
_EC_SlavErrTbl	从站异常表	以 EtherCAT 从站为单位通知异常状态。本异常状态表示实际系统构成中以从站为单位的异常状态，本变量的排列与 EtherCAT 从站的节点地址 （1 ～ 512）*1 对应，表示发生异常的从站。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	Array [1..512] OF WORD*1	16#0000 ～ 16#00F0	A-103
_EC_MacAdrErr	MAC 地址异常	MAC 地址不正确时，变为 TRUE。	BOOL	TRUE、FALSE	A-103
_EC_LanHwErr	通信控制器异常	检测到通信控制器的硬件异常时，变为 TRUE。	BOOL	TRUE、FALSE	A-103
_EC_LinkOffErr	链接断开异常	检测到通信控制器未建立链接时，变为 TRUE。	BOOL	TRUE、FALSE	A-103
_EC_NetCfgErr	网络配置信息异常	检测到网络配置信息不正确时，变为 TRUE。	BOOL	TRUE、FALSE	A-103
_EC_NetCfgCmpErr	网络配置核对异常	网络配置信息和网络实际配置不一致时，变为 TRUE。	BOOL	TRUE、FALSE	A-103
_EC_NetTopologyErr	网络配置异常	发生网络配置异常 （连接台数超限、环形连接）时，变为 TRUE。	BOOL	TRUE、FALSE	A-104
_EC_PDCommErr	过程数据通信异常	过程数据通信中，发生从站意外脱离、加入，或检测到从站 WDT 异常时，变为 TRUE。	BOOL	TRUE、FALSE	A-104
_EC_PDTimeoutErr	过程数据接收超时异常	若在接收过程数据时发生超时，则变为 TRUE。	BOOL	TRUE、FALSE	A-104
_EC_PDSendErr	过程数据发送异常	发生过程数据发送异常 （未在过程数据通信周期内发送、发送抖动超限）时，变为 TRUE。	BOOL	TRUE、FALSE	A-104
_EC_SlavAdrDupErr	从站节点地址重复异常	若从站的节点地址重复，则变为 TRUE。	BOOL	TRUE、FALSE	A-104
_EC_SlavInitErr	从站初始化异常	向从站发出初始化指令时若发生异常，则变为 TRUE。	BOOL	TRUE、FALSE	A-104
_EC_SlavAppErr	从站应用程序异常	检测到从站的应用程序状态寄存器异常时，变为 TRUE。	BOOL	TRUE、FALSE	A-105
_EC_MsgErr	EtherCAT 消息异常	向不支持消息功能的从站发送消息或向从站发送的消息返回的响应格式异常时，变为 TRUE。	BOOL	TRUE、FALSE	A-105

变量名称	名称	功能	数据类型	值的范围	参照页
_EC_SlavEmergErr	紧急 消息检测	主机检测到从站发出紧急消息时，变为 TRUE。	BOOL	TRUE、 FALSE	A-105
_EC_CommErrTbl	通信异常从站表	以从站节点地址顺序的表格表示。主机检测到异常的从站对应的位置变为 TRUE。	Array [1..512] OF BOOL*2	TRUE、 FALSE	A-105
_EC_CycleExceeded	EtherCAT 通信周期设定异常	启动时，若无法在设定的通信周期内进行通信，则变为 TRUE。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、 FALSE	A-105

*1. NJ 系列 CPU 单元时，节点地址为 1 ～ 192，数据类型为 Array [1..192] OF WORD。
*2. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

参考

内置 EtherCAT 异常标志的典型关系

变量名称	名称	变量名称	名称	变量名称	名称	事件重要程度
_EC_ErrSta	内置 EtherCAT 异常	_EC_PortErr	通信端口异常	_EC_MacAdrErr	MAC 地址异常	部分停止故障等级
				_EC_LanHwErr	通信控制器异常	
				_EC_LinkOffErr	链接断开异常	
		_EC_MstrErr	主机异常	_EC_NetCfgErr	网络 配置信息异常	轻度故障等级
				_EC_NetCfgCmpErr	网络 配置核对异常	
				_EC_NetTopologyErr	网络 配置异常	
				_EC_PDCommErr	过程数据通信异常	
				_EC_PDTimeoutErr	过程数据接收超时异常	
				_EC_PDSendErr	过程数据发送异常	
				_EC_SlavAdrDupErr	从站节点 节点地址重复	
				_EC_SlavInitErr	从站初始化异常	
				_EC_SlavAppErr	从站应用程序异常	
				_EC_CommErrTbl	通信异常从站表	
				_EC_CycleExceeded	EtherCAT 通信周期设定异常	
				_EC_MsgErr	EtherCAT 消息异常	监视信息
				_EC_SlavEmergErr	紧急消息检测	
		_EC_SlavErr	从站异常	_EC_SlavErrTbl	从站异常表	在从站上定义

（注）与 EtherCAT 通信异常相关的所有系统定义变量，在排除异常原因并解除发生的控制器异常 （Sysmac Studio 的故障排除 /ResetECError 指令）前，值不会变化。

● 功能分类：EtherCAT 通信的状态相关

变量名称	名称	功能	数据类型	值的范围	参照页
_EC_RegSlavTbl	登录从站表	网络配置信息中登录的从站一览。 以从站节点地址顺序的表格表示。 登录了从站时，对应的位置变为 TRUE。	Array [1..512] OF BOOL*1	TRUE、FALSE	A-106
_EC_EntrySlavTbl	网络加入从站表	通知网络加入从站一览。 以从站节点地址顺序的表格表示。 从站加入网络时，对应的位置变为 TRUE。	Array [1..512] OF BOOL*1	TRUE、FALSE	A-106
_EC_MBXSlavTbl	可进行信息通信的从站表	可进行信息通信的从站一览。以从站节点地址顺序的表格表示。 从站变为可进行信息通信的状态 （可预操作、可安全操作、可操作）时，对应的位置变为 TRUE。 （注）与 EtherCAT 从站进行信息通信时，请参照本数据，确认是否可与相应的从站进行信息通信。	Array [1..512] OF BOOL*1	TRUE、FALSE	A-106
_EC_PDSlavTbl	过程数据通信中从站表	正在进行过程数据通信的从站一览。以从站节点地址顺序的表格表示。 从站的 IN、OUT 均变为过程数据有效状态（可操作）时，对应的位置变为 TRUE。 （注）控制 EtherCAT 从站时，请参照本数据，确认对应从站数据的有效性。	Array [1..512] OF BOOL*1	TRUE、FALSE	A-106
_EC_DisconnSlavTbl	脱离指令中的从站表	以从站节点地址顺序的表格表示。 脱离指令中的从站对应的位置变为 TRUE。	Array [1..512] OF BOOL*1	TRUE、FALSE	A-107
_EC_DisableSlavTbl	无效设定从站表	以从站节点地址顺序的表格表示。 设定为无效的从站对应的位置变为 TRUE。	Array [1..512] OF BOOL*1	TRUE、FALSE	A-107
_EC_PDActive	过程数据通信状态	正在与所有从站 (*) 进行过程数据通信时，变为 TRUE。 * 不含设定为无效的从站。	BOOL	TRUE、FALSE	A-107
_EC_PktMonStop	信息包监视停止	信息包监视停止时变为 TRUE。	BOOL	TRUE、FALSE	A-107
_EC_LinkStatus	链接状态	通信控制器的链接状态为 Link ON 时，变为 TRUE。	BOOL	TRUE、FALSE	A-107
_EC_PktSaving	信息包数据文件保存中	表示是否正在执行信息包数据文件保存处理。 TRUE：正在执行信息包数据文件保存处理 FALSE：未执行信息包数据文件保存处理	BOOL	TRUE、FALSE	A-108
_EC_InDataInvalid	输入数据无效	主固定周期任务中执行的过程数据通信未正常进行，输入值不正确时，变为 TRUE。	BOOL	TRUE、FALSE	A-108
_EC_InData1Invalid	输入数据 1 无效	主固定周期任务中执行的过程数据通信未正常进行，输入值不正确时，变为 TRUE。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-108

变量名称	名称	功能	数据类型	值的范围	参照页
_EC_InData2Invalid	输入数据 2 无效	固定周期任务 （执行优先度 5）中执行的过程数据通信未正常进行，输入值不正确时，变为 TRUE。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-108

*1. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

（注）与 EtherCAT 通信的状态相关的所有系统定义变量均表示当前的状态。

● 功能分类：EtherCAT 通信的诊断统计信息相关

变量名称	名称	功能	数据类型	值的范围	参照页
_EC_StatisticsLogEnable (Ver.1.11)	诊断统计信息日志许可	开始诊断统计信息日志功能时，从 FALSE 变为 TRUE。 结束诊断统计信息日志功能时，从 TRUE 变为 FALSE。	BOOL	TRUE、FALSE	A-109
_EC_StatisticsLogCycleSec (Ver.1.11)	诊断统计信息日志保存周期	在诊断统计信息日志功能中，以秒为单位指定统计诊断信息的写入周期。 若指定为 “0”，则仅在结束诊断统计信息日志功能时写入 1 次。 （注）正在执行诊断统计信息日志功能时，即使变更本系统定义变量的值，正在执行的写入周期也不会发生变化。	UINT	0、30 ～ 1800	A-109
_EC_StatisticsLogBusy (Ver.1.11)	诊断统计信息日志执行状态标志	正在执行诊断统计信息日志功能时变为 TRUE。	BOOL	TRUE、FALSE	A-109
_EC_StatisticsLogErr (Ver.1.11)	诊断统计信息日志异常结束标志	开始诊断统计信息日志功能失败或执行过程中写入日志失败时，变为 TRUE。 开始诊断统计信息日志功能后，若 _EC_StatisticsLogBusy （诊断统计信息日志执行状态标志）变为 FALSE，则本标志的值确定。 异常结束可能有以下原因。 • SD 存储卡的容量达到上限，无法将记录添加到日志文件中。 • SD 存储卡有写保护。 • 未安装 SD 存储卡。 • 由于 _EC_StatisticsLogCycleSec （诊断统计信息日志保存周期）中指定的值不正确，因此无法开始。	BOOL	TRUE、FALSE	A-109

A-4-6 EtherNet/IP 功能模块 （类别名称：_EIP）

● 功能分类：EtherNet/IP 通信的异常相关

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_ErrSta	内置 EtherNet/IP 异常	内置 EtherNet/IP 端口的异常状态变量。 NX 系列 CPU 单元时，集合了以下异常标志。 • _EIP1_PortErr （通信端口 1 异常） • _EIP2_PortErr （通信端口 2 异常） • _EIP1_CipErr （CIP 通信 1 异常） • _EIP2_CipErr （CIP 通信 2 异常） • _EIP_TcpAppErr （TCP 应用通信异常） NJ 系列 CPU 单元时，集合了以下异常标志。 • _EIP_PortErr （通信端口异常） • _EIP_CipErr （CIP 通信异常） • _EIP_TcpAppErr （TCP 应用通信异常） （注）各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#00F0	A-110
_EIP_PortErr	通信端口异常	通信端口的异常状态变量。 NX 系列 CPU 单元时，集合了以下异常标志。 • _EIP1_MacAdrErr（通信端口 1 MAC 地址异常） • _EIP1_LanHwErr（通信端口 1 通信控制器异常） • _EIP1_EtnCfgErr（通信端口 1 Ethernet 基本设定异常） • _EIP1_IPAdrCfgErr （通信端口 1 IP 地址设定异常） • _EIP1_IPAdrDupErr （通信端口 1 IP 地址重复异常） • _EIP1_BootpErr （通信端口 1 BOOTP 服务器连接异常） • _EIP_DNSCfgErr （DNS 设定异常） • _EIP_DNSSrvErr （DNS 服务器连接异常） • _EIP_IPRTblErr （IP 路由表异常） NJ 系列 CPU 单元时，集合了以下异常标志。 • _EIP_MacAdrErr （MAC 地址异常） • _EIP_LanHwErr （通信控制器异常） • _EIP_EtnCfgErr （Ethernet 基本设定异常） • _EIP_IPAdrCfgErr （IP 地址设定异常） • _EIP_IPAdrDupErr （IP 地址重复异常） • _EIP_BootpErr（BOOTP 服务器连接异常） • _EIP_IPRTblErr （IP 路由表异常） （注）“ 链接断开检测 ” 或 “ 内置 EtherNet/IP 处理异常 ” 时，登录到事件日志中，本异常的位变为 ON。 各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#00F0	A-110

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP1_PortErr	通信端口 1 异常	通信端口的异常状态变量。 集合了以下异常标志。 • _EIP1_MacAdrErr（通信端口 1 MAC 地址异常） • _EIP1_LanHwErr（通信端口 1 通信控制器异常） • _EIP1_EtnCfgErr（通信端口 1 Ethernet 基本设定异常） • _EIP1_IPAdrCfgErr （通信端口 1 IP 地址设定异常） • _EIP1_IPAdrDupErr （通信端口 1 IP 地址重复异常） • _EIP1_BootpErr （通信端口 1 BOOTP 服务器连接异常） • _EIP_DNSCfgErr （DNS 设定异常） • _EIP_DNSSrvErr （DNS 服务器连接异常） • _EIP_IPRTblErr （IP 路由表异常） （注）“ 链接断开检测 ” 或 “ 内置 EtherNet/IP 处理异常 ” 时，登录到事件日志中，本异常的位变为 ON。各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	WORD	16#0000 ～ 16#00F0	A-111
_EIP2_PortErr	通信端口 2 异常	通信端口的异常状态变量。 集合了以下异常标志。 • _EIP2_MacAdrErr（通信端口 2 MAC 地址异常） • _EIP2_LanHwErr（通信端口 2 通信控制器异常） • _EIP2_EtnCfgErr （通信端口 2Ethernet 基本设定异常） • _EIP2_IPAdrCfgErr （通信端口 2 IP 地址设定异常） • _EIP2_IPAdrDupErr （通信端口 2 IP 地址重复异常） • _EIP2_BootpErr （通信端口 2 BOOTP 服务器连接异常） • _EIP_DNSCfgErr （DNS 设定异常） • _EIP_DNSSrvErr （DNS 服务器连接异常） • _EIP_IPRTblErr （IP 路由表异常） （注）“ 链接断开检测 ” 或 “ 内置 EtherNet/IP 处理异常 ” 时，登录到事件日志中，本异常的位变为 ON。各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	WORD	16#0000 ～ 16#00F0	A-111

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_CipErr	CIP 通信异常	CIP 通信的异常状态变量。 NX 系列 CPU 单元时，集合了以下异常标志。 • _EIP1_IdentityErr（CIP 通信 1Identity 信息不正确） • _EIP1_TDLinkCfgErr （CIP 通信 1 标签数据链接设定异常） • _EIP1_TDLinkOpnErr（CIP 通信 1 建立标签数据链接失败） • _EIP1_TDLinkErr（CIP 通信 1 标签数据链接通信异常） • _EIP1_TagAdrErr（CIP 通信 1 标签解决异常） • _EIP1_MultiSwONErr（CIP 通信 1 多个开关同时 ON 异常） NJ 系列 CPU 单元时，集合了以下异常标志。 • _EIP_IdentityErr（Identity 信息不正确） • _EIP_TDLinkCfgErr （标签数据链接设定异常） • _EIP_TDLinkOpnErr （建立标签数据链接失败） • _EIP_TDLinkErr （标签数据链接通信异常） • _EIP_TagAdrErr （标签解决异常） • _EIP_MultiSwONErr （多个开关同时 ON 异常） （注）“ 标签解决异常 ” 时登录到事件日志中，本异常变为 ON。各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#00F0	A-112
_EIP1_CipErr	CIP 通信 1 异常	CIP 通信 1 的异常状态变量。 集合了以下异常标志。 • _EIP1_IdentityErr（CIP 通信 1Identity 信息不正确） • _EIP1_TDLinkCfgErr （CIP 通信 1 标签数据链接设定异常） • _EIP1_TDLinkOpnErr（CIP 通信 1 建立标签数据链接失败） • _EIP1_TDLinkErr（CIP 通信 1 标签数据链接通信异常） • _EIP1_TagAdrErr（CIP 通信 1 标签解决异常） • _EIP1_MultiSwONErr（CIP 通信 1 多个开关同时 ON 异常） （注）“ 标签解决异常 ” 时登录到事件日志中，本异常变为 ON。各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	WORD	16#0000 ～ 16#00F0	A-112

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP2_CipErr	CIP 通信 2 异常	CIP 通信 1 的异常状态变量。 集合了以下异常标志。 • _EIP2_IdentityErr（CIP 通信 2Identity 信息不正确） • _EIP2_TDLinkCfgErr （CIP 通信 2 标签数据链接设定异常） • _EIP2_TDLinkOpnErr（CIP 通信 2 建立标签数据链接失败） • _EIP2_TDLinkErr（CIP 通信 2 标签数据链接通信异常） • _EIP2_TagAdrErr（CIP 通信 2 标签解决异常） • _EIP2_MultiSwONErr（CIP 通信 2 多个开关同时 ON 异常） （注）“ 标签解决异常 ” 时登录到事件日志中，本异常变为 ON。各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	WORD	16#0000 ～ 16#00F0	A-112
_EIP_TcpAppErr	TCP 应用通信异常	TCP 应用通信的异常状态变量。 集合了以下异常标志。 • _EIP_TcpAppCfgErr （TCP 应用程序设定异常） • _EIP_NTPSrvErr （NTP 服务器连接异常） （注）各个位的含义请参考后面介绍的 “ 异常状态中各个位的含义 ”。	WORD	16#0000 ～ 16#00F0	A-113
_EIP_MacAdrErr	MAC 地址异常	NX 系列 CPU 单元时，表示电源 ON 时发生通信端口 1 的 MAC 地址读取异常。 TRUE：发生异常 FALSE：正常 NJ 系列 CPU 单元时，表示电源 ON 时发生 MAC 地址读取异常。 TRUE：发生异常 FALSE：正常	BOOL	TRUE、FALSE	A-113
_EIP1_MacAdrErr	通信端口 1 MAC 地址异常	表示电源 ON 时发生通信端口 1 的 MAC 地址读取异常。 TRUE：发生异常 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-113
_EIP2_MacAdrErr	通信端口 2 MAC 地址异常	表示电源 ON 时发生通信端口 2 的 MAC 地址读取异常。 TRUE：发生异常 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-113
_EIP_LanHwErr	通信控制器异常	NX 系列 CPU 单元时，表示通信端口 1 上发生通信控制器故障。 TRUE：发生故障 FALSE：正常 NJ 系列 CPU 单元时，表示发生通信控制器故障。 TRUE：发生故障 FALSE：正常	BOOL	TRUE、FALSE	A-114

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP1_LanHwErr	通信端口 1 通信控制器异常	表示通信端口 1 上发生通信控制器故障。 TRUE：发生故障 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-114
_EIP2_LanHwErr	通信端口 2 通信控制器异常	表示通信端口 2 上发生通信控制器故障。 TRUE：发生故障 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-114
_EIP_EtnCfgErr	Ethernet 基本设定异常	NX 系列 CPU 单元时，表示通信端口 1 的 Ethernet 通信速度设定 （Speed ／ Duplex）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 NJ 系列 CPU 单元时，表示 Ethernet 通信速度设定 （Speed ／ Duplex）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常	BOOL	TRUE、FALSE	A-114
_EIP1_EtnCfgErr	通信端口 1 Ethernet 基本设定异常	表示通信端口 1 的 Ethernet 通信速度设定（Speed ／ Duplex）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-115
_EIP2_EtnCfgErr	通信端口 2 Ethernet 基本设定异常	表示通信端口 2 的 Ethernet 通信速度设定（Speed ／ Duplex）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-115
_EIP_IPAdrCfgErr	IP 地址设定异常	NX 系列 CPU 单元时，表示通信端口 1 的 IP 地址设定异常。 TRUE：• IP 地址的设定值不正确 • 读取失败 • 从BOOTP服务器获取的IP地址不匹配 FALSE：正常 NJ 系列 CPU 单元时，表示 IP 地址设定异常。 TRUE：• IP 地址的设定值不正确 • 读取失败 • 从BOOTP服务器获取的IP地址不匹配 • 默认网关设定不正确 FALSE：正常	BOOL	TRUE、FALSE	A-115
_EIP1_IPAdrCfgErr	通信端口 1 IP 地址设定异常	表示通信端口 1 的 IP 地址设定异常。 TRUE：• IP 地址的设定值不正确 • 读取失败 • 从BOOTP服务器获取的IP地址不匹配 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-115
_EIP2_IPAdrCfgErr	通信端口 2 IP 地址设定异常	表示通信端口 2 的 IP 地址设定异常。 TRUE：• IP 地址的设定值不正确 • 读取失败 • 从BOOTP服务器获取的IP地址不匹配 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-116

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_IPAdrDupErr	IP 地址重复异常	NX 系列 CPU 单元时，表示通信端口 1 上发生 IP 地址重复。 TRUE：发生重复 FALSE：上述以外 NJ 系列 CPU 单元时，表示发生 IP 地址重复。 TRUE：发生重复 FALSE：上述以外	BOOL	TRUE、FALSE	A-116
_EIP1_IPAdrDupErr	通信端口 1 IP 地址重复异常	表示通信端口 1 上发生 IP 地址重复。 TRUE：发生重复 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-116
_EIP2_IPAdrDupErr	通信端口 2 IP 地址重复异常	表示通信端口 2 上发生 IP 地址重复。 TRUE：发生重复 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-116
_EIP_DNSCfgErr*1	DNS 设定异常	DNS 设定值或 hosts 设定不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常	BOOL	TRUE、FALSE	A-117
_EIP_BootpErr	BOOTP 服务器连接异常	NX 系列 CPU 单元时，表示通信端口 1 上发生 BOOTP 服务器连接失败。 TRUE：BOOTP 服务器的连接失败（超时）。 FALSE：未进行 BOOTP 设定，或已设定 BOOTP，已正常从 BOOTP 服务器获取 IP 地址。 NJ 系列 CPU 单元时，表示发生 BOOTP 服务器连接失败。 TRUE：BOOTP 服务器的连接失败（超时）。 FALSE：未进行 BOOTP 设定，或已设定 BOOTP，已正常从 BOOTP 服务器获取 IP 地址。	BOOL	TRUE、FALSE	A-117
_EIP1_BootpErr	通信端口 1 BOOTP 服务器连接异常	表示通信端口 1 上发生 BOOTP 服务器连接失败。 TRUE：BOOTP 服务器的连接失败（超时）。 FALSE：未进行 BOOTP 设定，或已设定 BOOTP，已正常从 BOOTP 服务器获取 IP 地址。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-117

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP2_BootpErr	通信端口 2 BOOTP 服务器连接异常	表示通信端口 2 上发生 BOOTP 服务器连接失败。 TRUE：BOOTP 服务器的连接失败 （超时）。 FALSE：未进行 BOOTP 设定，或已设定 BOOTP，已正常从 BOOTP 服务器获取 IP 地址。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-117
_EIP_IPRTblErr	IP 路由表异常	NX 系列 CPU 单元时，表示默认网关的设定或 IP 路由表的设定不正确。 或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 NJ 系列 CPU 单元时，表示 IP 路由表的设定或 hosts 设定不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常	BOOL	TRUE、FALSE	A-118
_EIP_IdentityErr	Identity 信息不正确	NX 系列 CPU 单元时，表示 CIP 通信 1 的 Identity 信息 （用户不可改写）不正确。 或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 NJ 系列 CPU 单元时，表示 Identity 信息（用户不可改写）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常	BOOL	TRUE、FALSE	A-118
_EIP1_IdentityErr	CIP 通信 1 Identity 信息不正确	表示 CIP 通信 1 的 Identity 信息 （用户不可改写）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-118
_EIP2_IdentityErr	CIP 通信 2 Identity 信息不正确	表示 CIP 通信 2 的 Identity 信息 （用户不可改写）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-118
_EIP_TDLinkCfgErr	标签数据链接设定异常	NX 系列 CPU 单元时，表示 CIP 通信 1 的标签数据链接设定值不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 NJ 系列 CPU 单元时，表示标签数据链接的设定值不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常	BOOL	TRUE、FALSE	A-119
_EIP1_TDLinkCfgErr	CIP 通信 1 标签数据链接设定异常	表示 CIP 通信 1 的标签数据链接设定值不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-119

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP2_TDLinkCfgErr	CIP 通信 2 标签数据链接设定异常	表示 CIP 通信 2 的标签数据链接设定值不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-119
_EIP_TDLinkOpnErr	建立标签数据链接失败	NX 系列 CPU 单元时，表示建立 CIP 通信 1 的标签数据链接失败。 TRUE：因以下原因，导致建立标签数据链接失败。 • 标签数据连接参数中登录的配对节点信息与实际的节点信息不同。 • 配对节点无响应。 FALSE：上述以外 NJ 系列 CPU 单元时，表示建立标签数据链接失败。 TRUE：因以下原因，导致建立标签数据链接失败。 • 标签数据连接参数中登录的配对节点信息与实际的节点信息不同。 • 配对节点无响应。 FALSE：上述以外	BOOL	TRUE、FALSE	A-119
_EIP1_TDLinkOpnErr	CIP 通信 1 建立标签数据链接失败	表示建立 CIP 通信 1 的标签数据链接失败。 TRUE：因以下原因，导致建立标签数据链接失败。 • 标签数据连接参数中登录的配对节点信息与实际的节点信息不同。 • 配对节点无响应。 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-120
_EIP2_TDLinkOpnErr	CIP 通信 2 建立标签数据链接失败	表示建立 CIP 通信 2 的标签数据链接失败。 TRUE：因以下原因，导致建立标签数据链接失败。 • 标签数据连接参数中登录的配对节点信息与实际的节点信息不同。 • 配对节点无响应。 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-120

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_TDLinkErr	标签数据链接通信异常	NX 系列 CPU 单元时，表示建立 CIP 通信 1 的标签数据链接时发生超时。 TRUE：发生超时 FALSE：上述以外 NJ 系列 CPU 单元时，表示建立标签数据链接时发生超时。 TRUE：发生超时 FALSE：上述以外	BOOL	TRUE、FALSE	A-120
_EIP1_TDLinkErr	CIP 通信 1 标签数据链接通信异常	表示 CIP 通信 1 的标签数据链接发生超时。 TRUE：发生超时 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-120
_EIP2_TDLinkErr	CIP 通信 2 标签数据链接通信异常	表示 CIP 通信 2 的标签数据链接发生超时。 TRUE：发生超时 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-121
_EIP_TagAdrErr	标签解决异常	NX 系列 CPU 单元时，表示解决 CIP 通信 1 的标签 （根据标签名称找出对应的地址）失败。 TRUE：标签解决 （根据标签名称找出对应的地址）失败。原因如下： • 网络变量和标签设定之间的大小不同。 • 标签数据链接的设定输入输出方向和 CPU 单元中的变量输入输出方向不一致。 • CPU 单元中，与标签设定对应的网络变量不存在。 FALSE：上述以外 NJ 系列 CPU 单元时，表示解决标签 （根据标签名称找出对应的地址）失败。 TRUE：标签解决 （根据标签名称找出对应的地址）失败。原因如下： • 网络变量和标签设定之间的大小不同。 • 标签数据链接的设定输入输出方向和 CPU 单元中的变量输入输出方向不一致。 • CPU 单元中，与标签设定对应的网络变量不存在。 FALSE：上述以外	BOOL	TRUE、FALSE	A-121

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP1_TagAdrErr	CIP 通信 1 标签解决异常	表示解决 CIP 通信 1 的标签 （根据标签名称找出对应的地址）失败。 TRUE：标签解决 （根据标签名称找出对应的地址）失败。原因如下： • 网络变量和标签设定之间的大小不同。 • 标签数据链接的设定输入输出方向和 CPU 单元中的变量输入输出方向不一致。 • CPU 单元中，与标签设定对应的网络变量不存在。 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-121
_EIP2_TagAdrErr	CIP 通信 2 标签解决异常	表示解决 CIP 通信 2 的标签 （根据标签名称找出对应的地址）失败。 TRUE：标签解决 （根据标签名称找出对应的地址）失败。原因如下： • 网络变量和标签设定之间的大小不同。 • 标签数据链接的设定输入输出方向和 CPU 单元中的变量输入输出方向不一致。 • CPU 单元中，与标签设定对应的网络变量不存在。 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-121
_EIP_MultiSwONErr	多个开关同时 ON 异常	NX 系列 CPU 单元时，表示 CIP 通信 1 中有多个开关同时变为 ON。 TRUE：有多个数据链接开始 / 数据链接停止开关同时变为 TRUE。 FALSE：上述以外 NJ 系列 CPU 单元时，表示有多个开关同时变为 ON。 TRUE：有多个数据链接开始 / 数据链接停止开关同时变为 TRUE。 FALSE：上述以外	BOOL	TRUE、FALSE	A-122
_EIP1_MultiSwONErr	CIP 通信 1 多个开关同时 ON 异常	表示 CIP 通信 1 中有多个开关同时变为 ON。 TRUE：有多个数据链接开始 / 数据链接停止开关同时变为 TRUE。 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-122
_EIP2_MultiSwONErr	CIP 通信 2 多个开关同时 ON 异常	表示 CIP 通信 2 中有多个开关同时变为 ON。 TRUE：有多个数据链接开始 / 数据链接停止开关同时变为 TRUE。 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-122

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_TcpAppCfgErr	TCP 应用程序设定异常	TRUE：至少有 1 个或以上 TCP 应用程序功能（FTP、NTP、SNMP）的设定值不正确。或者读取失败。 FALSE：正常	BOOL	TRUE、FALSE	A-122
_EIP_NTPSrvErr	NTP 服务器连接异常	TRUE：NTP 客户端与服务器连接失败（超时）。 FALSE：未设定 NTP。或者已设定 NTP 且连接成功。	BOOL	TRUE、FALSE	A-122
_EIP_DNSSrvErr	DNS 服务器连接异常	TRUE：DNS 客户端与服务器连接失败（超时）。 FALSE：未设定 DNS。或者已设定 DNS 且连接成功。	BOOL	TRUE、FALSE	A-123

*1. NJ 系列 CPU 单元中， CPU 单元版本为 1.11 以上的环境下可以使用。

NJ 系列中与 EtherNet/IP 异常相关的系统定义变量的层级关系

与 EtherNet/IP 异常相关的系统定义变量的层级关系如下表所示。例如，第 2 层级中的 _EIP_PortErr、_EIP_CipErr、_EIP_TcpAppErr 有任意一个不是 0 时，其上一层中的 _EIP_ErrSta 不为 0。即通过参照上位层级中的系统定义变量值，可知道下面的层级是否发生异常。

1 级菜单		2 级菜单		3 级菜单	
变量名称	名称	变量名称	名称	变量名称	名称
_EIP_ErrSta	内置 EtherNet/IP 异常	_EIP_PortErr	通信端口异常	_EIP_MacAdrErr	MAC 地址异常
				_EIP_LanHwErr	通信控制器异常
				_EIP_EtnCfgErr	Ethernet 基本设定异常
				_EIP_IPAdrCfgErr	IP 地址设定异常
				_EIP_IPAdrDupErr	IP 地址重复异常
				_EIP_BootpErr	BOOTP 服务器连接异常
				_EIP_DNSSrvErr	DNS 服务器连接异常
				_EIP_IPRTblErr	IP 路由表异常
		_EIP_CipErr	CIP 通信异常	_EIP_IdentityErr	Identity 信息不正确
				_EIP_TDLinkCfgErr	标签数据链接设定异常
				_EIP_TDLinkOpnErr	建立标签数据链接失败
				_EIP_TDLinkErr	标签数据链接通信异常
				_EIP_TagAdrErr	标签解决异常
				_EIP_MultiSwONErr	多个开关同时 ON
		_EIP_TcpAppErr	TCP 应用通信异常	_EIP_TcpAppCfgErr	TCP 应用程序设定异常
				_EIP_NTPSrvErr	NTP 服务器连接异常

NX 系列中与 EtherNet/IP 异常相关的系统定义变量的层级关系

与EtherNet/IP异常相关的系统定义变量的层级关系如下表所示。例如，第2层级中的_EIP1_PortErr、_EIP2_PortErr、_EIP1_CipErr、_EIP2_CipErr、_EIP_TcpAppErr 有任意一个不是 0 时，其上一层中的 _EIP_ErrSta 不为 0。即通过参照上位层级中的系统定义变量值，可知道下面的层级是否发生异常。

1 级菜单		2 级菜单		3 级菜单	
变量名称	名称	变量名称	名称	变量名称	名称
_EIP_ErrSta	内置 EtherNet/IP 异常	_EIP1_PortErr	通信端口 1 异常	_EIP1_MacAdrErr	通信端口 1 MAC 地址异常
				_EIP1_LanHwErr	通信端口 1 通信控制器异常
				_EIP1_EtnCfgErr	通信端口 1 Ethernet 基本设定异常
				_EIP1_IPAdrCfgErr	通信端口 1 IP 地址设定异常
				_EIP1_IPAdrDupErr	通信端口 1 IP 地址重复异常
				_EIP1_BootpErr	通信端口 1 BOOTP 服务器连接异常
				_EIP_DNSCfgErr	DNS 设定异常
				_EIP_DNSSrvErr	DNS 服务器连接异常
				_EIP_IPRTblErr	IP 路由表异常
		_EIP2_PortErr	通信端口 2 异常	_EIP2_MacAdrErr	通信端口 2 MAC 地址异常
				_EIP2_LanHwErr	通信端口 2 通信控制器异常
				_EIP2_EtnCfgErr	通信端口 2 Ethernet 基本设定异常
				_EIP2_IPAdrCfgErr	通信端口 2 IP 地址设定异常
				_EIP2_IPAdrDupErr	通信端口 2 IP 地址重复异常
				_EIP2_BootpErr	通信端口 2 BOOTP 服务器连接异常
				_EIP_DNSCfgErr	DNS 设定异常
				_EIP_DNSSrvErr	DNS 服务器连接异常
				_EIP_IPRTblErr	IP 路由表异常
		_EIP1_CipErr	CIP 通信 1 异常	_EIP1_IdentityErr	CIP 通信 1 Identity 信息不正确
				_EIP1_TDLinkCfgErr	CIP 通信 1 标签数据链接设定异常
				_EIP1_TDLinkOpnErr	CIP 通信 1 建立标签数据链接失败
				_EIP1_TDLinkErr	CIP 通信 1 标签数据链接通信异常
				_EIP1_TagAdrErr	CIP 通信 1 标签解决异常
				_EIP1_MultiSwONErr	CIP 通信 1 多个开关同时 ON
		_EIP2_CipErr	CIP 通信 2 异常	_EIP2_IdentityErr	CIP 通信 2 Identity 信息不正确
				_EIP2_TDLinkCfgErr	CIP 通信 2 标签数据链接设定异常
				_EIP2_TDLinkOpnErr	CIP 通信 2 建立标签数据链接失败
				_EIP2_TDLinkErr	CIP 通信 2 标签数据链接通信异常
				_EIP2_TagAdrErr	CIP 通信 2 标签解决异常
				_EIP2_MultiSwONErr	CIP 通信 2 多个开关同时 ON
		_EIP_TcpAppErr	TCP 应用通信异常	_EIP_TcpAppCfgErr	TCP 应用程序设定异常
				_EIP_NTPSrvErr	NTP 服务器连接异常

（注）变量名称以 “_EIP1” 开始的系统定义变量与变量名称以 “_EIP” 开始的同名系统定义变量参照相同的值。例如，_EIP1_PortErr （通信端口 1 异常）与 _EIP_PortErr （通信端口异常）参照相同的值。

● 功能分类：EtherNet/IP 通信的状态相关

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_EtnOnlineSta	在线	NX 系列 CPU 单元时，表示可使用内置 EtherNet/IP 端口中通信端口 1 的通信功能（链接 ON、 IP 地址已确定。且未发生异常）。 TRUE：可使用 FALSE：因初始化处理中、重新启动中、发生异常、链接断开，导致内置 EtherNet/IP 端口的通信功能无法使用。 NJ 系列 CPU 单元时，表示可使用内置 EtherNet/IP 端口中通信端口的通信功能 （链接 ON、 IP 地址已确定。且未发生异常）。 TRUE：可使用 FALSE：因初始化处理中、重新启动中、发生异常、链接断开，导致内置 EtherNet/IP 端口的通信功能无法使用。	BOOL	TRUE、FALSE	A-123
_EIP1_EtnOnlineSta	通信端口 1 在线	表示可使用内置 EtherNet/IP 端口中通信端口 1 的通信功能 （链接 ON、 IP 地址已确定。且未发生异常）。 TRUE：可使用 FALSE：因初始化处理中、重新启动中、发生异常、链接断开，导致内置 EtherNet/IP 端口的通信功能无法使用。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-123
_EIP2_EtnOnlineSta	通信端口 2 在线	表示可使用内置 EtherNet/IP 端口中通信端口 2 的通信功能 （链接 ON、 IP 地址已确定。且未发生异常）。 TRUE：可使用 FALSE：因初始化处理中、重新启动中、发生异常、链接断开，导致内置 EtherNet/IP 端口的通信功能无法使用。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-123
_EIP_TDLinkRunSta	标签数据链接通信中	NX 系列 CPU 单元时，表示 CIP 通信 1 中至少有 1 个或以上的连接正在正常动作。 TRUE：正常动作 FALSE：上述以外 NJ 系列 CPU 单元时，表示至少有 1 个或以上的连接正在正常动作。 TRUE：正常动作 FALSE：上述以外	BOOL	TRUE、FALSE	A-124
_EIP1_TDLinkRunSta	CIP 通信 1 标签数据链接通信中	表示 CIP 通信 1 中至少有 1 个或以上的连接正在正常动作。 TRUE：正常动作 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-124

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP2_TDLinkRunSta	CIP 通信 2 标签数据链接通信中	表示 CIP 通信 2 中至少有 1 个或以上的连接正在正常动作。 TRUE：正常动作 FALSE：上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-124
_EIP_TDLinkAllRunSta	所有标签数据链接通信中	NX 系列 CPU 单元时，表示 CIP 通信 1 的所有标签数据链接正在通信。 TRUE：表示作为始发端，所有连接正在进行标签数据链接通信。 FALSE：至少有 1 个或以上的连接发生异常。 NJ 系列 CPU 单元时，表示所有标签数据链接正在通信。 TRUE：表示作为始发端，所有连接正在进行标签数据链接通信。 FALSE：至少有 1 个或以上的连接发生异常。	BOOL	TRUE、FALSE	A-124
_EIP1_TDLinkAllRunSta	CIP 通信 1 所有标签数据链接通信中	表示 CIP 通信 1 的所有标签数据链接正在通信。 TRUE：表示作为始发端，所有连接正在进行标签数据链接通信。 FALSE：至少有 1 个或以上的连接发生异常。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-125
_EIP2_TDLinkAllRunSta	CIP 通信 2 所有标签数据链接通信中	表示 CIP 通信 2 的所有标签数据链接正在通信。 TRUE：表示作为始发端，所有连接正在进行标签数据链接通信。 FALSE：至少有 1 个或以上的连接发生异常。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-125
_EIP_RegTargetSta［255］	登录目标节点信息	NX 系列 CPU 单元时，表示在内置 EtherNet/IP 的 CIP 通信 1 中登录了连接的节点一览。 本变量仅在始发端为内置 EtherNet/IP 时有效。 排列 [x] 为 TRUE： 已登录与目标节点 ID“x” 的节点之间的连接。 排列 [x] 为 FALSE： 未登录与目标节点 ID“x” 的节点之间的连接。 NJ 系列 CPU 单元时，表示在内置 EtherNet/IP 中登录了连接的节点一览。 本变量仅在始发端为内置 EtherNet/IP 时有效。 排列 [x] 为 TRUE： 已登录与目标节点 ID“x” 的节点之间的连接。 排列 [x] 为 FALSE： 未登录与目标节点 ID“x” 的节点之间的连接。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-125

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP1_RegTargetSta［255］	CIP 通信 1 登录目标节点信息	表示在内置 EtherNet/IP 的 CIP 通信 1 中登录了连接的节点一览。 本变量仅在始发端为内置 EtherNet/IP 时有效。 排列 [x] 为 TRUE： 已登录与目标节点 ID“x” 的节点之间的连接。 排列 [x] 为 FALSE： 未登录与目标节点 ID“x” 的节点之间的连接。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-125
_EIP2_RegTargetSta［255］	CIP 通信 2 登录目标节点信息	表示在内置 EtherNet/IP 的 CIP 通信 2 中登录了连接的节点一览。 本变量仅在始发端为内置 EtherNet/IP 时有效。 排列 [x] 为 TRUE： 已登录与目标节点 ID“x” 的节点之间的连接。 排列 [x] 为 FALSE： 未登录与目标节点 ID“x” 的节点之间的连接。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-126
_EIP_EstbTargetSta［255］	正常目标节点信息	NX 系列 CPU 单元时，表示在内置 EtherNet/IP 的 CIP 通信 1 中已正常建立连接的节点一览。 排列 [x] 为 TRUE： 已正常与目标节点 ID“x” 的节点建立连接。 排列 [x] 为 FALSE： 未与目标节点 ID“x” 的节点建立连接或发生异常。 NJ 系列 CPU 单元时，表示在内置 EtherNet/IP 中已正常建立连接的节点一览。 排列 [x] 为 TRUE： 已正常与目标节点 ID“x” 的节点建立连接。 排列 [x] 为 FALSE： 未与目标节点 ID“x” 的节点建立连接或发生异常。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-126
_EIP1_EstbTargetSta［255］	CIP 通信 1 正常目标节点信息	表示在内置 EtherNet/IP 的 CIP 通信 1 中已正常建立连接的节点一览。 排列 [x] 为 TRUE： 已正常与目标节点 ID“x” 的节点建立连接。 排列 [x] 为 FALSE： 未与目标节点 ID“x” 的节点建立连接或发生异常。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-126

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP2_EstbTargetSta［255］	CIP 通信 2 正常目标节点信息	表示在内置 EtherNet/IP 的 CIP 通信 2 中已正常建立连接的节点一览。 排列 [x] 为 TRUE： 已正常与目标节点 ID“x” 的节点建立连接。 排列 [x] 为 FALSE： 未与目标节点 ID“x” 的节点建立连接或发生异常。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-126
_EIP_TargetPLCModeSta［255］	目标 PLC 动作模式	NX 系列 CPU 单元时，表示在始发端为内置 EtherNet/IP 的 CIP 通信 1 的连接中，对应目标节点的控制器运行信息。 本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE： 目标节点 ID“x” 的目标控制器正在运行的状态。 排列 [x] 为 FALSE： 上述以外 NJ 系列 CPU 单元时，表示在始发端为内置 EtherNet/IP 的连接中，对应目标节点的控制器运行信息。 本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE： 目标节点 ID“x” 的目标控制器正在运行的状态。 排列 [x] 为 FALSE： 上述以外	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-127
_EIP1_TargetPLCModeSta［255］	CIP 通信 1 目标 PLC 动作模式	表示在始发端为内置 EtherNet/IP 的 CIP 通信 1 的连接中，对应目标节点的控制器运行信息。 本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE： 目标节点 ID“x” 的目标控制器正在运行的状态。 排列 [x] 为 FALSE： 上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-127
_EIP2_TargetPLCModeSta［255］	CIP 通信 2 目标 PLC 动作模式	表示在始发端为内置 EtherNet/IP 的 CIP 通信 2 的连接中，对应目标节点的控制器运行信息。 本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE： 目标节点 ID“x” 的目标控制器正在运行的状态。 排列 [x] 为 FALSE： 上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-127

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_TargetPLCErr［255］	目标 PLC 异常信息	NX 系列 CPU 单元时，表示在始发端为内置 EtherNet/IP 的 CIP 通信 1 的连接中，对应目标节点的控制器本体的异常状态（运行停止、持续异常的逻辑或）。本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE： 目标节点 ID“x” 的目标控制器继续运行或发生停止异常。 排列 [x] 为 FALSE： 上述以外 NJ 系列 CPU 单元时，表示在始发端为内置 EtherNet/IP 的连接中，对应目标节点的控制器本体的异常状态 （运行停止、持续异常的逻辑或）。本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE： 目标节点 ID“x” 的目标控制器继续运行或发生停止异常。 排列 [x] 为 FALSE： 上述以外	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-128
_EIP1_TargetPLCErr［255］	CIP 通信 1 目标 PLC 异常信息	表示在始发端为内置 EtherNet/IP 的 CIP 通信 1 的连接中，对应目标节点的控制器本体的异常状态 （运行停止、持续异常的逻辑或）。本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE： 目标节点 ID“x” 的目标控制器继续运行或发生停止异常。 排列 [x] 为 FALSE： 上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-128
_EIP2_TargetPLCErr［255］	CIP 通信 2 目标 PLC 异常信息	表示在始发端为内置 EtherNet/IP 的 CIP 通信 2 的连接中，对应目标节点的控制器本体的异常状态 （运行停止、持续异常的逻辑或）。本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE： 目标节点 ID“x” 的目标控制器继续运行或发生停止异常。 排列 [x] 为 FALSE： 上述以外 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-128

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_TargetNodeErr［255］	目标节点异常信息	NX 系列 CPU 单元时，表示与 CIP 通信 1 的登录目标节点信息对应的连接未建立或目标控制器发生异常。 本区域仅在登录目标节点信息为 TRUE 时有效。 排列 [x] 为 TRUE： 未正常与目标节点 ID“x” 的目标节点建立连接 （登录目标节点信息为 TRUE，正常目标节点信息为 FALSE）， 已与目标节点建立连接，但目标控制器发生异常。 排列 [x] 为 FALSE： 目标节点 ID“x” 的目标节点未登录 （登录目标节点信息为 FALSE），或已与目标节点建立连接，（登录目标节点信息为 TRUE、正常目标节点信息为 TRUE）但目标控制器发生异常 （目标 PLC 异常信息为 TRUE）。 NJ 系列 CPU 单元时，表示与登录目标节点信息对应的连接未建立或目标控制器发生异常。 本区域仅在登录目标节点信息为 TRUE 时有效。 排列 [x] 为 TRUE： 未正常与目标节点 ID“x” 的目标节点建立连接 （登录目标节点信息为 TRUE，正常目标节点信息为 FALSE）， 已与目标节点建立连接，但目标控制器发生异常。 排列 [x] 为 FALSE： 目标节点 ID“x” 的目标节点未登录 （登录目标节点信息为 FALSE），或已与目标节点建立连接，（登录目标节点信息为 TRUE、正常目标节点信息为 TRUE）但目标控制器发生异常 （目标 PLC 异常信息为 TRUE）。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-129

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP1_TargetNodeErr［255］	CIP 通信 1 目标节点异常信息	表示与 CIP 通信 1 的登录目标节点信息对应的连接未建立或目标控制器发生异常。 本区域仅在登录目标节点信息为 TRUE 时有效。 排列 [x] 为 TRUE： 未正常与目标节点 ID“x” 的目标节点建立连接 （登录目标节点信息为 TRUE，正常目标节点信息为 FALSE）， 已与目标节点建立连接，但目标控制器发生异常。 排列 [x] 为 FALSE： 目标节点 ID“x” 的目标节点未登录 （登录目标节点信息为 FALSE），或已与目标节点建立连接，（登录目标节点信息为 TRUE、正常目标节点信息为 TRUE）但目标控制器发生异常 （目标 PLC 异常信息为 TRUE）。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-129
_EIP2_TargetNodeErr［255］	CIP 通信 2 目标节点异常信息	表示与 CIP 通信 2 的登录目标节点信息对应的连接未建立或目标控制器发生异常。 本区域仅在登录目标节点信息为 TRUE 时有效。 排列 [x] 为 TRUE： 未正常与目标节点 ID“x” 的目标节点建立连接 （登录目标节点信息为 TRUE，正常目标节点信息为 FALSE）， 已与目标节点建立连接，但目标控制器发生异常。 排列 [x] 为 FALSE： 目标节点 ID“x” 的目标节点未登录 （登录目标节点信息为 FALSE），或已与目标节点建立连接，（登录目标节点信息为 TRUE、正常目标节点信息为 TRUE）但目标控制器发生异常 （目标 PLC 异常信息为 TRUE）。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	ARRAY [0..255] OF BOOL	TRUE、FALSE	A-129
_EIP_NTPResult	NTP 的动作信息	通过用户程序参照 NTP 的动作信息时，请使用 GetNTPStatus 指令。 无法直接访问。	_sNTP_RESULT		A-130

变量名称	名称	功能	数据类型	值的范围	参照页
.ExecTime	NTP 的最终动作时间	登录 NTP 正常结束的时间。 从 NTP 服务器正常获取时间后，登录从 NTP 服务器获取的时间。 若未从 NTP 服务器正常获取时间，则不登录。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。	DATE_AND_TIME	遵照数据类型	A-130
.ExecNormal	NTP 的动作结果	TRUE：表示 NTP 正常结束。 FALSE：表示 NTP 异常结束或 NTP 一次都未执行。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。	BOOL	TRUE、FALSE	A-130

参考

与目标节点的通信状态

NJ/NX 系列控制器和目标节点的通信状态以以下 4 个系统定义变量值的组合，按下表决定。

- _EIP_RegTargetSta （登录目标节点信息）
- _EIP_EstbTargetSta （正常目标节点信息）
- _EIP_TargetPLCErr （目标 PLC 异常信息）
- _EIP_TargetNodeErr （目标节点异常信息）

_EIP_Reg TargetSta 的值	_EIP_Estb TargetSta 的值	_EIP_Target PLCErr 的值	_EIP_Target NodeErr 的值	与目标节点的通信状态
TRUE	TRUE	FALSE	FALSE	已与目标节点正常建立连接，且目标 PLC 未发生异常。
		TRUE	TRUE	已与目标节点建立连接，但目标 PLC 发生异常。
	FALSE	—	TRUE	未正常与目标节点建立连接。
FALSE	—	—	—	由于未登录目标节点，因此信息无效。

NX 系列控制器时，CIP 通信 1 和 CIP 通信 2 的通信状态分别以以下 4 个系统定义变量值的组合，如上表决定。

- CIP 通信 1 时
  - _EIP1_RegTargetSta （CIP 通信 1 登录目标节点信息）
  - _EIP1_EstbTargetSta （CIP 通信 1 正常目标节点信息）
  - _EIP1_TargetPLCErr （CIP 通信 1 目标 PLC 异常信息）
  - _EIP_TargetNodeErr （目标节点异常信息）
- CIP 通信 2 时
  - _EIP2_RegTargetSta （CIP 通信 2 登录目标节点信息）
  - _EIP2_EstbTargetSta （CIP 通信 2 正常目标节点信息）
  - _EIP2_TargetPLCErr （CIP 通信 2 目标 PLC 异常信息）
  - _EIP_TargetNodeErr （目标节点异常信息）

● 功能分类：EtherNet/IP 通信的开关相关

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP_TDLinkStartCmd	标签数据链接通信开始开关	NX 系列 CPU 单元时，从 FALSE 变为 TRUE 后，开始 CIP 通信 1 的标签数据链接。 开始标签数据链接动作后，自动变为 FALSE。 NJ 系列 CPU 单元时，从 FALSE 变为 TRUE 后，开始标签数据链接。 开始标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或Sysmac Studio等强制操作本开关。	BOOL	TRUE、FALSE	A-130

变量名称	名称	功能	数据类型	值的范围	参照页
_EIP1_TDLinkStartCmd	CIP 通信 1 标签数据链接通信开始开关	从 FALSE 变为 TRUE 后，开始 CIP 通信 1 的标签数据链接。 开始标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或Sysmac Studio等强制操作本开关。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-130
_EIP2_TDLinkStartCmd	CIP 通信 2 标签数据链接通信开始开关	从 FALSE 变为 TRUE 后，开始 CIP 通信 2 的标签数据链接。 开始标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或Sysmac Studio等强制操作本开关。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-131
_EIP_TDLinkStopCmd	标签数据链接通信停止开关	NX 系列 CPU 单元时，从 FALSE 变为 TRUE 后，停止 CIP 通信 1 的标签数据链接。 停止标签数据链接动作后，自动变为 FALSE。 NJ 系列 CPU 单元时，从 FALSE 变为 TRUE 后，停止标签数据链接。 停止标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或Sysmac Studio等强制操作本开关。	BOOL	TRUE、FALSE	A-131
_EIP1_TDLinkStopCmd	CIP 通信 1 标签数据链接通信停止开关	从 FALSE 变为 TRUE 后，停止 CIP 通信 1 的标签数据链接。 停止标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或Sysmac Studio等强制操作本开关。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-131
_EIP2_TDLinkStopCmd	CIP 通信 2 标签数据链接通信停止开关	从 FALSE 变为 TRUE 后，停止 CIP 通信 2 的标签数据链接。 停止标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或Sysmac Studio等强制操作本开关。 （注）本系统定义变量仅可在NX系列CPU单元中使用。	BOOL	TRUE、FALSE	A-131

A-4-7 异常状态中各个位的含义

以下异常状态中各个位的含义共通。

- “_ErrSta”（控制器异常状态）
- “_PLC_ErrSta”（PLC 功能模块异常状态）
- “_CJB_ErrSta”（I/O 总线异常状态）
- “_CJB_MstrErrSta”（I/O 总线主机异常状态）
- “_CJB_UnitErrSta”（I/O 总线单元异常状态）
- “_MC_ErrSta”（MC 异常状态）
- “_MC_ComErrSta”（MC 共通异常状态）
- “_MC_AX_ErrSta”（轴异常状态）
- “_MC_GRP_ErrSta”（轴组异常状态）
- “_EC_ErrSta”（内置 EtherCAT 异常）
- “_EC_PortErr”（通信端口异常）
- “_EC_MstrErr”（主机异常）
- “_EC_SlavErr”（从站异常）
- “_EC_SlavErrTbl”（从站异常表）
- “_EIP_ErrSta”（内置 EtherNet/IP 异常）
- “_EIP_PortErr”（通信端口异常）、“_EIP1_PortErr”（通信端口 1 异常）、“_EIP2_PortErr”（通信端口 2 异常）
- “_EIP_CipErr”（CIP通信异常）、“_EIP1_CipErr”（CIP通信1异常）、“_EIP2_CipErr”（CIP通信2异常）
- “_EIP_TcpAppErr”（TCP 应用通信异常）

如下所示。但是，“_ErrSta”（控制器异常状态）、“_CJB_ErrSta”（I/O 总线异常状态）、“_CJB_MstrErrSta”（I/O 总线主机异常状态）、“_CJB_UnitErrSta”（I/O 总线单元异常状态）出于实时性及与表示各功能模块异常原因的状态之间的同时性考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。

位：	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WORD			-	-	-	-	-	-					-	-	-	-

位	内容
15	主机检测：表示对于控制器异常的异常状态对象单元 / 从站，主机是否检测到控制器异常。 TRUE: 有主机检测到的控制器异常 FALSE: 无主机检测到的控制器异常 （_CJB_UnitErrSta 有效。）
14	从站集：表示在事件发生源（各功能模块）的下位层级（单元 / 从站 / 轴 / 轴组等）中，是否发生控制器异常。 TRUE: 下位层级中有控制器异常 FALSE: 下位层级中无控制器异常 （_CJB_ErrSta/_MC_ErrSta/_EC_ErrSta 有效。）
13 ～ 8	保留
7	表示是否发生全部停止故障（Major fault）等级的控制器异常。 TRUE: 发生全部停止故障等级的控制器异常 FALSE: 未发生全部停止故障等级的控制器异常
6	表示是否发生部分停止故障（Partial fault）等级的控制器异常。 TRUE: 发生部分停止故障等级的控制器异常 FALSE: 未发生部分停止故障等级的控制器异常
5	表示是否发生轻度故障（Minor fault）等级的控制器异常。 TRUE: 发生轻度故障等级的控制器异常 FALSE: 未发生轻度故障等级的控制器异常
4	表示是否发生监视信息（Observation）等级的控制器异常。 TRUE: 发生监视信息等级的控制器异常 FALSE: 未发生监视信息等级的控制器异常
3 ～ 0	保留

A-5 系统定义变量的个别规格

系统定义变量的个别规格表的阅读方法如下。

<table>
<tr><td>变量名称</td><td colspan="3">[系统定义变量名称。开头带类别名称]</td><td>成员 （结构体型时）</td><td>[仅在结构体型时表示成员名称]</td></tr>
<tr><td>名称</td><td colspan="3">[名称]</td><td>全局 / 本地</td><td>“ 全局：全局变量、本地：本地变量 ”</td></tr>
<tr><td>功能</td><td colspan="5">[介绍功能]</td></tr>
<tr><td>数据类型</td><td colspan="3">[表示变量的数据类型]</td><td>值的范围</td><td>[表示变量值的范围]</td></tr>
<tr><td>R/W 访问</td><td>[R：只读
RW：可读写]</td><td>保持</td><td>［表示变量的保持属性］</td><td>网络公开</td><td>[表示变量的网络公开属性]</td></tr>
<tr><td>是否可在用户程序中使用</td><td>［表示是否可在用户程序上直接指定后使用］</td><td>相关指令</td><td colspan="3">[表示与变量相关的指令。
“ 用户程序中的使用 ” 为不可时，表示用于访问的专用指令］</td></tr>
</table>

A-5-1 NJ/NX 系列控制器的系统整体 （类别名称：无）

● 功能分类：时钟相关

变量名称	_CurrentTime				
名称	系统时刻			**全局 / 本地**	全局
功能	保存 CPU 单元的内部时钟数据。				
数据类型	DATE_AND_TIME			**值的范围**	• NX 系列 CPU 单元时 DT#1970-01-01-00:00:00 ～ DT#2069-12-31-23:59:59 • NJ 系列 CPU 单元时 DT#1970-01-01-00:00:00 ～ DT#2106-02-06-23:59:59
R/W 访问	R	**保持**	非保持	**网络公开**	公开
是否可在用户程序中使用	可以	**相关指令**	时钟功能用指令		

● 功能分类：任务相关

变量名称	_< 任务名称 >_Active				
名称	任务执行中标志			**全局 / 本地**	全局
功能	正在执行任务时为 TRUE。 任务非执行中时为 FALSE。 （注） 本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。				
数据类型	BOOL			**值的范围**	TRUE、 FALSE
R/W 访问	R	**保持**	非保持	**网络公开**	非公开
是否可在用户程序中使用	不可	**相关指令**	• ActEventTask 在用户程序中，只能通过以下专用指令访问。 • Task_IsActive		

变量名称	_< 任务名称 >_LastExecTime				
名称	上次任务执行时间			**全局 / 本地**	全局
功能	表示上次执行任务的时间 （以 0.1μs 为单位）。 （注） 本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。				
数据类型	TIME			**值的范围**	遵照数据类型
R/W 访问	R	**保持**	非保持	**网络公开**	非公开
是否可在用户程序中使用	不可	**相关指令**	在用户程序中，只能通过以下专用指令访问。 • GetMyTaskStatus		

变量名称	_< 任务名称 >_MaxExecTime				
名称	任务执行时间最大值			**全局 / 本地**	全局
功能	表示任务执行时间的最大值 （以 0.1μs 为单位）。 （注） 本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。				
数据类型	TIME			**值的范围**	遵照数据类型
R/W 访问	R	**保持**	非保持	**网络公开**	非公开
是否可在用户程序中使用	不可	**相关指令**	在用户程序中，只能通过以下专用指令访问。 • GetMyTaskStatus		

变量名称	_< 任务名称 >_MinExecTime				
名称	任务执行时间最小值			**全局 / 本地**	全局
功能	表示任务执行时间的最小值 （以 0.1μs 为单位）。 （注） 本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。				
数据类型	TIME			**值的范围**	遵照数据类型
R/W 访问	R	**保持**	非保持	**网络公开**	非公开
是否可在用户程序中使用	不可	**相关指令**	在用户程序中，只能通过以下专用指令访问。 • GetMyTaskStatus		

变量名称	_< 任务名称 >_ExecCount				
名称	任务执行次数			全局 / 本地	全局
功能	表示任务的执行次数。 若超出 4294967295 次，将恢复为 0，然后继续计数。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。				
数据类型	UDINT			值的范围	遵照数据类型
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	不可	相关指令	在用户程序中，只能通过以下专用指令访问。 • GetMyTaskStatus		

变量名称	_< 任务名称 >_Exceeded				
名称	任务周期超出标志			全局 / 本地	全局
功能	超出任务周期后，变为 TRUE。 若在任务周期内完成任务，则变为 FALSE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	不可	相关指令	在用户程序中，只能通过以下专用指令访问。 • GetMyTaskStatus		

变量名称	_< 任务名称 >_ExceedCount				
名称	任务周期超出次数			全局 / 本地	全局
功能	保存超出周期的次数。 若当前值超出数据类型的最大值，当前值将恢复为 0，然后继续计数。 若超出 4294967295 次，将恢复为 0，然后继续计数。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 的数据追踪功能浏览任务状态时使用。				
数据类型	UDINT			值的范围	遵照数据类型
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	不可	相关指令	在用户程序中，只能通过以下专用指令访问。 • GetMyTaskStatus		

● 功能分类：异常相关

变量名称	_ErrSta				
名称	控制器异常状态			全局 / 本地	全局
功能	正在发生控制器异常时为 TRUE。 未发生控制器异常时为 FALSE。 （注）出于实时性及与表示各功能模块异常原因的状态之间的同时性考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。各位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#C0F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	• ResetPLCError • ResetCJBError • ResetECError • ResetMCError • MC_Reset • MC_GroupReset 在用户程序中，只能通过以下专用指令访问。 • GetPLCError • GetCJBError • GetECError • GetMCError • GetEIPError		

变量名称	_AlarmFlag				
名称	用户异常状态			全局 / 本地	全局
功能	发生用户异常时，与事件重要程度对应的位将变为 TRUE。 User fault Level 1 ～ 8 对应位 00 ～ 07。 未发生用户异常时，为 16 进制的 0000。				
数据类型	WORD			值的范围	16#0000 ～ 16#00FF
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	• SetAlarm • ResetAlarm • GetAlarm		

● 功能分类：SD 存储卡相关

变量名称	_Card1Ready				
名称	SD 存储卡可使用标志			全局 / 本地	全局
功能	识别到 SD 存储卡时，变为 TRUE。 未识别到时，为 FALSE。 TRUE：可使用 FALSE：不可使用				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_Card1Protect				
名称	SD 存储卡写保护标志			全局 / 本地	全局
功能	SD 存储卡的 LOCK 开关为写保护时，变为 TRUE。 TRUE：有写保护 FALSE：无写保护				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_Card1Err				
名称	SD 存储卡错误标志			全局 / 本地	全局
功能	安装了不可使用的 SD 存储卡或格式异常时，变为 TRUE。 TRUE：有错误 FALSE：无错误				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_Card1Access				
名称	SD 存储卡访问中标志			全局 / 本地	全局
功能	正在访问 SD 存储卡时，变为 TRUE。 TRUE：访问中 FALSE：非访问中 本标志以 100ms 为周期由系统更新。因此， SD 存储卡的访问状态最多会延迟 100ms 反映到本标志中。所以，不建议在用户程序上使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_Card1Deteriorated				
名称	SD 存储卡寿命警告标志			全局 / 本地	全局
功能	检测到 SD 存储卡达到使用寿命时，变为 TRUE。 本标志变为 TRUE 时，请更换 SD 存储卡。 若不更换，可能导致读取 / 写入失败。 TRUE：检测到使用寿命 FALSE：未检测到使用寿命				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_Card1PowerFail				
名称	SD 存储卡访问中断电标志			全局 / 本地	全局
功能	正在访问 SD 存储卡的过程中，若 CPU 单元发生断电，则变为 TRUE。 TRUE：正在访问 SD 存储卡时发生断电 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	保持*1	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_Card1BkupCmd*1			成员名称	.ExecBkup
名称	备份启动标志			全局 / 本地	全局
功能	向 SD 存储卡备份控制器的数据时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。				
数据类型	结构体：_sBKUP_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1BkupCmd*1			成员名称	.CancelBkup
名称	备份取消标志			全局 / 本地	全局
功能	停止向 SD 存储卡备份时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。				
数据类型	结构体：_sBKUP_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1BkupCmd*1			成员名称	.ExecVefy
名称	核对启动标志			全局 / 本地	全局
功能	核对控制器和 SD 存储卡中的备份文件时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。				
数据类型	结构体：_sBKUP_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1BkupCmd*1			成员名称	.CancelVefy
名称	核对取消标志			全局 / 本地	全局
功能	停止核对控制器和 SD 存储卡中的备份文件时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。				
数据类型	结构体：_sBKUP_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1BkupCmd*1			成员名称	.DirName
名称	目录名			全局 / 本地	全局
功能	指定备份对象的 SD 存储卡目录名称。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，由显示器或上位计算机发出指令时使用。				
数据类型	结构体：_sBKUP_CMD、成员：STRING(64)			值的范围	遵照数据类型
R/W 访问	RW	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1BkupSta*1			成员名称	.Done
名称	完成标志			全局 / 本地	全局
功能	备份执行完成时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，从显示器或上位计算机读取时使用。				
数据类型	结构体：_sBKUP_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1BkupSta*1			成员名称	.Active
名称	执行中标志			全局 / 本地	全局
功能	正在执行备份时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，从显示器或上位计算机读取时使用。				
数据类型	结构体：_sBKUP_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1BkupSta*1			成员名称	.Err
名称	错误标志			全局 / 本地	全局
功能	备份处理异常结束时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，从显示器或上位计算机读取时使用。				
数据类型	结构体：_sBKUP_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1VefySta*1			成员名称	.Done
名称	完成标志			全局 / 本地	全局
功能	核对执行完成时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，从显示器或上位计算机读取时使用。				
数据类型	结构体：_sVEFY_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1VefySta*1			成员名称	.Active
名称	执行中标志			全局 / 本地	全局
功能	正在执行核对时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，从显示器或上位计算机读取时使用。				
数据类型	结构体：_sVEFY_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1VefySta*1			成员名称	.VefyRslt
名称	核对结果标志			全局 / 本地	全局
功能	核对一致时为 TRUE，核对不一致时为 FALSE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，从显示器或上位计算机读取时使用。				
数据类型	结构体：_sVEFY_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1VefySta*1			成员名称	.Err
名称	错误标志			全局 / 本地	全局
功能	核对处理异常结束时为 TRUE。 （注）本系统定义变量在用户程序中不可使用。通过 CIP 信息通信，从显示器或上位计算机读取时使用。				
数据类型	结构体：_sVEFY_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

变量名称	_Card1PrgTransferCmd*1			成员名称	.Exec
名称	程序传送启动标志			全局 / 本地	全局
功能	利用 SD 存储卡程序传送功能，向控制器传送 SD 存储卡中的备份文件数据时，变为 TRUE。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	保持*2	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。
*2. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_Card1PrgTransferCmd*1			成员名称	.DirName
名称	目录名			全局 / 本地	全局
功能	指定保存有程序传送对象备份文件的 SD 存储卡目录名称。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_CMD、成员：STRING(64)			值的范围	遵照数据类型
R/W 访问	RW	保持	保持*2	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。
*2. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_Card1PrgTransferCmd*1			成员名称	.Password
名称	密码			全局 / 本地	全局
功能	指定开始传送程序时要核对的密码。 每次开始 SD 存储卡程序传送时，密码将初始化。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_CMD、成员：STRING(33)			值的范围	遵照数据类型
R/W 访问	RW	保持	保持*2	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。
*2. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_Card1PrgTransferCmd*1			成员名称	.TargetUserProgram
名称	用户程序、设定的传送标志			全局 / 本地	全局
功能	程序传送时的传送对象为用户程序或设定时，变为 TRUE。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	保持*2	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。
*2. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_Card1PrgTransferCmd*1			成员名称	.TargetIPAdr
名称	IP 地址的传送标志			全局 / 本地	全局
功能	程序传送时的传送对象为内置 EtherNet/IP 端口的 IP 地址时，变为 TRUE。 IP 地址中包括设定种类、 IP 地址、子网掩码、默认网关。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	保持*2	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。
*2. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_Card1PrgTransferCmd*1			成员名称	.TargetVariable
名称	变量当前值 （仅保持属性）的传送标志			全局 / 本地	全局
功能	程序传送时的传送对象为变量的当前值 （仅保持属性）时，变为 TRUE。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	保持*2	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。
*2. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_Card1PrgTransferCmd*1			成员名称	.TargetMemory
名称	CJ 单元用存储器当前值 （仅保持存储器）的传送标志			全局 / 本地	全局
功能	程序传送时的传送对象为 CJ 单元用存储器的当前值 （仅保持存储器）时，变为 TRUE。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_CMD、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	保持*2	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。
*2. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_Card1PrgTransferSta*1			成员名称	.Done
名称	完成标志			全局 / 本地	全局
功能	程序传送执行完成时为 TRUE。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。

变量名称	_Card1PrgTransferSta*1			成员名称	.Active
名称	执行中标志			全局 / 本地	全局
功能	正在执行程序传送时为 TRUE。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。

变量名称	_Card1PrgTransferSta*1			成员名称	.Err
名称	错误标志			全局 / 本地	全局
功能	程序传送异常结束时为 TRUE。 （注）本系统定义变量仅可在 NJ/NX 系列 CPU 单元中使用。				
数据类型	结构体：_sPRGTRANSFER_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。

● 功能分类：备份相关

变量名称	_BackupBusy*1				
名称	备份相关功能执行中标志			全局 / 本地	全局
功能	正在执行备份、恢复、核对时为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.03 中追加的系统定义变量。

● 功能分类：电源管理相关

变量名称	_PowerOnHour				
名称	通电时间			全局 / 本地	全局
功能	表示通电时间。 以 1 小时为单位保存 CPU 单元的通电时间。 重置时，请保存 0。 变为 4294967295 后，不会继续更新。 电源 ON 时不会初始化。				
数据类型	UDINT			值的范围	0 ～ 4294967295
R/W 访问	RW	保持	保持*1	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_PowerOnCount				
名称	断电发生次数			全局 / 本地	全局
功能	表示发生断电的次数。 从第一次打开 CPU 单元的电源后，每次发生断电时累加 （+1）。 重置时，请保存 0。 变为 4294967295 后，不会继续更新。 电源 ON 时不会初始化。				
数据类型	UDINT			值的范围	0 ～ 4294967295
R/W 访问	RW	保持	保持*1	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. 有保持属性的系统定义变量，但不在备份功能的对象范围内。

变量名称	_RetainFail				
名称	断电保持失败标志			全局 / 本地	全局
功能	以下情况下，将变为 TRUE （断电保持失败）。 • 接通电源时，电池备份存储器检查错误时 以下情况下，将变为 FALSE （无断电保持失败）。 • 接通电源时，电池备份存储器检查 OK 时 • 下载了用户程序时 • 执行了存储器全部清除时 （注）无法保持绝对编码器原点位置偏置数据时，将反映到轴变量的异常状态中，而不是本标志。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	—		

● 功能分类：编程相关

变量名称	P_On				
名称	始终 ON 标志			全局 / 本地	全局
功能	表示始终为 ON 的标志。				
数据类型	BOOL			值的范围	TRUE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	P_Off				
名称	始终 OFF 标志			全局 / 本地	全局
功能	表示始终为 OFF 的标志。				
数据类型	BOOL			值的范围	FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	P_CY				
名称	进位标志			全局 / 本地	本地
功能	在部分指令中更新的标志。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	P_First_RunMode				
名称	运行开始时 1 个周期 ON 标志			全局 / 本地	本地
功能	控制器的动作模式从程序模式变更为运行模式时，若该程序正在执行，则在 1 个任务周期内为 TRUE。 若该程序不是正在执行，则保持 FALSE。 用于在 CPU 单元开始运行时进行初始处理等情况。 （注）本系统定义变量在函数中不可使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	P_First_Run*1				
名称	程序启动时 1 个周期 ON 标志			全局 / 本地	本地
功能	该程序开始执行后，仅在 1 个任务周期内为 TRUE。 用于每次开始执行程序时需要进行初始处理的情况。 （注）本系统定义变量在函数中不可使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.08 中追加的系统定义变量。

变量名称	P_PRGER				
名称	指令错误发生标志			全局 / 本地	本地
功能	相应的程序或通过该程序调用的 FB 或 FUN 发生指令错误时，持续变为 TRUE 的标志。一旦变为 TRUE，在通过用户程序变为 FALSE 前，将保持 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	—		

● 功能分类：通信相关

变量名称	_Port_numUsingPort				
名称	使用端口数量			全局 / 本地	全局
功能	表示当前正在使用的内部逻辑端口数量。 用于用户程序的调试等。				
数据类型	USINT			值的范围	0 ～ 32
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	通信类指令 （ExecPMCR、 SerialSend、 SerialRcv、通信指令、 SendCmd）		

变量名称	_Port_isAvailable				
名称	网络通信指令可执行标志			全局 / 本地	全局
功能	表示是否有可使用的内部逻辑端口。 有可使用的内部逻辑端口时为 TRUE，没有时为 FALSE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	可以	相关指令	通信类指令 （ExecPMCR、 SerialSend、 SerialRcv、通信指令、 SendCmd）		

变量名称	_FINSTCPConnSta				
名称	FINS/TCP 连接状态			全局 / 本地	全局
功能	表示 FINS/TCP 连接的状态。				
数据类型	WORD			值的范围	16#0000 ～ 16#FFFF
R/W 访问	R	保持	非保持	网络公开	公开*1
是否可在用户程序中使用	可以	相关指令	—		

*1. 单元版本 1.07 以下的 CPU 单元中，不在网络上公开。

● 功能分类：版本管理相关

变量名称	_UnitVersion*1				
名称	单元版本			全局 / 本地	全局
功能	保存 CPU 单元的单元版本。 在要素编号 0 中保存单元版本的整数部分。 在要素编号 1 中保存单元版本的小数部分。 例 1) 单元版本为 1.08 时，要素编号 0 中保存 1、要素编号 1 中保存 8。 例 2) 单元版本为 1.10 时，要素编号 0 中保存 1、要素编号 1 中保存 10。				
数据类型	ARRAY[0..1] OF USINT			值的范围	0 ～ 99
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.08 中追加的系统定义变量。

变量名称	_HardwareRevision*1				
名称	硬件修订版本			全局 / 本地	全局
功能	保存 CPU 单元的硬件修订版本。 硬件修订版本为 “ 无 ” 时，保存 “'-'”，其他情况下保存 “'A' ～ 'Z'”。				
数据类型	STRING[2]			值的范围	'-'、 'A' ～ 'Z'
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。

● 功能分类：自诊断相关

变量名称	_SelfTest_HighTemperature*1				
名称	CPU 单元内部温度超限标志			全局 / 本地	全局
功能	CPU 单元内部温度为高温时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.10 中追加的系统定义变量。

变量名称	_SelfTest_LowBattery*1				
名称	电池电压低标志			全局 / 本地	全局
功能	电池拆下或电池电压低时变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.10 中追加的系统定义变量。

变量名称	_SelfTest_LowFanRevolution*1				
名称	风扇转速降低标志			全局 / 本地	全局
功能	风扇拆下或风扇的转速下降时变为 TRUE。 （注）NJ 系列 CPU 单元时，始终为 FALSE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.10 中追加的系统定义变量。

A-5-2 PLC 功能模块（类别名称：_PLC）

● 功能分类：调试相关

变量名称	_PLC_TraceSta[0..3]			成员名称	.IsStart
名称	追踪执行中 ON 标志			全局 / 本地	全局
功能	开始追踪时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 监视数据追踪状态时使用。				
数据类型	结构体：_sTRACE_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	不可	相关指令	• TraceTrig • TraceSamp 在用户程序中，只能通过以下专用指令访问。 • GetTraceStatus		

变量名称	_PLC_TraceSta[0..3]			成员名称	.IsComplete
名称	追踪完成时 ON 标志			全局 / 本地	全局
功能	追踪执行完成时，变为 TRUE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 监视数据追踪状态时使用。				
数据类型	结构体：_sTRACE_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	不可	相关指令	• TraceTrig • TraceSamp 在用户程序中，只能通过以下专用指令访问。 • GetTraceStatus		

变量名称	_PLC_TraceSta[0..3]			成员名称	.IsTrigger
名称	追踪触发监视标志			全局 / 本地	全局
功能	追踪触发条件成立时，变为 TRUE。 下次开始追踪时，变为 FALSE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 监视数据追踪状态时使用。				
数据类型	结构体：_sTRACE_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	保持	网络公开	非公开
是否可在用户程序中使用	不可	相关指令	• TraceTrig • TraceSamp 在用户程序中，只能通过以下专用指令访问。 • GetTraceStatus		

变量名称	_PLC_TraceSta[0..3]			成员名称	.ParamErr
名称	追踪参数异常标志			全局 / 本地	全局
功能	开始追踪时，若追踪设定异常，则变为 TRUE。 正常时，为 FALSE。 （注）本系统定义变量在用户程序中不可使用。仅在通过 Sysmac Studio 监视数据追踪状态时使用。				
数据类型	结构体：_sTRACE_STA、成员：BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	非公开
是否可在用户程序中使用	不可	相关指令	在用户程序中，只能通过以下专用指令访问。 • GetTraceStatus		

● 功能分类：异常相关

变量名称	_PLC_ErrSta				
名称	PLC 功能模块异常状态			全局 / 本地	全局
功能	发生与 PLC 功能模块相关的控制器异常时，变为 TRUE。 未发生与 PLC 功能模块相关的控制器异常时，变为 FALSE。 各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	• GetPLCError 可通过以下专用指令解除 • ResetPLCError		

A-5-3 PLC 功能模块（类别名称：_CJB）

● 功能分类：I/O 总线的状态相关

变量名称	_CJB_MaxRackNo				
名称	最大装置编号			全局 / 本地	全局
功能	表示控制器识别到的扩展装置的最大装置编号。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。				
数据类型	UINT			值的范围	0 ～ 3 0 表示无扩展装置
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_CJB_MaxSlotNo				
名称	最大插槽编号			全局 / 本地	全局
功能	表示控制器识别到的 CJ 单元各装置的 （最大插槽编号 +1）。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。				
数据类型	ARRAY [0..3] OF UINT			值的范围	0 ～ 10 为 0 时，表示未安装 CJ 单元
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

● 功能分类：I/O 总线的异常相关

变量名称	_CJB_ErrSta				
名称	I/O 总线异常状态			全局 / 本地	全局
功能	I/O 总线的异常状态。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。各位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。				
数据类型	WORD			值的范围	16#0000 ～ 16#C0F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	在用户程序中，只能通过以下专用指令访问。 • GetCJBError 可通过以下专用指令解除 • ResetCJBError		

变量名称	_CJB_MstrErrSta				
名称	I/O 总线主机异常状态			全局 / 本地	全局
功能	I/O 总线主机的异常状态。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。各位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	在用户程序中，只能通过以下专用指令访问。 • GetCJBError 可通过以下专用指令解除 • ResetCJBError		

变量名称	_CJB_UnitErrSta				
名称	I/O 总线单元异常状态			全局 / 本地	全局
功能	I/O 总线单元的异常状态。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。各位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。				
数据类型	ARRAY [0..3, 0..9] OF WORD			值的范围	16#0000 ～ 16#80F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	在用户程序中，只能通过以下专用指令访问。 • GetCJBError 可通过以下专用指令解除 • ResetCJBError		

变量名称	_CJB_InRespTm				
名称	基本输入单元的输入时间常数			全局 / 本地	全局
功能	表示基本输入单元的输入时间常数。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。				
数据类型	ARRAY [0..3, 0..9] OF UINT			值的范围	0 ～ 320
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

● 功能分类：CJ 单元用存储器的特殊辅助继电器相关

变量名称		_CJB_IOUnitInfo				
名称		基本 I/O 单元信息区域			全局 / 本地	全局
功能		显示基本 I/O 单元 （带负载短路保护功能）的警报输出。 TRUE：有负载短路 FALSE：无负载短路 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。				
数据类型		ARRAY [0..3, 0..9, 0..7] of BOOL			值的范围	TRUE、 FALSE
R/W 访问		R	保持	非保持	网络公开	非公开
是否可在用户程序中使用		可以	相关指令	—		
特殊辅助继电器地址	CH	A50 ～ A69				
	位	A50.00 ～ A69.15				

变量名称		_CJB_CBU00InitSta ・・・ _CJB_CBU15InitSta				
名称		CPU 总线单元初始化中标志			全局 / 本地	全局
功能		CPU 总线单元正在初始化时，变为 TRUE。 初始化完成时，变为 FALSE。 各个位与单元编号对应。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。				
数据类型		BOOL			值的范围	TRUE、 FALSE
R/W 访问		R	保持	非保持	网络公开	公开
是否可在用户程序中使用		可以	相关指令	• ResetUnit		
特殊辅助继电器地址	CH	A302				
	位	A302.00 ～ A302.15				

变量名称		_CJB_SIO00InitSta ・・・ _CJB_SIO95InitSta			
名称		高功能 I/O 单元初始化中标志		全局 / 本地	全局
功能		高功能 I/O 单元正在初始化时，变为 TRUE。 初始化完成时，变为 FALSE。 各个位与号机 No. 对应。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。			
数据类型		BOOL		值的范围	TRUE、 FALSE
R/W 访问		R	保持	非保持	网络公开	公开
是否可在用户程序中使用		可以	相关指令	• ResetUnit	
特殊辅助继电器地址	CH	A330 ～ A335			
	位	A330.00 ～ A335.15			

变量名称		_CJB_CBU00Restart ・・・ _CJB_CBU15Restart			
名称		CPU 总线单元重启标志		全局 / 本地	全局
功能		上升时，重启 CPU 总线单元 （重启处理后，在系统中变为 FALSE）。 各个位与单元编号对应。 执行指令时，若重启标志变为 TRUE，将在执行下一任务周期的刷新处理时，开始重启处理。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。			
数据类型		BOOL		值的范围	TRUE、 FALSE
R/W 访问		RW	保持	非保持	网络公开	公开
是否可在用户程序中使用		可以	相关指令	• ResetUnit	
特殊辅助继电器地址	CH	A501			
	位	A501.00 ～ A501.15			

变量名称		_CJB_SIO00Restart ・・・ _CJB_SIO95Restart			
名称		高功能 I/O 单元重启标志		全局 / 本地	全局
功能		上升时，重启高功能 I/O 单元 （重启处理后，在系统中变为 FALSE）。 各个位与号机 No. 对应。 执行指令时，若重启标志变为 TRUE，将在执行下一任务周期的刷新处理时，开始重启处理。 （注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。			
数据类型		BOOL		值的范围	TRUE、 FALSE
R/W 访问		RW	保持	非保持	网络公开	公开
是否可在用户程序中使用		可以	相关指令	• ResetUnit	
特殊辅助继电器地址	CH	A502 ～ A507			
	位	A502.00 ～ A507.15			

<table>
<tr><td colspan="2">变量名称</td><td colspan="5">_CJB_SCU00P1ChgSta
_CJB_SCU00P2ChgSta
・・・
_CJB_SCU15P1ChgSta
_CJB_SCU15P2ChgSta</td></tr>
<tr><td colspan="2">名称</td><td colspan="2">串行通讯单元 0 号机端口 1/2 用串行端口设定变更中标志
・・・
串行通讯单元 1-15 号机端口 1/2 用串行端口设定变更中标志</td><td>全局 / 本地</td><td colspan="2">全局</td></tr>
<tr><td colspan="2">功能</td><td colspan="5">正在变更该端口的设定时，变为 TRUE。
设定变更完成时，变为 FALSE。
通过执行指令或用户操作将该位变为 TRUE 后，可通知串行端口设定的变更。
（注）本系统定义变量仅可在 NJ 系列 CPU 单元中使用。</td></tr>
<tr><td colspan="2">数据类型</td><td colspan="2">BOOL</td><td>值的范围</td><td colspan="2">TRUE、 FALSE</td></tr>
<tr><td colspan="2">R/W 访问</td><td>RW</td><td>保持</td><td>非保持</td><td>网络公开</td><td>公开</td></tr>
<tr><td colspan="2">是否可在用户程序中使用</td><td>可以</td><td>相关指令</td><td colspan="3">—</td></tr>
<tr><td rowspan="2">特殊辅助继电器地址</td><td>CH</td><td colspan="5">串行通讯单元 0 号机端口时：A620
串行通讯单元 1-15 号机端口时：A621 ～ A635</td></tr>
<tr><td>位</td><td colspan="5">串行通讯单元 0 号机端口时：A620.01 ～ A620.02
串行通讯单元 1-15 号机端口时：A621.01 ～ A635.02</td></tr>
</table>

A-5-4 运动控制功能模块（类别名称：_MC）

● 功能分类：运动控制功能相关

变量名称	_MC_ErrSta				
名称	运动控制功能模块的异常状态			全局 / 本地	全局
功能	表示运动控制功能模块中检测到的异常状态。 可在用户程序中使用（直接参照）。各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#40F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	• GetMCError • ResetMCError • MC_Reset • MC_GroupReset		

变量名称	_MC_ComErrSta				
名称	共通异常状态			全局 / 本地	全局
功能	表示在运动控制的共通处理中检测到的异常状态。 可在用户程序中使用（直接参照）。各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	• GetMCError • ResetMCError		

变量名称	_MC_AX_ErrSta				
名称	轴异常状态			全局 / 本地	全局
功能	表示以轴为单位检测到的异常状态。最多可显示 256 个轴[*1]。 可在用户程序中使用（直接参照）。各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	ARRAY [0..255] OF WORD[*1]			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	• GetMCError • ResetMCError • MC_Reset		

*1. NJ 系列 CPU 单元时，最多 64 个轴，数据类型为 ARRAY [0..63] OF WORD。

变量名称	_MC_GRP_ErrSta				
名称	轴组异常状态			全局 / 本地	全局
功能	表示以轴组为单位检测到的异常状态。最多可显示 64 个轴组[*1]。 可在用户程序中使用（直接参照）。各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	ARRAY [0..63] OF WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	• GetMCError • ResetMCError • MC_GroupReset		

*1. NJ 系列 CPU 单元时，最多 32 个轴组，数据类型为 ARRAY [0..31] OF WORD。

变量名称	_MC_COM				
名称	通用变量			全局 / 本地	全局
功能	进行运动控制功能模块共通的状态监视。 结构体的成员详情请参考 📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。				
数据类型	_sCOMMON_REF			值的范围	—
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_MC_GRP				
名称	轴组变量			全局 / 本地	全局
功能	NX 系列 CPU 单元时，进行运动控制 1 中使用的运动控制指令的轴组指定、多轴协调控制的状态监视、多轴协调控制的设定监视。 NJ 系列 CPU 单元时，进行运动控制指令的轴组指定、多轴协调控制的状态监视、多轴协调控制的设定监视。 若用 Sysmac Studio 创建 “ 轴组 ”，将生成用户自定义名称的轴组变量。 通常使用自定义名称的轴组变量。 结构体的成员详情请参考 📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。				
数据类型	ARRAY[0..63] OF _sGROUP_REF*1			值的范围	—
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. NJ 系列 CPU 单元时，为 ARRAY[0..31] OF _sGROUP_REF。

变量名称	_MC1_GRP				
名称	轴组变量			全局 / 本地	全局
功能	进行运动控制 1 中使用的运动控制指令的轴组指定、多轴协调控制的状态监视、多轴协调控制的设定监视。 若用 Sysmac Studio 创建 “ 轴组 ”，将生成用户自定义名称的轴组变量。 通常使用自定义名称的轴组变量。 结构体的成员详情请参考 📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。若 “_MC1_GRP” 和 “_MC_GRP” 的排列要素编号相同，则参照相同的值。				
数据类型	ARRAY[0..63] OF _sGROUP_REF			值的范围	—
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_MC2_GRP				
名称	轴组变量			全局 / 本地	全局
功能	进行运动控制 2 中使用的运动控制指令的轴组指定、多轴协调控制的状态监视、多轴协调控制的设定监视。 若用 Sysmac Studio 创建 “ 轴组 ”，将生成用户自定义名称的轴组变量。 通常使用自定义名称的轴组变量。 结构体的成员详情请参考 📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY[0..63] OF _sGROUP_REF			值的范围	—
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_MC_AX				
名称	轴变量			全局 / 本地	全局
功能	NX 系列 CPU 单元时，进行运动控制 1 中使用的运动控制指令的轴指定、单轴控制的状态监视、单轴控制的设定监视。 NJ 系列 CPU 单元时，进行运动控制指令的轴指定、单轴控制的状态监视、单轴控制的设定监视。 若用 Sysmac Studio 创建 “ 轴 ”，将生成用户自定义名称的轴变量。 通常使用自定义名称的轴变量。 结构体的成员详情请参考 📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。				
数据类型	ARRAY[0..255] OF _sAXIS_REF*1			值的范围	—
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. NJ 系列 CPU 单元时，为 ARRAY[0..63] OF _sAXIS_REF。

变量名称	_MC1_AX				
名称	轴变量			全局 / 本地	全局
功能	进行运动控制 1 中使用的运动控制指令的轴指定、单轴控制的状态监视、单轴控制的设定监视。 若用 Sysmac Studio 创建 “ 轴 ”，将生成用户自定义名称的轴变量。 通常使用自定义名称的轴变量。 结构体的成员详情请参考 📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。若 “_MC1_AX” 和 “_MC_AX” 的排列要素编号相同，则参照相同的值。				
数据类型	ARRAY[0..255] OF _sAXIS_REF			值的范围	—
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_MC2_AX				
名称	轴变量			全局 / 本地	全局
功能	进行运动控制 2 中使用的运动控制指令的轴指定、单轴控制的状态监视、单轴控制的设定监视。 若用 Sysmac Studio 创建 “ 轴 ”，将生成用户自定义名称的轴变量。 通常使用自定义名称的轴变量。 结构体的成员详情请参考 📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY[0..255] OF _sAXIS_REF			值的范围	—
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

A-5-5 EtherCAT 主机功能模块（类别名称：_EC）

● 功能分类：EtherCAT 通信的异常相关

变量名称	_EC_ErrSta				
名称	内置 EtherCAT 异常			全局 / 本地	全局
功能	集合了 EtherCAT 主机功能模块异常的系统定义变量。 各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#40F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常状态获取指令 • GetECError EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_PortErr				
名称	通信端口异常			全局 / 本地	全局
功能	集合了 EtherCAT 主机用通信端口异常的系统定义变量。 各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常状态获取指令 • GetECError EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_MstrErr				
名称	主机异常			全局 / 本地	全局
功能	集合了 EtherCAT 主机异常和 EtherCAT 主机检测到的从站异常的系统定义变量。 各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常状态获取指令 • GetECError EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_SlavErr				
名称	从站异常			全局 / 本地	全局
功能	集合了 EtherCAT 从站整体异常状态的系统定义变量。 各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常状态获取指令 • GetECError EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_SlavErrTbl				
名称	从站异常表			全局 / 本地	全局
功能	以 EtherCAT 从站为单位通知异常状态。 本异常状态表示实际系统构成中以从站为单位的异常状态，本变量的排列与 EtherCAT 从站的节点地址 （1 ～ 512）*1 对应，表示发生异常的从站。 各个位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	Array [1..512] OF WORD*1			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常状态获取指令 • GetECError EtherCAT 异常解除指令 • ResetECError		

*1. NJ 系列 CPU 单元时，节点地址为 1 ～ 192，数据类型为 Array [1..192] OF WORD。

变量名称	_EC_MacAdrErr				
名称	MAC 地址异常			全局 / 本地	全局
功能	MAC 地址不正确时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_LanHwErr				
名称	通信控制器异常			全局 / 本地	全局
功能	检测到通信控制器的硬件异常时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_LinkOffErr				
名称	链接断开异常			全局 / 本地	全局
功能	检测到通信控制器未建立链接时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_NetCfgErr				
名称	网络配置信息异常			全局 / 本地	全局
功能	检测到网络配置信息不正确时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_NetCfgCmpErr				
名称	网络配置核对异常			全局 / 本地	全局
功能	网络配置信息和网络实际配置不一致时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_NetTopologyErr				
名称	网络配置异常			全局 / 本地	全局
功能	发生网络配置异常 （连接台数超限、环形连接）时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_PDCommErr				
名称	过程数据通信异常			全局 / 本地	全局
功能	过程数据通信中，发生从站意外脱离、加入，或检测到从站 WDT 异常时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_PDTimeoutErr				
名称	过程数据接收超时异常			全局 / 本地	全局
功能	若在接收过程数据时发生超时，则变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_PDSendErr				
名称	过程数据发送异常			全局 / 本地	全局
功能	发生过程数据发送异常 （未在过程数据通信周期内发送、发送抖动超限）时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_SlavAdrDupErr				
名称	从站节点地址重复异常			全局 / 本地	全局
功能	若从站的节点地址重复，则变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_SlavInitErr				
名称	从站初始化异常			全局 / 本地	全局
功能	向从站发出初始化指令时若发生异常，则变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_SlavAppErr				
名称	从站应用程序异常			全局 / 本地	全局
功能	检测到从站的应用程序状态寄存器异常时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_MsgErr				
名称	EtherCAT 消息异常			全局 / 本地	全局
功能	向不支持消息功能的从站发送消息或向从站发送的消息返回的响应格式异常时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	CoE 消息 （CoE SDO 读取） • EC_CoESDORead CoE 消息 （CoE SDO 写入） • EC_CoESDOWrite		

变量名称	_EC_SlavEmergErr				
名称	紧急消息检测			全局 / 本地	全局
功能	主机检测到从站发出紧急消息时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

变量名称	_EC_CommErrTbl				
名称	通信异常从站表			全局 / 本地	全局
功能	以从站节点地址顺序的表格表示。主机检测到异常的从站对应的位置变为 TRUE。				
数据类型	Array [1..512] OF BOOL*1			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 异常解除指令 • ResetECError		

*1. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

（注）与 EtherCAT 通信异常相关的所有系统定义变量，在排除异常原因并解除发生的控制器异常（Sysmac Studio 的故障排除 /ResetECError 指令）前，值不会变化。

变量名称	_EC_CycleExceeded				
名称	EtherCAT 通信周期设定异常			全局 / 本地	全局
功能	启动时，若无法在设定的通信周期内进行通信，则变为 TRUE。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

● 功能分类：EtherCAT 通信的状态相关

变量名称	_EC_RegSlavTbl				
名称	登录从站表			全局 / 本地	全局
功能	网络配置信息中登录的从站一览。 以从站节点地址顺序的表格表示。 登录了从站时，对应的位置变为 TRUE。				
数据类型	Array [1..512] OF BOOL*1			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

变量名称	_EC_EntrySlavTbl				
名称	网络加入从站表			全局 / 本地	全局
功能	通知网络加入从站一览。以从站节点地址顺序的表格表示。 从站加入网络时，对应的位置变为 TRUE。				
数据类型	Array [1..512] OF BOOL*1			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

变量名称	_EC_MBXSlavTbl				
名称	可进行信息通信的从站表			全局 / 本地	全局
功能	可进行信息通信的从站一览。以从站节点地址顺序的表格表示。 从站变为可进行信息通信的状态 （可预操作、可安全操作、可操作）时，对应的位置变为 TRUE。 （注）与 EtherCAT 从站进行信息通信时，请参照本数据，确认是否可与相应的从站进行信息通信。				
数据类型	Array [1..512] OF BOOL*1			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 从站脱离 • EC_DisconnectSlave EtherCAT 从站再加入 • EC_ConnectSlave		

*1. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

变量名称	_EC_PDSlavTbl				
名称	过程数据通信中从站表			全局 / 本地	全局
功能	正在进行过程数据通信的从站一览。以从站节点地址顺序的表格表示。 从站的 IN、 OUT 均变为过程数据有效状态 （可操作）时，对应的位置变为 TRUE。 （注）控制 EtherCAT 从站时，请参照本数据，确认对应从站数据的有效性。				
数据类型	Array [1..512] OF BOOL*1			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 从站脱离 • EC_DisconnectSlave EtherCAT 从站再加入 • EC_ConnectSlave		

*1. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

变量名称	_EC_DisconnSlavTbl				
名称	脱离指令中的从站表			全局 / 本地	全局
功能	以从站节点地址顺序的表格表示。 脱离指令中的从站对应的位置变为 TRUE。				
数据类型	Array [1..512] OF BOOL*1			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 从站脱离 • EC_DisconnectSlave EtherCAT 从站再加入 • EC_ConnectSlave		

*1. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

变量名称	_EC_DisableSlavTbl				
名称	无效设定从站表			全局 / 本地	全局
功能	以从站节点地址顺序的表格表示。 设定为无效的从站对应的位置变为 TRUE。				
数据类型	Array [1..512] OF BOOL*1			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. NJ 系列 CPU 单元时，为 Array [1..192] OF BOOL。

变量名称	_EC_PDActive				
名称	过程数据通信状态			全局 / 本地	全局
功能	正在与所有从站*1 进行过程数据通信时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	EtherCAT 从站脱离 • EC_DisconnectSlave EtherCAT 从站再加入 • EC_ConnectSlave		

*1. 不含设定为无效的从站。

变量名称	_EC_PktMonStop				
名称	信息包监视停止			全局 / 本地	全局
功能	信息包监视停止时变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	信息包监视停止 • EC_StopMon 信息包监视启动 • EC_StartMon		

变量名称	_EC_LinkStatus				
名称	链路状态			全局 / 本地	全局
功能	通信控制器的链接状态为 Link ON 时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EC_PktSaving				
名称	信息包数据文件保存中			全局 / 本地	全局
功能	表示是否正在执行信息包数据文件保存处理。 TRUE：正在执行信息包数据文件保存处理 FALSE：未执行信息包数据文件保存处理				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	信息包数据文件保存 • EC_SaveMon		

变量名称	_EC_InDataInvalid				
名称	输入数据无效			全局 / 本地	全局
功能	主固定周期任务中执行的过程数据通信未正常进行，输入值不正确时，变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

（注） 与 EtherCAT 通信的状态相关的所有系统定义变量均表示当前的状态。

变量名称	_EC_InData1Invalid				
名称	输入数据 1 无效			全局 / 本地	全局
功能	主固定周期任务中执行的过程数据通信未正常进行，输入值不正确时，变为 TRUE。 （注） 本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

（注） 与 EtherCAT 通信的状态相关的所有系统定义变量均表示当前的状态。

变量名称	_EC_InData2Invalid				
名称	输入数据 2 无效			全局 / 本地	全局
功能	固定周期任务 （执行优先度 5）中执行的过程数据通信未正常进行，输入值不正确时，变为 TRUE。 （注） 本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

（注） 与 EtherCAT 通信的状态相关的所有系统定义变量均表示当前的状态。

● 功能分类：EtherCAT 通信的诊断统计信息相关

变量名称	_EC_StatisticsLogEnable*1				
名称	诊断统计信息日志许可			全局 / 本地	全局
功能	开始诊断统计信息日志功能时，从 FALSE 变为 TRUE。 结束诊断统计信息日志功能时，从 TRUE 变为 FALSE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	RW	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。

变量名称	_EC_StatisticsLogCycleSec*1				
名称	诊断统计信息日志保存周期			全局 / 本地	全局
功能	在诊断统计信息日志功能中，以秒为单位指定统计诊断信息的写入周期。 若指定为 “0”，则仅在结束诊断统计信息日志功能时写入 1 次。 （注）正在执行诊断统计信息日志功能时，即使变更本系统定义变量的值，正在执行的写入周期也不会发生变化。				
数据类型	UINT			值的范围	0、 30 ～ 1800
R/W 访问	RW	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。

变量名称	_EC_StatisticsLogBusy*1				
名称	诊断统计信息日志执行状态标志			全局 / 本地	全局
功能	正在执行诊断统计信息日志功能时变为 TRUE。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。

变量名称	_EC_StatisticsLogErr*1				
名称	诊断统计信息日志异常结束标志			全局 / 本地	全局
功能	开始诊断统计信息日志功能失败或执行过程中写入日志失败时，变为 TRUE。 开始诊断统计信息日志功能后，若 _EC_StatisticsLogBusy （诊断统计信息日志执行状态标志）变为 FALSE，则本标志的值确定。 异常结束可能有以下原因。 • SD 存储卡的容量达到上限，无法将记录添加到日志文件中。 • SD 存储卡有写保护。 • 未安装 SD 存储卡。 • 由于 _EC_StatisticsLogCycleSec （诊断统计信息日志保存周期）中指定的值不正确，因此无法开始。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. CPU 单元的单元版本 1.11 中追加的系统定义变量。

A-5-6 EtherNet/IP 功能模块（类别名称：_EIP）

● 功能分类：EtherNet/IP 通信的异常相关

变量名称	_EIP_ErrSta				
名称	内置 EtherNet/IP 异常			全局 / 本地	全局
功能	内置 EtherNet/IP 端口的异常状态变量。 NX 系列 CPU 单元时，集合了以下异常标志。 • _EIP1_PortErr （通信端口 1 异常） • _EIP2_PortErr （通信端口 2 异常） • _EIP1_CipErr （CIP 通信 1 异常） • _EIP2_CipErr （CIP 通信 2 异常） • _EIP_TcpAppErr （TCP 应用通信异常） NJ 系列 CPU 单元时，集合了以下异常标志。 • _EIP_PortErr （通信端口异常） • _EIP_CipErr （CIP 通信异常） • _EIP_TcpAppErr （TCP 应用通信异常） （注）各位的含义请参考 📖 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	在用户程序中，可通过以下专用指令访问。 • GetEIPError		

变量名称	_EIP_PortErr				
名称	通信端口异常			全局 / 本地	全局
功能	通信端口的异常状态变量。 NX 系列 CPU 单元时，集合了以下异常标志。 • _EIP1_MacAdrErr （通信端口 1MAC 地址异常） • _EIP1_LanHwErr （通信端口 1 通信控制器异常） • _EIP1_EtnCfgErr （通信端口 1 Ethernet 基本设定异常） • _EIP1_IPAdrCfgErr （通信端口 1 IP 地址设定异常） • _EIP1_IPAdrDupErr （通信端口 1 IP 地址重复异常） • _EIP1_BootpErr （通信端口 1 BOOTP 服务器连接异常） • _EIP_DNSCfgErr （DNS 设定异常） • _EIP_DNSSrvErr （DNS 服务器连接异常） • _EIP_IPRTblErr （IP 路由表异常） NJ 系列 CPU 单元时，集合了以下异常标志。 • _EIP_MacAdrErr （MAC 地址异常） • _EIP_LanHwErr （通信控制器异常） • _EIP_EtnCfgErr （Ethernet 基本设定异常） • _EIP_IPAdrCfgErr （IP 地址设定异常） • _EIP_IPAdrDupErr （IP 地址重复异常） • _EIP_BootpErr （BOOTP 服务器连接异常） • _EIP_DNSSrvErr （DNS 服务器连接异常） • _EIP_IPRTblErr （IP 路由表异常） （注）“ 链接断开检测 ” 或 “ 内置 EtherNet/IP 异常 ” 时，登录到事件日志中，本异常的位变为 ON。各位的含义请参考 📖 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	在用户程序中，可通过以下专用指令访问。 • GetEIPError		

变量名称	_EIP1_PortErr				
名称	通信端口 1 异常			全局 / 本地	全局
功能	通信端口 1 的异常状态变量。 集合了以下异常标志。 • _EIP1_MacAdrErr （通信端口 1MAC 地址异常） • _EIP1_LanHwErr （通信端口 1 通信控制器异常） • _EIP1_EtnCfgErr （通信端口 1 Ethernet 基本设定异常） • _EIP1_IPAdrCfgErr （通信端口 1 IP 地址设定异常） • _EIP1_IPAdrDupErr （通信端口 1 IP 地址重复异常） • _EIP1_BootpErr （通信端口 1 BOOTP 服务器连接异常） • _EIP_DNSCfgErr （DNS 设定异常） • _EIP_DNSSrvErr （DNS 服务器连接异常） • _EIP_IPRTblErr （IP 路由表异常） （注）“ 链接断开检测 ” 或 “ 内置 EtherNet/IP 异常 ” 时，登录到事件日志中，本异常的位变为 ON。各位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	在用户程序中，可通过以下专用指令访问。 • GetEIPError		

变量名称	_EIP2_PortErr				
名称	通信端口 2 异常			全局 / 本地	全局
功能	通信端口 2 的异常状态变量。 集合了以下异常标志。 • _EIP2_MacAdrErr （通信端口 1MAC 地址异常） • _EIP2_LanHwErr （通信端口 1 通信控制器异常） • _EIP2_EtnCfgErr （通信端口 1 Ethernet 基本设定异常） • _EIP2_IPAdrCfgErr （通信端口 1 IP 地址设定异常） • _EIP2_IPAdrDupErr （通信端口 1 IP 地址重复异常） • _EIP2_BootpErr （通信端口 1 BOOTP 服务器连接异常） • _EIP_DNSCfgErr （DNS 设定异常） • _EIP_DNSSrvErr （DNS 服务器连接异常） • _EIP_IPRTblErr （IP 路由表异常） （注）“ 链接断开检测 ” 或 “ 内置 EtherNet/IP 异常 ” 时，登录到事件日志中，本异常的位变为 ON。各位的含义请参考 「A-4-7 异常状态中各个位的含义」（P.A-78）。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	在用户程序中，可通过以下专用指令访问。 • GetEIPError		

变量名称	_EIP_CipErr				
名称	CIP 通信异常			全局 / 本地	全局
功能	CIP 通信的异常状态变量。 NX 系列 CPU 单元时，集合了以下异常标志。 • _EIP1_IdentityErr （CIP 通信 1Identity 信息不正确） • _EIP1_TDLinkCfgErr （CIP 通信 1 标签数据链接设定异常） • _EIP1_TDLinkOpnErr （CIP 通信 1 建立标签数据链接失败） • _EIP1_TDLinkErr （CIP 通信 1 标签数据链接通信异常） • _EIP1_TagAdrErr （CIP 通信 1 标签解决异常） • _EIP1_MultiSwONErr （CIP 通信 1 多个开关同时 ON 异常） NJ 系列 CPU 单元时，集合了以下异常标志。 • _EIP_IdentityErr （Identity 信息不正确） • _EIP_TDLinkCfgErr （标签数据链接设定异常） • _EIP_TDLinkOpnErr （建立标签数据链接失败） • _EIP_TDLinkErr （标签数据链接通信异常） • _EIP_TagAdrErr （标签解决异常） • _EIP_MultiSwONErr （多个开关同时 ON 异常） （注）“ 标签解决异常 ” 时登录到事件日志中，本异常变为 ON。各位的含义请参考 📖 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	在用户程序中，可通过以下专用指令访问。 • GetEIPError		

变量名称	_EIP1_CipErr				
名称	CIP 通信 1 异常			全局 / 本地	全局
功能	CIP 通信 1 的异常状态变量。 集合了以下异常标志。 • _EIP1_IdentityErr （CIP 通信 1Identity 信息不正确） • _EIP1_TDLinkCfgErr （CIP 通信 1 标签数据链接设定异常） • _EIP1_TDLinkOpnErr （CIP 通信 1 建立标签数据链接失败） • _EIP1_TDLinkErr （CIP 通信 1 标签数据链接通信异常） • _EIP1_TagAdrErr （CIP 通信 1 标签解决异常） • _EIP1_MultiSwONErr （CIP 通信 1 多个开关同时 ON 异常） （注）“ 标签解决异常 ” 时登录到事件日志中，本异常变为 ON。各位的含义请参考 📖 「A-4-7 异常状态中各个位的含义」（P.A-78）。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	在用户程序中，可通过以下专用指令访问。 • GetEIPError		

变量名称	_EIP2_CipErr				
名称	CIP 通信 2 异常			全局 / 本地	全局
功能	CIP 通信 2 的异常状态变量。 集合了以下异常标志。 • _EIP2_IdentityErr （CIP 通信 2Identity 信息不正确） • _EIP2_TDLinkCfgErr （CIP 通信 2 标签数据链接设定异常） • _EIP2_TDLinkOpnErr （CIP 通信 2 建立标签数据链接失败） • _EIP2_TDLinkErr （CIP 通信 2 标签数据链接通信异常） • _EIP2_TagAdrErr （CIP 通信 2 标签解决异常） • _EIP2_MultiSwONErr （CIP 通信 2 多个开关同时 ON 异常） （注）“ 标签解决异常 ” 时登录到事件日志中，本异常变为 ON。各位的含义请参考 📖 「A-4-7 异常状态中各个位的含义」（P.A-78）。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	在用户程序中，可通过以下专用指令访问。 • GetEIPError		

变量名称	_EIP_TcpAppErr				
名称	TCP 应用通信异常			全局 / 本地	全局
功能	TCP 应用通信的异常状态变量。 集合了以下异常标志。 • _EIP_TcpAppCfgErr （TCP 应用程序设定异常） • _EIP_NTPSrvErr （NTP 服务器连接异常） （注）各位的含义请参考 📖 「A-4-7 异常状态中各个位的含义」（P.A-78）。				
数据类型	WORD			值的范围	16#0000 ～ 16#00F0
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	在用户程序中，可通过以下专用指令访问。 • GetEIPError		

变量名称	_EIP_MacAdrErr				
名称	MAC 地址异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示电源 ON 时发生通信端口 1 的 MAC 地址读取异常。 TRUE：发生异常 FALSE：正常 NJ 系列 CPU 单元时，表示电源 ON 时发生 MAC 地址读取异常。 TRUE：发生异常 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_MacAdrErr				
名称	通信端口 1 MAC 地址异常			全局 / 本地	全局
功能	表示电源 ON 时发生通信端口 1 的 MAC 地址读取异常。 TRUE：发生异常 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_MacAdrErr				
名称	通信端口 2 MAC 地址异常			全局 / 本地	全局
功能	表示电源 ON 时发生通信端口 2 的 MAC 地址读取异常。 TRUE：发生异常 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_LanHwErr				
名称	通信控制器异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示通信端口 1 上发生通信控制器故障。 TRUE：发生故障 FALSE：正常 NJ 系列 CPU 单元时，表示发生通信控制器故障。 TRUE：发生故障 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_LanHwErr				
名称	通信端口 1 通信控制器异常			全局 / 本地	全局
功能	表示通信端口 1 上发生通信控制器故障。 TRUE：发生故障 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_LanHwErr				
名称	通信端口 2 通信控制器异常			全局 / 本地	全局
功能	表示通信端口 2 上发生通信控制器故障。 TRUE：发生故障 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_EtnCfgErr				
名称	Ethernet 基本设定异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示通信端口 1 的 Ethernet 通信速度设定 （Speed ／ Duplex）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 NJ 系列 CPU 单元时，表示 Ethernet 通信速度设定 （Speed ／ Duplex）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_EtnCfgErr				
名称	通信端口 1 Ethernet 基本设定异常			全局 / 本地	全局
功能	表示通信端口 1 的 Ethernet 通信速度设定 （Speed / Duplex）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_EtnCfgErr				
名称	通信端口 2 Ethernet 基本设定异常			全局 / 本地	全局
功能	表示通信端口 2 的 Ethernet 通信速度设定 （Speed / Duplex）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_IPAdrCfgErr				
名称	IP 地址设定异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示通信端口 1 的 IP 地址设定异常。 TRUE：•IP 地址的设定值不正确 •读取失败 •从 BOOTP 服务器获取的 IP 地址不匹配 FALSE：正常 NJ 系列 CPU 单元时，表示 IP 地址设定异常。 TRUE：•IP 地址的设定值不正确 •读取失败 •从 BOOTP 服务器获取的 IP 地址不匹配 •默认网关设定不正确 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_IPAdrCfgErr				
名称	通信端口 1 IP 地址设定异常			全局 / 本地	全局
功能	表示通信端口 1 的 IP 地址设定异常。 TRUE：•IP 地址的设定值不正确 •读取失败 •从 BOOTP 服务器获取的 IP 地址不匹配 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_IPAdrCfgErr				
名称	通信端口 2 IP 地址设定异常			全局 / 本地	全局
功能	表示通信端口 2 的 IP 地址设定异常。 TRUE：•IP 地址的设定值不正确 •读取失败 •从 BOOTP 服务器获取的 IP 地址不匹配 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_IPAdrDupErr				
名称	IP 地址重复异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示通信端口 1 上发生 IP 地址重复。 TRUE：发生重复 FALSE：上述以外 NJ 系列 CPU 单元时，表示发生 IP 地址重复。 TRUE：发生重复 FALSE：上述以外				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_IPAdrDupErr				
名称	通信端口 1 IP 地址重复异常			全局 / 本地	全局
功能	表示通信端口 1 上发生 IP 地址重复。 TRUE：发生重复 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_IPAdrDupErr				
名称	通信端口 2 IP 地址重复异常			全局 / 本地	全局
功能	表示通信端口 2 上发生 IP 地址重复。 TRUE：发生重复 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_DNSCfgErr*1				
名称	DNS 设定异常			全局 / 本地	全局
功能	DNS 设定值或 hosts 设定不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

*1. NJ 系列 CPU 单元中， CPU 单元版本为 1.11 以上的环境下可以使用。

变量名称	_EIP_BootpErr				
名称	BOOTP 服务器连接异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示通信端口 1 上发生 BOOTP 服务器连接失败。 TRUE：BOOTP 服务器的连接失败 （超时）。 FALSE：未进行 BOOTP 设定，或已设定 BOOTP，已正常从 BOOTP 服务器获取 IP 地址。 NJ 系列 CPU 单元时，表示发生 BOOTP 服务器连接失败。 TRUE：BOOTP 服务器的连接失败 （超时）。 FALSE：未进行 BOOTP 设定，或已设定 BOOTP，已正常从 BOOTP 服务器获取 IP 地址。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_BootpErr				
名称	通信端口 1 BOOTP 服务器连接异常			全局 / 本地	全局
功能	表示通信端口 1 上发生 BOOTP 服务器连接失败。 TRUE：BOOTP 服务器的连接失败 （超时）。 FALSE：未进行 BOOTP 设定，或已设定 BOOTP，已正常从 BOOTP 服务器获取 IP 地址。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_BootpErr				
名称	通信端口 2 BOOTP 服务器连接异常			全局 / 本地	全局
功能	表示通信端口 2 上发生 BOOTP 服务器连接失败。 TRUE：BOOTP 服务器的连接失败 （超时）。 FALSE：未进行 BOOTP 设定，或已设定 BOOTP，已正常从 BOOTP 服务器获取 IP 地址。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_IPRTblErr				
名称	IP 路由表异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示默认网关的设定或 IP 路由表的设定不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 NJ 系列 CPU 单元时，表示 IP 路由表的设定或 hosts 设定不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_IdentityErr				
名称	Identity 信息不正确			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示 CIP 通信 1 的 Identity 信息 （用户不可改写）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 NJ 系列 CPU 单元时，表示 Identity 信息 （用户不可改写）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_IdentityErr				
名称	CIP 通信 1 Identity 信息不正确			全局 / 本地	全局
功能	表示 CIP 通信 1 的 Identity 信息 （用户不可改写）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_IdentityErr				
名称	CIP 通信 2 Identity 信息不正确			全局 / 本地	全局
功能	表示 CIP 通信 2 的 Identity 信息 （用户不可改写）不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TDLinkCfgErr				
名称	标签数据链接设定异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示 CIP 通信 1 的标签数据链接设定值不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 NJ 系列 CPU 单元时，表示标签数据链接的设定值不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TDLinkCfgErr				
名称	CIP 通信 1 标签数据链接设定异常			全局 / 本地	全局
功能	表示 CIP 通信 1 的标签数据链接设定值不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TDLinkCfgErr				
名称	CIP 通信 2 标签数据链接设定异常			全局 / 本地	全局
功能	表示 CIP 通信 2 的标签数据链接设定值不正确。或者读取失败。 TRUE：不正确或读取失败 FALSE：正常 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TDLinkOpnErr				
名称	建立标签数据链接失败			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示建立 CIP 通信 1 的标签数据链接失败。 TRUE：因以下原因，导致建立标签数据链接失败。 •标签数据连接参数中登录的配对节点信息与实际的节点信息不同。 •配对节点无响应。 FALSE：上述以外 NJ 系列 CPU 单元时，表示建立标签数据链接失败。 TRUE：因以下原因，导致建立标签数据链接失败。 •标签数据连接参数中登录的配对节点信息与实际的节点信息不同。 •配对节点无响应。 FALSE：上述以外				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TDLinkOpnErr				
名称	CIP 通信 1 建立标签数据链接失败			全局 / 本地	全局
功能	表示建立 CIP 通信 1 的标签数据链接失败。 TRUE：因以下原因，导致建立标签数据链接失败。 •标签数据连接参数中登录的配对节点信息与实际的节点信息不同。 •配对节点无响应。 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TDLinkOpnErr				
名称	CIP 通信 2 建立标签数据链接失败			全局 / 本地	全局
功能	表示建立 CIP 通信 2 的标签数据链接失败。 TRUE：因以下原因，导致建立标签数据链接失败。 •标签数据连接参数中登录的配对节点信息与实际的节点信息不同。 •配对节点无响应。 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TDLinkErr				
名称	标签数据链接通信异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示建立 CIP 通信 1 的标签数据链接时发生超时。 TRUE：发生超时 FALSE：上述以外 NJ 系列 CPU 单元时，表示建立标签数据链接时发生超时。 TRUE：发生超时 FALSE：上述以外				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TDLinkErr				
名称	CIP 通信 1 标签数据链接通信异常			全局 / 本地	全局
功能	表示 CIP 通信 1 的标签数据链接发生超时。 TRUE：发生超时 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TDLinkErr				
名称	CIP 通信 2 标签数据链接通信异常			全局 / 本地	全局
功能	表示 CIP 通信 2 的标签数据链接发生超时。 TRUE：发生超时 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TagAdrErr				
名称	标签解决异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示解决 CIP 通信 1 的标签 （根据标签名称找出对应的地址）失败。 TRUE：标签解决 （根据标签名称找出对应的地址）失败。原因如下： •网络变量和标签设定之间的大小不同。 •标签数据链接的设定输入输出方向和 CPU 单元中的变量输入输出方向不一致。 •CPU 单元中，与标签设定对应的网络变量不存在。 FALSE：上述以外 NJ 系列 CPU 单元时，表示解决标签 （根据标签名称找出对应的地址）失败。 TRUE：标签解决 （根据标签名称找出对应的地址）失败。原因如下： •网络变量和标签设定之间的大小不同。 •标签数据链接的设定输入输出方向和 CPU 单元中的变量输入输出方向不一致。 •CPU 单元中，与标签设定对应的网络变量不存在。 FALSE：上述以外				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TagAdrErr				
名称	CIP 通信 1 标签解决异常			全局 / 本地	全局
功能	表示解决 CIP 通信 1 的标签 （根据标签名称找出对应的地址）失败。 TRUE：标签解决 （根据标签名称找出对应的地址）失败。原因如下： •网络变量和标签设定之间的大小不同。 •标签数据链接的设定输入输出方向和 CPU 单元中的变量输入输出方向不一致。 •CPU 单元中，与标签设定对应的网络变量不存在。 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TagAdrErr				
名称	CIP 通信 2 标签解决异常			全局 / 本地	全局
功能	表示解决 CIP 通信 2 的标签 （根据标签名称找出对应的地址）失败。 TRUE：标签解决 （根据标签名称找出对应的地址）失败。原因如下： •网络变量和标签设定之间的大小不同。 •标签数据链接的设定输入输出方向和 CPU 单元中的变量输入输出方向不一致。 •CPU 单元中，与标签设定对应的网络变量不存在。 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_MultiSwONErr				
名称	多个开关同时 ON 异常			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示 CIP 通信 1 中有多个开关同时变为 ON。 TRUE：有多个数据链接开始 / 数据链接停止开关同时变为 TRUE。 FALSE：上述以外 NJ 系列 CPU 单元时，表示有多个开关同时变为 ON。 TRUE：有多个数据链接开始 / 数据链接停止开关同时变为 TRUE。 FALSE：上述以外				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_MultiSwONErr				
名称	CIP 通信 1 多个开关同时 ON 异常			全局 / 本地	全局
功能	表示 CIP 通信 1 中有多个开关同时变为 ON。 TRUE：有多个数据链接开始 / 数据链接停止开关同时变为 TRUE。 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_MultiSwONErr				
名称	CIP 通信 2 多个开关同时 ON 异常			全局 / 本地	全局
功能	表示 CIP 通信 2 中有多个开关同时变为 ON。 TRUE：有多个数据链接开始 / 数据链接停止开关同时变为 TRUE。 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TcpAppCfgErr				
名称	TCP 应用程序设定异常			全局 / 本地	全局
功能	TRUE：至少有 1 个或以上 TCP 应用程序功能 （FTP、 NTP、 SNMP）的设定值不正确。或者读取失败。 FALSE：正常				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_NTPSrvErr				
名称	NTP 服务器连接异常			全局 / 本地	全局
功能	TRUE：NTP 客户端与服务器连接失败 （超时）。 FALSE：未设定 NTP。或者已设定 NTP 且连接成功。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_DNSSrvErr				
名称	DNS 服务器连接异常			全局 / 本地	全局
功能	TRUE：DNS 客户端与服务器连接失败 （超时）。 FALSE：未设定 DNS。或者已设定 DNS 且连接成功。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

● 功能分类：EtherNet/IP 通信的状态相关

变量名称	_EIP_EtnOnlineSta				
名称	在线			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示可使用内置 EtherNet/IP 端口中通信端口 1 的通信功能 （链接 ON、 IP 地址已确定。且未发生异常）。 TRUE：可使用 FALSE：因初始化处理中、重新启动中、发生异常、链接断开，导致内置 EtherNet/IP 端口的通信功能无法使用。 NJ 系列 CPU 单元时，表示可使用内置 EtherNet/IP 端口中通信端口的通信功能 （链接 ON、 IP 地址已确定。且未发生异常）。 TRUE：可使用 FALSE：因初始化处理中、重新启动中、发生异常、链接断开，导致内置 EtherNet/IP 端口的通信功能无法使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_EtnOnlineSta				
名称	通信端口 1 在线			全局 / 本地	全局
功能	表示可使用内置 EtherNet/IP 端口中通信端口 1 的通信功能 （链接 ON、 IP 地址已确定。且未发生异常）。 TRUE：可使用 FALSE：因初始化处理中、重新启动中、发生异常、链接断开，导致内置 EtherNet/IP 端口的通信功能无法使用。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_EtnOnlineSta				
名称	通信端口 2 在线			全局 / 本地	全局
功能	表示可使用内置 EtherNet/IP 端口中通信端口 2 的通信功能 （链接 ON、 IP 地址已确定。且未发生异常）。 TRUE：可使用 FALSE：因初始化处理中、重新启动中、发生异常、链接断开，导致内置 EtherNet/IP 端口的通信功能无法使用。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TDLinkRunSta				
名称	标签数据链接通信中			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示 CIP 通信 1 中至少有 1 个或以上的连接正在正常动作。 TRUE：正常动作 FALSE：上述以外 NJ 系列 CPU 单元时，表示至少有 1 个或以上的连接正在正常动作。 TRUE：正常动作 FALSE：上述以外				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TDLinkRunSta				
名称	CIP 通信 1 标签数据链接通信中			全局 / 本地	全局
功能	表示 CIP 通信 1 中至少有 1 个或以上的连接正在正常动作。 TRUE：正常动作 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TDLinkRunSta				
名称	CIP 通信 2 标签数据链接通信中			全局 / 本地	全局
功能	表示 CIP 通信 2 中至少有 1 个或以上的连接正在正常动作。 TRUE：正常动作 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TDLinkAllRunSta				
名称	所有标签数据链接通信中			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示 CIP 通信 1 的所有标签数据链接正在通信。 TRUE：表示作为始发端，所有连接正在进行标签数据链接通信。 FALSE：至少有 1 个或以上的连接发生异常。 NJ 系列 CPU 单元时，表示所有标签数据链接正在通信。 TRUE：表示作为始发端，所有连接正在进行标签数据链接通信。 FALSE：至少有 1 个或以上的连接发生异常。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TDLinkAllRunSta				
名称	CIP 通信 1 所有标签数据链接通信中			全局 / 本地	全局
功能	表示 CIP 通信 1 的所有标签数据链接正在通信。 TRUE：表示作为始发端，所有连接正在进行标签数据链接通信。 FALSE：至少有 1 个或以上的连接发生异常。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TDLinkAllRunSta				
名称	CIP 通信 2 所有标签数据链接通信中			全局 / 本地	全局
功能	表示 CIP 通信 2 的所有标签数据链接正在通信。 TRUE：表示作为始发端，所有连接正在进行标签数据链接通信。 FALSE：至少有 1 个或以上的连接发生异常。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_RegTargetSta ［255］				
名称	登录目标节点信息			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示在内置 EtherNet/IP 的 CIP 通信 1 中登录了连接的节点一览。 本变量仅在始发端为内置 EtherNet/IP 时有效。 排列 [x] 为 TRUE：已登录与目标节点 ID“x” 的节点之间的连接。 排列 [x] 为 FALSE：未登录与目标节点 ID“x” 的节点之间的连接。 NJ 系列 CPU 单元时，表示在内置 EtherNet/IP 中登录了连接的节点一览。 本变量仅在始发端为内置 EtherNet/IP 时有效。 排列 [x] 为 TRUE：已登录与目标节点 ID“x” 的节点之间的连接。 排列 [x] 为 FALSE：未登录与目标节点 ID“x” 的节点之间的连接。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_RegTargetSta ［255］				
名称	CIP 通信 1 登录目标节点信息			全局 / 本地	全局
功能	表示在内置 EtherNet/IP 的 CIP 通信 1 中登录了连接的节点一览。 本变量仅在始发端为内置 EtherNet/IP 时有效。 排列 [x] 为 TRUE：已登录与目标节点 ID“x” 的节点之间的连接。 排列 [x] 为 FALSE：未登录与目标节点 ID“x” 的节点之间的连接。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_RegTargetSta ［255］				
名称	CIP 通信 2 登录目标节点信息			全局 / 本地	全局
功能	表示在内置 EtherNet/IP 的 CIP 通信 2 中登录了连接的节点一览。 本变量仅在始发端为内置 EtherNet/IP 时有效。 排列 [x] 为 TRUE：已登录与目标节点 ID“x” 的节点之间的连接。 排列 [x] 为 FALSE：未登录与目标节点 ID“x” 的节点之间的连接。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_EstbTargetSta ［255］				
名称	正常目标节点信息			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示在内置 EtherNet/IP 的 CIP 通信 1 中已正常建立连接的节点一览。 排列 [x] 为 TRUE：已正常与目标节点 ID“x” 的节点建立连接。 排列 [x] 为 FALSE：未与目标节点 ID“x” 的节点建立连接或发生异常。 NJ 系列 CPU 单元时，表示在内置 EtherNet/IP 中已正常建立连接的节点一览。 排列 [x] 为 TRUE：已正常与目标节点 ID“x” 的节点建立连接。 排列 [x] 为 FALSE：未与目标节点 ID“x” 的节点建立连接或发生异常。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_EstbTargetSta ［255］				
名称	CIP 通信 1 正常目标节点信息			全局 / 本地	全局
功能	表示在内置 EtherNet/IP 的 CIP 通信 1 中已正常建立连接的节点一览。 排列 [x] 为 TRUE：已正常与目标节点 ID“x” 的节点建立连接。 排列 [x] 为 FALSE：未与目标节点 ID“x” 的节点建立连接或发生异常。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_EstbTargetSta ［255］				
名称	CIP 通信 2 正常目标节点信息			全局 / 本地	全局
功能	表示在内置 EtherNet/IP 的 CIP 通信 2 中已正常建立连接的节点一览。 排列 [x] 为 TRUE：已正常与目标节点 ID“x” 的节点建立连接。 排列 [x] 为 FALSE：未与目标节点 ID“x” 的节点建立连接或发生异常。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TargetPLCModeSta ［255］				
名称	目标 PLC 动作模式			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示在始发端为内置 EtherNet/IP 的 CIP 通信 1 的连接中，对应目标节点的控制器运行信息。 本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE：目标节点 ID“x” 的目标控制器正在运行的状态。 排列 [x] 为 FALSE：上述以外 NJ 系列 CPU 单元时，表示在始发端为内置 EtherNet/IP 的连接中，对应目标节点的控制器运行信息。 本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE：目标节点 ID“x” 的目标控制器正在运行的状态。 排列 [x] 为 FALSE：上述以外				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TargetPLCModeSta ［255］				
名称	CIP 通信 1 目标 PLC 动作模式			全局 / 本地	全局
功能	表示在始发端为内置 EtherNet/IP 的 CIP 通信 1 的连接中，对应目标节点的控制器运行信息。 本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE：目标节点 ID“x” 的目标控制器正在运行的状态。 排列 [x] 为 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TargetPLCModeSta ［255］				
名称	CIP 通信 2 目标 PLC 动作模式			全局 / 本地	全局
功能	表示在始发端为内置 EtherNet/IP 的 CIP 通信 2 的连接中，对应目标节点的控制器运行信息。 本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE：目标节点 ID“x” 的目标控制器正在运行的状态。 排列 [x] 为 FALSE：上述以外 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TargetPLCErr ［255］				
名称	目标 PLC 异常信息			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示在始发端为内置 EtherNet/IP 的 CIP 通信 1 的连接中，对应目标节点的控制器本体的异常状态 （运行停止、持续异常的逻辑或）。本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE：目标节点 ID“x” 的目标控制器继续运行或发生停止异常。 排列 [x] 为 FALSE：上述以外 NJ 系列 CPU 单元时，表示在始发端为内置 EtherNet/IP 的连接中，对应目标节点的控制器本体的异常状态 （运行停止、持续异常的逻辑或）。本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE：目标节点 ID“x” 的目标控制器继续运行或发生停止异常。 排列 [x] 为 FALSE：上述以外				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TargetPLCErr ［255］				
名称	CIP 通信 1 目标 PLC 异常信息			全局 / 本地	全局
功能	表示在始发端为内置 EtherNet/IP 的 CIP 通信 1 的连接中，对应目标节点的控制器本体的异常状态 （运行停止、持续异常的逻辑或）。本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE：目标节点 ID“x” 的目标控制器继续运行或发生停止异常。 排列 [x] 为 FALSE：上述以外 （注） 本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TargetPLCErr ［255］				
名称	CIP 通信 2 目标 PLC 异常信息			全局 / 本地	全局
功能	表示在始发端为内置 EtherNet/IP 的 CIP 通信 2 的连接中，对应目标节点的控制器本体的异常状态 （运行停止、持续异常的逻辑或）。本区域的信息仅在正常目标节点信息为 TRUE 时有效。若为 FALSE，则保持之前的值。 排列 [x] 为 TRUE：目标节点 ID“x” 的目标控制器继续运行或发生停止异常。 排列 [x] 为 FALSE：上述以外 （注） 本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TargetNodeErr ［255］				
名称	目标节点异常信息			全局 / 本地	全局
功能	NX 系列 CPU 单元时，表示与 CIP 通信 1 的登录目标节点信息对应的连接未建立或目标控制器发生异常。 本区域仅在登录目标节点信息为 TRUE 时有效。 排列 [x] 为 TRUE：未正常与目标节点 ID“x” 的目标节点建立连接 （登录目标节点信息为 TRUE，正常目标节点信息为 FALSE），或已与目标节点建立连接，但目标控制器发生异常。 排列 [x] 为 FALSE：目标节点 ID“x” 的目标节点未登录 （登录目标节点信息为 FALSE），或已与目标节点建立连接，（登录目标节点信息为 TRUE、正常目标节点信息为 TRUE），但目标控制器发生异常 （目标 PLC 异常信息为 TRUE）。 NJ 系列 CPU 单元时，表示与登录目标节点信息对应的连接未建立或目标控制器发生异常。 本区域仅在登录目标节点信息为 TRUE 时有效。 排列 [x] 为 TRUE：未正常与目标节点 ID“x” 的目标节点建立连接 （登录目标节点信息为 TRUE，正常目标节点信息为 FALSE），或已与目标节点建立连接，但目标控制器发生异常。 排列 [x] 为 FALSE：目标节点 ID“x” 的目标节点未登录 （登录目标节点信息为 FALSE），或已与目标节点建立连接，（登录目标节点信息为 TRUE、正常目标节点信息为 TRUE），但目标控制器发生异常 （目标 PLC 异常信息为 TRUE）。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TargetNodeErr ［255］				
名称	CIP 通信 1 目标节点异常信息			全局 / 本地	全局
功能	表示与 CIP 通信 1 的登录目标节点信息对应的连接未建立或目标控制器发生异常。 本区域仅在登录目标节点信息为 TRUE 时有效。 排列 [x] 为 TRUE：未正常与目标节点 ID“x” 的目标节点建立连接 （登录目标节点信息为 TRUE，正常目标节点信息为 FALSE），或已与目标节点建立连接，但目标控制器发生异常。 排列 [x] 为 FALSE：目标节点 ID“x” 的目标节点未登录 （登录目标节点信息为 FALSE），或已与目标节点建立连接，（登录目标节点信息为 TRUE、正常目标节点信息为 TRUE），但目标控制器发生异常 （目标 PLC 异常信息为 TRUE）。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TargetNodeErr ［255］				
名称	CIP 通信 2 目标节点异常信息			全局 / 本地	全局
功能	表示与 CIP 通信 2 的登录目标节点信息对应的连接未建立或目标控制器发生异常。 本区域仅在登录目标节点信息为 TRUE 时有效。 排列 [x] 为 TRUE：未正常与目标节点 ID“x” 的目标节点建立连接 （登录目标节点信息为 TRUE，正常目标节点信息为 FALSE），或已与目标节点建立连接，但目标控制器发生异常。 排列 [x] 为 FALSE：目标节点 ID“x” 的目标节点未登录 （登录目标节点信息为 FALSE），或已与目标节点建立连接，（登录目标节点信息为 TRUE、正常目标节点信息为 TRUE），但目标控制器发生异常 （目标 PLC 异常信息为 TRUE）。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	ARRAY [0..255] OF BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_NTPResult			成员名称	.ExecTime
名称	NTP 的最终动作时间			全局 / 本地	全局
功能	登录 NTP 正常结束的时间。 从 NTP 服务器正常获取时间后，登录从 NTP 服务器获取的时间。 若未从 NTP 服务器正常获取时间，则不登录。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。				
数据类型	结构体：_sNTP_RESULT 成员：DATE_AND_TIME			值的范围	遵照数据类型
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	可通过 GetNTPStatus 指令读取		

变量名称	_EIP_NTPResult			成员名称	.ExecNormal
名称	NTP 的动作结果			全局 / 本地	全局
功能	表示 NTP 是否正常结束。 TRUE：表示 NTP 正常结束。 FALSE：表示 NTP 异常结束或 NTP 一次都未执行。 （注）出于实时性的考虑，在用户程序中无法使用。仅在介由通信从外部参考状态时使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R	保持	非保持	网络公开	公开
是否可在用户程序中使用	不可	相关指令	可通过 GetNTPStatus 指令读取		

● 功能分类：EtherNet/IP 通信的开关相关

变量名称	_EIP_TDLinkStartCmd				
名称	标签数据链接通信开始开关			全局 / 本地	全局
功能	NX 系列 CPU 单元时，从 FALSE 变为 TRUE 后，开始 CIP 通信 1 的标签数据链接。 开始标签数据链接动作后，自动变为 FALSE。 NJ 系列 CPU 单元时，从 FALSE 变为 TRUE 后，开始标签数据链接。 开始标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或 Sysmac Studio 等强制操作本开关。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R/W	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TDLinkStartCmd				
名称	CIP 通信 1 标签数据链接通信开始开关			全局 / 本地	全局
功能	从 FALSE 变为 TRUE 后，开始 CIP 通信 1 的标签数据链接。 开始标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或 Sysmac Studio 等强制操作本开关。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R/W	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TDLinkStartCmd				
名称	CIP 通信 2 标签数据链接通信开始开关			全局 / 本地	全局
功能	从 FALSE 变为 TRUE 后，开始 CIP 通信 2 的标签数据链接。 开始标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或 Sysmac Studio 等强制操作本开关。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R/W	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP_TDLinkStopCmd				
名称	标签数据链接通信停止开关			全局 / 本地	全局
功能	NX 系列 CPU 单元时，从 FALSE 变为 TRUE 后，停止 CIP 通信 1 的标签数据链接。 停止标签数据链接动作后，自动变为 FALSE。 NJ 系列 CPU 单元时，从 FALSE 变为 TRUE 后，停止标签数据链接。 停止标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或 Sysmac Studio 等强制操作本开关。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R/W	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP1_TDLinkStopCmd				
名称	CIP 通信 1 标签数据链接通信停止开关			全局 / 本地	全局
功能	从 FALSE 变为 TRUE 后，停止 CIP 通信 1 的标签数据链接。 停止标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或 Sysmac Studio 等强制操作本开关。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R/W	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

变量名称	_EIP2_TDLinkStopCmd				
名称	CIP 通信 2 标签数据链接通信停止开关			全局 / 本地	全局
功能	从 FALSE 变为 TRUE 后，停止 CIP 通信 2 的标签数据链接。 停止标签数据链接动作后，自动变为 FALSE。 （注）自动变为 FALSE 前，请勿通过用户程序或 Sysmac Studio 等强制操作本开关。 （注）本系统定义变量仅可在 NX 系列 CPU 单元中使用。				
数据类型	BOOL			值的范围	TRUE、 FALSE
R/W 访问	R/W	保持	非保持	网络公开	公开
是否可在用户程序中使用	可以	相关指令	—		

A-6 CPU 单元内数据的属性

断电时、电源接通时、动作模式变更时、发生全部停止等级的控制器异常时， CPU 单元内数据的保持 / 非保持属性如下。

CPU 单元内数据项目			断电时的保持有无和方法 〇 : 断电保持 ×: 断电非保持	电源接通时	状态变更		覆盖保护中的写入	Sysmac Studio 的传送功能	可写入的动作模式	运行模式下的重写
					程序模式 ←→ 运行模式变更时	发生全部停止故障等级的控制器异常时		同步功能的对象		
用户程序	POUs、用户程序执行用 ID （用户程序内）		〇（通过非易失性存储器）	与断电前一样	保持	保持	不可	〇	程序模式 / 运行模式 （在线编辑）	〇 在线编辑
任务设定	任务设置		〇（通过非易失性存储器）	与断电前一样	保持	保持	不可	〇	程序模式	×
变量	变量表（变量的值除外）	设备变量	〇（通过非易失性存储器）	与断电前一样	保持	保持	不可	〇	程序模式	×
		用户定义变量	〇（通过非易失性存储器）	与断电前一样	保持	保持	不可	〇	程序模式 / 运行模式 （在线编辑）	〇 在线编辑
数据类型		用户定义数据类型	〇（通过非易失性存储器）	与断电前一样	保持	保持	不可	〇	程序模式 / 运行模式 （在线编辑）	〇 在线编辑
控制器名称		CPU 单元名称	〇（通过非易失性存储器）	与断电前一样	保持	保持	不可	〇	程序模式 / 运行模式	〇
		内置 EtherNet/IP 端口名称	〇（通过非易失性存储器）	与断电前一样	保持	保持	可	×	程序模式 / 运行模式	〇

CPU 单元内数据项目			断电时的保持有无和方法 ○：断电保持 ×：断电非保持	电源接通时	状态变更		覆盖保护中的写入	Sysmac Studio 的传送功能	可写入的动作模式	运行模式下的重写
					程序模式 ←→ 运行模式变更时	发生全部停止故障等级的控制器异常时		同步功能的对象		
控制器设定	动作设定	动作设定 异常设定	○（通过非易失性存储器）	与断电前一样	保持	保持	不可	○	CPU 单元名称：运行模式 / 程序模式 其他：程序模式	×
	安全设定	电源接通时的保护设定	○（通过非易失性存储器）	与断电前一样	保持	保持	不可	○	写保护，其他：程序模式	○
	内置 EtherNet/IP 端口设定	TCP/IP 内置 EtherNet/IP 端口 LINK 设定 服务设定 SNMP 设定 SNMP 捕捉设定 NTP 设定 FTP 设定 IP 路由表	○（通过非易失性存储器）	与断电前一样	保持	保持	不可	○	程序模式	×
		内置 EtherNet/IP 端口用标签数据链接设定	○（通过非易失性存储器）	与断电前一样	保持	保持	可	×	程序模式 / 运行模式	×
	FINS 设定	节点地址设定 FINS/UDP FINS/TCP FINS 路由表	○（通过非易失性存储器）	与断电前一样	保持	保持	不可	○	程序模式	×
运动设定	轴分配 轴参数设定 轴组参数设定 MC 共通参数设定		○（通过非易失性存储器）	与断电前一样	保持	保持	不可	○	程序模式	×
凸轮数据			○（通过非易失性存储器）	与断电前一样	保持	保持	不可	○	程序模式	×
事件设定表	事件设定表	用户异常消息	○（通过非易失性存储器）	与断电前一样	保持	保持	不可	○	运行模式 / 程序模式	×
总线构成	CJ 总线构成	I/O 表	○（通过非易失性存储器）	与断电前一样	保持	保持	不可	○	程序模式	×

CPU 单元内数据项目			断电时的保持有无和方法 ○：断电保持 ×：断电非保持	电源接通时	状态变更		覆盖保护中的写入	Sysmac Studio 的传送功能	可写入的动作模式	运行模式下的重写
					程序模式 ⟷ 运行模式变更时	发生全部停止故障等级的控制器异常时		同步功能的对象		
高功能 I/O 单元设定 / CPU 总线单元设定	CJ 单元设定	CJ 单元内例）协议宏	○ 取决于 CJ 单元	—	保持	保持	可	×	因单元而异	
		CPU 总线单元区域 例）Controller Link 数据链接表	○ （通过非易失性存储器）	与断电前一样	保持	保持	可	×	运行模式 / 程序模式	○
		分配 DM 区域	○ （通过电池）	与断电前一样	保持	保持	可	○	运行模式 / 程序模式	○
EtherCAT 构成	EtherCAT 网络配置	网络配置信息	○ （通过非易失性存储器）	与断电前一样	保持	保持	不可	○	程序模式	×
EtherCAT 设定	EtherCAT 设定	主机	○ （通过非易失性存储器）	与断电前一样	保持	保持	不可	○	程序模式	×
		从站自身的设定	○ 取决于从站	—	保持	保持	可	○	运行模式 / 程序模式	○
操作权限认证			○ （通过非易失性存储器）	与断电前一样	保持	保持	不可	×	程序模式	×
用户程序执行用 ID（CPU 单元内）			○ （通过非易失性存储器）	与断电前一样	保持	保持	不可	×	程序模式	×
变量的值	非保持变量的值	用户定义变量 / 设备变量	×	初始值	初始值	初始值	可	×	运行模式 / 程序模式	○
	保持变量的值	用户定义变量 / 设备变量	○ （通过电池）	与断电前一样	保持	保持	可	×	运行模式 / 程序模式	○
CJ 单元用存储器的值	CIO/WR		×	16#0000	16#0000	16#0000	可	×	运行模式 / 程序模式	○
	HR/DM/EM		○ （通过电池）	与断电前一样	保持	保持	可	×	运行模式 / 程序模式	○
事件日志	履历	系统日志 用户日志	○ （通过电池）	与断电前一样	保持	保持	可	×		○
内置时钟	系统时钟的当前时刻		○ （通过电池）	有电池时：保持（继续） 无电池时：不确定 （也可能停止）	保持（继续）	保持（继续）	可	×	运行模式 / 程序模式	○
绝对编码器原点位置偏置			○ （通过电池）	与断电前一样	保持（继续）	保持（继续）	可	×		×

A-7 CJ 单元用存储器的详情

CJ 单元用存储器为变量的 AT （分配对象）指定对象。
以下表示各区域类型的详情。
（注）NX 系列 CPU 单元中，不可使用 CJ 单元用存储器。

A-7-1 CIO 区域

输入输出继电器

● **说明**

可分配 CJ 系列基本 I/O 单元的输入输出端子的继电器区域。
根据连接位置（从距离 CPU 单元最近的位置从左往右），各 CJ 系列基本 I/O 单元的占用通道数以追加编号且以 16 点为单位进行分配。
断电恢复时 （ON → OFF → ON）、动作模式变更 （程序模式←→运行模式）时，将清除。

● **地址**

地址	通道地址	位地址
范围	0 ～ 159CH	0.00 ～ 159.15

参考

NJ 系列 CPU 单元中，通过 I/O 端口中分配的设备变量访问本输入输出继电器区域。所以，不建议使用 AT （分配对象）指定。
仅在部分高功能单元的分配中指定了任意区域时，对本输入输出继电器区域进行 AT （分配对象）指定。

CPU 总线单元继电器

● **说明**

可分配 CJ 系列 CPU 总线单元的状态信息的继电器区域。每个单元 25CH，根据单元编号分配。
断电恢复时 （ON → OFF → ON）、动作模式变更 （程序模式←→运行模式）时，将清除。

● **地址**

地址	通道地址	位地址	每 1 个单元
范围	1500 ～ 1899CH	1500.00 ～ 1899.15	25CH

分配区域如下所示。

通道编号	单元编号
1500 ～ 1524CH	0
1525 ～ 1549CH	1
≀	≀
1875 ～ 1899CH	F

本区域的详细使用方法请参考各 CPU 总线单元的用户手册。

参考

NJ 系列 CPU 单元中，通过 I/O 端口中分配的设备变量访问本 CPU 总线单元继电器区域。所以，不建议使用 AT （分配对象）指定。
仅在部分高功能单元的分配中指定了任意区域时，对本输入输出继电器区域进行 AT （分配对象）指定。

高功能 I/O 单元继电器

● 说明

可分配 CJ 系列高功能 I/O 单元的状态信息的继电器区域。每个单元 10CH，总共 96 个单元（0 ～ 95 号机），根据号机 No. 分配。
断电恢复时 （ON ➞ OFF ➞ ON）、动作模式变更 （程序模式←➞运行模式）时，将清除。

● 地址

地址	通道地址	位地址	每 1 个单元
范围	2000 ～ 2959CH （10CH×96 号机）	2000.00 ～ 2959.15	10CH

分配区域如下所示。

通道编号	单元编号
2000 ～ 2009CH	0 号机
2010 ～ 2019CH	1 号机
≀	≀
2950 ～ 2959CH	95 号机

本区域的详细使用方法请参考各高功能 I/O 单元的用户手册。

参考

NJ 系列 CPU 单元中，通过 I/O 端口中分配的设备变量访问本高功能 I/O 单元继电器区域。所以，不建议使用 AT （分配对象）指定。

DeviceNet 继电器

● 说明

使用 DeviceNet 单元的远程 I/O 主机功能时，分配各从站的继电器区域 （固定分配时）。
断电恢复时 （ON ➞ OFF ➞ ON）、动作模式变更 （程序模式←➞运行模式）时，将清除。

● 地址

地址	通道地址	位地址
范围	3200 ～ 3799CH	3200.00 ～ 3799.15

利用分配继电器区域软开关中的固定分配区域设定 1/2/3 开关，可选择以下固定分配区域 1/2/3 中的任意一个。

地址	主机➞从站输出 （OUT） 区域	从站➞主机输入 （IN） 区域
固定分配区域 1	3200 ～ 3263CH	3300 ～ 3363CH
固定分配区域 2	3400 ～ 3463CH	3500 ～ 3563CH
固定分配区域 3	3600 ～ 3663CH	3700 ～ 3763CH

以固定分配使用 DeviceNet 单元的远程 I/O 从站功能时，也将分配到本 DeviceNet 继电器中。

地址	主机➞从站输出 （OUT） 区域	从站➞主机输入 （IN） 区域
固定分配区域 1	3370CH	3270CH
固定分配区域 2	3570CH	3470CH
固定分配区域 3	3770CH	3670CH

详情请参考📖《CJ 系列 DeviceNet 单元 用户手册 NJ 系列连接篇 （SBCD-CN5-357）》。

CIO 区域的内部辅助继电器

● 说明

仅在程序上可使用的继电器。不可与外部输入输出端子进行输入输出。内部辅助继电器基本上请通过本区域使用。
断电恢复时 （ON ➞ OFF ➞ ON）、动作模式变更 （程序模式←➞运行模式）时，将清除。

● 地址

地址	通道地址	位地址
范围	1300 ～ 1499CH、 3800 ～ 6143CH	1300.00 ～ 1499.15、 3800.00 ～ 6143.15

A-7-2 内部辅助继电器

● 说明

仅在程序上可使用的继电器。不可与外部输入输出端子进行输入输出。内部辅助继电器基本上请通过本区域使用。
断电恢复时 （ON ➞ OFF ➞ ON）、动作模式变更 （程序模式←→运行模式）时，将清除。

● 地址

地址	通道地址	位地址
范围	W0 ～ W511CH	W0.00 ～ W511.15

A-7-3 保持继电器

● 说明

该继电器仅在程序上可使用，断电恢复时 （ON ➞ OFF ➞ ON）或动作模式变更 （程序模式←→运行模式）时，仍保持断电前或模式变更前的 ON/OFF 状态。

● 地址

地址	通道地址	位地址
范围	H0 ～ H511CH	H0.00 ～ H511.15

A-7-4 数据存储器

● 说明

以字节 （16 位）为单位进行读写的通用数据区域。
此外，在地址上添加位编号后，可以以位为单位进行指定 （读写）。
断电恢复时 （ON ➞ OFF ➞ ON）或动作模式变更 （程序模式←→运行模式）时，仍保持断电前或模式变更前的数据。

● 地址

地址	通道地址	位地址
范围	D0 ～ D32767	D0.00 ～ D32767.15

高功能单元用 DM 区域

● 说明

在数据存储器区域中，以下区域用于分配高功能单元的初始设定数据等。

● 地址

地址	CJ 系列高功能单元的种类	通道地址	每 1 个单元
范围	CJ 系列高功能 I/O 单元	D20000 ～ 29599（100CH×96 号机）	100CH
	CJ 系列 CPU 高功能单元	D30000 ～ 31599（100CH×16 单元编号）	100CH

分配区域如下所示。

CJ 系列高功能 I/O 单元

通道编号	号机 No.
D20000 ～ D20099	0 号机
D20100 ～ D20199	1 号机
≀	≀
D29500 ～ D29599	95 号机

CJ 系列 CPU 总线单元

通道编号	单元编号
D30000 ～ D30099	0
D30100 ～ D30199	1
≀	≀
D31500 ～ D31599	F

本区域的详细使用方法请参考各高功能单元的用户手册。

参考

NJ 系列 CPU 单元中，通过 I/O 端口中分配的设备变量访问本高功能单元用 DM 区域。所以，不建议使用 AT （分配对象）指定。

A-7-5 扩展数据存储器

● 说明

以字节 （16 位）为单位进行读写的通用数据区域。

此外，在地址上添加位编号后，可以以位为单位进行指定 （读写）。

断电恢复时 （ON ➞ OFF ➞ ON）或动作模式变更 （程序模式←➞运行模式）时，仍保持断电前或模式变更前的数据。

● 地址

地址	通道地址	位地址
范围	NJ501- □□□□：E0_0 ～ E18_32767 NJ301- □□□□：E0_0 ～ E3_32767 NJ101- □□□□：E0_0 ～ E3_32767	NJ501- □□□□：E0_0.00 ～ E18_3276.15 NJ301- □□□□：E0_0.00 ～ E3_32767.15 NJ101- □□□□：E0_0.00 ～ E3_32767.15

（注）BANK 数以 16 进制数表示。

A-8 变量的存储器确保方法

将结构体型或联合体型变量成员在存储器上的位置与其他设备统一时，需要注意变量的存储器确保方法。主要是在使用结构体型和联合体型变量与其他设备进行以下通信时，需要调整。

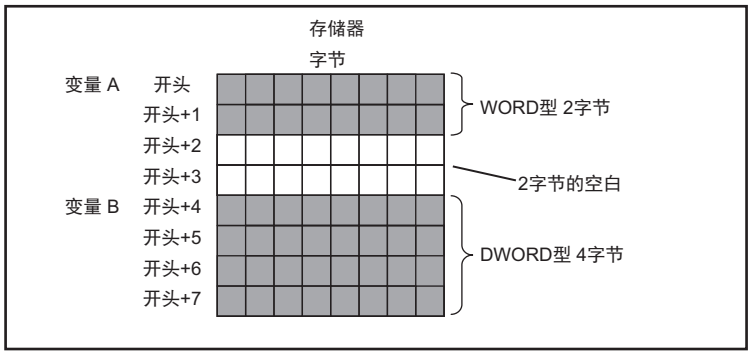
- NJ/NX系列CPU单元和其他CPU单元之间，进行EtherNet/IP的标签数据链接或通过CIP消息访问变量时
- 通过结构体型、联合体型变量与 ID 标签等 CPU 单元以外的设备进行数据交换时

A-8-1 变量的存储器确保规则

为变量保留的存储器量和存储器上的位置因变量的数据类型而变化。此外，为排列的要素、结构体的成员、联合体的成员保留的存储器量和存储器上的位置，除了数据类型之外，还因排列、结构体、联合体的声明内容而变化。

数据类型的校准和保留的存储器量

不同的数据类型有规定的数据大小。数据大小是指保存该数据类型的值需要的最小存储器容量。
另外，为了最有效地访问存储器，与变量对应的存储器由控制器自动配置。因此，为变量保留的存储器总量和这些变量的数据大小总和肯定不一致。例如，声明了 WORD 型和 DWORD 型变量时，数据大小总和为 6 字节，但存储器应如下图所示保留 8 字节。



变量表

名称	数据类型
A	WORD
B	DWORD

这种决定存储器上变量配置的信息称为校准。校准根据数据类型而定，为具体变量保留的存储器量和存储器上的位置如下所示。

项目	规格
保留的存储器量	校准的整数倍。但应大于数据大小。
存储器上的位置	从存储器起点起，校准的整数倍位置。

基本数据型和列举型的校准和保留的存储器量如下所示。

数据类型	校准 [字节]	保留的存储器量 [字节]
BOOL	2	2
BYTE,USINT,SINT	1	1
WORD,UINT,INT	2	2
DWORD,UDINT,DINT	4	4
LWORD,ULINT,LINT	8	8
REAL	4	4
LREAL	8	8
TIME,DATE,TIME_OF_DAT,DATE_ANT_TIME	8	8
STRING[N+1][*1]	1	N+1
列举型	4	4

*1. N 表示处理的最大字符数。例如，半角英文或数字的最大字符数为 10 时，字符串末尾附加 NULL，需要保留 11 个字符的存储器。

为了最有效地访问，排列、结构体型、联合体型的各要素和各成员也分别配置到存储器上。排列、结构体型、联合体型的校准和保留的存储器量根据其声明确定如下。

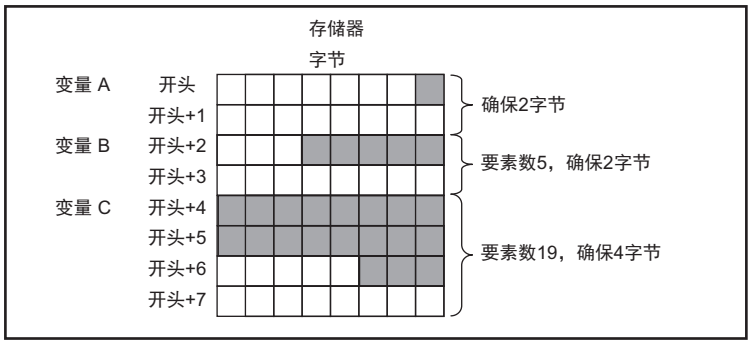
数据类型	校准	保留的存储器量
排列	与要素的数据类型的校准相同	（为要素的数据类型保留的存储器量）× 要素数[*1]
结构体型	所有成员的校准中，最大的值	将各成员依次配置到该成员数据类型校准的整数倍位置时保留的存储器的总量以上且校准的整数倍
联合体型	所有成员的校准中，最大的值	所有成员保留的存储器量中，最大的值

*1. BOOL 型排列为例外。为 BOOL 型排列保留的存储器量请参考后面的使用注意事项。

使用注意事项

为 BOOL 型排列保留的存储器量

为 BOOL 型的单独变量、结构体的成员、联合体的成员保留的存储器量为 2 字节。
但是， BOOL 型排列并非每 1 个要素确保 2 字节的存储器。各要素对应的存储器 1 位 1 位排列。排列整体保留 2 字节整数倍的存储空间，包括空格在内。



变量表

名称	数据类型
A	BOOL
B	ARRAY［1..5］OF BOOL
C	ARRAY［0..18］OF BOOL

因此，为 BOOL 型排列保留的存储器量可按以下公式计算。要素数为 1 ～ 16 时，保留 2 字节，17 ～ 32 时，保留 4 字节。

$$\text{存储器的量}=2\left[\frac{\text{要素数}-1}{16}\right]+2$$

括号内的计算结果中，小数点以下舍去

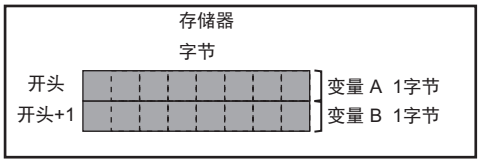
下面以具体案例为例，介绍各数据类型的存储器保留规则。

基本数据型

● BYTE 型等校准为 1 字节的变量

以 1 字节为单位的校准，保留 1 字节的存储器。

例） BYTE 型变量为连续 2 个时



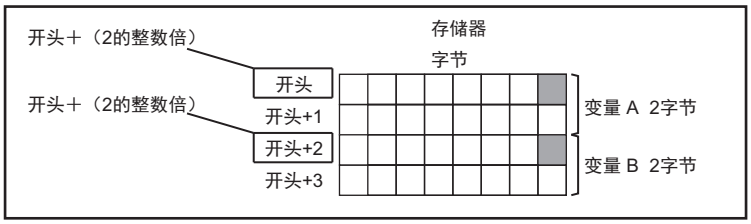
变量表

名称	数据类型
A	BYTE
B	BYTE

● BOOL 型、 WORD 型等校准为 2 字节的变量

以 2 字节为单位的校准，保留 2 字节的存储器。

例） BOOL 型变量为连续 2 个时



变量表

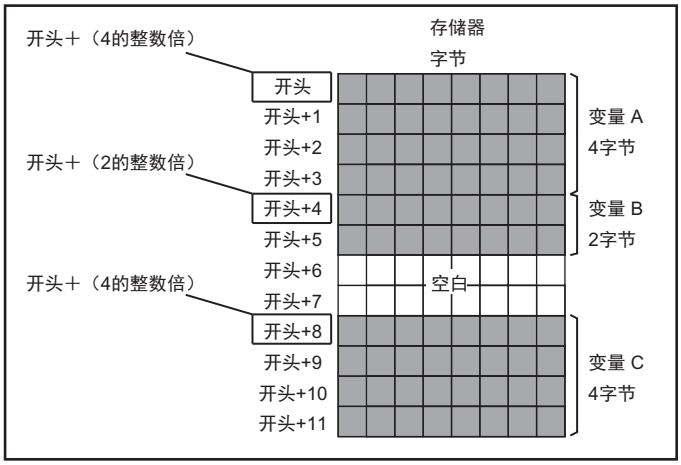
名称	数据类型
A	BOOL
B	BOOL

● DWORD 型等校准为 4 字节的变量

以 4 字节为单位的校准，保留 4 字节的存储器。

配置了数据了存储器开头位置为 4 字节的整数倍，因此若插入 WORD 型等校准为 2 字节的变量，将出现 2 字节的空白。

例）变量按 DWORD 型、 WORD 型、 DWORD 型的顺序连续时



变量表

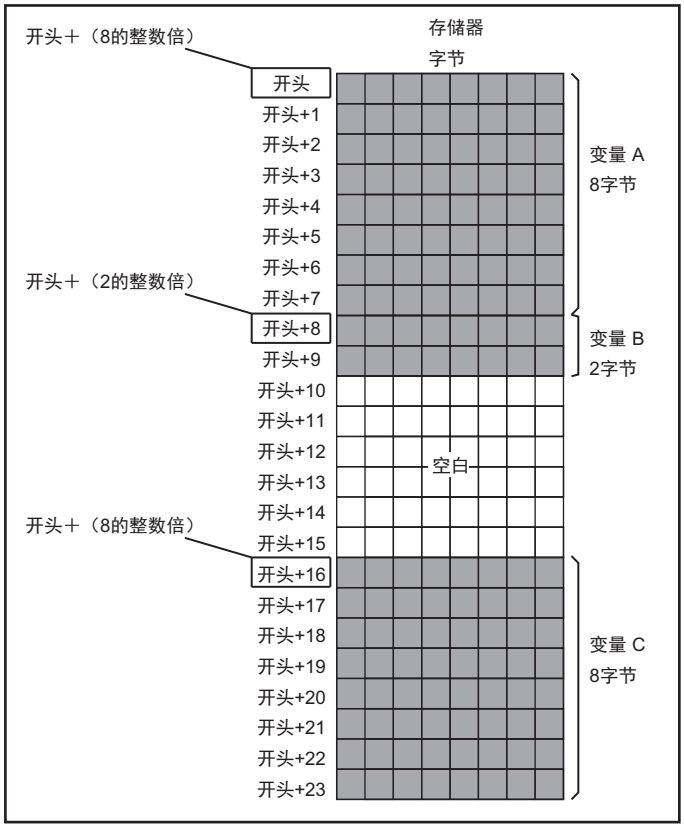
名称	数据类型
A	DWORD
B	WORD
C	DWORD

● LWORD 型等校准为 8 字节的变量

以 8 字节为单位的校准，保留 8 字节的存储器。

配置了数据了存储器开头位置为 8 字节的整数倍，因此若插入 WORD 型等校准为 2 字节的变量，将出现 6 字节的空白，若插入 DWORD 型等校准为 4 字节的变量，将出现 4 字节的空白。

例）变量按 LWORD 型、 WORD 型、 LWORD 型的顺序连续时



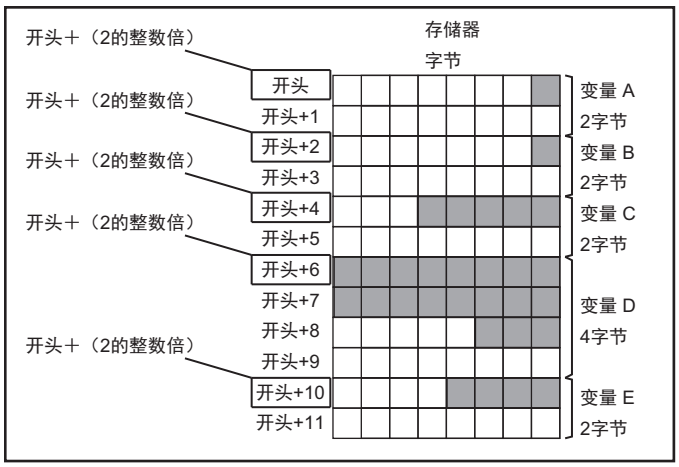
变量表

名称	数据类型
A	LWORD
B	WORD
C	LWORD

排列

排列变量以排列要素数据类型的数据大小为单位，按照要素数连续保留存储空间。排列的校准与要素的数据类型的校准相同。

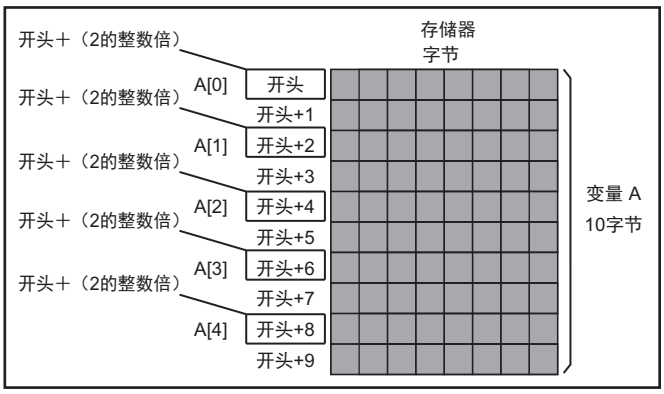
例）变量按照 BOOL 型变量 2 个、要素数 5 的 BOOL 型排列、要素数 19 的 BOOL 型排列、要素数 4 的 BOOL 型排列的顺序连续时



变量表

名称	数据类型
A	BOOL
B	BOOL
C	ARRAY［1..5］OF BOOL
D	ARRAY［0..18］OF BOOL
E	ARRAY［5..8］OF BOOL

例）要素数 5 的 INT 型排列时



变量表

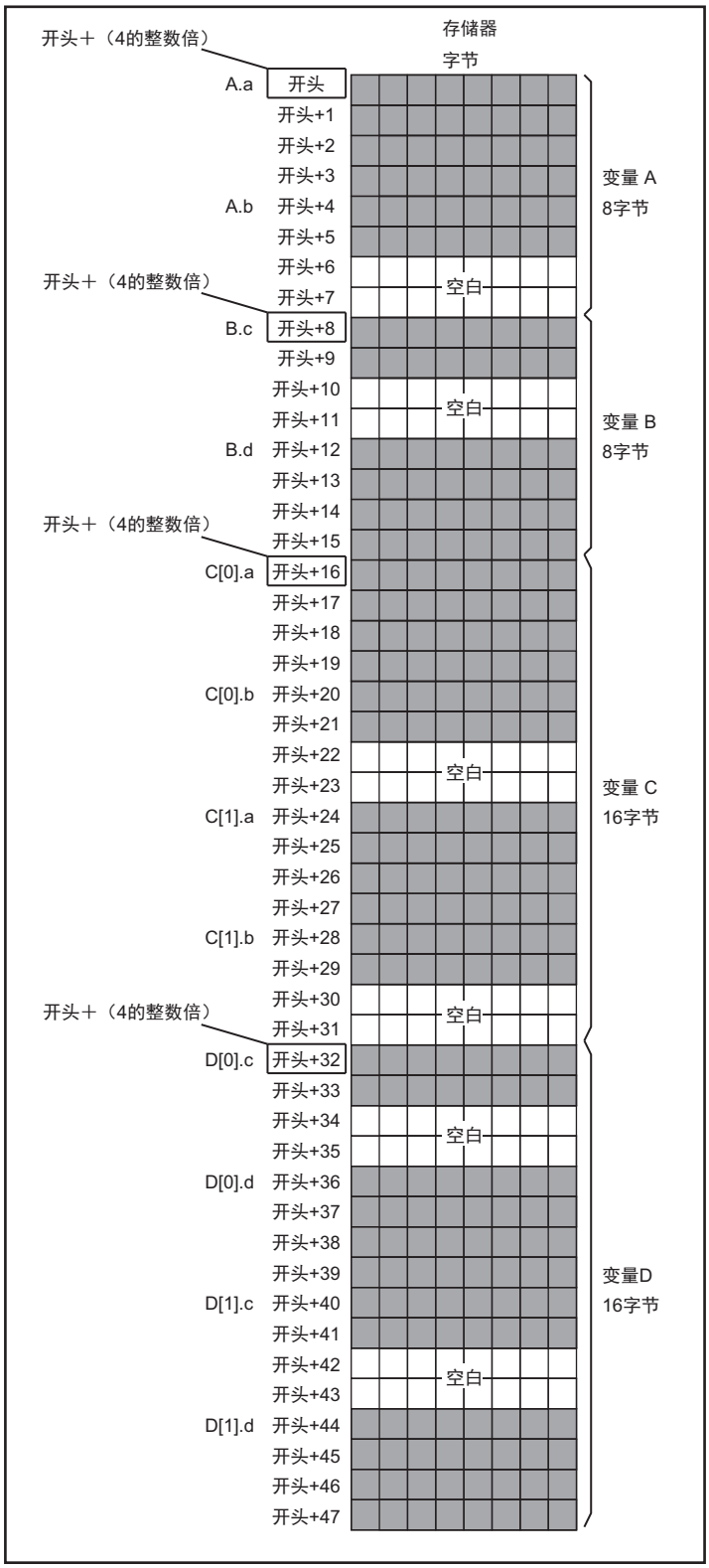
名称	数据类型
变量 A	ARRAY[0..4] OF INT

结构体型

结构体变量按照各成员声明的顺序配置到存储器中。或者将各成员依次配置到该成员数据类型校准的整数倍位置。因此，成员之间或成员末尾可能会插入存储器的空白。结构体型校准为所有成员的校准中最大的值。保留的存储器量：将各成员依次配置到该成员数据类型校准的整数倍位置时保留的存储器的总量以上且校准的整数倍。

例）下图 4 种变量声明时，各自的校准和保留的存储器量为以下值。

变量	校准 [字节]	保留的存储器量 [字节]
A	4	8
B	4	8
C	4	16
D	4	16



数据类型定义

名称	数据类型
结构体 STR_A	STRUCT
a	DINT
b	INT

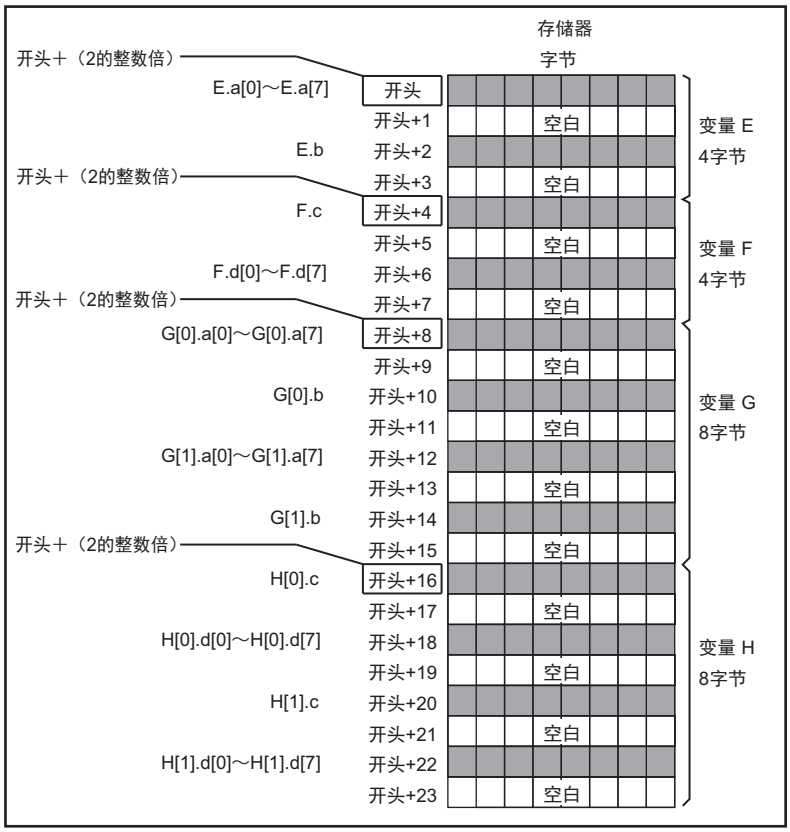
名称	数据类型
结构体 STR_B	STRUCT
c	INT
d	DINT

变量表

名称	数据类型
变量 A	结构体 STR_A
变量 B	结构体 STR_B
变量 C	ARRAY[0..1] OF STR_A
变量 D	ARRAY[0..1] OF STR_B

例）下图 4 种变量声明时，各自的校准和保留的存储器量为以下值。

变量	校准 [字节]	保留的存储器量 [字节]
E	2	4
F	2	4
G	2	8
H	2	8



数据类型定义

名称	数据类型
结构体 STR_C	STRUCT
a	ARRAY[0..7] OF BOOL
b	BYTE

名称	数据类型
结构体 STR_D	STRUCT
c	BYTE
d	ARRAY[0..7] OF BOOL

变量表

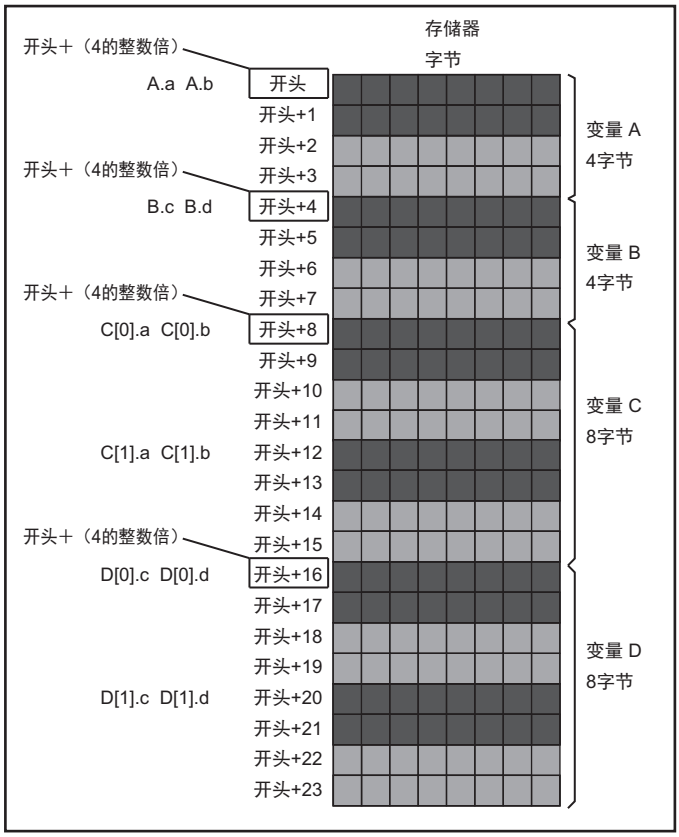
名称	数据类型
变量 E	结构体 STR_C
变量 F	结构体 STR_D
变量 G	ARRAY[0..1] OF STR_C
变量 H	ARRAY[0..1] OF STR_D

联合体型

联合体型变量的各成员重叠配置到存储器上相同的位置。
联合体型校准为所有成员的校准中最大的值。保留的存储器量为所有成员保留的存储器量中最大的值。

例）下图 4 种变量声明时，各自的校准和保留的存储器量为以下值。

变量	校准 [字节]	保留的存储器量 [字节]
A	4	4
B	4	4
C	4	8
D	4	8



数据类型定义

名称	数据类型
联合体 UNI_A	UNION
a	DWORD
b	WORD

名称	数据类型
联合体 UNI_B	UNION
c	WORD
d	DWORD

变量表

名称	数据类型
变量 A	联合体 UNI_A
变量 B	联合体 UNI_B
变量 C	ARRAY[0..1] OF UNI_A
变量 D	ARRAY[0..1] OF UNI_B

A-8-2 应注意的案例

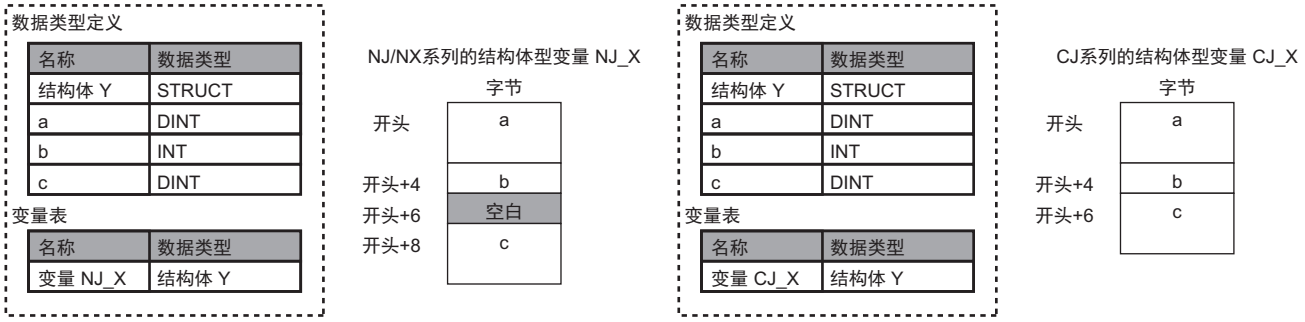
NJ/NX 系列 CPU 单元的结构体型变量，通过与其他设备的通信进行数据交换时，结构体型变量的成员的存储器配置需要与进行通信的对象设备一致。
此时，请在 NJ/NX 系列 CPU 单元或通信对象的设备上采取本项目中介绍的措施。

参考

NJ/NX 系列 CPU 单元之间进行数据交换时，无需特别注意成员的存储器配置。

与通信对象设备统一存储器配置的方法

要与通信对象设备统一存储器配置，有两种方法。
例如，在 NJ/NX 系列 CPU 单元和 CJ 系列的 CPU 中，结构体型变量的存储器配置有以下不同。
本项中举例介绍统一其存储器配置的方法。



● 对策 1：变更 NJ/NX 系列 CPU 单元的结构体型变量成员的存储器配置

NJ/NX 系列 CPU 单元中，可通过成员的偏置指定，对各结构体型变量设定成员的存储器配置。在 NJ/NX 系列 CPU 单元上，根据通信对象设备，变更和调整结构体型变量成员的存储器配置。
结构体型变量成员的偏置指定在登录结构体型时进行。
进行通信的对象设备为CJ系列CPU单元时，若将偏置类型选择为“CJ”，可自动变为CJ的存储器配置。
若将偏置类型选择为 “ 任意 ”，则可自由决定偏置。

版本相关信息

可否使用成员的偏置指定，根据 CPU 单元的单元版本与 Sysmac Studio 版本的组合，发生如下变化。

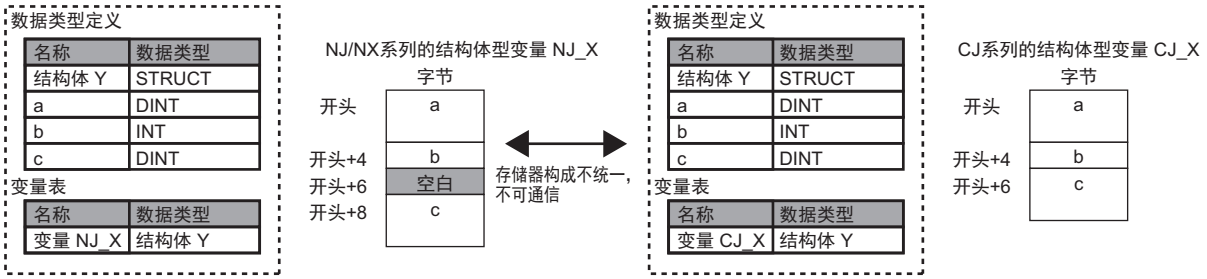
CPU 单元的单元版本	Sysmac Studio 的版本		
	Ver.1.01 以下	Ver.1.02	Ver.1.03 以上
Ver.1.01 以上	不可	可*1	可
Ver.1.00	不可	不可	不可

*1. 无偏置类型的选择功能。任意设定成员的偏置。

在本设定中调整结构体型变量的存储器配置后，请在联网的其他 NJ/NX 系列 CPU 单元上也进行相同的设定变更。

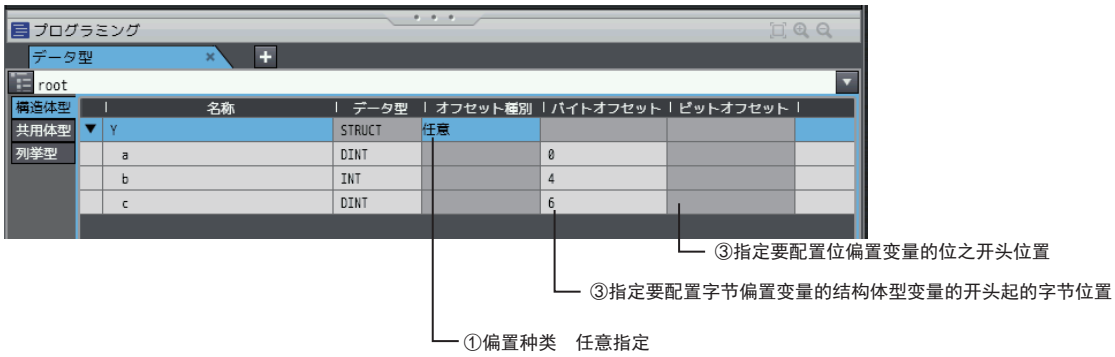
结构体型变量的存储器配置操作方法请参考📖《Sysmac Studio Version 1 操作手册（SBCA-CN5-362C 以上）》。

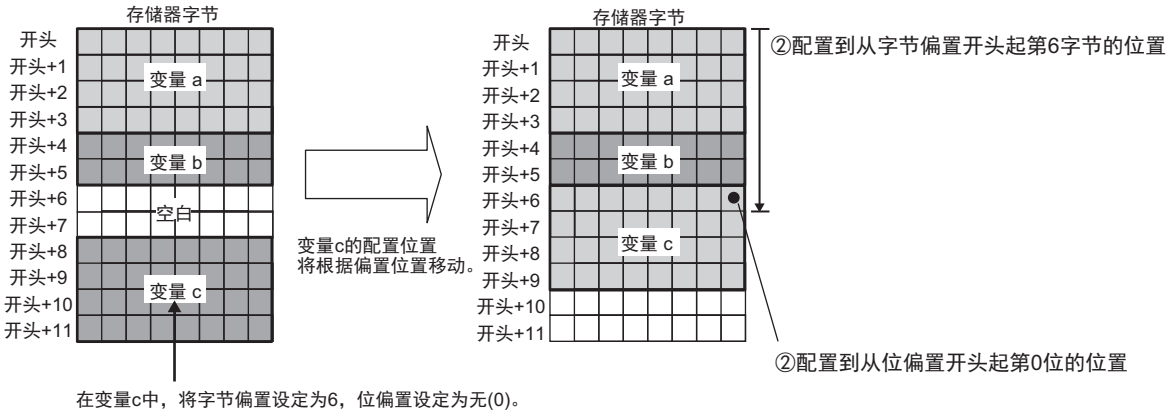
例）根据 CJ 系列 CPU 单元的结构体型变量的成员变更 NJ/NX 系列 CPU 单元的结构体型变量成员的存储器配置



为了统一NJ/NX系列CPU单元和CJ系列CPU单元的存储器构成，在Sysmac Studio中指定偏置。

在结构体型变量NJ_X的数据类型Y的成员c中指定以下偏置值。

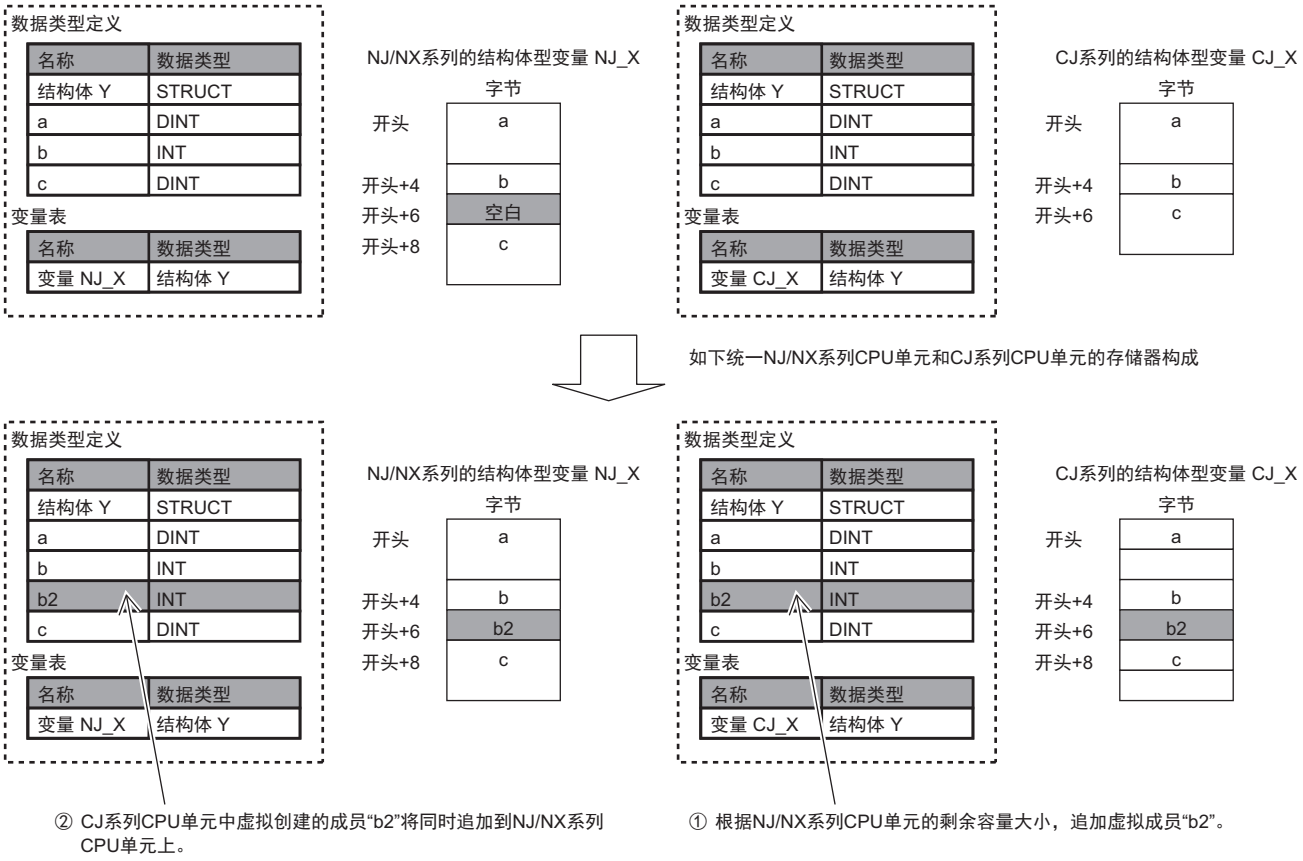




● 对策 2：变更通信对象设备的结构体型变量的存储器配置

根据 NJ/NX 系列 CPU 单元的结构体型变量的存储器配置，插入配置调整用的成员等，来调整通信对象设备的结构体型变量的存储器配置。
此外，除了存储器配置之外，还需要统一数据类型，因此需要同时创建通信对象设备、 NJ/NX 系列 CPU 单元的配置调整用成员。

例）根据 NJ/NX 系列 CPU 单元的结构体型变量的存储器配置变更 CJ 系列 CPU 单元的结构体型变量的存储器配置



A-9 CX-Designer 的变量表登录方法

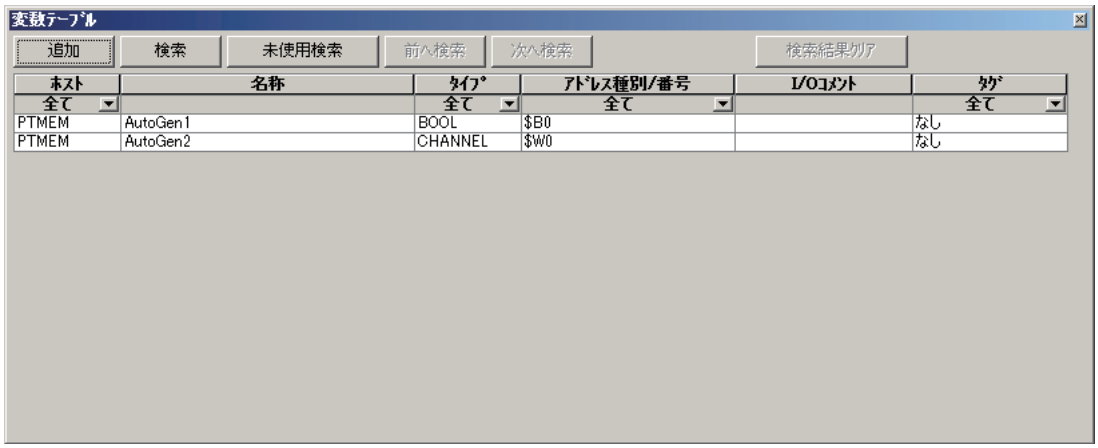
连接 NJ/NX 系列控制器和 NS 系列显示器使用时，在 CX-Designer 上，可将变量作为各功能部件中设定的地址使用。变量通过 [变量表] 管理。下面介绍通过 Microsoft Excel 复制变量表后，将多个变量批量登录到变量表的方法。 CX-Designer 的详情请参考📖《CX-Designer 用户手册 （SBSA-CN5-532）》。

1 **在 Microsoft Excel 中创建以下格式的变量表。**

表格的列数和配置请遵照以下格式。如 [地址类型 / 编号] 或 [I/O 注释] 等，全是空白单元格的列也不可省略。

主机	名称	类型	地址类型 / 编号	I/O 注释	标签
HOST3	_Card1BkupCmd.ExecBkup	BOOL			TRUE
HOST3	_Card1BkupCmd.CancelBkup	BOOL			TRUE
HOST3	_Card1BkupCmd.ExecVefy	BOOL			TRUE
HOST3	_Card1BkupCmd.CancelVefy	BOOL			TRUE
HOST3	_Card1BkupCmd.DirName	STRING(64)			TRUE
HOST3	_Card1BkupSta.Done	BOOL			TRUE
HOST3	_Card1BkupSta.Active	BOOL			TRUE
HOST3	_Card1BkupSta.Err	BOOL			TRUE
HOST3	_Card1VefySta.Done	BOOL			TRUE
HOST3	_Card1VefySta.Active	BOOL			TRUE
HOST3	_Card1VefySta.VefyRslt	BOOL			TRUE
HOST3	_Card1VefySta.Err	BOOL			TRUE
HOST3	_BackupBusy	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.Exec	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.DirName	STRING(64)			TRUE
HOST3	_Card1PrgTransferCmd.Password	STRING(33)			TRUE
HOST3	_Card1PrgTransferCmd.TargetUserProgram	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.TargetIPAdr	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.TargetVariable	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.TargetMemory	BOOL			TRUE
HOST3	_Card1PrgTransferSta.Done	BOOL			TRUE
HOST3	_Card1PrgTransferSta.Active	BOOL			TRUE
HOST3	_Card1PrgTransferSta.Err	BOOL			TRUE

2 **启动 CX-Designer，打开变量表画面。**

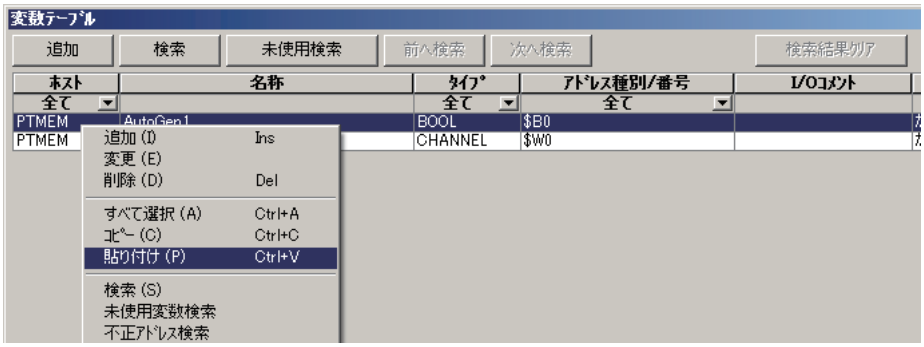


3 **复制 Microsoft Excel 变量表的阴影部分。**

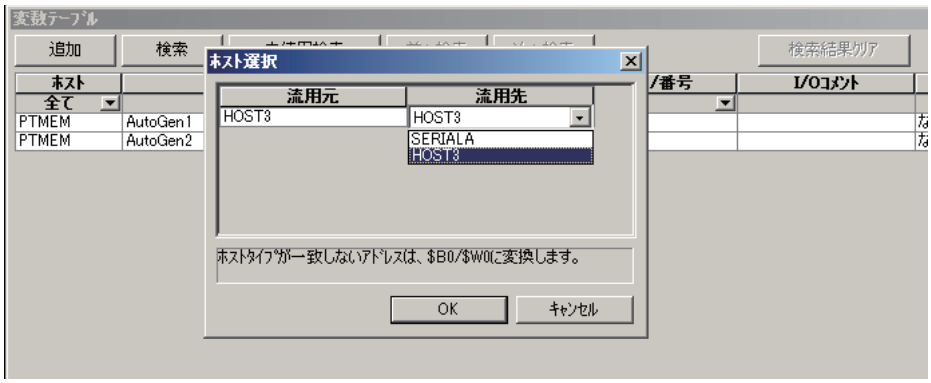
请务必复制所有的列。

主机	名称	类型	地址类型 / 编号	I/O 注释	标签
HOST3	_Card1BkupCmd.ExecBkup	BOOL			TRUE
HOST3	_Card1BkupCmd.CancelBkup	BOOL			TRUE
HOST3	_Card1BkupCmd.ExecVefy	BOOL			TRUE
HOST3	_Card1BkupCmd.CancelVefy	BOOL			TRUE
HOST3	_Card1BkupCmd.DirName	STRING(64)			TRUE
HOST3	_Card1BkupSta.Done	BOOL			TRUE
HOST3	_Card1BkupSta.Active	BOOL			TRUE
HOST3	_Card1BkupSta.Err	BOOL			TRUE
HOST3	_Card1VefySta.Done	BOOL			TRUE
HOST3	_Card1VefySta.Active	BOOL			TRUE
HOST3	_Card1VefySta.VefyRslt	BOOL			TRUE
HOST3	_Card1VefySta.Err	BOOL			TRUE
HOST3	_BackupBusy	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.Exec	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.DirName	STRING(64)			TRUE
HOST3	_Card1PrgTransferCmd.Password	STRING(33)			TRUE
HOST3	_Card1PrgTransferCmd.TargetUserProgram	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.TargetIPAdr	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.TargetVariable	BOOL			TRUE
HOST3	_Card1PrgTransferCmd.TargetMemory	BOOL			TRUE
HOST3	_Card1PrgTransferSta.Done	BOOL			TRUE
HOST3	_Card1PrgTransferSta.Active	BOOL			TRUE
HOST3	_Card1PrgTransferSta.Err	BOOL			TRUE

4 在 CX-Designer 的变量表画面上右击，执行 [粘贴]。



5 在 CX-Designer 的主机选择画面中，选择 NJ/NX 系列的主机，然后按下 [OK] 按钮。



至此，已将变量登录到 CX-Designer 的变量表画面中。

A-10 EtherCAT 从站或轴的有效 / 无效切换

EtherCAT 从站或轴的有效 / 无效切换可通过指令执行。利用本功能，可实现以下用途。

- 可在一个 Sysmac Studio 项目中管理 EtherCAT 从站构成或轴构成不同的多个装置。
- 可在一条生产线保持运行的状态下，变更其他生产线的 EtherCAT 从站构成或轴构成。

下面介绍本功能中使用的指令、系统定义变量及应用示例。

版本相关信息

利用指令执行 EtherCAT 从站或轴的有效 / 无效切换，需要 CPU 单元版本 1.04 以上和 Sysmac Studio Ver.1.05 以上的组合。

A-10-1 使用 EtherCAT 从站和轴时的项目设定

接通电源时或下载项目时，如果存在可能未安装到 EtherCAT 网络的 EtherCAT 从站，请事先设定为无效。同样，轴使用也请设定为未使用轴。若未安装到 EtherCAT 网络的 EtherCAT 从站设定为有效，或轴使用设定为使用轴，执行运行时可能发生异常。

参考

- EtherCAT 从站的有效 / 无效切换还可通过 Sysmac Studio 的 [配置和设定] ｜ [EtherCAT] ｜ [网络配置] ｜ [有效 / 无效设定] 的设定进行。但是，若设定方法为后面介绍的「应用 1　对 EtherCAT 从站构成和轴构成不同的装置进行统一管理」（P.A-160），可能因装置原因，需要每次手动更改 Sysmac Studio 的设定或更改项目文件。
- 设定为无效的 EtherCAT 从站可从 EtherCAT 网络上拆下或安装到网络上。

A-10-2 执行 EtherCAT 从站或轴的有效 / 无效切换的指令

EtherCAT 从站或轴的有效 / 无效切换通过用户程序中的指令执行。 EtherCAT 从站的有效 / 无效切换和轴的有效 / 无效切换使用不同的指令。指令名称分别如下。

切换对象	指令名称
EtherCAT 从站	EC_ChangeEnableSetting(EtherCAT 从站有效 / 无效切换) 指令
轴	MC_ChangeAxisUse(轴使用变更）指令

版本相关信息

EC_ChangeEnableSetting 指令、 MC_ChangeAxisUse 指令都可在 CPU 单元版本 1.04 以上和 Sysmac Studio Ver.1.05 以上的组合中使用。

EC_ChangeEnableSetting 指令

EtherCAT 从站的有效 / 无效切换使用 EC_ChangeEnableSetting(EtherCAT 从站有效无效切换）指令。EC_ChangeEnableSetting 指令可将指定节点地址的 EtherCAT 从站设定为无效或有效。执行本指令后，若重新接通控制器的电源，将恢复为指令执行前的设定。

EC_ChangeEnableSetting 指令的详细规格请参考📖《NJ/NX 系列 指令基准手册 基本篇（SBCA-CN5-360）》。

MC_ChangeAxisUse 指令

轴的有效 / 无效切换使用 MC_ChangeAxisUse （轴使用变更）指令。 MC_ChangeAxisUse 指令可将指定的轴参数 [轴使用] 设定为使用轴或未使用轴。执行本指令后，若重新接通控制器的电源，将恢复为指令执行前的设定。

MC_ChangeAxisUse 指令的详细规格请参考📖《NJ/NX 系列 指令基准手册 运动篇 (SBCE-CN5-364)》。

A-10-3 表示 EtherCAT 从站或轴状态的系统定义变量

EtherCAT 从站或轴的当前状态可通过系统定义变量的值来判断。各自的系统定义变量如下所示。

参照状态的对象	系统定义变量名称
EtherCAT 从站	_EC_DisableSlavTbl[] （无效设定从站表）
轴	_MC_AX[].Cfg.AxEnable （轴使用）*1

*1. NX 系列 CPU 单元中， _MC1_AX[].Cfg.AxEnable、 _MC2_AX[].Cfg.AxEnable 也可使用。

_EC_DisableSlavTbl[] （无效设定从站表）

_EC_DisableSlavTbl[] （无效设定从站表）为表示 EtherCAT 从站的当前状态是否为无效的系统定义变量。在排列的后缀中指定节点地址。 _EC_DisableSlavTbl[] （无效设定从站表）的值含义如下。

值	含义
TRUE	指定节点地址的 EtherCAT 从站为无效
FALSE	指定节点地址的 EtherCAT 从站为有效

_MC_AX[].Cfg.AxEnable （轴使用）

_MC_AX[].Cfg.AxEnable （轴使用）为表示轴是否是未创建轴以及是否使用轴的系统定义变量。在排列的后缀中指定轴编号。 _MC_AX[].Cfg.AxEnable （轴使用）的值含义如下。

值	含义
0:_mcNoneAxis	指定的轴为未创建轴
1:_mcUnusedAxis	指定的轴为未使用轴
2:_mcUsedAxis	指定的轴为使用轴

A-10-4 程序的执行 / 停止指令

进行有效 / 无效切换的 EtherCAT 从站或轴中可能有附带的特定程序。此时，在切换 EtherCAT 从站或轴的有效 / 无效的同时，需要执行或停止附带的程序。特定程序的执行 / 停止指令可在用户程序中使用以下指令执行。

功能	指令名称
程序的执行指令	PrgStart 指令
程序的停止指令	PrgStop 指令

PrgStart 指令、PrgStop 指令的详细规格请参考📖《NJ/NX 系列 指令基准手册 基本篇（SBCA-CN5-360K 以后）》。

使用注意事项

要停止程序时，请在程序停止前，将附带该程序的 EtherCAT 从站或轴设为无效。

版本相关信息

PrgStart 指令、PrgStop 指令都可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。

A-10-5 确认程序的执行 / 停止

对于进行有效 / 无效切换的 EtherCAT 从站或轴中附带的程序，可通过 PrgStatus 指令确认对其发出的指令时执行还是停止。 PrgStatus 指令的详细规格请参考📖《NJ/NX 系列 指令基准手册 基本篇（SBCA-CN5-360K 以后）》。

版本相关信息

PrgStatus 指令可在 CPU 单元版本 1.08 以上和 Sysmac Studio Ver.1.09 以上的组合中使用。

A-10-6 Sysmac Studio 上的设定

EtherCAT 从站的有效 / 无效、轴的有效 / 无效及程序开始运行时的执行 / 停止等设定在 Sysmac Studio 上也可变更。根据应用不同，有些需要在 Sysmac Studio 上事先设定 EtherCAT 从站、轴及开始运行时程序的状态。

Sysmac Studio 上的 EtherCAT 从站有效 / 无效设定

若要在 Sysmac Studio 上将 EtherCAT 从站设为有效，应按照以下步骤进行设定。

1 **右击 [配置和设定] ｜ [EtherCAT]，从菜单中选择 [编辑]。**
显示 EtherCAT 配置编辑画面。

2 **在工具箱中右击要连接的 EtherCAT 从站，从菜单中选择 [插入]。**
在 EtherCAT 配置编辑画面中，选择的 EtherCAT 从站将显示在 EtherCAT 主机下方。此外，EtherCAT 配置编辑画面的右侧将显示 EtherCAT 从站的参数设定画面。

3 **在 EtherCAT 从站的参数设定画面中，将 [有效 / 无效设定] 的设定值设为 “ 有效 ”。**

Sysmac Studio 上的轴有效 / 无效设定

若要在 Sysmac Studio 上将轴设为有效，应按照以下步骤进行设定。

1 **右击 [配置和设定] ｜ [运动控制设定] ｜ [轴设定]，从菜单中选择 [添加] ｜ [轴设定]。**
将在 [轴设定] 下方添加 “MC_Axis000(0)” 等轴变量。

2 **右击 “MC_Axis000(0)” 等轴变量名称，从菜单中选择 [编辑]。**
显示轴参数设定的 “ 轴基本设定 ” 画面。

3 **将 [轴使用] 设定为 “ 使用轴 ”。**

Sysmac Studio 上的开始运行时程序启动 / 停止设定

若要在 Sysmac Studio 上、在开始运行时执行程序，应按照以下步骤进行设定。

1 **右击 [配置和设定] ｜ [任务设定]，从菜单中选择 [编辑]。**
显示任务设定画面。

2 **单击 [程序的分配设定] 按钮。**
显示任务设定的程序分配设定画面。

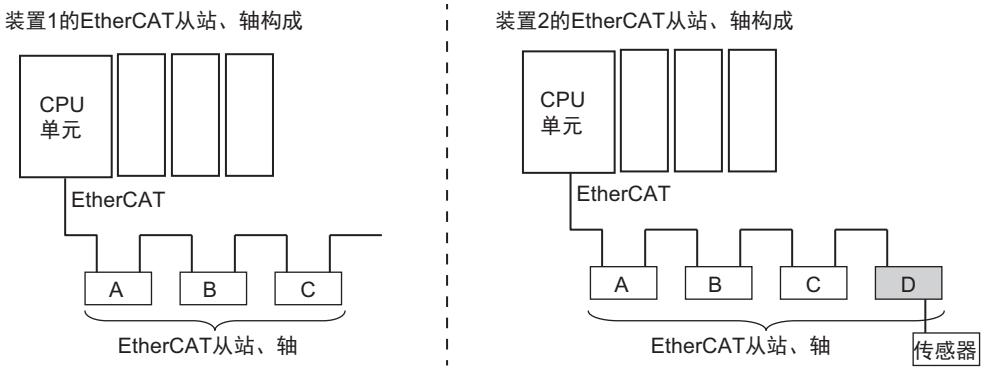
3 **在程序的分配设定画面中，将该程序的 [初始状态] 设为 “ 启动 ”。**

A-10-7 EtherCAT 从站或轴的有效 / 无效切换应用示例

下面以具体的应用为例，介绍 EtherCAT 从站或轴的有效 / 无效切换使用方法。

应用 1 对 EtherCAT 从站构成和轴构成不同的装置进行统一管理

NJ 控制器的 EtherCAT 从站或轴构成在装置 1 和装置 2 中不同，具体如下图所示。用 Sysmac Studio 上的一个项目对其进行统一管理。



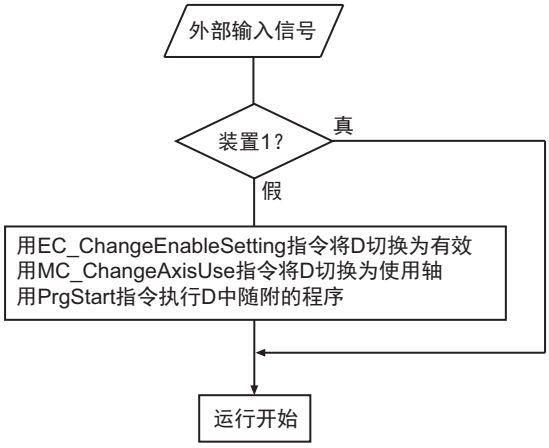
在 Sysmac Studio 的项目中，对图中的 A、B、C、D 4 个 EtherCAT 从站或轴创建 EtherCAT 从站构成。然后，如下在 Sysmac Studio 上根据装置 1 设定各 EtherCAT 从站的有效 / 无效、轴使用的使用轴 / 未使用轴及随附程序开始运行时的启动 / 停止状态。

EtherCAT 从站	安装到 EtherCAT 网络	有效 / 无效	轴使用	随附程序
A、B、C	有	有效	使用轴	运行开始时、启动
D	无	无效	未使用轴	运行开始时、停止

对应装置 2 时，可利用指令如下切换 EtherCAT 从站的有效 / 无效、轴使用的使用轴 / 未使用轴及随附程序的启动 / 停止设定。

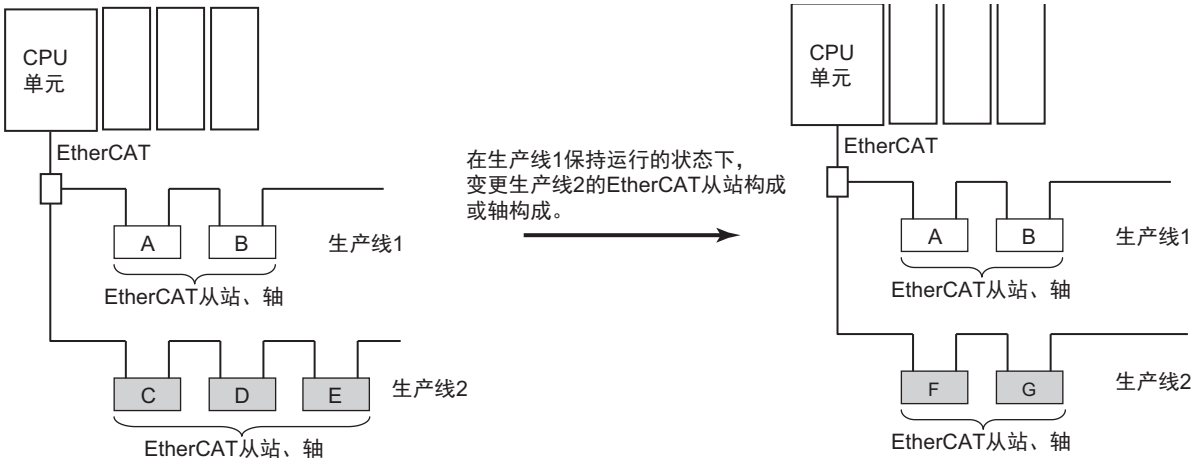
EtherCAT 从站	安装到 EtherCAT 网络	有效 / 无效	轴使用	随附程序
A、B、C	有	有效	使用轴	执行
D	有	有效	使用轴	执行

用户程序的算法如下图所示。假设从控制器外部输入表示运行装置为装置 1 还是装置 2 的信号。



应用 2　在运行过程中变更 EtherCAT 从站构成和轴构成

在下图生产线 1 保持运行的状态下，变更生产线 2 的 EtherCAT 从站构成或轴构成。



在 Sysmac Studio 的项目中，对图中的 A ～ G 7 个 EtherCAT 从站或轴创建 EtherCAT 从站构成。使用 Sysmac Studio，事先设定构成变更前 A ～ G EtherCAT 从站的有效 / 无效、轴使用的使用轴 / 未使用轴及随附程序开始运行时的启动 / 停止状态。

EtherCAT 从站	安装到 EtherCAT 网络	有效 / 无效	轴使用	随附程序
A、 B	有	有效	使用轴	运行开始时、启动
C、 E、 D	有	有效	使用轴	运行开始时、启动
F、 G	无	无效	未使用轴	运行开始时、停止

将使用的 EtherCAT 从站或轴从 C、 D、 E 变更为 F、 G 的步骤如下。

1 **停止生产线 2。**

2 **使用 MC_ChangeAxisUse 指令，将 C、 D、 E 的 [轴使用] 设为未使用轴。**

3 **使用 EC_ChangeEnableSetting 指令，将 C、 D、 E 的 EtherCAT 从站设定设为无效。**

4 **使用 PrgStop 指令，停止 C、 D、 E 随附的程序。**

5 **将 EtherCAT 从站 C、 D、 E 从生产线 2 上排除。**

6 **将 EtherCAT 从站 F、 G 安装到生产线 2 上。**

7 **使用 EC_ChangeEnableSetting 指令，将 F、 G 的 EtherCAT 从站设定设为有效。**

8 **使用 MC_ChangeAxisUse 指令，将 F、 G 的 [轴使用] 设为使用轴。**

9 **使用 PrgStart 指令，执行 F、 G 随附的程序。**

10 **再次启动生产线 2。**

执行上述操作后， A ～ G EtherCAT 从站的有效 / 无效、轴使用的使用轴 / 未使用轴及随附程序的启动 / 停止设定变化如下。

EtherCAT 从站	安装到 EtherCAT 网络	有效 / 无效	轴使用	随附程序
A、 B	有	有效	使用轴	执行
C、 E、 D	无	无效	未使用轴	停止
F、 G	有	有效	使用轴	执行

使用注意事项

要停止程序时，请在程序停止前，将附带该程序的 EtherCAT 从站或轴设为无效。

A-11 用户程序的大小相关的限制

受 CPU 单元内存储器容量的制约，用户程序的大小有限制。若超出限制，运行时会发生异常。下面按照不同项目介绍 1 台 CPU 单元上可创建的用户程序大小限制。

用户程序或变量大小的近似值可使用 Sysmac Studio 的存储器显示功能确认。

创建用户程序时请注意限制。但是，标示的限制值仅供参考。考虑到将来用户程序可能会扩展，创建用户程序时，建议留出一定的余量。

使用注意事项

使用在线编辑时，即使用户程序的大小在限制值以内，也可能发生异常。这是因为通过在线编辑变更用户程序后， CPU 单元的存储器中可能还有保留的数据。发生异常时，若将控制器设为程序模式，再次向控制器传送用户程序，有时可以解除异常。

A-11-1 用户程序的各部件限制

下面介绍用户程序的各部件限制。有限制的部件如下所示。

- POU
- 变量
- 数据类型定义
- 常数 （文本）

POU 的限制

POU 的限制包括 POU 定义的限制和 POU 实例的限制。

● POU 定义的限制

POU 的定义有以下限制。

限制项目	CPU 单元的型号			
	NX701-□□□□	NJ501-□□□□	NJ301-□□□□	NJ101-□□□□
程序数的最大值	1,000	500	500	500
（FB 定义数＋ FUN 定义数＋梯形图程序的段落数）的最大值	6,000	3,000	750	450
（FB 定义和 FUN 定义的输入 / 输出 / 输入输出变量的总个数）的最大值	64	64	64	64

● POU 实例的限制

POU 的实例有以下限制。因 CPU 单元的型号及 CPU 单元和 Sysmac Studio 的版本不同，POU 实例数的最大值不同。

NX701- □□□□的 POU 实例数最大值

限制项目	CPU 单元的型号
	NX701- □□□□
POU 实例数的最大值	48,000

NJ501- □□□□的 POU 实例数最大值

限制项目	版本	CPU 单元的型号
	Sysmac Studio	NJ501- □□□□
POU 实例数的最大值	1.05 以下	6,000
	1.06 以上	9,000

NJ301- □□□□的 POU 实例数最大值

限制项目	版本		CPU 单元的型号
	CPU 单元	Sysmac Studio	NJ301- □□□□
POU 实例数的最大值	—	1.04 以下	1,500
	1.03 以下	1.05 以上	2,400
	1.04 以上		3,000

NJ101- □□□□的 POU 实例数最大值

限制项目	CPU 单元的型号
	NJ101- □□□□
POU 实例数的最大值	1,800

POU 实例的计数方法请参考📖「POU 实例的数量」（P.A-167）。

变量的限制

变量的限制包括变量使用量的限制和变量定义的限制。

● 变量使用量的限制

变量的使用量有以下限制。

限制项目	CPU 单元的型号			
	NX701-□□□□	NJ501-□□□□	NJ301-□□□□	NJ101-□□□□
无保持属性的变量总大小 [MB] 的最大值*1	256	4	2	2
有保持属性的变量总大小 [MB] 的最大值 *1	4	2	0.5	0.5
无保持属性的变量数*2 最大值	360,000	90,000	22,500	22,500
有保持属性的变量数*3 最大值	40,000	10,000	*4	5,000
网络变量的数量最大值	40,000	40,000	27,500	27,500

*1. 每个变量的数据大小因数据类型不同而异。各数据类型的大小请参考📖「6-3-5 数据类型」（P.6-31）。
*2. 计数方法请参考📖「无保持属性的变量数」（P.A-167）。
*3. 计数方法请参考📖「有保持属性的变量数」（P.A-168）。
*4. 因 Sysmac Studio 的版本和 CPU 单元的单元版本不同，限制值变化如下。
Sysmac Studio Ver.1.05 以上且 CPU 单元版本 1.04 以上时：5000 个
上述之外的组合：2500 个

● 变量定义的限制

变量定义有以下限制。

限制项目	CPU 单元的型号			
	NX701-□□□□	NJ501-□□□□	NJ301-□□□□	NJ101-□□□□
1 个排列的要素数最大值	65,535	65,535	65,535	65,535
排列的维数最大值	3	3	3	3
1 个排列的大小 [MB] 最大值	8	4	2	2
排列后缀 （要素编号）值的最大值	65,535	65,535	65,535	65,535
字符串型变量的大小 [字节]*1 最大值	1,986	1,986	1,986	1,986

*1. 包含结束 NULL 字符。因此，大小为 1986 字节的字符串中，半角英文及数字为 1985 字符。

数据类型定义的限制

数据类型定义有以下限制。

限制项目	CPU 单元的型号			
	NX701-□□□□	NJ501-□□□□	NJ301-□□□□	NJ101-□□□□
数据类型定义数*1 的最大值	8,000	2,000	1,000	1,000
每个结构体型定义的层级数最大值	8	8	8	8
每个结构体型定义的成员数最大值	2,048	2,048	2,048	2,048
每个结构体型变量的大小 [MB] 最大值	8	4	2	2
每个联合体型定义的成员数最大值	4	4	4	4
每个列举型定义的列举值数最大值	2,048	2,048	2,048	2,048

*1. 计数方法请参考📖「数据类型定义的数量」（P.A-168）。

常数 （文本）的限制

常数 （文本）有以下限制。

限制项目	CPU 单元的型号			
	NX701-□□□□	NJ501-□□□□	NJ301-□□□□	NJ101-□□□□
字符串型常数 （文本）的大小 [字节] 最大值	1,985	1,985	1,985	1,985

A-11-2 用户程序的各部件数量

表示 POU 实例、无保持属性的变量、有保持属性的变量及数据类型定义的计数方法。但是，这里记载的内容仅供参考。因 CPU 单元的单元版本不同，部件的技术方法略有不同。因此，请务必在 Sysmac Studio 上确认用户程序的大小是否合适。

POU 实例的数量

POU 实例的计数方法如下。

● POU 实例的计数对象

POU 实例的计数对象如下。

- 程序
- FB 实例 （包括用户创建和指令）
- FUN （包括用户创建和指令）

● POU 实例的计数注意

计算 POU 实例的数量时，请注意以下几点。

- 使用 n 个与同一 FB 定义对应的 FB 实例时，计为 n 个。
- 在同一任务中使用同一 FUN 时，即使 FUN 有多个，也算作 1 个。
- 在不同任务中使用同一 FUN 时，每个任务计为 1 个。

无保持属性的变量数

无保持属性的变量计数方法如下。

● 无保持属性的变量计数对象

无保持属性的变量计数对象如下。

- 无保持属性的全局变量
- 程序及 FB 实例 （包括用户创建和指令）的无保持属性的本地变量

● 无保持属性的变量计数注意

计算无保持属性变量的数量时，请注意以下几点。

- 排列始终计为 1 个，与要素数无关。
- FB 实例计为 1 个。但是， FB 实例包括用户创建和指令。
- 将FB实例作为要素的排列始终计为1个，与要素数无关。但是，该FB中使用的无保持属性的变量数按照排列的要素数计算。

有保持属性的变量数

有保持属性的变量计数方法如下。

● 有保持属性的变量计数对象

有保持属性的变量计数对象如下。

- 有保持属性的全局变量
- 程序及 FB 实例 （包括用户创建和指令）的有保持属性的本地变量

● 有保持属性的变量计数注意

计算有保持属性变量的数量时，请注意以下几点。

- 排列始终计为 1 个，与要素数无关。
- 将 FB 实例作为要素的排列不计数。但是，该 FB 中使用的有保持属性的变量数按照排列的要素数计算。

数据类型定义的数量

数据类型定义的计数方法如下。

● 数据类型定义的计数对象

数据类型定义的计数对象如下。

- 用户创建的结构体型定义
- 用户创建的联合体型定义
- 用户创建的列举型定义

A-12 Ver.1.02 以下的 CPU 单元更换方法

单元版本 1.02 以下的 NJ 系列 CPU 单元不支持 SD 存储卡备份功能、Sysmac Studio 控制器备份功能。因此，由于 CPU 单元故障或版本升级等需要更换 CPU 单元时，需要进行以下作业。

作业	内容
CPU 单元内数据的上传	将更换前 CPU 单元内的数据分为以下 3 类，上传到计算机中。 • 项目 • 变量 / 存储器的当前值 • 标签数据链接表
连接新的 CPU 单元	从控制器上拆下旧的 CPU 单元，连接新的 CPU 单元。
将数据下载到 CPU 单元	将计算机中保存的 3 类数据下载到更换后的 CPU 单元中。

下面介绍上述作业的详情。

A-12-1 CPU 单元内数据的上传

将更换前 CPU 单元内的数据分为以下 3 类，上传到计算机中。使用支持软件 Sysmac Studio 和 Network Configurator。

要上传的数据	数据的内容	支持软件
项目	• 单元构成和单元设定 • I/O 映射 • 控制器设定 （动作设定、内置 EtherNet/IP 端口设定） • 运动控制设定 • 凸轮数据设定 • 事件设定 • 任务设置 • POU • 数据类型定义、全局变量定义	Sysmac Studio
变量 / 存储器的当前值	• 有保持属性的变量值 • CJ 单元用存储器中的 HR、 DM、 EM 区域的值 • 绝对编码器原点位置偏置	Sysmac Studio
标签数据链接表*1	• EtherNet/IP 上的标签数据链接设定	Sysmac Studio Network Configurator*2

*1. 仅在设定了标签数据链接时需要上传标签数据链接表。
*2. 使用 Ver.1.09 以下的 Sysmac Studio 时，使用 Network Configurator。

使用注意事项

项目、变量 / 存储器的当前值、标签数据链接表中不含 CPU 单元内的以下数据。因此，更换 CPU 单元后，需要重新设定。

- 数据追踪设定
- 控制器名称
- 操作权限认证
- 内置时钟的时区设定

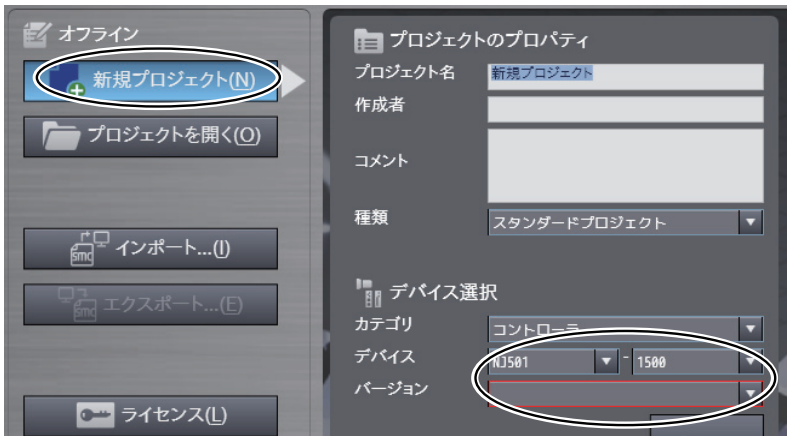
项目的上传

项目的上传步骤如下。

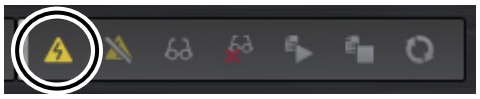
1 **在连接了 NJ 系列控制器的计算机上启动 Sysmac Studio。**

2 **单击 [新项目] 按钮，创建新项目。**

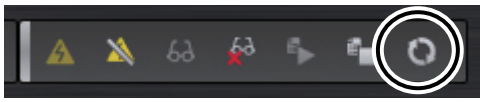
3 **选择更换前 CPU 单元的 [设备] 和 [版本]。**



4 **单击工具栏的 [在线] 按钮。**



5 **单击工具栏的 [同步] 按钮。**



6 **取消勾选 [EtherNet/IP 连接设定（不传送内置端口、单元）]，单击[传送[控制器➞计算机]]按钮。**

控制器中的项目将上传到计算机中。

7 **单击 [重新核对] 按钮。**

将核对上传的项目和控制器中的项目。

8 **单击 [保存] 按钮。**

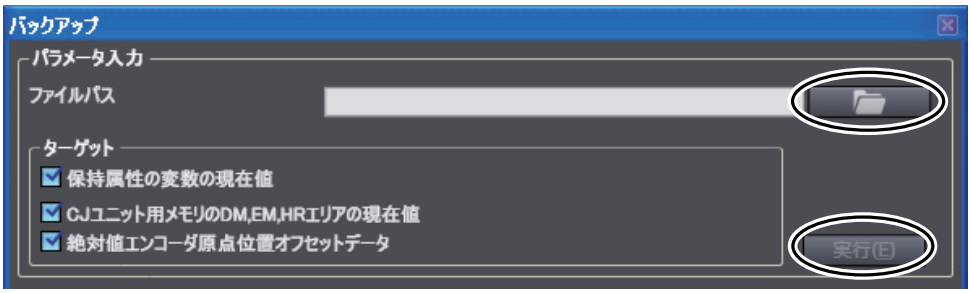
项目将保存到计算机中。

上传变量 / 存储器的当前值

上传变量 / 存储器的当前值时，使用 “Sysmac Studio 变量 / 存储器的备份功能 ”。 “Sysmac Studio 变量 / 存储器的备份功能 ” 详情请参考「9-8 Sysmac Studio 变量 / 存储器的备份功能」（P.9-39）。
步骤如下所示。

1 **从 Sysmac Studio 的工具栏中选择 [工具] ｜ [备份] ｜ [变量・存储器的备份]。**

2 **选择目标 “ 保持属性的变量当前值 ”、“CJ 单元用存储器的 DM、EM、HR 区域当前值 ”、“ 绝对编码器原点位置偏置数据 ”，单击 [执行] 按钮。**
变量 / 存储器的各数据将上传到计算机中。



标签数据链接表的上传

使用 Ver.1.09 以下的 Sysmac Studio 时，使用 Network Configurator。
标签数据链接表的上传步骤如下。

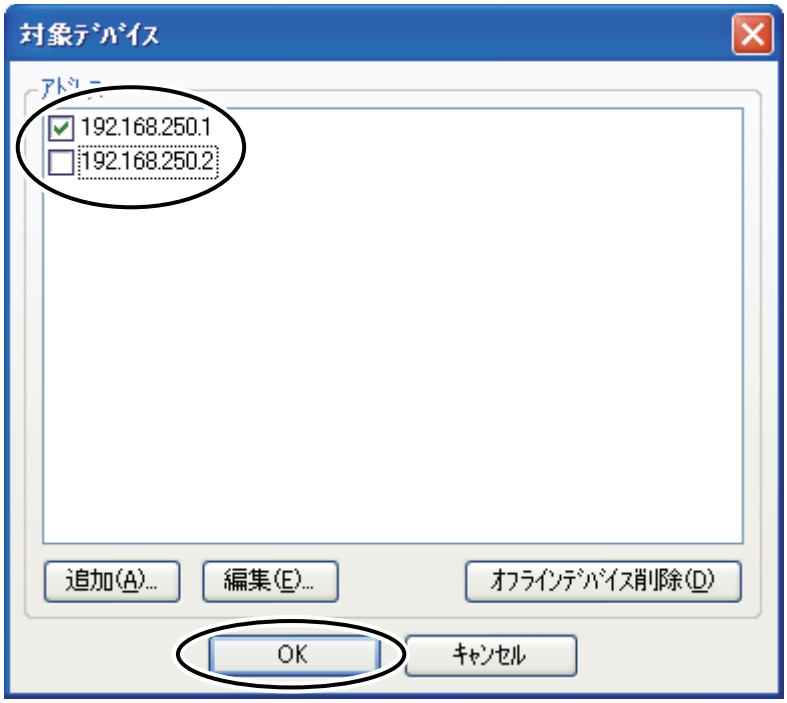
1 **在连接了 NJ 系列控制器的计算机上启动 Network Configurator。**

2 **选择工具栏中的 [网络] | [连接]。**

3 **选择工具栏中的 [网络] | [上传]。**
显示 “ 以当前文档开始上传。确定吗？ ”。

4 **单击 [是] 按钮。**

5 **“ 对象设备 ” 仅选择所连接的 CPU 单元的 IP 地址，然后单击 [OK] 按钮。**



6 **单击 [文件] | [另存为]。**
标签数据链接表将上传到计算机中。

A-12-2 连接新的 CPU 单元

拆下更换前的 CPU 单元，连接新的 CPU 单元。连接方法请参考📖《NJ 系列 CPU 单元 用户手册 硬件篇 (SBCA-CN5-358)》。

请对新的 CPU 单元单元执行以下操作。

- 安装更换前的 CPU 单元上安装的 SD 存储卡。
- 将拨动开关设定为与更换前的 CPU 单元相同。

A-12-3 将数据下载到 CPU 单元

将已上传到计算机的项目、变量 / 存储器的当前值及标签数据链接表下载到更换后的 CPU 单元中。

安全要点

请对下载的项目进行充分的动作确认后再转移到正式运行。

项目的下载

项目的下载步骤如下。

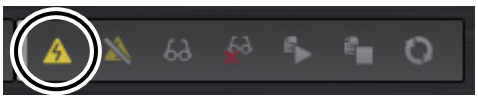
1 **在连接了 NJ 系列控制器的计算机上启动 Sysmac Studio。**

2 **单击［打开项目］按钮。**

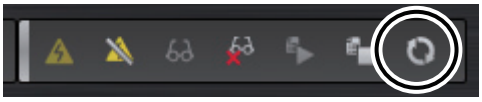
3 **从项目列表中选择更换 CPU 单元前上传的项目。**

下图为更换 CPU 单元前上传的项目名称为 " 备份数据 " 时的示例。

4 **单击工具栏的 [在线] 按钮。**



5 **单击工具栏的 [同步] 按钮。**



6 **取消勾选 [EtherNet/IP连接设定（不传送内置端口、单元）], 单击[传送[计算机→控制器]]按钮。**

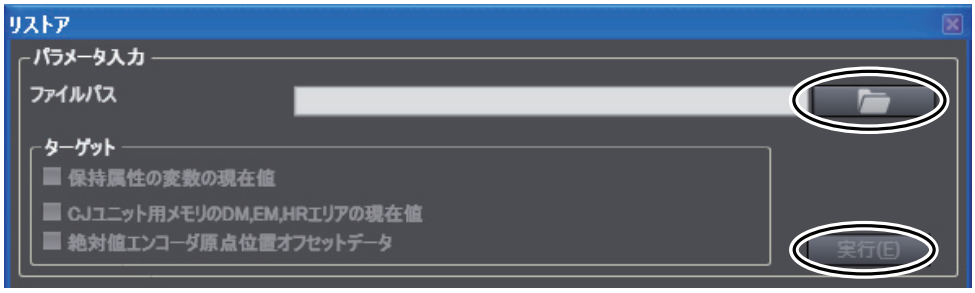
计算机中的项目将下载到控制器中。

下载变量 / 存储器的当前值

下载变量 / 存储器的当前值时，使用 “Sysmac Studio 变量 / 存储器的备份功能 ”。 “Sysmac Studio 变量 / 存储器的备份功能 ” 详情请参考「9-8 Sysmac Studio 变量 / 存储器的备份功能」（P.9-39）。
步骤如下所示。

1 **从 Sysmac Studio 的工具栏中选择 [工具] 丨 [备份] 丨 [变量・存储器的恢复]。**

2 **选择更换 CPU 单元前上传的数据文件，单击 [执行] 按钮。**
变量 / 存储器的各数据将下载到控制器中。



标签数据链接表的下载

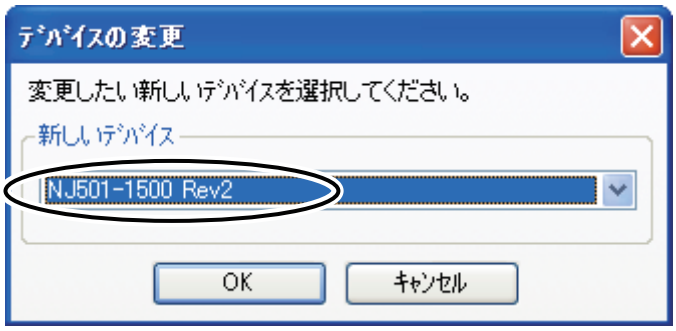
使用 Ver.1.09 以下的 Sysmac Studio 时，使用 Network Configurator。
标签数据链接表的下载步骤如下。

1 **在连接了 NJ 系列控制器的计算机上启动 Network Configurator。**

2 **选择工具栏中的 [网络] ｜ [连接]。**

3 **选择 [文件] ｜ [打开]，打开更换 CPU 单元前上传的标签数据链接表文件。**

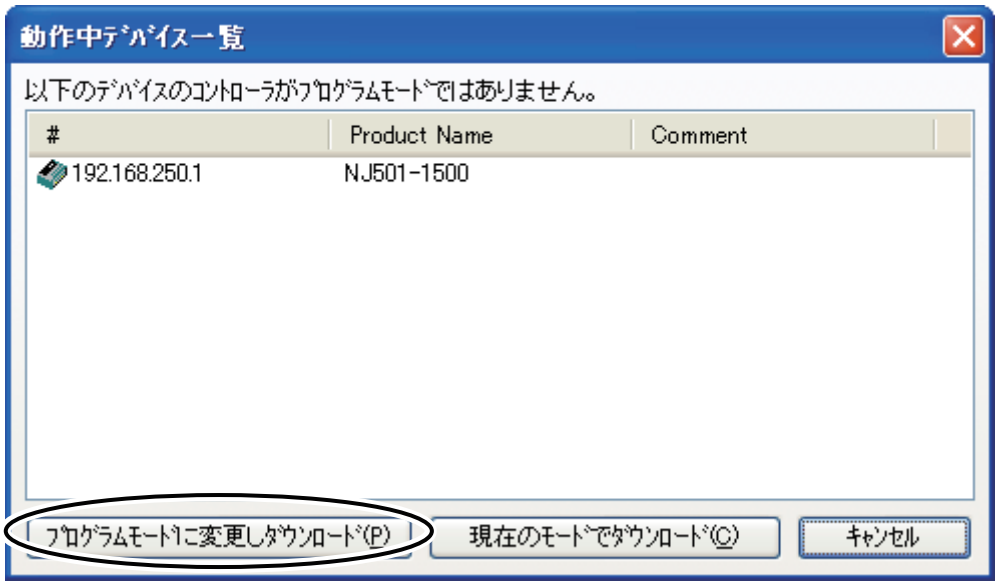
4 **将单元版本 1.02 以下的 CPU 单元上传的标签数据链接表下载到单元版本 1.03 以上的 CPU 单元中时，选择 [设备] ｜ [设备变更]，将 “ 新设备 ” 设定为 “NJ □□□ -1 □□□ Rev2”。**



5 **选择工具栏中的 [网络] ｜ [下载]。**
显示 “ 写入所有设备的参数，将新设定设为有效。确定吗？ ”。

6 **单击 [是] 按钮。**
显示动作中的设备一览。

7 **单击 [变更为程序模式并下载] 按钮。**
标签数据链接表将下载到控制器中。



使用注意事项

上述步骤4中，若“新设备”的选项中未显示“NJ□□□-1□□□ Rev2”，请将Network Configurator更新为最新版本。

A-13 NX 系列的版本相关信息

下面介绍 NX 系列 CPU 单元和 Sysmac Studio 的版本对应及各单元版本的支持功能。

A-13-1 单元版本和 Sysmac Studio 版本之间的关系

NX 系列CPU 单元的单元版本和Sysmac Studio的版本有对应关系。一般情况下请组合使用对应的版本。

单元版本和 Sysmac Studio 的对应版本

NX 系列 CPU 单元的单元版本与 Sysmac Studio 的对应版本表如下所示。

CPU 单元的单元版本	Sysmac Studio 的对应版本
Ver.1.11	Ver.1.15
Ver.1.10	Ver.1.13

单元版本和 Sysmac Studio 的版本不匹配时的规格

NX 系列 CPU 单元的单元版本和 Sysmac Studio 的版本不匹配时的规格如下。

- **Sysmac Studio 的版本为 Ver.1.12 以下时**

  Ver.1.12 以下的 Sysmac Studio 中，不可使用 NX 系列 CPU 单元。

- **Sysmac Studio 的版本较旧时**

  如果 Sysmac Studio 的版本较旧，可使用的功能为 Sysmac Studio 版本所对应的 CPU 单元的单元版本所配备的功能。

  例） CPU 单元的单元版本：Ver.1.11
  Sysmac Studio 的版本：Ver.1.13

与 Syamac Studio Ver.1.13 对应的 CPU 单元的单元版本为 Ver.1.10。因此，可使用的功能为 CPU 单元版本 1.10 所配备的功能。 CPU 单元版本 1.11 及以上版本中追加的功能将无法使用。

● CPU 单元的单元版本较旧时

NX 系列 CPU 单元的单元版本较旧时，请在 Sysmac Studio 项目的设备选择画面中，选择所连接 CPU 单元的单元版本。可使用的功能为 CPU 单元的单元版本所配备的功能。

例） CPU 单元的单元版本：Ver.1.10

Sysmac Studio 的版本：Ver.1.15

与 Syamac Studio Ver.1.15 对应的 CPU 单元的单元版本为 Ver.1.11。但是，连接的 CPU 单元的单元版本为 Ver.1.10，因此在项目的设备选择画面的 “ 版本 ” 中，请选择 “Ver.1.10”。

在设备选择画面的 “ 版本 ” 中选择了 “Ver.1.10” 时，可使用的功能为 CPU 单元版本 1.10 所配备的功能。 CPU 单元版本 1.11 及以上版本中追加的功能将无法使用。

参考

项目的设备中设定的单元版本

- Sysmac Studio Ver.1.02 以上时，在设备选择画面中设定单元版本的选择。
- 可设定的单元版本为所用 CPU 单元的单元版本或更低版本。例如， CPU 单元的单元版本为 Ver.1.11 时，请选择 “1.10” 或 “1.11”。
- Sysmac Studio 中，项目的设备中设定的单元版本将视为 CPU 单元的单元版本。例如，项目的设备中设定的单元版本为 Ver.1.10 时，Sysmac Studio 中可使用的功能为单元版本 1.10 及更低版本的功能。
- 项目的设备中设定的单元版本低于传送目标 CPU 单元的单元版本时，可从 Sysmac Studio 传送项目。
- 设备选择画面的详情请参考📖《Sysmac Studio Version1 操作手册 (SBCA-CN5-362C 以上)》。

A-13-2 通过更新单元版本追加和变更支持功能

下面介绍 NX 系列 CPU 单元的单元版本更新后，发生追加或变更的支持功能。

● 功能规格的追加和变更

按追加或变更的功能规格，分别列出 CPU 单元的单元版本和 Sysmac Studio 版本。

功能			追加 / 变更	单元版本	Sysmac Studio 的版本	参照处
通信功能	通信用指令		变更	Ver.1.11	Ver.1.15	📖《NJ/NX 系列 指令基准手册 基本篇（SBCA-CN5-360）》
SD 存储卡功能	应用程序	SD 存储卡程序传送功能	追加	Ver.1.11	Ver.1.15	📖「9-5 SD 存储卡程序传送功能」（P.9-28）

（注）因 CPU 单元的单元版本更新而发生的各功能模块功能的追加和变更，请参考各功能模块的手册。

● 基本指令、运动控制指令的追加和变更

CPU 单元的单元版本更新后，可使用的基本指令及运动控制指令可能发生追加或变更。

详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》及📖《NJ/NX 系列 指令基准手册 运动篇 （SBCE-CN5-364）》。

● 控制器事件的追加和变更

CPU 单元的单元版本更新后，发生的事件可能发生追加或变更。此外，发生异常时的处理方法也可能变更。

详情请参考📖《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。

● 系统定义变量的追加和变更

CPU 单元的单元版本更新后，可使用的系统定义变量可能发生追加或变更。

详情请参考📖「A-4 系统定义变量一览」（P.A-38）。

A-14 NJ 系列的版本相关信息

下面介绍 NJ 系列 CPU 单元和 Sysmac Studio 的版本对应及各单元版本的支持功能。

A-14-1 单元版本和 Sysmac Studio 版本之间的关系

NJ 系列 CPU 单元的单元版本和 Sysmac Studio 的版本有对应关系。一般情况下请组合使用对应的版本。

单元版本和 Sysmac Studio 的对应版本

NJ 系列 CPU 单元的单元版本与 Sysmac Studio 的对应版本表如下所示。

CPU 单元的单元版本	Sysmac Studio 的对应版本
Ver.1.11	Ver.1.15
Ver.1.10*1	Ver.1.13*2
	Ver.1.12
Ver.1.09	Ver.1.10
Ver.1.08	Ver.1.09
Ver.1.07	Ver.1.08
Ver.1.06	Ver.1.07
Ver.1.05	Ver.1.06
Ver.1.04	Ver.1.05
Ver.1.03	Ver.1.04
Ver.1.02	Ver.1.03
Ver.1.01	Ver.1.02
Ver.1.00*3	Ver.1.01
	Ver.1.00

*1. CPU 单元 NJ101- □□□□中无 1.09 以下的版本。
*2. CPU 单元 NJ101- □□□□中请使用 Sysmac Studio Ver.1.13 或更高版本。Sysmac Studio Ver.1.12 或更低版本中无法使用 NJ101- □□□□。
*3. CPU 单元 NJ301- □□□□中无 1.00 版本。因此，Sysmac StudioVer.1.01 或更低版本中无法使用 NJ301- □□□□。

单元版本和 Sysmac Studio 的版本不匹配时的规格

NJ 系列 CPU 单元的单元版本和 Sysmac Studio 的版本不匹配时的规格如下。

● Sysmac Studio 的版本较旧时

如果 Sysmac Studio 的版本较旧，可使用的功能为 Sysmac Studio 版本所对应的 CPU 单元的单元版本所配备的功能。

例） CPU 单元的单元版本：Ver.1.04
Sysmac Studio 的版本：Ver.1.04

与 Syamac Studio Ver.1.04 对应的 CPU 单元的单元版本为 Ver.1.03。因此，可使用的功能为 CPU 单元版本 1.03 所配备的功能。 CPU 单元版本 1.04 及以上版本中追加的功能将无法使用。

● CPU 单元的单元版本较旧时

NJ 系列 CPU 单元的单元版本较旧时，请在 Sysmac Studio 项目的设备选择画面中，选择所连接 CPU 单元的单元版本。可使用的功能为 CPU 单元的单元版本所配备的功能。

例） CPU 单元的单元版本：Ver.1.03

Sysmac Studio 的版本：Ver.1.05

与 Syamac Studio Ver.1.05 对应的 CPU 单元的单元版本为 Ver.1.04。但是，连接的 CPU 单元的单元版本为Ver.1.03，因此在项目的设备选择画面的“版本”中，请选择“Ver.1.03”或更低版本。在设备选择画面的 “ 版本 ” 中选择了 “Ver.1.03” 时，可使用的功能为 CPU 单元版本 1.03 所配备的功能。 CPU 单元版本 1.04 及以上版本中追加的功能将无法使用。

使用注意事项

若执行以下条件中的操作，将发生异常。敬请注意。

- Sysmac Studio Ver.1.02以上时，在项目的设备选择画面的“版本”中选择了“1.01”以上，并创建项目。
- 将该项目上传到 Sysmac Studio Ver.1.01。

参考

项目的设备中设定的单元版本

- Sysmac Studio Ver.1.02 以上时，在设备选择画面中设定单元版本的选择。
- 可设定的单元版本为所用 CPU 单元的单元版本或更低版本。例如， CPU 单元的单元版本为 Ver.1.01 时，请选择 “1.00” 或 “1.01”。
- Sysmac Studio 中，项目的设备中设定的单元版本将视为CPU单元的单元版本。例如，项目的设备中设定的单元版本为 Ver.1.00 时，Sysmac Studio 中可使用的功能为单元版本 1.00 及更低版本的功能。
- 项目的设备中设定的单元版本低于传送目标 CPU 单元的单元版本时，可从 Sysmac Studio 传送项目。
- 设备选择画面的详情请参考📖《Sysmac Studio Version1 操作手册 (SBCA-CN5-362C 以上)》。

A-14-2 硬件修订版本和 Sysmac Studio 版本之间的关系

NJ 系列 CPU 单元的硬件修订版本和 Sysmac Studio 的版本有对应关系。在执行时间预测模式下执行 Sysmac Studio 的模拟时，请使用对应版本以上的 Sysmac Studio。Sysmac Studio 的版本较旧时，无法选择该硬件修订版本。

型号	CPU 单元的硬件修订版本	Sysmac Studio 的对应版本
NJ501- □□□□	A	Ver.1.14

A-14-3 通过更新单元版本追加和变更支持功能

下面介绍 NJ 系列 CPU 单元的单元版本更新后，发生追加或变更的支持功能。

● 功能规格的追加和变更

按追加或变更的功能规格，分别列出 CPU 单元的单元版本和 Sysmac Studio 版本。

功能				追加 / 变更	单元版本	Sysmac Studio 的版本	参照处
任务功能	功能	条件成立时执行的任务		追加	Ver.1.03	Ver.1.04	「5-2 任务的概要」（P.5-5）
编程功能	名称空间			追加	Ver.1.01	Ver.1.02	「6-7 名称空间」（P.6-137）
	数据类型	结构体型	成员的偏置指定	追加	Ver.1.01	Ver.1.02	「结构体成员的偏置指定」（P.6-43）
				变更		Ver.1.03	
	库			追加	Ver.1.01	Ver.1.02	「6-8 库」（P.6-142）
运动控制功能	单轴	单轴位置控制	周期性同步绝对定位	追加	Ver.1.03	Ver.1.04	《NJ/NX 系列 CPU 单元 用户手册 运动控制篇（SBCE-CN5-363）》
		单轴控制辅助	参数指定原点复位	追加	Ver.1.03	Ver.1.04	
			数字凸轮开关有效	追加	Ver.1.06	Ver.1.07	
			指令位置补偿	追加	Ver.1.10	Ver.1.12	
			启动速度	追加	Ver.1.05	Ver.1.06	
	轴组	多轴协调控制	轴组循环周期绝对位置控制	追加	Ver.1.01	Ver.1.02	
		多轴协调控制辅助	轴组位置获取	追加	Ver.1.01	Ver.1.02	
			轴组构成轴写入	追加	Ver.1.01	Ver.1.02	
	通用	凸轮	凸轮表生成	追加	Ver.1.08	Ver.1.09	
		参数	轴参数的变更	追加	Ver.1.08	Ver.1.09	
	辅助功能	输入信号的逻辑反转		追加	Ver.1.05	Ver.1.06	
单元（输入输出）管理功能	NX 单元			追加	Ver.1.05	Ver.1.06	《NX 系列 EtherCAT 耦合器单元 用户手册（SBCD-CN5-361）》

功能				追加 / 变更	单元版本	Sysmac Studio 的版本	参照处
通信功能	Ethernet/IP 端口	TCP/IP 应用	FTP 客户端	追加	Ver.1.08	Ver.1.09	📖《NJ/NX 系列 CPU 单元 内置 EtherNet/IP 端口 用户手册 (SBCD-CN5-359)》
	EtherCAT 端口	Packet 监视器*1 (NJ301- □□□□)		追加	Ver.1.10	Ver.1.12	📖《NJ/NX 系列 CPU 单元 内置 EtherCAT 端口 用户手册 (SBCD-CN5-358)》
	通信用指令			变更	Ver.1.08 Ver.1.11	Ver.1.09 Ver.1.15	📖《NJ/NX 系列 指令基准手册 基本篇（SBCA-CN5-360）》
调试功能	微分监控			追加	Ver.1.03	Ver.1.04	📖「8-5-5 微分监控」（P.8-46）
高可靠性功能	自诊断	控制器异常	重要度的变更	追加	Ver.1.03	Ver.1.04	📖「8-7 事件重要程度变更功能」（P.8-64）
安全功能	顾客资产保护 / 防止误操作	保护功能	数据保护功能	追加	Ver.1.01	Ver.1.02	📖「8-4-4 数据保护功能」（P.8-25）
		操作权限的认证功能	组数	变更	Ver.1.01	Ver.1.02	📖「8-4-5 操作权限的认证功能」（P.8-27）
SD 存储卡功能	应用程序	SD 存储卡自动传送功能		追加	Ver.1.03	Ver.1.04	📖「9-4 SD 存储卡自动传送功能」（P.9-26）
		SD 存储卡程序传送功能		追加	Ver.1.11	Ver.1.15	📖「9-5 SD 存储卡程序传送功能」（P.9-28）

功能				追加 / 变更	单元版本	Sysmac Studio 的版本	参照处
备份功能	SD 存储卡的备份功能	操作方法	CPU 单元正面开关	追加	Ver.1.03	Ver.1.04	📖「9-2 SD 存储卡的备份功能」（P.9-12）
			通过系统定义变量进行指示	追加	Ver.1.03	Ver.1.04	
			Sysmac Studio 存储卡画面	追加	Ver.1.03	Ver.1.04	
			专用指令	追加	Ver.1.08	Ver.1.09	
		保护功能	禁止向 SD 存储卡备份	追加	Ver.1.03	Ver.1.04	📖「9-3 SD 存储卡备份禁止功能」（P.9-25）
	Sysmac Studio 控制器备份功能			追加	Ver.1.03	Ver.1.04	📖「9-6 Sysmac Studio 控制器备份功能」（P.9-34）

*1. 仅 CPU 单元 NJ301- □□□□追加。 NJ501- □□□□、 NJ101- □□□□可在所有版本中使用信息包监视。

（注）因 CPU 单元的单元版本更新而发生的各功能模块功能的追加和变更，请参考各功能模块的手册。

● 追加可安装的 CJ 单元

CPU 单元的单元版本更新后，可安装的 CJ 系列单元可能发生追加。

详情请参考📖《NJ 系列 CPU 单元 用户手册 硬件篇 (SBCA-CN5-358)》。

● 基本指令、运动控制指令的追加和变更

CPU 单元的单元版本更新后，可使用的基本指令及运动控制指令可能发生追加或变更。

详情请参考📖《NJ/NX 系列 指令基准手册 基本篇 （SBCA-CN5-360）》及📖《NJ/NX 系列 指令基准手册 运动篇 （SBCE-CN5-364）》。

● 控制器事件的追加和变更

CPU 单元的单元版本更新后，发生的事件可能发生追加或变更。此外，发生异常时的处理方法也可能变更。

详情请参考📖《NJ/NX 系列 故障排除手册 （SBCA-CN5-361）》。

● 系统定义变量的追加和变更

CPU 单元的单元版本更新后，可使用的系统定义变量可能发生追加或变更。

详情请参考📖「A-4 系统定义变量一览」（P.A-38）。

A-14-4 通过单元版本更新改善性能

通过更新 NJ 系列 CPU 单元的单元版本及 Sysmac Studio 的版本，性能得到改善的功能如下所示。

功能				性能值	单元版本	SysmacStudio 的版本
编程	程序容量	数量	POU 实例数 (NJ501- □□□□)	9,000	—	Ver.1.06 以上
				6,000	—	Ver.1.05 以下
			POU 实例数 (NJ301- □□□□)	3,000	Ver.1.04 以上	Ver.1.05 以上
				1,500		Ver.1.04 以下
				2,400	Ver.1.03 以下	Ver.1.05 以上
				1,500		Ver.1.04 以下
	变量容量	有保存属性	变量数量*1 (NJ301- □□□□)	5,000	Ver.1.04 以上	Ver.1.05 以上
				2,500		Ver.1.04 以下
				2,500	Ver.1.03 以下	—
运动控制	控制轴数	最大控制轴数*2*3*4 (NJ301- □□□□)		15 轴	Ver.1.06 以上	Ver.1.07 以上
				8 轴 （NJ301-1200） 4 轴 （NJ301-1100）	上述之外的组合	
		最大单轴控制数 *4*5 (NJ301- □□□□)		单轴 15 轴	Ver.1.06 以上	Ver.1.07 以上
				单轴 8 轴 （NJ301-1200） 单轴 4 轴 （NJ301-1100）	上述之外的组合	
内置 EtherNet/IP 端口	CIP 服务：标签数据链接 （周期链接通信）	Packet 间隔		各连接可设定 1 ～ 10000ms （以 1ms 为单位）	Ver.1.03 以上	—
				各连接可设定 10 ～ 10000ms （以 1ms 为单位）	Ver.1.02 以下	
		单元允许通信带宽		3000pps*6 （含 Heartbeat）	Ver.1.03 以上	—
				1000pps （含 Heartbeat）	Ver.1.02 以下	
	TCP Socket 数			30	Ver.1.03 以上	—
				16	Ver.1.02 以下	
内置 EtherCAT 端口	通信周期*7 (NJ301- □□□□)			500/1000/2000/4000μs	Ver.1.03 以上	—
				1000/2000/4000μs	Ver.1.02 以下	

*1. 仅 CPU 单元 NJ301- □□□□的性能得到改善。 NJ501- □□□□中有保持属性的变量最大数量为 10,000。
*2. 所有轴种类的合计。
*3. 仅 CPU 单元 NJ301- □□□□的性能得到改善。 NJ501- □□□□的最大控制轴数如下。
NJ501- □ 5 □□ : 64 轴、 NJ501- □ 4 □□ :32 轴、 NJ501- □ 3 □□ :16 轴
*4. 使用实轴最大数无变化。
*5. 仅 CPU 单元 NJ301- □□□□的性能得到改善。 NJ501- □□□□的最大单轴控制数如下。
NJ501- □ 5 □□ : 单轴 64 轴、 NJ501- □ 4 □□ : 单轴 32 轴、 NJ501- □ 3 □□ : 单轴 16 轴
*6. 代表 Packet Per Second，表示 1 秒内可处理的收发信息包数。
*7. 仅 CPU 单元 NJ301- □□□□的性能得到改善。 NJ501- □□□□可使用 500/1000/2000/4000μs 的通信周期，NJ101- □□□□可使用 1000/2000/4000μs 的通信周期。

I

索引

1

索引

A

_AlarmFlag （用户异常状态） ....8-59，8-63，A-40，A-82
AT （分配目标）指定 6-57
安全设定 4-5

B

_BackupBusy （备份相关功能执行中标志） .....A-43，A-88
BOOL ....6-32，6-45
BOOTP 服务器连接异常 ....A-61，A-117
BYTE ....6-32，6-45
版本相关信息 A-180
保持 6-59
保持条件 6-66
备份 9-3
备份相关功能执行中标志 ....A-43，A-88
本地变量 6-29
　内部变量 6-29
　输出变量 6-29
　输入变量 6-29
　输入输出变量 6-29
　外部变量 6-29
本地变量表 ....6-10，6-19
编程语言 6-84
变更了保持变量的数据类型时变量的值 6-69
变更 POU 名称或新建 POU 等时保持变量的值 6-67
变量
　种类 6-28
　概要 6-28
变量的种类
　系统定义变量 2-10
　用户定义变量 2-10
　准用户定义变量 2-10
变量的存储器确保方法 A-140
变量的任务间排他控制设定 4-10
变量的属性 6-30
变量访问时间 4-7
变量访问时间设定 5-77
变量和 I/O 的访问 2-13
变量名称 6-57
变量名称等程序相关名称的限制 6-77
边缘 6-64
标签解决异常 ....A-64，A-121
标签数据链接服务 5-47
标签数据链接设定异常 ....A-62，A-119
标签数据链接通信开始开关 ....A-76，A-130
标签数据链接通信停止开关 ....A-77，A-131
标签数据链接通信异常 ....A-64，A-120
标签数据链接通信中 ....A-68，A-124
布尔型 ....6-32，6-45
部分停止故障等级 12-6

C

_Card1Access （SD 存储卡访问中标志）
　....8-15，A-40，A-83
_Card1BkupCmd （SD 存储卡备份指示） ....A-41，A-83
_Card1BkupSta （SD 存储卡备份状态） ....A-41，A-84
_Card1Deteriorated （SD 存储卡使用寿命警告标志）
　....8-14，8-15，A-40，A-83
_Card1Err （SD 存储卡错误标志） ....8-11，8-15，A-40，A-83
_Card1PowerFail （SD 存储卡访问中断电标志）
　....8-15，A-40，A-83
_Card1PrgTransferCmd （SD 存储卡程序传送指示）
　....A-42，A-86
_Card1PrgTransferSta （SD 存储卡程序传送状态）
　....A-42，A-88
_Card1Protect （SD 存储卡写保护标志） ....8-15，A-40，A-82
_Card1Ready （SD 存储卡可使用标志） ....8-15，A-40，A-82
_Card1VefySta （SD 存储卡核对状态） ....A-42，A-85
_CJB_CBU □□ InitSta （CPU 总线单元初始化中标志）
　....8-9，A-48，A-96
_CJB_CBU □□ Restart （CPU 总线单元重启标志）
　....8-9，A-48，A-97
_CJB_ErrSta （I/O 总线异常状态） ....A-47，A-95
_CJB_InRespTm （基本输入单元的输入时间常数）
　....8-8，A-47，A-96
_CJB_IOUnitInfo （基本 I/O 单元信息区域） ....A-48，A-96
_CJB_MaxRackNo （最大装置编号） ....A-47，A-95
_CJB_MaxSlotNo （最大插槽编号） ....A-47，A-95
_CJB_MstrErrSta （I/O 总线主机异常状态） ....A-47，A-95
_CJB_SCU00P □ ChgSta （串行通讯单元 0 号机端口 1/2 用串行端口设定变更中标志） ....A-48，A-98
_CJB_SCU15P □ ChgSta（串行通讯单元 1-15 号机端口 1/2 用串行端口设定变更中标志） ....A-48，A-98
_CJB_SIO □□ InitSta （高功能 I/O 单元初始化中标志）
　....8-9，A-48，A-97
_CJB_SIO □□ Restart （高功能 I/O 单元重启标志）
　....8-9，A-48，A-97
_CJB_UnitErrSta （I/O 总线单元异常状态） ....A-47，A-96
CJ 单元构成 ....1-5，1-7
CJ 单元用存储器 A-135
CIP 通信 1 Identity 信息不正确 ....A-62，A-118
CIP 通信 1 标签解决异常 ....A-65，A-121
CIP 通信 1 标签数据链接设定异常 ....A-62，A-119
CIP 通信 1 标签数据链接通信开始开关 ....A-77，A-130
CIP 通信 1 标签数据链接通信停止开关 ....A-77，A-131
CIP 通信 1 标签数据链接通信异常 ....A-64，A-120
CIP 通信 1 标签数据链接通信中 ....A-68，A-124
CIP 通信 1 登录目标节点信息 ....A-70，A-125
CIP 通信 1 多个开关同时 ON 异常 ....A-65，A-122
CIP 通信 1 建立标签数据链接失败 ....A-63，A-120
CIP 通信 1 目标节点异常信息 ....A-74，A-129
CIP 通信 1 目标 PLC 动作模式 ....A-71，A-127
CIP 通信 1 目标 PLC 异常信息 ....A-72，A-128
CIP 通信 1 所有标签数据链接通信中 ....A-69，A-125
CIP 通信 1 异常 ....A-58，A-112

CIP 通信 1 正常目标节点信息 A-70, A-126
CIP 通信 2 Identity 信息不正确 A-62
CIP 通信 2Identity 信息不正确 A-118
CIP 通信 2 标签解决异常 A-65, A-121
CIP 通信 2 标签数据链接设定异常 A-63, A-119
CIP 通信 2 标签数据链接通信开始开关 A-77, A-131
CIP 通信 2 标签数据链接通信停止开关 A-77, A-131
CIP 通信 2 标签数据链接通信异常 A-64, A-121
CIP 通信 2 标签数据链接通信中 A-69, A-124
CIP 通信 2 登录目标节点信息 A-70, A-126
CIP 通信 2 多个开关同时 ON 异常 A-65, A-122
CIP 通信 2 建立标签数据链接失败 A-63, A-120
CIP 通信 2 目标节点异常信息 A-74, A-129
CIP 通信 2 目标 PLC 动作模式 A-71, A-127
CIP 通信 2 目标 PLC 异常信息 A-72, A-128
CIP 通信 2 所有标签数据链接通信中 A-69, A-125
CIP 通信 2 异常 A-59, A-112
CIP 通信 2 正常目标节点信息 A-71, A-126
CIP 通信异常 A-58, A-112
CPU 单元的动作状态 12-2, 12-3
CPU 单元的写入保护功能 8-29
CPU 单元的状态 2-26
CPU 单元名称 8-30
CPU 单元名称功能 8-30
CPU 单元内部温度超限标志 A-45, A-92
CPU 单元内的数据 2-25
CPU 复位 12-4
CPU 异常 12-4
CPU 总线单元初始化中标志 8-9, A-48, A-96
CPU 总线单元重启标志 8-9, A-48, A-97
_CurrentTime （系统时刻） 8-5, A-38, A-80
CX-Designer 的变量表登录方法 A-153
参数 6-10, 6-19
参照任务 5-69
操作权限 8-27
操作权限的认证功能 8-27
常量 6-62
常数 6-79
布尔型 6-79
持续时间型 6-80
列举型 6-83
日期时刻型 6-81
日期型 6-81
时刻型 6-81
实型数据 6-80
位列型 6-79
整数 6-80
字符串型 6-82
程序 6-8
程序的初始状态 4-9
程序的分配设定 4-9
程序的运行开始时初始状态 5-65
程序的执行顺序 5-65
程序模式 2-29
程序启动时 1 个周期 ON 标志 6-8, A-44, A-90
程序执行顺序 4-9
持续时间型 6-32, 6-45
触发跟踪 8-41
初始设定
EtherCAT 主机功能模块的初始设定 4-17
EtherNet/IP 功能模块的初始设定 4-18
高功能单元的初始设定
高功能单元的初始设定 4-13
运动控制功能模块的初始设定 4-15
初始值 6-60
串行通讯单元 0 号机端口 1/2 用串行端口设定变更中标志 A-48, A-98
串行通讯单元 1-15 号机端口 1/2 用串行端口设定变更中标志 A-48, A-98
创建用户程序时的注意事项 6-146
从站初始化异常 A-52, A-104
从站节点地址重复异常 A-52, A-104
从站异常 A-52, A-102
从站异常表 A-52, A-103
从站应用程序异常 A-52, A-105
存储器全部清除 8-3

D

DATE 6-32, 6-45
DATE_ AND_ TIME 6-33, 6-45
DINT 6-32
DNS 服务器连接异常 A-66, A-123
DNS 设定异常 A-61, A-117
DWORD 6-32
单纯名称 6-138
单元版本 A-45, A-91
单元构成 3-6
当前值变更 8-37
登录从站表 A-54, A-106
登录目标节点信息 A-69, A-125
电池电压低标志 A-45, A-92
电源部异常 12-4
电源连接不正确 12-4
动作模式 2-29
动作模式的确认 2-30
动作模式切换 2-30
动作模式切换时变量值的保持 2-29
动作设定 4-5
动作设定标签 4-4
动作选项 6-129, 6-130
断电保持失败标志 A-43, A-89
断电发生次数 A-43, A-89
多个开关同时 ON 异常 A-65, A-122

E

_EC1_InDataInvalid （输入数据 1 无效） A-54
_EC2_InDataInvalid （输入数据 2 无效） A-55
_EC_CommErrTbl （通信异常从站表） A-53, A-105
_EC_CycleExceeded （EtherCAT 通信周期设定异常） A-53, A-105
_EC_DisableSlavTbl （无效设定从站表） A-54, A-107
_EC_DisconnSlavTbl （脱离指令中的从站表） A-54, A-107
_EC_EntrySlavTbl （网络加入从站表） A-54, A-106
_EC_ErrSta （内置 EtherCAT 异常） A-52, A-102
_EC_InDataInvalid （输入数据无效） A-54, A-108
_EC_InData1Invalid （输入数据 1 无效） A-108

_EC_InData2Invalid （输入数据 2 无效） A-108
_EC_LanHwErr （通信控制器异常） A-52, A-103
_EC_LinkOffErr （链接断开异常） A-52, A-103
_EC_LinkStatus （链接状态） A-54, A-107
_EC_MacAdrErr （MAC 地址异常） A-52, A-103
_EC_MBXSlavTbl （可进行信息通信的从站表）
.....A-54, A-106
_EC_MsgErr （EtherCAT 消息异常） A-52, A-105
_EC_MstrErr （主机异常） A-52, A-102
_EC_NetCfgCmpErr （网络配置核对异常） ...A-52, A-103
_EC_NetCfgErr （网络配置信息异常） A-52, A-103
_EC_NetTopologyErr （网络配置异常） A-52, A-104
_EC_PDActive （过程数据通信状态） A-54, A-107
_EC_PDCommErr （过程数据通信异常） A-52, A-104
_EC_PDSendErr （过程数据发送异常） A-52, A-104
_EC_PDSlavTbl （过程数据通信中从站表） ..A-54, A-106
_EC_PDTimeoutErr （过程数据接收超时异常）
.....A-52, A-104
_EC_PktMonStop （信息包监视停止） A-54, A-107
_EC_PktSaving （信息包数据文件保存中） ...A-54, A-108
_EC_PortErr （通信端口异常） A-52, A-102
_EC_RegSlavTbl （登录从站表） A-54, A-106
_EC_SlavAdrDupErr （从站节点地址重复异常）
.....A-52, A-104
_EC_SlavAppErr （从站应用程序异常） A-52, A-105
_EC_SlavEmergErr （紧急消息检测） A-53, A-105
_EC_SlavErr （从站异常） A-52, A-102
_EC_SlavErrTbl （从站异常表） A-52, A-103
_EC_SlavInitErr （从站初始化异常） A-52, A-104
_EC_StatisticsLogBusy （诊断统计信息日志执行状态标志）
.....A-55, A-109
_EC_StatisticsLogErr （诊断统计信息日志异常结束标志）
.....A-55
_EC_StatisticsLogCycleSec （诊断统计信息日志保存周期）
.....A-55, A-109
_EC_StatisticsLogEnable （诊断统计信息日志许可）
.....A-55, A-109
_EC_StatisticsLogErr （诊断统计信息日志异常结束标志）
.....A-109
_EIP1_BootpErr （通信端口 1 BOOTP 服务器连接异常）
.....A-61, A-117
_EIP1_CipErr （CIP 通信 1 异常） A-58, A-112
_EIP1_EstbTargetSta （CIP 通信 1 正常目标节点信息）
.....A-70, A-126
_EIP1_EtnCfgErr （通信端口 1 Ethernet 基本设定异常）
.....A-60, A-115
_EIP1_EtnOnlineSta （通信端口 1 在线） A-68, A-123
_EIP1_IdentityErr （CIP 通信 1 Identity 信息不正确）
.....A-62, A-118
_EIP1_IPAdrCfgErr （通信端口 1 IP 地址设定异常）
.....A-60, A-115
_EIP1_IPAdrDupErr （通信端口 1 IP 地址重复异常）
.....A-61, A-116
_EIP1_LanHwErr （通信端口 1 通信控制器异常）
.....A-60, A-114
_EIP1_MacAdrErr （通信端口 1 MAC 地址异常）
.....A-59, A-113
_EIP1_MultiSwONErr （CIP 通信 1 多个开关同时 ON 异常）
.....A-65, A-122
_EIP1_PortErr （通信端口 1 异常） A-57, A-111
_EIP1_RegTargetSta （CIP 通信 1 登录目标节点信息）
.....A-70, A-125
_EIP1_TagAdrErr （CIP 通信 1 标签解决异常）
.....A-65, A-121
_EIP1_TargetNodeErr （CIP 通信 1 目标节点异常信息）
.....A-74, A-129
_EIP1_TargetPLCErr （CIP 通信 1 目标 PLC 异常信息）
.....A-72, A-128
_EIP1_TargetPLCModeSta（CIP 通信 1 目标 PLC 动作模式）
.....A-71, A-127
_EIP1_TDLinkAllRunSta（CIP 通信 1 所有标签数据链接通信中） A-69, A-125
_EIP1_TDLinkCfgErr （CIP 通信 1 标签数据链接设定异常）
.....A-62, A-119
_EIP1_TDLinkErr （CIP 通信 1 标签数据链接通信异常）
.....A-64, A-120
_EIP1_TDLinkOpnErr （CIP 通信 1 建立标签数据链接失败）
.....A-63, A-120
_EIP1_TDLinkRunSta （CIP 通信 1 标签数据链接通信中）
.....A-68, A-124
_EIP1_TDLinkStartCmd （CIP 通信 1 标签数据链接通信开始开关） A-77, A-130
_EIP1_TDLinkStopCmd （CIP 通信 1 标签数据链接通信停止开关） A-77, A-131
_EIP2_BootpErr （通信端口 2 BOOTP 服务器连接异常）
.....A-62, A-117
_EIP2_CipErr （CIP 通信 2 异常） A-59, A-112
_EIP2_EstbTargetSta （CIP 通信 2 正常目标节点信息）
.....A-71, A-126
_EIP2_EtnCfgErr （通信端口 2 Ethernet 基本设定异常）
.....A-60, A-115
_EIP2_EtnOnlineSta （通信端口 2 在线） A-68, A-123
_EIP2_IdentityErr （CIP 通信 2 Identity 信息不正确）
.....A-62, A-118
_EIP2_IPAdrCfgErr （通信端口 2 IP 地址设定异常）
.....A-60, A-116
_EIP2_IPAdrDupErr （通信端口 2 IP 地址重复异常）
.....A-61, A-116
_EIP2_LanHwErr （通信端口 2 通信控制器异常）
.....A-60, A-114
_EIP2_MacAdrErr （通信端口 2 MAC 地址异常） A-59
_EIP2_MultiSwONErr （CIP 通信 2 多个开关同时 ON 异常）
.....A-65, A-122
_EIP2_PortErr （通信端口 2 异常） A-57, A-111
_EIP2_RegTargetSta （CIP 通信 2 登录目标节点信息）
.....A-70, A-126
_EIP2_TagAdrErr （CIP 通信 2 标签解决异常）
.....A-65, A-121
_EIP2_TargetNodeErr （CIP 通信 2 目标节点异常信息）
.....A-74, A-129
_EIP2_TargetPLCErr （CIP 通信 2 目标 PLC 异常信息）
.....A-72, A-128
_EIP2_TargetPLCModeSta（CIP 通信 2 目标 PLC 动作模式）
.....A-71, A-127
_EIP2_TDLinkAllRunSta（CIP 通信 2 全所有标签数据链接通信中） A-69, A-125
_EIP2_TDLinkCfgErr （CIP 通信 2 标签数据链接设定异常）
.....A-63, A-119

_EIP2_TDLinkErr （CIP 通信 2 标签数据链接通信异常） A-64, A-121
_EIP2_TDLinkOpnErr （CIP 通信 2 建立标签数据链接失败） A-63, A-120
_EIP2_TDLinkRunSta （CIP 通信 2 标签数据链接通信中） A-69, A-124
_EIP2_TDLinkStartCmd （CIP 通信 2 标签数据链接通信开始开关） A-77, A-131
_EIP2_TDLinkStopCmd （CIP 通信 2CIP 标签数据链接通信停止开关） A-77, A-131
_EIP_BootpErr （BOOTP 服务器连接异常） . A-61, A-117
_EIP_CipErr （CIP 通信异常） A-58, A-112
_EIP_DNSCfgErr （DNS 设定异常） A-61, A-117
_EIP_DNSSrvErr （DNS 服务器连接异常） ... A-66, A-123
_EIP_ErrSta （内置 EtherNet/IP 异常） A-56, A-110
_EIP_EstbTargetSta （正常目标节点信息） ... A-70, A-126
_EIP_EtnCfgErr （Ethernet 基本设定异常） .. A-60, A-114
_EIP_EtnOnlineSta （在线） A-68, A-123
_EIP_IdentityErr （Identity 信息不正确） A-62, A-118
_EIP_IPAdrCfgErr （IP 地址设定异常） A-60, A-115
_EIP_IPAdrDupErr （IP 地址重复异常） A-61, A-116
_EIP_IPRTblErr （TCP/IP 扩展设定异常） A-62, A-118
_EIP_LanHwErr （通信控制器异常） A-59, A-114
_EIP_MacAdrErr （MAC 地址异常） A-59, A-113
_EIP_MultiSwONErr （多个开关同时 ON 异常） A-65, A-122
_EIP_NTPResult （NTP 的动作信息） A-74
_EIP_NTPResult.ExecNormal （NTP 的动作结果） A-75, A-130
_EIP_NTPResult.ExecTime （NTP 的最终动作时间） A-75, A-130
_EIP_NTPSrvErr （NTP 服务器连接异常） ... A-66, A-122
_EIP_PortErr （通信端口异常） A-56, A-110
_EIP_RegTargetSta （登录目标节点信息） A-69, A-125
_EIP_TagAdrErr （标签解决异常） A-64, A-121
_EIP_TargetNodeErr （目标节点异常信息） .. A-73, A-129
_EIP_TargetPLCErr （目标 PLC 异常信息） .. A-72, A-128
_EIP_TargetPLCModeSta （目标 PLC 动作模式） A-71, A-127
_EIP_TcpAppCfgErr （TCP/IP 设定异常） A-66, A-122
_EIP_TcpAppErr （TCP 应用通信异常） A-59, A-113
_EIP_TDLinkAllRunSta （所有标签数据链接通信中） A-69, A-124
_EIP_TDLinkCfgErr （标签数据链接设定异常） A-62, A-119
_EIP_TDLinkErr （标签数据链接通信异常） .. A-64, A-120
_EIP_TDLinkOpnErr （建立标签数据链接失败） A-63, A-119
_EIP_TDLinkRunSta （标签数据链接通信中） A-68, A-124
_EIP_TDLinkStartCmd （标签数据链接通信开始开关） A-76, A-130
_EIP_TDLinkStopCmd （标签数据链接通信停止开关） A-77, A-131
EN 6-20
ENO 6-21
_ErrSta （控制器异常状态） 5-79, A-40, A-82
EtherCAT 从站构成 3-4
EtherCAT 从站或轴的有效 / 无效切换 A-156
EtherCAT 通信周期设定异常 A-53, A-105
EtherCAT 网络配置 1-5, 1-7
EtherCAT 消息异常 A-52, A-105
EtherCAT 主机功能模块的初始设定 4-17
EtherNet/IP 功能模块的初始设定 4-18
Ethernet 基本设定异常 A-60, A-114

F

FB 名称或指令名称 6-10
FINS/TCP 连接状态 A-44, A-91
_FINSTCPConnSta （FINS/TCP 连接状态） .. A-44, A-91
发生原因 12-16, 12-17, 12-28
FTP 服务器功能 8-14
FTP 客户端通信指令 8-13
FUN 定义名称或指令名称 6-18
返回值 6-21
范围指定 ((,)) 6-55
风扇转速降低标志 A-45, A-92

G

GetCJBError （获取 I/O 总线异常状态） 12-12
GetECError （获取 EtherCAT 异常状态） 12-12
GetEIPError （获取 EtherNet/IP 异常状态） 12-12
GetMCError （获取运动控制异常状态） 12-12
GetPLCError （获取 PLC 异常状态） 12-12
高功能单元的初始设定 4-13
高功能 I/O 单元初始化中标志 8-9, A-48, A-97
高功能 I/O 单元重启标志 8-9, A-48, A-97
各种状态下的动作 2-28
更新任务 5-69
功能块
  从外部参照变量 6-17
  内部变量的规格 6-12
  启动条件 6-16
  ST 语言的调用表述 6-11
  实例 6-14
功能块 (FB)
  参数省略时的动作 6-24
  参数异常时的动作 6-25
  EN、 ENO 的使用 / 省略可否 6-24
功能块 (FB) 构成 6-9
功能块 (FB) 详情 6-9
功能模块
  EtherCAT 主机功能模块 2-4
  EtherNet/IP 功能模块 2-4
  PLC 功能模块 2-4
  运动控制功能模块 2-4
共通异常状态 A-49, A-99
固定周期任务 5-35
固定周期任务 （执行优先度 16、 17、 18）的设定项目 5-67
固定周期任务 （执行优先度 5）的设定项目 5-67
固定周期任务 （执行优先度 16、 17、 18） 5-17
固定周期任务 （执行优先度 5） 5-16
故障 12-75
故障排除 12-12, 12-13
故障排除功能 12-11
规格
  功能规格 A-7
  性能规格 A-4
  一般规格 A-3

1

规格概要 1-9
过程数据发送异常 A-52, A-104
过程数据接收超时异常 A-52, A-104
过程数据通信异常 A-52, A-104
过程数据通信中从站表 A-54, A-106
过程数据通信状态 A-54, A-107

H

_HardwareRevision （硬件修订版本） A-45, A-91
函数 （FUN）
参数省略时的动作 6-24
参数异常时的动作 6-25
EN、 ENO 的使用 / 省略可否 6-24
构成 6-18
内部变量的规格 6-20
启动条件 6-23
ST 语言的表述 6-19
详情 6-18
核对 9-3
恢复 9-3
获取 EtherCAT 异常状态 12-12
获取 EtherNet/IP 异常状态 12-12
获取 I/O 总线异常状态 12-12
获取 PLC 异常状态 12-12
获取运动控制异常状态 12-12

I

I/O 端口 3-8
I/O 端口名称 3-9
I/O 刷新周期超限 5-80
I/O 总线单元异常状态 A-47, A-96
I/O 总线异常状态 A-47, A-95
I/O 总线主机异常状态 A-47, A-95
Identity 信息不正确 A-62, A-118
INT 6-32
IP 地址设定异常 A-60, A-115
IP 地址重复异常 A-61, A-116

J

基本构成 1-5, 1-7
基本 I/O 单元信息区域 A-48, A-96
基本数据型 6-31
基本输入单元的输入时间常数 8-8, A-47, A-96
建立标签数据链接失败 A-63, A-119
监视信息 12-6
接点 6-85
结构体成员的偏置指定 6-43
结构体型 6-41
紧急消息检测 A-53, A-105
进位标志 6-8, 6-130, A-44, A-90

K

可进行信息通信的从站表 A-54, A-106
可在线连接的构成 10-5
控制器设定 4-4
控制器事件 8-52, 8-57
控制器异常状态 5-79, A-40, A-82
库 6-142

L

LD 6-84
LED 12-2, 12-10
LINT 6-32
LREAL 6-32
LWORD 6-32
类别 8-53
离线调试 7-3
联合体型 6-46
联机 ST 6-89
链接断开异常 A-52, A-103
连接线 [连接线] 6-85
链接状态 A-54, A-107
连续跟踪 8-41
列举型 （ENUM） 6-49

M

MAC 地址异常 A-52, A-59, A-103, A-113
_MC1_AX （轴变量） A-50, A-101
_MC1_GRP （轴组变量） A-50, A-100
_MC2_AX （轴变量） A-51, A-101
_MC2_GRP （轴组变量） A-50, A-100
_MC_AX （轴变量） A-50, A-101
_MC_AX_ErrSta （轴异常状态） A-49, A-99
_MC_COM （通用变量） A-49, A-100
_MC_ComErrSta （共通异常状态） A-49, A-99
MC 调试 7-9
_MC_ErrSta （运动控制功能模块的异常状态） A-49, A-99
_MC_GRP （轴组变量） A-49, A-100
_MC_GRP_ErrSta （轴组异常状态） A-49, A-99
名称空间 6-137
名称空间的使用声明 6-139
模拟程序 7-6
模拟功能 7-3
模拟器的启动和停止 7-3
模拟速度 7-7
默认的 CAST 6-121
目标节点异常信息 A-73, A-129
目标 PLC 动作模式 A-71, A-127
目标 PLC 异常信息 A-72, A-128
母线 6-84

N

NTP 8-5
NTP 的动作结果 A-75, A-130
NTP 的动作信息 A-74
NTP 的最终动作时间 A-75, A-130
NTP 服务器连接异常 A-66, A-122
内部变量 6-20
内置 EtherCAT 异常 A-52, A-102
内置 EtherNet/IP 异常 A-56, A-110

P

P_CY （进位标志） ..... 6-8，6-130，A-44，A-90
P_First_Run （程序启动时 1 个周期 ON 标志）
..... 6-8，A-44，A-90
P_First_RunMode （运行开始时 1 个周期 ON 标志）
..... 6-8，A-44，A-90
_PLC_ErrSta （PLC 功能模块异常状态） ..... A-46，A-94
PLC 功能模块 4-4
PLC 功能模块异常状态 ..... A-46，A-94
_PLC_TraceSta.IsComplete （追踪完成时 ON 标志）
..... 8-46，A-46，A-93
_PLC_TraceSta.IsStart （追踪执行中 ON 标志）
..... 8-46，A-46，A-93
_PLC_TraceSta.IsTrigger （追踪触发监视标志）
..... 8-46，A-46，A-93
_PLC_TraceSta.ParamErr （追踪参数异常标志）
..... 8-46，A-46，A-93
P_Off （始终 OFF 标志） ..... A-44，A-90
P_On （始终 ON 标志） ..... A-44，A-89
_Port_isAvailable （网络通信指令可执行标志） .. A-44，A-91
_Port_numUsingPort （使用端口数量） ..... A-44，A-91
POU
程序 6-6
功能块 (FB) 6-6
函数 (FUN) 6-6
POU （程序构成单位） 6-5
POU 的限制事项 6-26
_PowerOnCount （断电发生次数） ..... A-43，A-89
_PowerOnHour （通电时间） ..... A-43，A-89
P_PRGER （指令错误发生标志）
.....6-8，6-132，6-135，A-44，A-90
排列指定 （ARRAY OF） 6-51
派生数据型 6-41

Q

嵌套的层级 6-27
强制值刷新 8-33
轻度故障等级 12-6
全部停止故障等级 12-6
全局变量 6-29
确保任务间变量值的同时性 5-69

R

REAL .....6-32，6-45
_RetainFail （断电保持失败标志） ..... A-43，A-89
RUN 接点 8-6
_ ＜任务名称＞ _Active （任务执行中标志）
..... 5-7，A-39，A-80
_ ＜任务名称＞ _ExceedCount （任务周期超出次数）
..... 5-7，5-79，A-39，A-81
_ ＜任务名称＞ _Exceeded （任务周期超出标志）
.....5-7，5-79，5-85，A-39，A-81
_ ＜任务名称＞ _ExecCount （任务执行次数）
..... 5-7，5-85，A-39，A-81
_ ＜任务名称＞ _LastExecTime （上次任务执行时间）
..... 5-7，A-39，A-80
_ ＜任务名称＞ _MaxExecTime （任务执行时间最大值）
..... 5-7，A-39，A-80
_ ＜任务名称＞ _MinExecTime （任务执行时间最小值）
..... 5-7，A-39，A-80
任务
程序的分配 5-65
分配到 I/O 刷新的任务中 5-57
NJ 系列的规格 5-28
NX 系列的规格 5-9
执行优先度 .....5-11，5-29
任务间排他控制设定 5-76
任务间排他控制指令 5-72
任务类型 .....4-6，5-15，5-34
任务名称 .....4-7，4-8
任务设计示例 5-90
任务设置 4-6
任务实际处理时间 5-86
任务相关的系统定义变量 5-7
任务相关的指令 5-7
任务执行超时 5-80
任务执行超时时间 4-7
任务执行次数 .....5-7，5-85，A-39，A-81
任务执行时间 5-86
任务执行时间的监视 5-83
任务执行时间的确认 5-87
任务执行时间最大值 ..... 5-7，A-39，A-80
任务执行时间最小值 ..... 5-7，A-39，A-80
任务执行中标志 ..... 5-7，A-39，A-80
任务执行状态的监视 5-82
任务执行状态监视 4-10
任务中可分配的 POU 5-65
任务周期超出标志 .....5-7，5-79，5-85，A-39，A-81
任务周期超出次数 .....5-7，5-79，A-39，A-81
任务周期超限 5-79
任务周期超限检测 4-7
日期时刻型 .....6-33，6-45
日期型 .....6-32，6-45
软件的动作 2-5
软件构成 2-4

S

SD 存储卡
操作指令 8-13
功能一览 8-10
规格 8-11
寿命检测功能 8-14
文件访问的排他控制 8-18
自检功能 8-16
SD 存储卡备份禁止功能 9-25
SD 存储卡备份指示 ..... A-41，A-83
SD 存储卡备份状态 ..... A-41，A-84
SD 存储卡程序传送功能 .....4-5，9-28
SD 存储卡程序传送指示 ..... A-42，A-86
SD 存储卡程序传送状态 ..... A-42，A-88
SD 存储卡错误标志 .....8-11，8-15，A-40，A-83
SD 存储卡的备份功能 9-12
SD 存储卡访问中标志 ..... 8-15，A-40，A-83
SD 存储卡访问中断电标志 ..... 8-15，A-40，A-83

1

SD 存储卡核对状态 A-42, A-85
SD 存储卡可使用标志 8-15, A-40, A-82
SD 存储卡设定 4-5
SD 存储卡使用寿命警告标志 8-14, 8-15, A-40, A-83
SD 存储卡写保护标志 8-15, A-40, A-82
SD 存储卡自动传送功能 9-26
_SelfTest_HighTemperature （CPU 单元内部温度超限标志）
.....A-45, A-92
_SelfTest_LowBattery （电池电压低标志） A-45, A-92
_SelfTest_LowFanRevolution （风扇转速降低标志）
.....A-45, A-92
SINT 6-32, 6-45
STRING 6-33, 6-45
ST 语言 6-90
CASE 6-103
代入 6-97
EXIT 6-114
FB 调用 6-115
FUN 调用 6-118
FOR 6-106
各运算符的操作对象中可使用的数据类型 6-96
IF （1 个条件） 6-99
IF （多个条件） 6-101
REPEAT 6-112
RETURN 6-98
WHILE 6-110
语句的关键词 6-93
运算符 6-94
Sysmac Studio 备份文件的导入 / 导出功能 9-38
Sysmac Studio 变量 / 存储器的备份功能 9-39
Sysmac Studio 控制器备份功能 9-34
上次任务执行时间 5-7, A-39, A-80
设备变量 2-11, 3-10
设定控制 I/O 的任务 4-8
事件代码 8-53, 12-16, 12-17, 12-28
事件发生位置 8-52
事件名称 12-16, 12-17, 12-28
事件任务
设定项目 5-18, 5-36, 5-68
ActEventTask 指令 5-19, 5-37
变量的条件式一致 5-20, 5-39
事件日志功能 8-51
事件日志设定 4-5
事件设定表 8-60
事件重要程度 8-54
事件重要程度变更功能 8-64
时刻型 6-32, 6-45
实例 6-14
实例名称 6-10
实型数据 6-32, 6-45
使用端口数量 A-44, A-91
时钟功能 8-3
始终 OFF 标志 A-44, A-90
始终 ON 标志 A-44, A-89
输出变量 6-20
数据保护功能 8-25
数据的保持 / 非保持属性一览 A-132
数据类型 6-31
数据类型的转换 6-40
数据追踪 8-41
数据追踪的动作 8-43
数据追踪的规格 8-42
输入变量 6-20
输入输出变量 6-20
输入输出响应时间 5-91
输入数据 1 无效 A-54, A-108
输入数据 2 无效 A-55, A-108
输入数据无效 A-54, A-108
顺序控制的原理 2-21
顺序控制和运动控制的同步 2-23
顺序控制和运动控制的原理 2-19
算法 6-10, 6-18
所有标签数据链接通信中 A-69, A-124

T

TCP/IP 扩展设定异常 A-62, A-118
TCP/IP 设定异常 A-66, A-122
TCP 应用通信异常 A-59, A-113
TIME 6-32, 6-45
TIME_ OF_DAY 6-32, 6-45
梯形图语言 6-84
梯形图中 FB、 FUN 的连接 6-86
条件标志 6-130
调试程序 7-6
通电时间 A-43, A-89
通过当前值变更，变更向外部的输出状态时的注意事项
..... 8-38
通信端口 1 BOOTP 服务器连接异常 A-61, A-117
通信端口 1 Ethernet 基本设定异常 A-60, A-115
通信端口 1 IP 地址设定异常 A-60, A-115
通信端口 1 IP 地址重复异常 A-61
通信端口 1 IP 地址重复异常 A-116
通信端口 1 MAC 地址异常 A-59, A-113
通信端口 1 通信控制器异常 A-60, A-114
通信端口 1 异常 A-57, A-111
通信端口 1 在线 A-123
通信端口 2 BOOTP 服务器连接异常 A-62, A-117
通信端口 2 Ethernet 基本设定异常 A-60, A-115
通信端口 2 IP 地址设定异常 A-60, A-116
通信端口 2 IP 地址重复异常 A-61
通信端口 2 IP 地址重复异常 A-116
通信端口 2 MAC 地址异常 A-59
通信端口 2 通信控制器异常 A-60, A-114
通信端口 2 异常 A-57, A-111
通信端口 2 在线 A-123
通信端口异常 A-52, A-56, A-102, A-110
通信控制器异常 A-52, A-59, A-103, A-114
通信异常从站表 A-53, A-105
通用变量 A-49, A-100
凸轮数据变量 2-11
推测原因 12-16, 12-17, 12-28
脱离指令中的从站表 A-54, A-107

U

UDINT 6-32

UINT 6-32
ULINT 6-32
_UnitVersion （单元版本） A-45, A-91
USINT 6-32

W

WDT 异常 12-4
WORD 6-32
外部变量 6-20
完全限定名称 6-138
网络公开 6-63
网络加入从站表 A-54, A-106
网络配置 1-6, 1-8
网络配置核对异常 A-52, A-103
网络配置信息异常 A-52, A-103
网络配置异常 A-52, A-104
网络通信指令可执行标志 A-44, A-91
微分监控 8-46
位列型 6-32, 6-45
位列型、实型数据、字符串型的数据格式 6-36
文本 6-79
无法在线连接的构成 10-6
无效设定从站表 A-54, A-107
无用户程序复原信息传送功能 8-23

X

系统初始化异常 12-4
系统定义变量 2-12, 6-28, 12-12
系统定义变量的个别规格 A-79
系统定义变量一览 A-38
系统服务 5-47
系统服务监视设定 4-5
系统服务时间不足 5-81
系统时刻 8-5, A-38, A-80
线圈 6-85
项目文件整体的保护功能 8-24
信息包监视停止 A-54, A-107
信息包数据文件保存中 A-54, A-108
序列 ID 功能 8-32

Y

一般信息 12-6
异常的确认 12-5, 12-10
异常的重要程度 12-6
异常内容 12-30
异常一览 12-16
异常状态 A-78
CPU 单元可动作的异常 12-6
CPU 单元可动作的异常确认方法 12-10
CPU 单元无法动作的异常 12-4
CPU 单元以外的异常及其处理 12-75
NJ 系列的状态概要 12-2
通过各功能模块的异常状态获取指令确认 [异常状态]
通过各功能模块的异常状态获取指令确认 12-12
通过 LED 确认 12-10
通过 Sysmac Studio 的故障排除功能确认 12-11
通过系统定义变量确认 12-12
通过显示器的故障排除确认 12-12
异常状态变量 12-12
异常状态获取指令 12-12
硬件修订版本 A-45, A-91
用户程序 / 控制器配置和设定传送错误 12-36
用户程序执行用 ID 8-20
用户定义变量 6-28
种类 6-29
用户事件 8-52, 8-58
用户异常状态 8-59, 8-63, A-40, A-82
与 EtherCAT 从站的访问 2-13
语法错误 6-134
与高功能单元的访问 2-17
与基本 I/O 单元的访问 2-16
运动控制的原理 2-22
运动控制功能模块的初始设定 4-15
运动控制功能模块的异常状态 A-49, A-99
运动控制系统变量 2-12
运动控制周期超限 5-80
运行开始时 1 个周期 ON 标志 6-8, A-44, A-90
运行模式 2-29
运行模式改写 4-6
运行中输出功能 8-6

Z

在程序中使用强制值时的注意事项 8-36
在线 A-68, A-123
在线编辑 8-39
诊断统计信息日志保存周期 A-55, A-109
诊断统计信息日志许可 A-55, A-109
诊断统计信息日志异常结束标志 A-55, A-109
诊断统计信息日志执行状态标志 A-55, A-109
正常目标节点信息 A-70, A-126
整数型 6-32, 6-45
支持软件 1-5, 1-6, 1-7, 1-8
指令 6-128
指令错误 6-131
指令错误发生标志 6-8, 6-132, 6-135, A-44, A-90
轴 3-17
轴变量 3-17, A-50, A-51, A-101
轴及轴变量的创建和使用方法 3-19
周期 / 执行条件 4-7
轴异常状态 A-49, A-99
轴组变量 3-17, A-49, A-50, A-100
轴组异常状态 A-49, A-99
主固定周期任务 5-15, 5-34
设定项目 5-66
主机异常 A-52, A-102
主控 6-147
追踪参数异常标志 8-46, A-46, A-93
追踪触发监视标志 8-46, A-46, A-93
追踪完成时 ON 标志 8-46, A-46, A-93
追踪执行中 ON 标志 8-46, A-46, A-93
准用户定义变量 2-11, 6-28
字符串型 6-33, 6-45
最大插槽编号 A-47, A-95
最大装置编号 A-47, A-95

购买欧姆龙产品的客户须知

承诺事项

承蒙对欧姆龙株式会社(以下简称“本公司”)产品的一贯厚爱和支持，藉此机会再次深表谢意。

如果未特别约定，无论贵司从何处购买的产品，都将适用本承诺事项中记载的事项。

请在充分了解这些注意事项基础上订购。

1. 定义

本承诺事项中的术语定义如下。

(1) “本公司产品”：是指“本公司”的FA系统机器、通用控制器、传感器、电子/结构部件。

(2) “产品目录等”：是指与“本公司产品”有关的欧姆龙综合产品目录、FA系统设备综合产品目录、安全组件综合产品目录、电子/机构部件综合产品目录以及其他产品目录、规格书、使用说明书、操作指南等，包括以电子数据方式提供的资料。

(3) “使用条件等”：是指在“产品目录等”资料中记载的“本公司产品”的使用条件、额定值、性能、运行环境、操作使用方法、使用时的注意事项、禁止事项以及其他事项。

(4) “客户用途”：是指客户使用“本公司产品”的方法，包括将“本公司产品”组装或运用到客户生产的部件、电子电路板、机器、设备或系统等产品中。

(5) “适用性等”：是指在“客户用途”中“本公司产品”的(a)适用性、(b)动作、(c)不侵害第三方知识产权、(d)法规法令的遵守以及(e)满足各种规格标准。

2. 关于记载事项的注意事项

对“产品目录等”中的记载内容，请理解如下要点。

(1) 额定值及性能值是在单项试验中分别在各条件下获得的值，并不构成对各额定值及性能值的综合条件下获得值的承诺。

(2) 提供的参考数据仅作为参考，并非可在该范围内一直正常运行的保证。

(3) 应用示例仅作参考，不构成对“适用性等”的保证。

(4) 如果因技术改进等原因，“本公司”可能会停止“本公司产品”的生产或变更“本公司产品”的规格。

3. 使用时的注意事项

选用及使用本公司产品时请理解如下要点。

(1) 除了额定值、性能指标外，使用时还必须遵守“使用条件等”。

(2) 客户应事先确认“适用性等”，进而再判断是否选用“本公司产品”。“本公司”对“适用性等”不做任何保证。

(3) 对于“本公司产品”在客户的整个系统中的设计用途，客户应负责事先确认是否已进行了适当配电、安装等事项。

(4) 使用“本公司产品”时，客户必须采取如下措施：(i)相对额定值及性能指标，必须在留有余量的前提下使用“本公司产品”，并采用冗余设计等安全设计(ii)所采用的安全设计必须确保即使“本公司产品”发生故障时也可将“客户用途”中的危险降到最小程度、(iii)构建随时提示使用者危险的完整安全体系、(iv)针对“本公司产品”及“客户用途”定期实施各项维护保养。

(5) “本公司产品”是作为应用于一般工业产品的通用产品而设计生产的。如果客户将“本公司产品”用于以下所列用途，则本公司对产品不作任何保证。但“本公司”已表明可用于特殊用途，或已与客户有特殊约定时，另行处理。

(a) 必须具备很高安全性的用途(例：核能控制设备、燃烧设备、航空/宇宙设备、铁路设备、升降设备、娱乐设备、医疗设备、安全装置、其他可能危及生命及人身安全的用途)

(b) 必须具备很高可靠性的用途(例：燃气、自来水、电力等供应系统、24小时连续运行系统、结算系统、以及其他处理权利、财产的用途等)

(c) 具有苛刻条件或严酷环境的用途(例：安装在室外的设备、会受到化学污染的设备、会受到电磁波影响的设备、会受到振动或冲击的设备等)

(d) “产品目录等”资料中未记载的条件或环境下的用途

(6) 除了不适用于上述3.(5)(a)至(d)中记载的用途外，“本产品目录等资料中记载的产品”也不适用于汽车(含二轮车，以下同)。请勿配置到汽车上使用。关于汽车配置用产品，请咨询本公司销售人员。

4. 保修条件

“本公司产品”的保修条件如下。

(1) 保修期限　自购买之日起1年。(但是，“产品目录等”资料中有明确说明时除外。)

(2) 保修内容　对于发生故障的“本公司产品”，由“本公司”判断并可选择以下其中之一方式进行保修。

(a) 在本公司的维修保养服务点对发生故障的“本公司产品”进行免费修理(但是对于电子、结构部件不提供修理服务。)

(b) 对发生故障的“本公司产品”免费提供同等数量的替代品

(3) 当故障因以下任何一种情形引起时，不属于保修的范围。

(a) 将“本公司产品”用于原本设计用途以外的用途

(b) 超过“使用条件等”范围的使用

(c) 违反本注意事项“3.使用时的注意事项”的使用

(d) 非因“本公司”进行的改装、修理导致故障时

(e) 非因“本公司”出品的软件导致故障时

(f) “本公司”生产时的科学、技术水平无法预见的原因

(g) 除上述情形外的其它原因，如“本公司”或“本公司产品”以外的原因(包括天灾等不可抗力)

5. 责任限制

本承诺事项中记载的保修是关于“本公司产品”的全部保证。对于因“本公司产品”而发生的其他损害，“本公司”及“本公司产品”的经销商不负任何责任。

6. 出口管理

客户若将“本公司产品”或技术资料出口或向境外提供时，请遵守中国及各国关于安全保障进出口管理方面的法律、法规。否则，“本公司”有权不予提供“本公司产品”或技术资料。

201610

注：规格如有变更，恕不另行通知。请以最新产品说明书为准。

欧姆龙自动化(中国)有限公司

http://www.fa.omron.com.cn　　咨询热线：**400-820-4535**

欧姆龙自动化(中国)有限公司　版权所有　2016