

欢迎参加 **OMRON** NJ专题班



第一章 EtherCAT

- 第一节 EtherCAT概述
- 第二节 EtherCAT机制
- 第三节 EtherCAT协议
- 第四节 EtherCAT通信类型
- 第五节 PDO mapping
- 第六节 DC功能
- 第七节 EtherCAT通讯的控制状态
- 第八节 拓扑结构
- 第九节 通信建立所需文件

第一章 EtherCAT

- EtherCAT概述
- EtherCAT(Ethernet Control Automation Technology)是基于以太网系统但是动作更快，通讯性能更高效的一种高性能工作网络系统。
- EtherCAT网络的特点：
 - 能处理100Mbps超高速的通讯。
 - 与以太网极高的兼容性

第一章 EtherCAT

- EtherCAT机制

- (1) EtherCAT的网络中，数据并不是一一对应发送给个别节点的，而是通过以太网帧将这些数据传送到从站节点去的
- (2) 以太网帧是通过EtherCAT主站向EtherCAT从站发送，在通过从站时是不停留等待的。每当到达整个网络最后一台从站，帧就会被从站发回，直接穿过所有的从站并最终回到EtherCAT主站

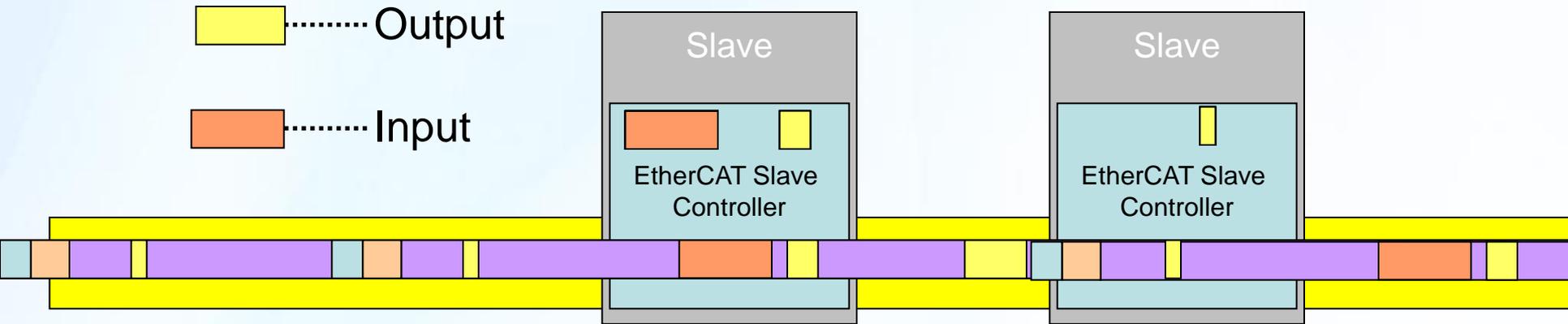
第一章 EtherCAT

数据的读写都是直接从以太网帧里操作。

在以太网帧里会分配出一个区域（最大1498字节）用来给所有的从站做数据交换（读和写）。

每一个从站注册后都会在帧里分到属于其自己的数据地址位置

示意图



第一章 EtherCAT

- EtherCAT协议
 - (1) NJ系列CPU内置EtherCAT口使用COE(CAN application protocol over EtherCAT)协议
 - (2) COE协议能在EtherCAT网络中实现CANOpen通讯。（CANopen是一种由Cia规定的独立协议）
 - (3) COE协议里，从站的参数和控制信息被定义成一种数据规格，这种规格叫做对象字典（object dictionary/OD）

第一章 EtherCAT

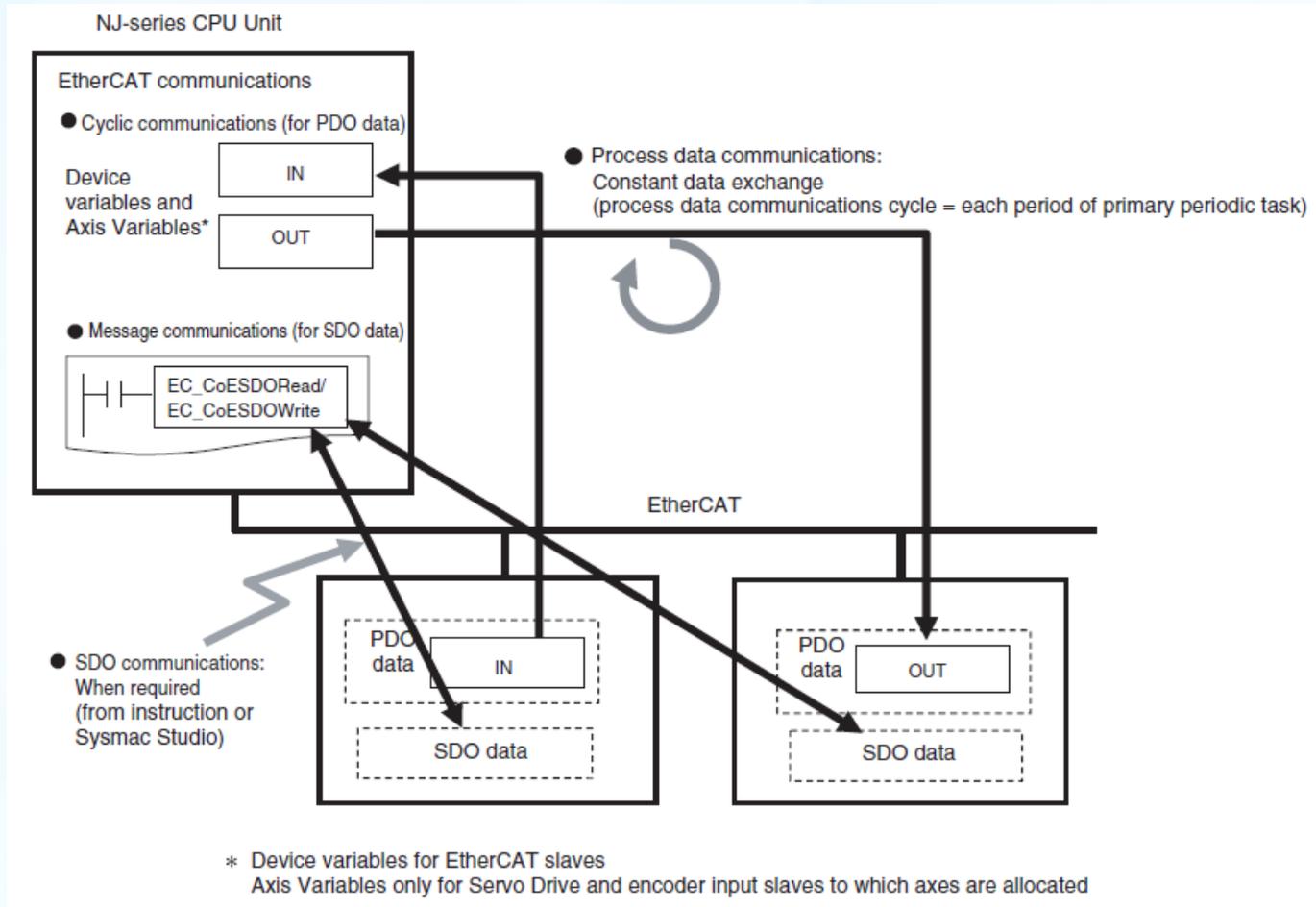
- EtherCAT通信类型
- 分两种
- (1) PDO(Process Data Objects)通讯
- (2) SDO(Service Data Objects) 通讯

The following two methods are used to exchange data between master and slaves in EtherCAT communications.

Communication type	Name of communication type	Timing of processing	Type of data
Cyclic communications	Process data communications (PDO communications)	Constant (process data communications period)	• PDO data
Message communications	SDO communications	When required.	• SDO data

第一章 EtherCAT

- 两种通讯类型的通讯时机不同

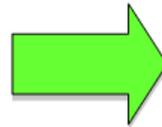
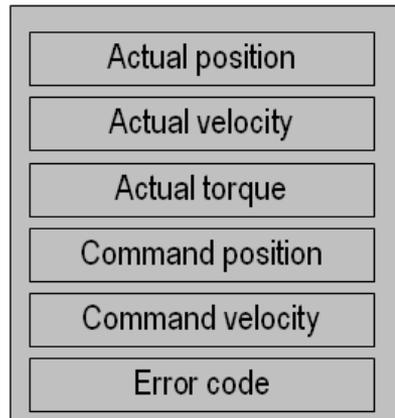


第一章 EtherCAT

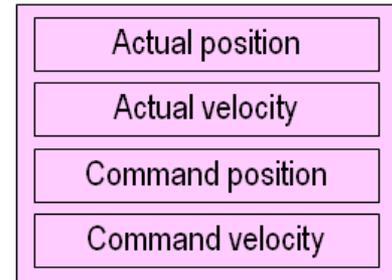
- PDO mapping
 - (1) PDO mapping 的用途是登记每个从站的PDO通讯数据
 - (2) 可以只登记需要那些用来做PDO通讯的数据，提高通讯性能

E.g. Servo Drive

PDO communication data (before)



PDO communications data (after)



第一章 EtherCAT

- 默认设置以外的数据需要改变PDO Map。
- 举例说明如何设置PDO mapping。

(1)默认设置直接选择

单击

Process Data Size : Input 208 [bit] / 240 [bit]
Output 184 [bit] / 192 [bit]

Selection	Input/Output	Name	Flag
<input type="radio"/>	---	No option	---
<input type="radio"/>	Output	1st receive PDO Mapping	Editable
<input type="radio"/>	Output	258th receive PDO Mapping	---
<input type="radio"/>	Output	259th receive PDO Mapping	---
<input type="radio"/>	Output	260th receive PDO Mapping	---
<input checked="" type="radio"/>	Output	261th receive PDO Mapping	---
<input type="radio"/>	Output	262th receive PDO Mapping	---
<input type="radio"/>	---	No option	---
<input type="radio"/>	Input	1st transmit PDO Mapping	Editable
<input type="radio"/>	Input	258th transmit PDO Mapping	---
<input checked="" type="radio"/>	Input	259th transmit PDO Mapping	---
<input type="radio"/>	Input	260th transmit PDO Mapping	---
<input type="radio"/>	Input	261th transmit PDO Mapping	---
<input type="radio"/>	---	No option	---
<input checked="" type="radio"/>	Input	512th transmit PDO Mapping	---

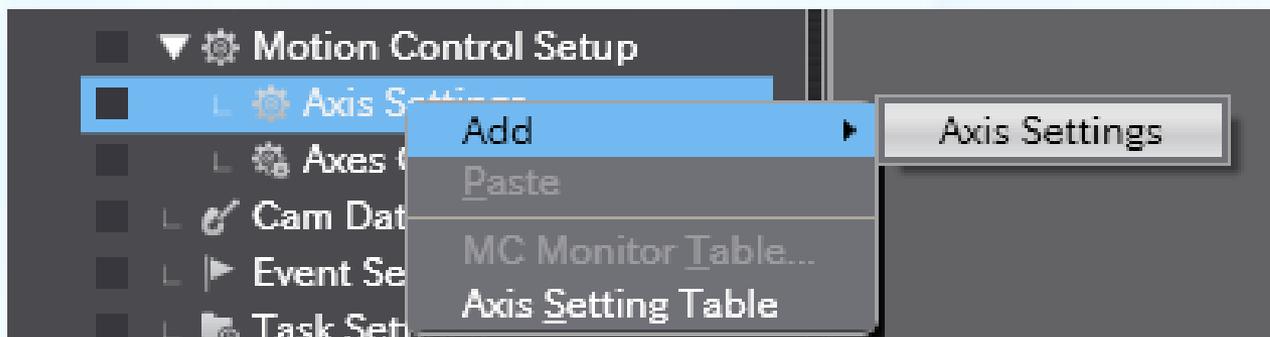
PDO entries included in 259th transmit PDO Mapping

Index	Size	Data type	PDO entry name	Comment
0x603F:00	16 [bit]	WORD	Error code	Error code
0x6041:00	16 [bit]	WORD	Statusword	Statusword
0x6064:00	32 [bit]	DINT	Position actual v...	Position act
0x6077:00	16 [bit]	INT	Torque actual value	Torque actua
0x6061:00	8 [bit]	SINT	Modes of operatio...	Modes of ope
0x60B9:00	16 [bit]	WORD	Touch probe status	Touch probe
0x60BA:00	32 [bit]	DINT	Touch probe pos1...	The latch po
0x60BC:00	32 [bit]	DINT	Touch probe pos2...	The latch po
0x60FD:00	32 [bit]	DWORD	Digital inputs	Digital inpu

Add PDO Entry Delete PDO Entry
OK Cancel Apply

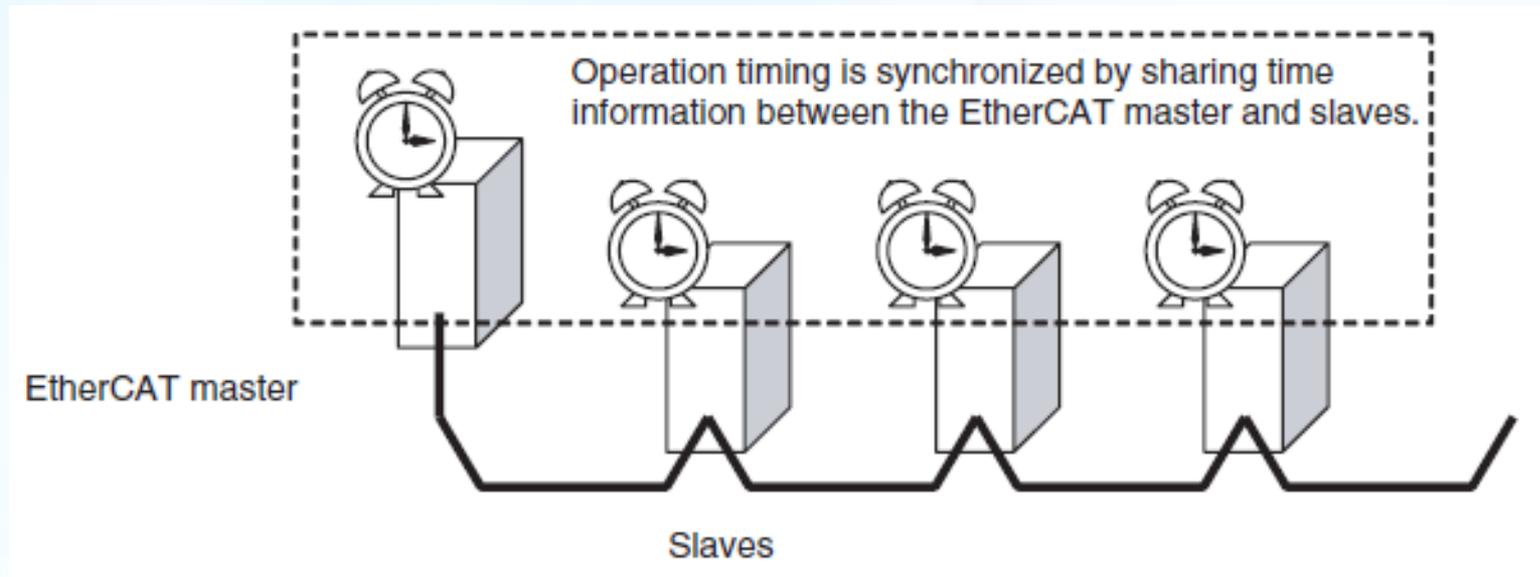
第一章 EtherCAT

- (2) 设置非默认设置



第一章 EtherCAT

- DC(Distributed Clock)功能
- DC功能整合在EtherCAT功能内，这个功能能使通讯的主站和从站之间的时间实现精确同步。

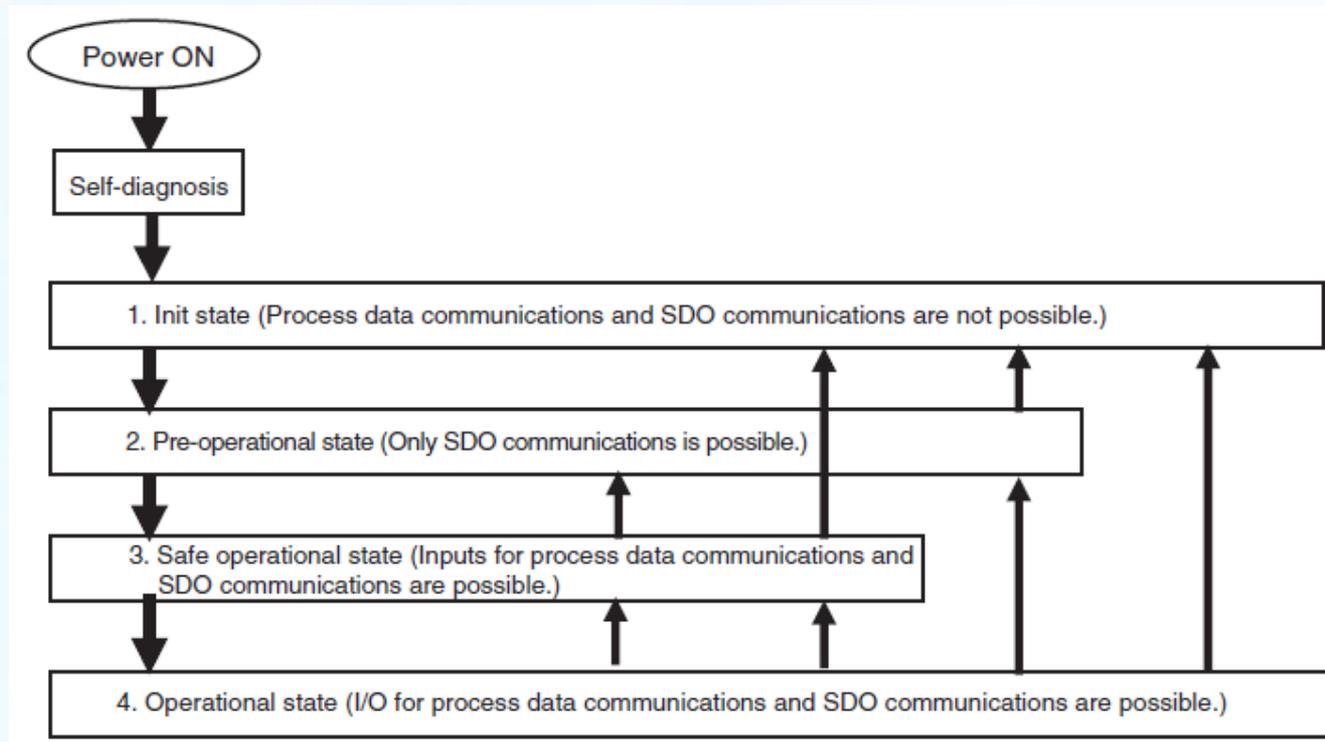


第一章 EtherCAT

- 参考时间
- 在一个系统中需要有一个基准时间，被称为参考时间。
- 在一个EtherCAT系统中，靠主站最近的一台从站如果有DC功能，那么这个从站的时间就被作为参考时间

第一章 EtherCAT

- EtherCAT通讯的控制状态
- EtherCAT通讯会经过四个控制状态，当条件满足时通讯会自动在这四个状态间自动转换



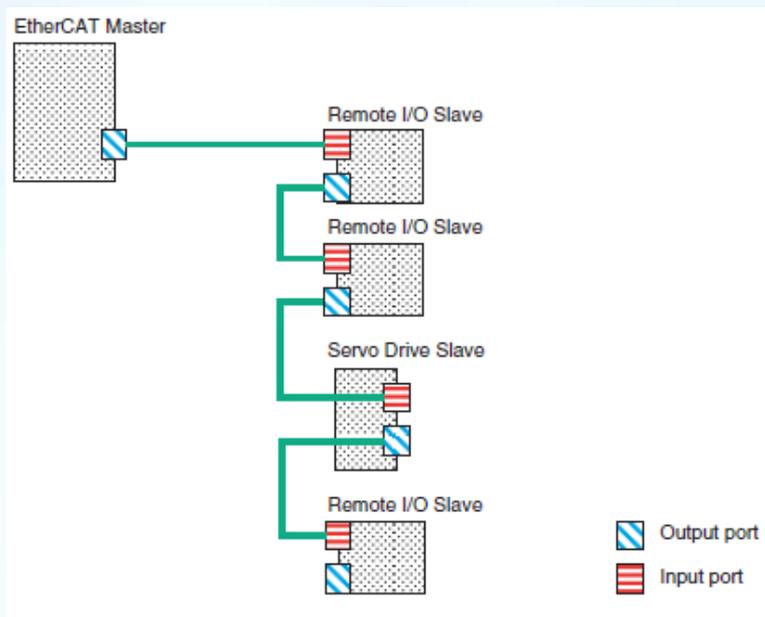
- 1: 初始状态
- 2: 预操作状态
- 3: 安全操作状态
- 4: 操作状态

第一章 EtherCAT

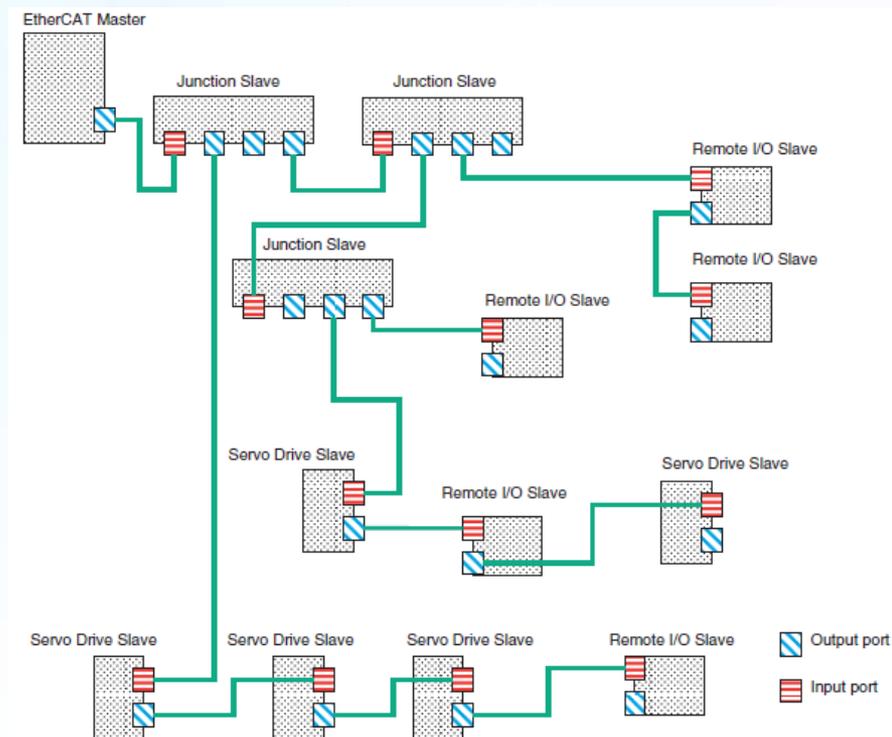
- 拓扑结构
- **NJ系列的EtherCAT支持以下两种拓扑**
 - (1) 菊花链总线连接方式
 - (2) 分支连接方式
- 最大从站数**192**个。
- 从站之间导线长度最大**100**米。
- 导线总长度取决于网络拓扑（由**Sysmac Studio**软件得到）。

第一章 EtherCAT

- (1) 菊花链总线连接方式

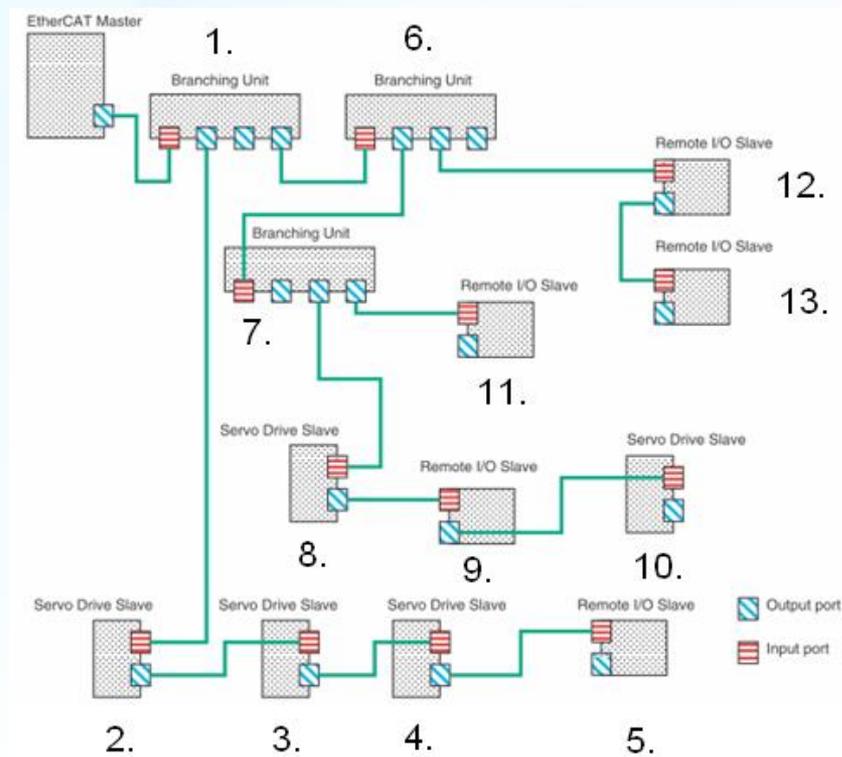
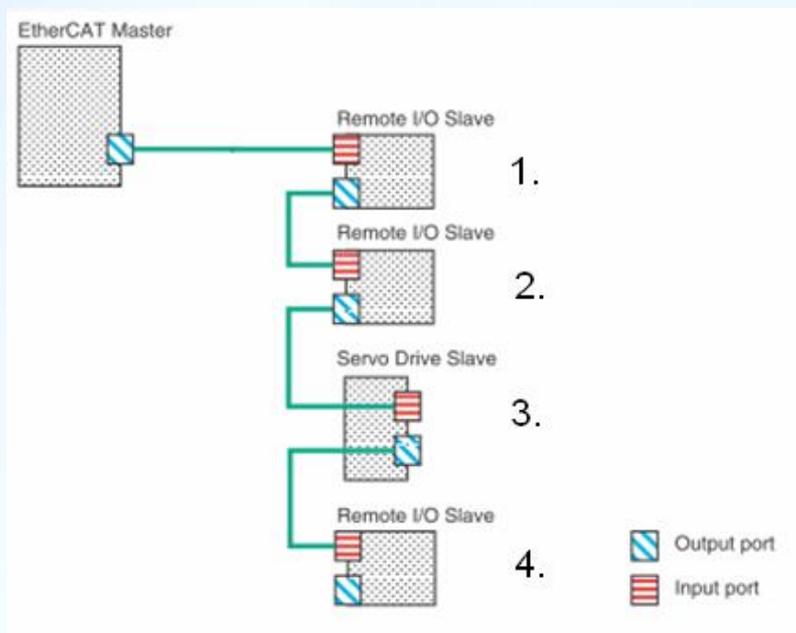


- (2) 分支连接方式



第一章 EtherCAT

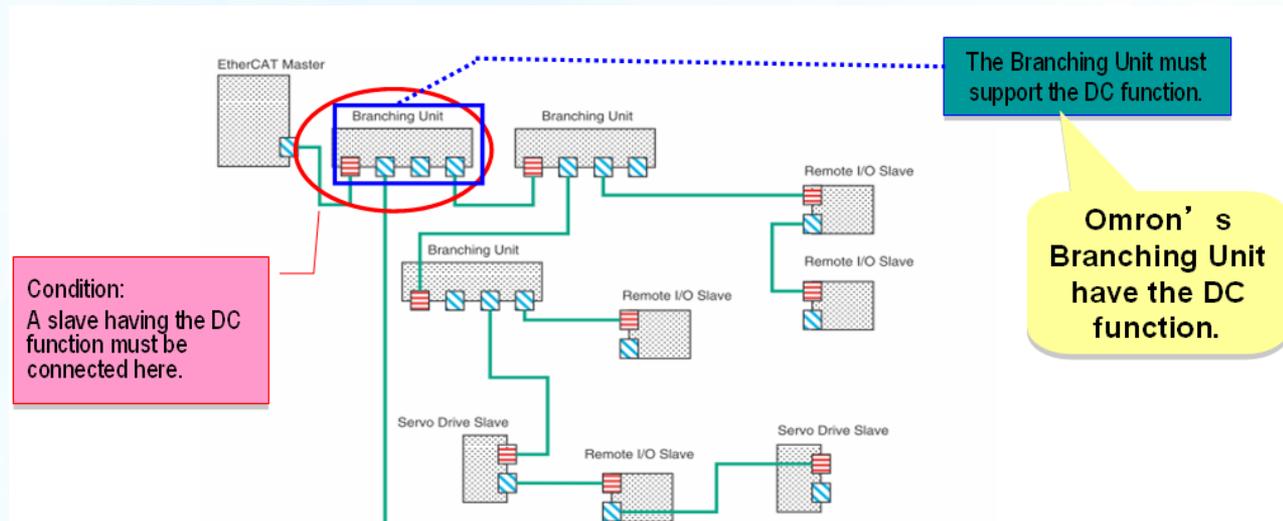
- 菊花链总线连接方式 从站编号
- 分支连接方式从站编号



第一章 EtherCAT

▪ 注意点:

- (1) 一个具有**DC**功能的从站必须被连接在主站和第一个分支器之间。（即第一个分支器之前）
- (2) 分支器必须是专用分支器支持**DC**功能



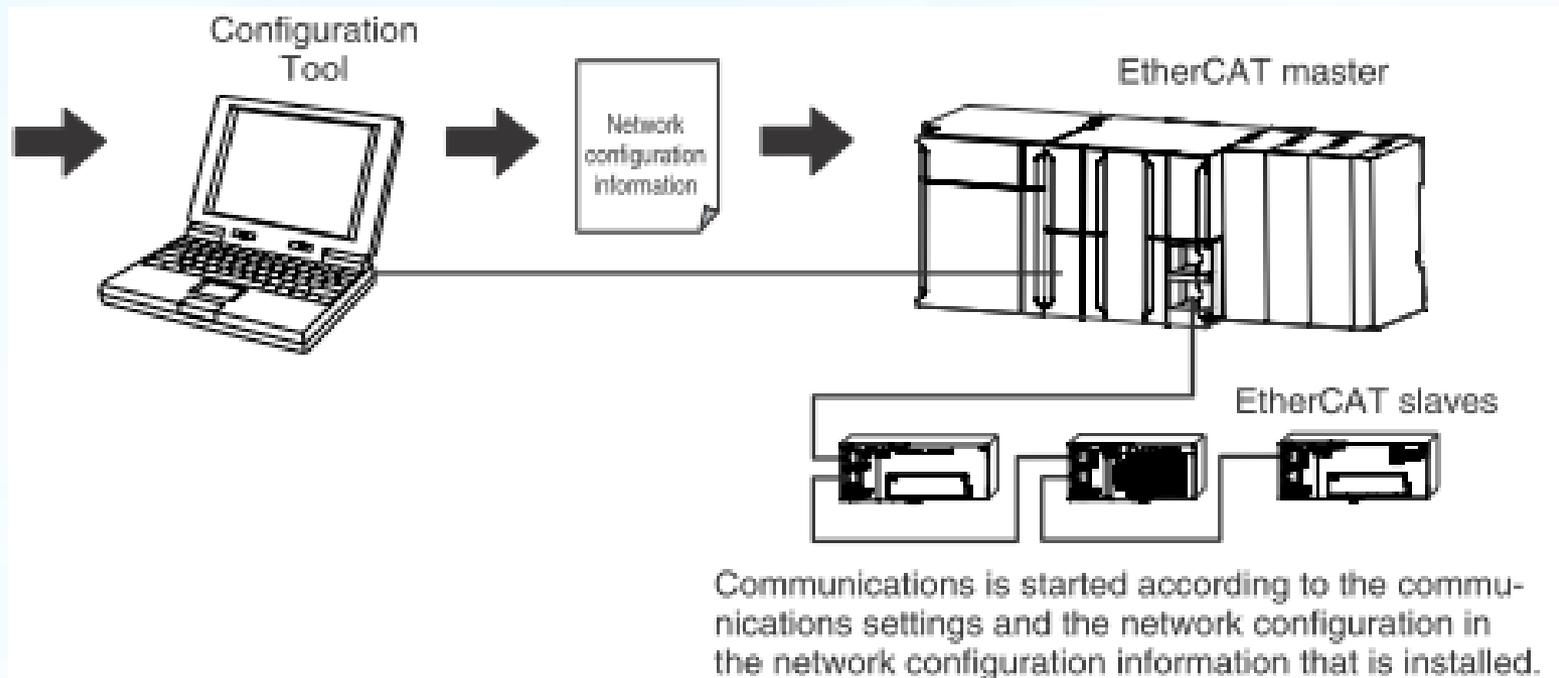
第一章 EtherCAT

- ESI

- (1) EtherCAT从站信息会被做成一个文件，这个文件就叫“ESI”（EtherCAT Slave Information）文件
- (2) ESI文件包括设备名，对分布时钟的设置信息等
- (3) ESI文件由从站厂家提供（欧姆龙产品ESI文件会由Sysmac Studio中集成）

第一章 EtherCAT

- 网络配置信息会被做成另一个文件，这个文件叫作“ENI”（EtherCAT Network Information）文件。
- ENI文件包括主站设置信息，从站设置信息，网络配置等。



第二章 NJ内置EtherCAT通信

- 第一节 CPU单元的显示
- 第二节 通信周期和I/O刷新的关系
- 第三节 Message通讯动作周期
- 第四节 通过用户程序访问从站
- 第五节 通过用户程序的要点
- 第六节 登记网络构成信息
- 第七节 从站的使能与禁用
- 第八节 从站的断开连接与重新连接
- 第九节 监视包功能
- 第十节 与第三方设备从站连接

第二章 NJ内置EtherCAT通信

- 一个NJ系列CPU单元会内建了一个USB口，一个EtherNet/IP口，一个EtherCAT口和一个SD卡槽



第二章 NJ内置EtherCAT通信

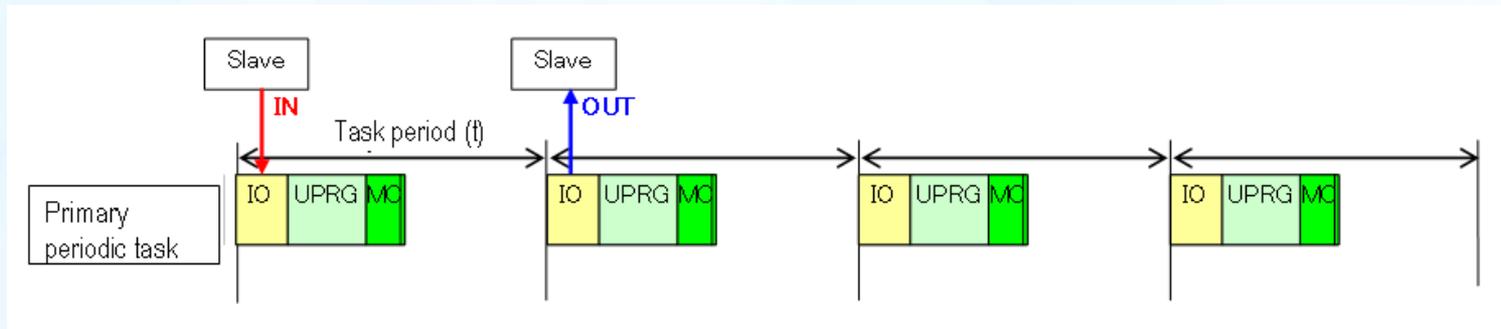
Indicator	Color	Status	Description
EtherCAT NET RUN	■ green	Lit	EtherCAT communications is in progress. • I/O data is being input and output: Operational state
		Flashing (1 second cycle)	EtherCAT communications is established. (One of the following states) • Only message communications: Pre-Operational state • Message communications, and inputs of I/O data: Safe-Operational state
		Not lit	EtherCAT communications is stopped. • Power-off state, or the Unit is being reset. • There is a MAC address error, communications controller error, or other error.
EtherCAT NET ERR	■ red	Lit	An unrecoverable error such as a hardware error or exceptional process occurs
		Flashing (1 second cycle)	A recoverable error occurs
		Not lit	No error
EtherCAT LINK/ACT	■ yellow	Lit	Link is established
		Flashing	Data being transmitted and reprieved after link is established (The indicator flashes every time data is sent or received.)
		Not lit	Link is not established

第二章 NJ内置EtherCAT通信

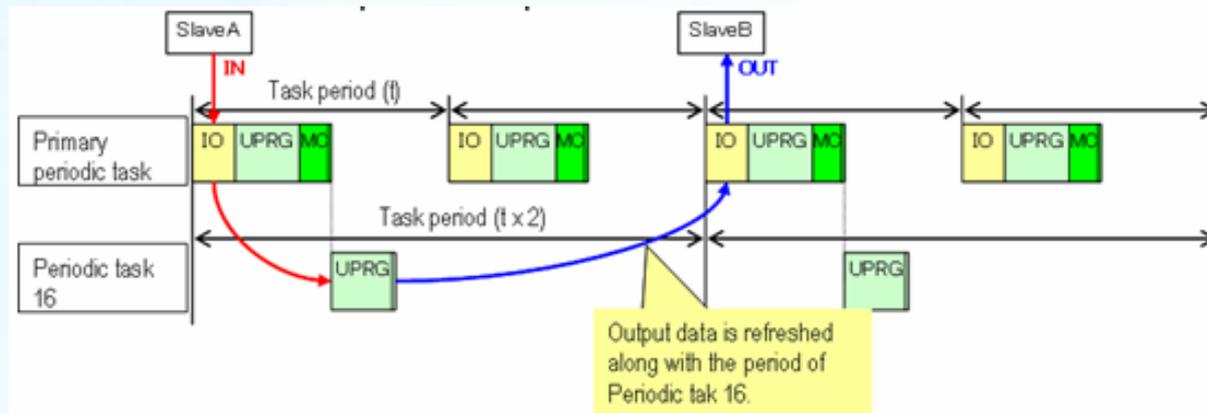
- 通信周期与I/O刷新的关系
- NJ系列允许用户将EtherCAT网络从站的I/O刷新设置分配给EtherCAT的任务。共有两类任务可以选择：
 - (1) 主要周期任务
 - (2) 周期任务（16优先级）

第二章 NJ内置EtherCAT通信

- 第一种情况是将I/O刷新分配在主要周期任务内

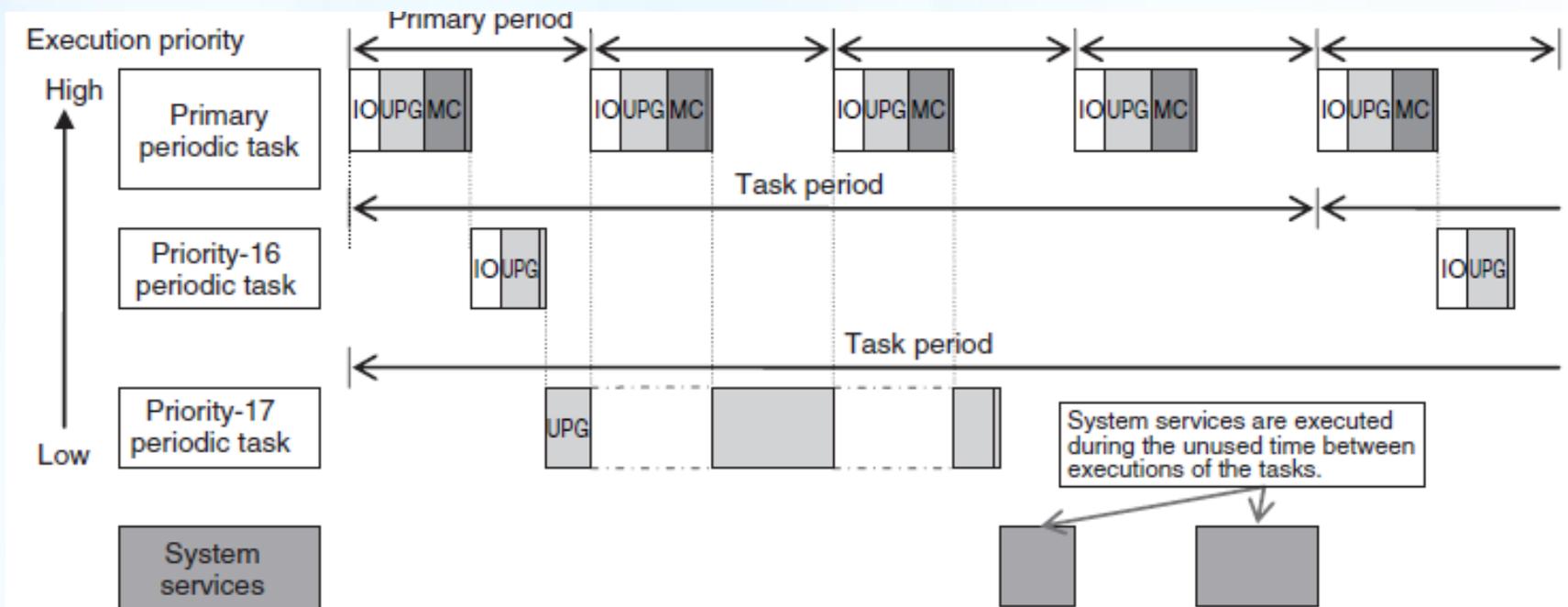


- 第二种情况是将I/O刷新分配在周期任务（16号）中



第二章 NJ内置EtherCAT通信

- **Message**通讯动作周期
- **Message**通讯（**SDO**通讯）是在NJ系列CPU的系统服务时间内动作的。
- 系统服务时间是在所有任务的间隙中动作执行的



第二章 NJ内置EtherCAT通信

- 如果发生由于系统服务时间过短而导致不足以动作的情况（默认时间是 $10\%*10ms$ ）。则根据任务号分为两种情况，如果是周期任务17以及17以下的任务动作停止用以执行系统服务，而如果是周期任务是16则不停止。
- NJ系统没有SDO Message缓冲功能，在NJ系统中SDO Message是通过软件或者指令一个接一个地传送。

第二章 NJ内置EtherCAT通信

- 通过用户程序访问从站
- 设置相关变量目的是用户程序可以对从站进行访问

Slave for	Slave type	Used variable type	Method to create variables
Other than motion control	EtherCAT slave	Device variable	I/O map setting
Motion control	Servo Drive or Encoder	Axis variable, Axes group variable	Axis setting

- 变量类型根据从站的使用情况不同而不同

第二章 NJ内置EtherCAT通信

- I/O map setting

Pos	Port	Description	R/W	Data Ty	Variable
	▼ CPU/Expansion Racks				
CP	CPU Rack 0				
	▼ EtherCAT Network Configuration				
Et	Master				
Nc	▼ R88D-KN01H-ECT				
	Controlword	Controlword	W	WORD	E001_Controlword
	Target position	Target position	W	DINT	E001_Target_position
	Target velocity	Command speed for th	W	DINT	E001_Target_velocity
	Target torque	Target torque	W	INT	E001_Target_torque
	Modes of operation	Modes of operation	W	SINT	E001_Modes_of_operation
	Touch probe function	Touch probe function	W	WORD	E001_Touch_probe_functi

第二章 NJ内置EtherCAT通信

- 轴和轴组里的变量
- 当一个轴（**Axis**）或者一个轴组(**Axis group**)建立后，关于这个轴或者轴组的变量就会自动生成
- 新建的轴和轴组名称上具有规律。
- 如 **MC_Axis***** （***为具体轴号）
- 如**MC_Group***** （***为具体轴组号）



第二章 NJ内置EtherCAT通信

- 制作用户程序的要点
- 启动EtherCAT网络和启动用户的动作实际上彼此之间是没有关联的，显然他们也并非同步启动的，因此当用户程序中需要使用到EtherCAT网络中的某些数据的时候请务必确认EtherCAT网络的状态

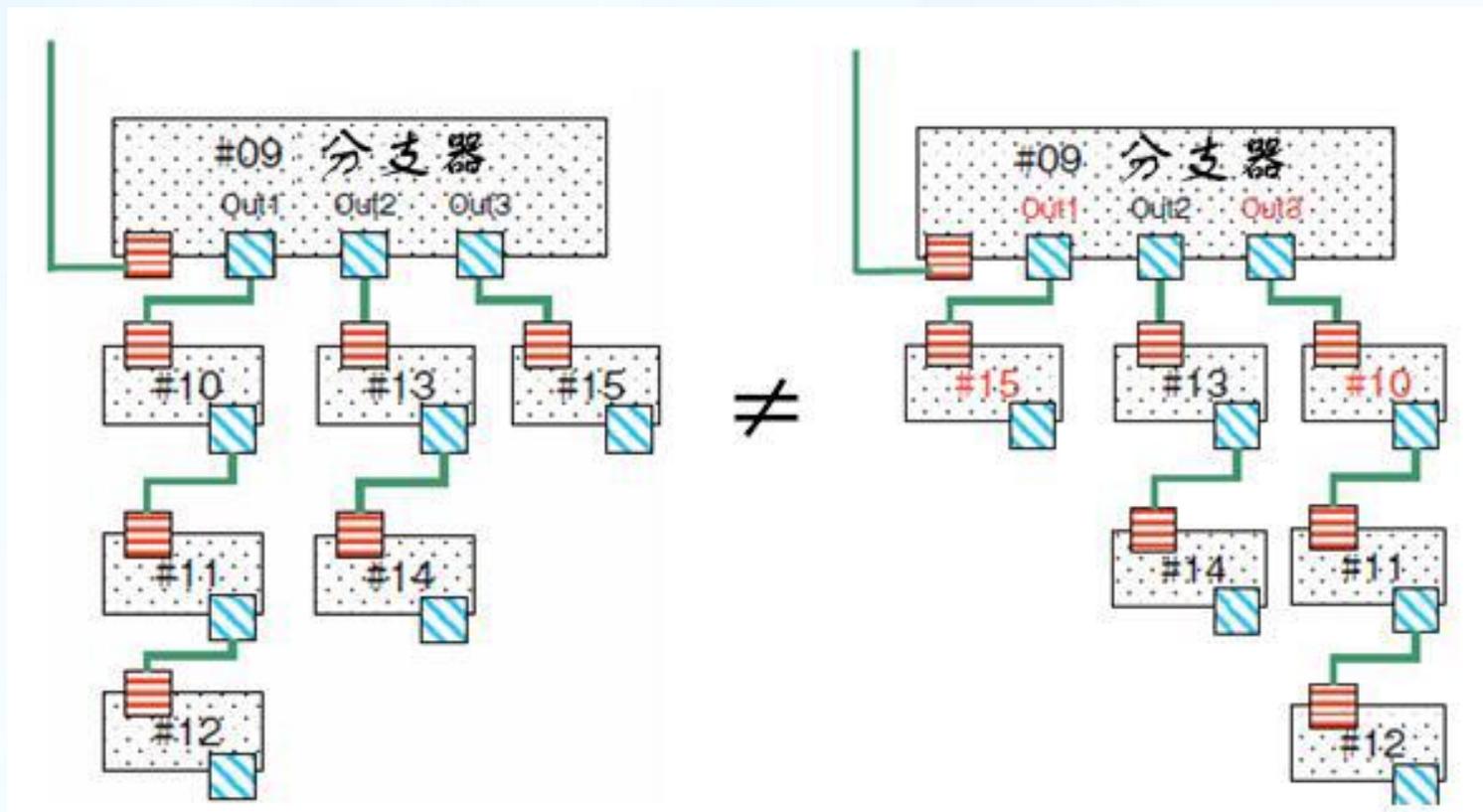
第二章 NJ内置EtherCAT通信

- 登记网络构成信息
- 网络构成信息是由Sysmac Studio设置的，用以对实际网络连接结构进行验证。当结构是完全正确的情况下，那么当发生一些本地错误后这些错误能被识别到

Verification information	Description	Verified by	Required or optional
Number of slaves connected	Number of slaves in the network	Master, Sysmac Studio	Required
Slave information	Vender ID and product code of each slave	Master, Sysmac Studio	Required
Slave information	Revision number and serial number of each slave	Master, Sysmac Studio	Optional
Connection port	Location of port on a branching Unit	Sysmac Studio only	Required

第二章 NJ内置EtherCAT通信

- 例如以下网络结构实际情况和设置不符合。此时Sysmac Studio能立刻发现



第二章 NJ内置EtherCAT通信

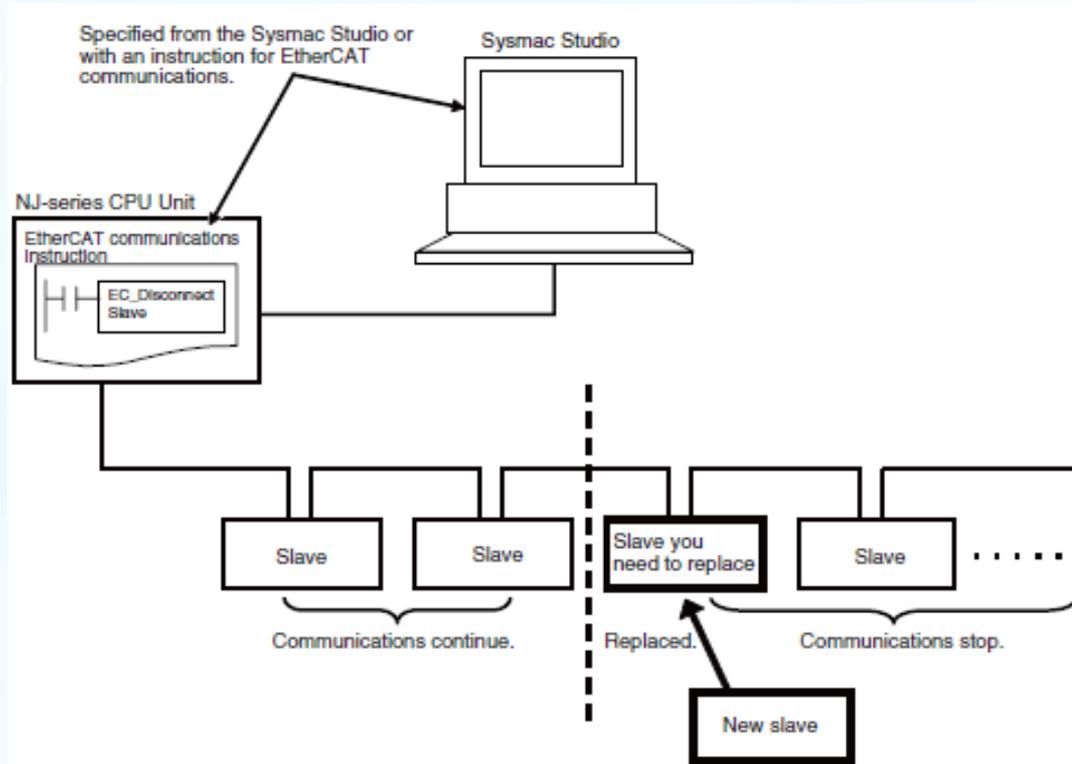
- 从站的使能与禁用
- 通过Sysmac Studio可以使能或禁用一个从站
- 使能从站意味着从站能操作
- 禁用从站意味着从站不能操作
- 当现在有一个从站没有连接进实际的EtherCAT网络，但是如果将来要拓展，你可以先把它登记成为一个禁用从站

注意：

即便是禁用从站，但是一样可以设置变量。一样可以在用户程序里使用这个禁用从站默认变量或设备变量等。

第二章 NJ内置EtherCAT通信

- 从站的断开连接与重新连接
- 这个功能允许断开或者重新连接被选中的从站，而不需要停下整个通讯系统
- 断开连接或者重新连接功能可以通过Sysmac Studio软件来操作或者通过用户程序来操作

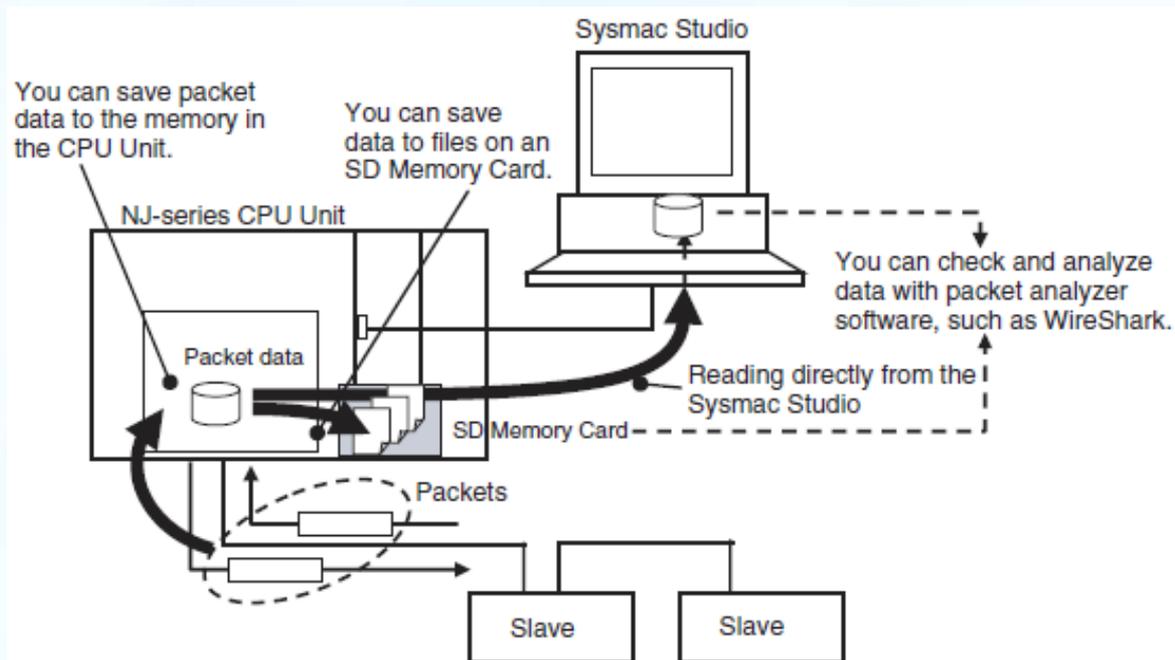


第二章 NJ内置EtherCAT通信

- 监视包功能
- 这个功能的目的是用来直接监控那些由EtherCAT主站发送和接受的帧
- 通过使用Sysmac Studio或者专用指令，可以把电脑包数据保存在CPU单元的内存里

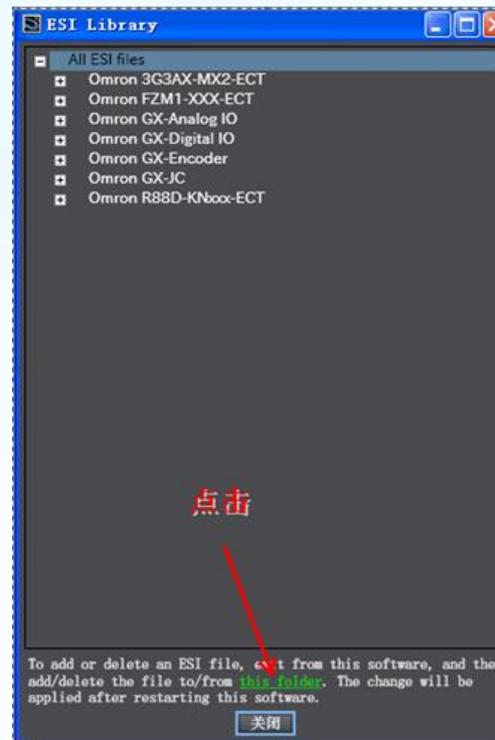
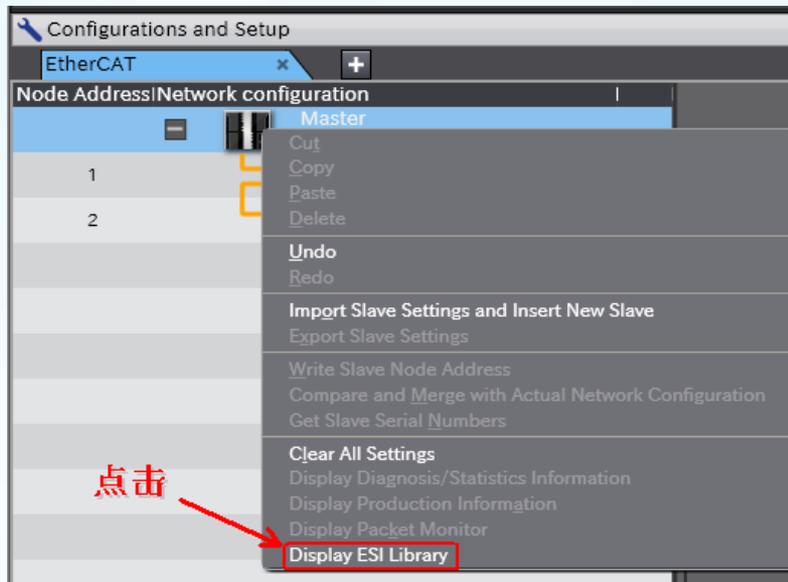
第二章 NJ内置EtherCAT通信

- 这些保存在CPU内存里的数据可以被以下两种方式获取：
 - （1）直接通过Sysmac Studio读取。
 - （2）转移进CPU单元上插着的SD内存卡



第二章 NJ内置EtherCAT通信

- 与第三方设备从站连接
- 为了在Sysmac Studio里使用第三方设备从站，必须导入ESI文件

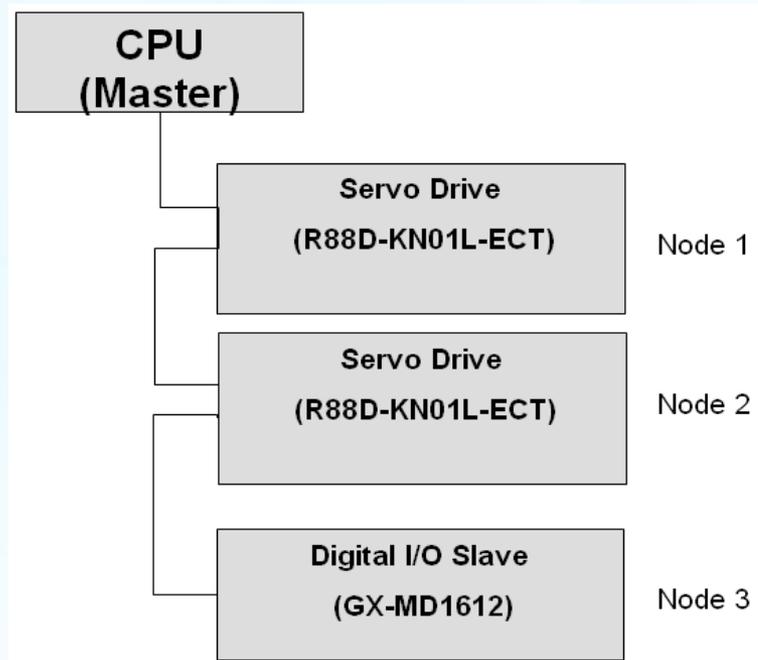


第三章 EtherCAT的单元设置

- 第一节 系统配置
- 第二节 配线
- 第三节 硬件设定
- 第四节 软件设定
- 第五节 动作确认

第三章 EtherCAT的单元设置

- 系统配置
- 配置案例： 一个NJ系列PLC做为主站， 下带两个支持ECT功能的伺服和一个ECT网络I/O从站。 伺服节点号分别为1和2， I/O从站节点号为3。

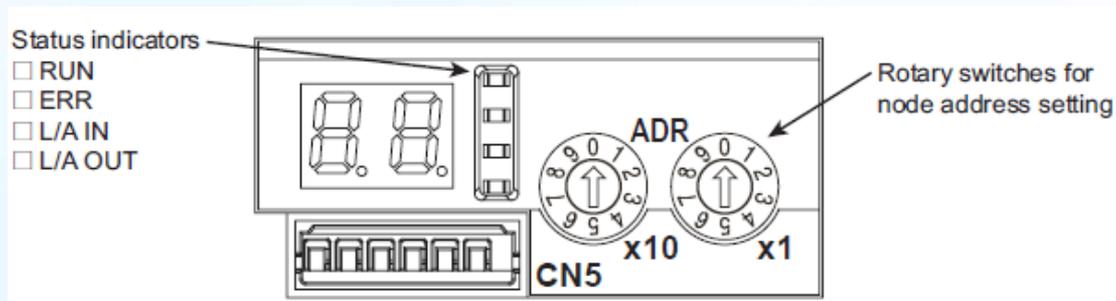


第三章 EtherCAT的单元设置

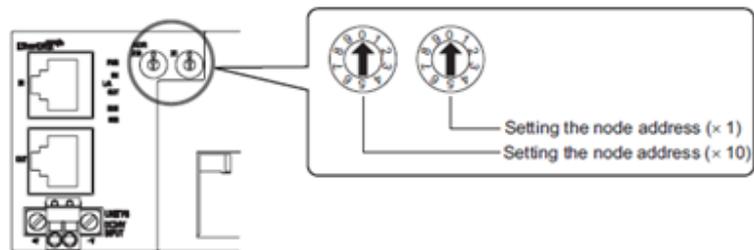
- 设置EtherCAT从站号
- 一般如果要分配超过99，那么必须用软件来分配，最大到192（尤其是从站数超过99个的情况下）
- 并不是所有的从站都在硬件上有这类开关。没有的话用Sysmac Studio来设置
- 硬件节点只有在上电时刻生效，所以请在上电前请确认是否设置正确并且确认在要设置的时候关闭电源

注意：

另外某些从站会有其他的限制具体参考各从站手册



These switches are used to set node addresses of Slave Units in the EtherCAT network (decimal). The 10s digit is set on the left rotary switch and the 1s digit is set on the right rotary switch. Setting range is 00 to 99. (Default setting: 00)



第三章 EtherCAT的单元设置

- 主站登记
- 从站登记

Item name	Value
Device name	Master
Model	Master
Product name	Master
Number of Slaves	2
PDO Communications...	1000 us
Total Cable Length	1000 m
Fail-soft Operation Set...	Fail-soft operation ▼
Wait Time for Slave St...	30 s
PDO communications...	2 times
Revision Check Method	Setting <= Actual device ▼
Serial Number Check...	No check ▼

Device name
Set a name for the master.

第三章 EtherCAT的单元设置

- 1 单击EtherCAT主站会出现相应的选项
- 2 设置Device name。在此处输入主站单元名称，最多64个字符。
- 3 设置Total cable length
- 4 设置 Fail-soft Operation
- 5 设置 Wait Time for Slave Startup 设置主站需要等待所有参与在网络里的从站花费的时间。

第三章 EtherCAT的单元设置

- 6 设置POD通讯超时次数 如果通讯连续次数超出设定值会发生对应的报错。设置范围1~8。
- 7 设置Revision Check Method
- 8 设置Serial Number Cherk Method 设置是否检测串口号， 如果选择检测要求一致（Set value=Actual value），一旦有从站重置，会无法辨识串口就必须重启网络
- 9 其他皆为自动设置

第三章 EtherCAT的单元设置

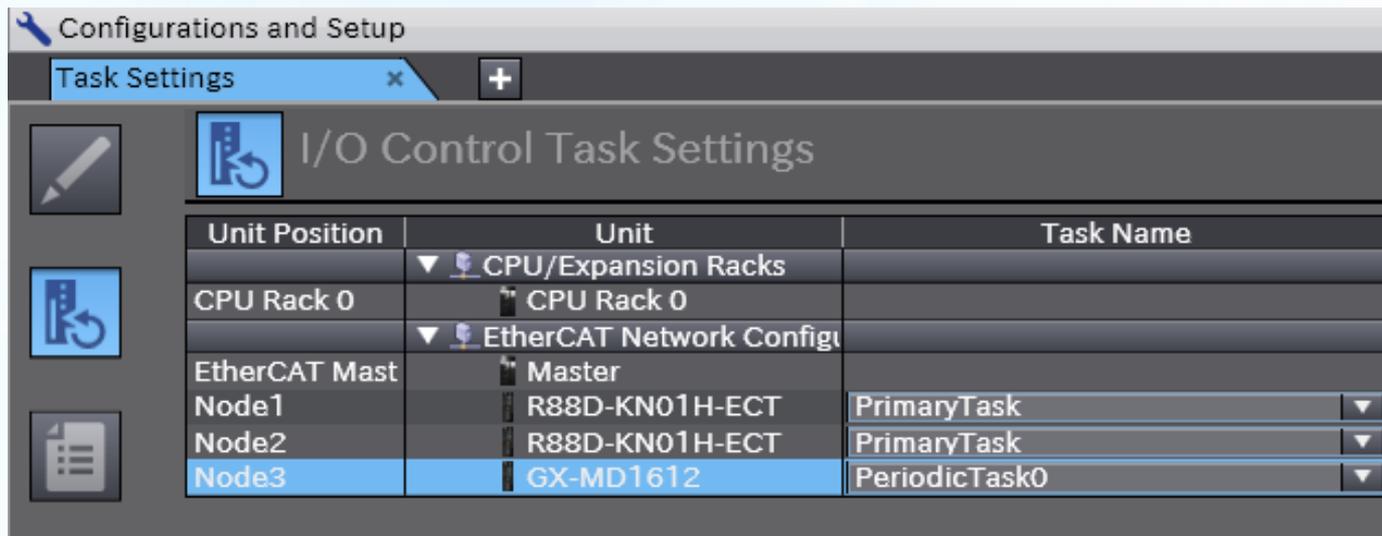
- 1 单击需要设置的从站
- 2 设置**Device name**。在此处输入主站单元名称，最多64个字符
- 3 设置**Node address**。设置一个未分配的节点地址，当实物硬件设置非0的时候，此处设置必须与实物的开关上设置的节点地址一致。如果不一致会导致报错。设置范围1~192
- 4 设置**Enable/Disable Settings**，选择**enable**或者**disable** 来使能和禁止操作，当选择使能的时候，从站会动作。当选择禁止时候有两种可能性，一 从站即便连接也不需启动；二 即便从站没有连接也不报错

第三章 EtherCAT的单元设置

- **5 设置PDO Map**，设置通过PDO通讯发送和接受的数据，根据具体从站不同，如果这个从站是有PDO内容的，列表会显示出来，如果本身这个从站就没有PDO会显示“---”
- **6 设置Setting Parameters** 设置从站的初始设置参数，如果这个从站没有需要设置的初始设置参数会显示“---”需要设置的内容具体以各类从站为准。
- **7 设置Backup Parameter Settings** 设置从站的备份参数设置，这个和“Setting Parameters”选型一样，也是要以从站为主。不同的从站会不同，例如G5伺服的Backup Partameter是其各个菜单选项。

第三章 EtherCAT的单元设置

- 任务设置
- 需要对各个通讯从站分配任务。
- 有两个任务可以被设置
- 主要周期任务
- 周期任务（16优先级）

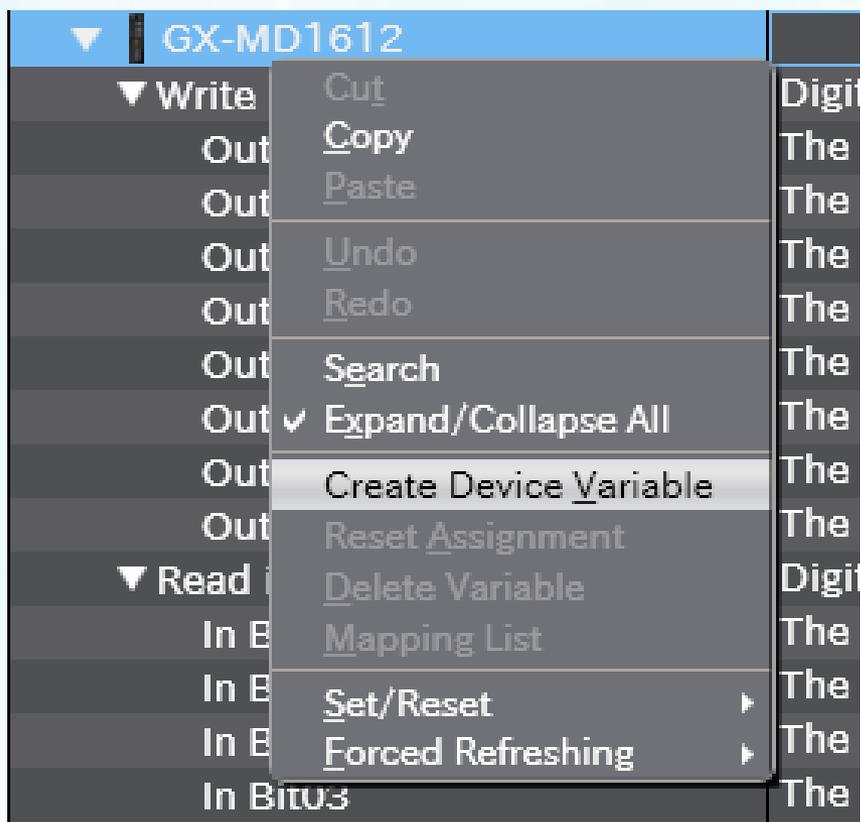


第三章 EtherCAT的单元设置

- I/O Map
- “I/O口” 是用以CPU单元和外部设备（比如从站或者模块）交互数据用。
- 当EtherCAT从站和模块在Sysmac Studio中被建立后，“I/O口”就会自动创建。
- 之后“I/O口”还会自动被注册在Sysmac Studio内。

第三章 EtherCAT的单元设置

- (1) 自动创建变量的做法
- 右键从站模块选择 “Create Device Variable”



第三章 EtherCAT的单元设置

- (2) 创建轴变量
- 选择“Motion Control Setup”下的“Axis Settings”右键选择“Add”的“Axis Settings”



第三章 EtherCAT的单元设置

- 如何动作确认
- (1) 观察工作指示灯

Check	Indicator	Description
<input type="checkbox"/>	NET RUN	OK when green light is on.
<input type="checkbox"/>	NET ERR	OK when it is unlit.
<input type="checkbox"/>	LINK/ACT	OK when yellow light is on or flashes.

Check	Indicator	Description
<input type="checkbox"/>	RUN	OK when green light is on.
<input type="checkbox"/>	ERR	OK when it is unlit.
<input type="checkbox"/>	LA IN	OK when yellow light is on or flashes.
<input type="checkbox"/>	LA OUT	OK when yellow light is on or flashes.

第三章 EtherCAT的单元设置

- (2) 通过Sysmac Studio设置强制值
- 使用设置强制值功能在从站的输出上设置任意数据。
- 可以通过一个I/O从站的指示灯来确认设置强制值的通讯是否成功

第四章 报错异常的应对

- 第一节 报错时的应对
- 第二节 报错的种类
- 第三节 报错发生时的对应
- 第四节 清楚报错

第四章 报错异常的应对

- 当EtherCAT网络的从站发生错误时候可以对主站的输入和输出操作选择以下两种应对方式。
 - (1) 停止通讯
 - (2) 与尚可通讯的从站继续进行通讯

第四章 报错异常的应对

- 报错的种类
- EtherCAT网络的报错可以根据它们的类型和级别进行分类
- 根据不同的报错级别，对应选择有停止所有操作或是仅部分停止，可以在“fail-soft operation”设置里设置。

第四章 报错异常的应对

Category	Error Type	Event Level
Communications Port Errors	MAC Address Error	Partial fault level Controller error, Communications stop.
	Communications Controller Error	
	Link OFF Error	
EtherCAT Master Error	Network Configuration Information Error	Minor fault level Controller error, Communications continues when the Fail-soft operation is selected.
	Network Configuration Verification Error	
	Network Configuration Error	
	Process Data Communications Error	Communications stops when the Stop is selected.
	Process Data Reception Timeout Error	
	Process Data Send Error	
	Slave Node Address Duplicated Error	
	Slave Initialization Error	
	Slave Application Error	
	Emergency Message Detection	

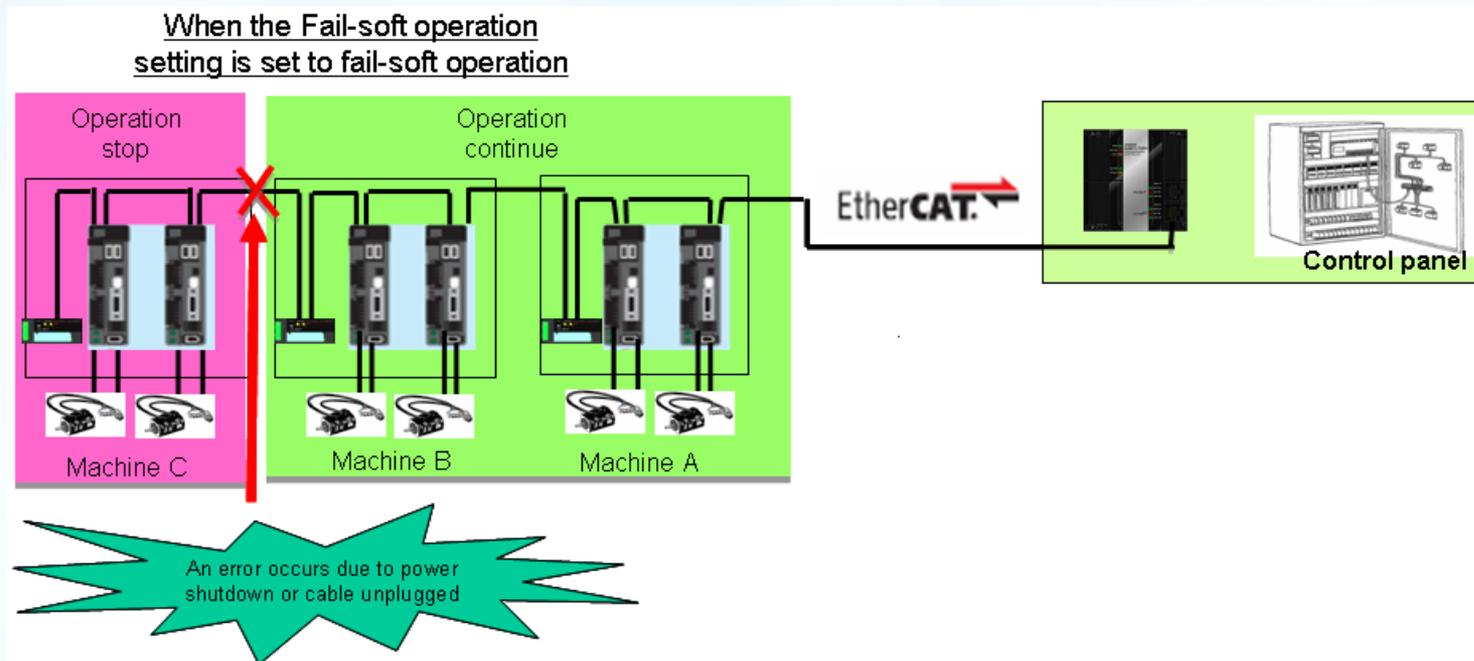
Note The above levels of EtherCAT slave errors are those defined by the slaves. On the EtherCAT master, slave error levels are indicated by the slave error (`_EC_SlavErr`) or the slave error table (`_EC_SlavErrTbl`).

第四章 报错异常的应对

- Fail-soft Operation
- Fail-soft Operation是一个专门关于EtherCAT报错的设置选项，用以决定当EtherCAT网络发生一个轻度错误（minor fault）时是所有从站都停止通讯还是继续和出错从站之前的从站设备继续通讯

第四章 报错异常的应对

- 一个菊花链连接方式的网络拓扑，当一个通讯错误发生后，出错从站之后的从站也会停下来。
- 如果使用分支器去连接机器和从站，那么在机器运行动作的时候也能维护或者纠正出现的错误。



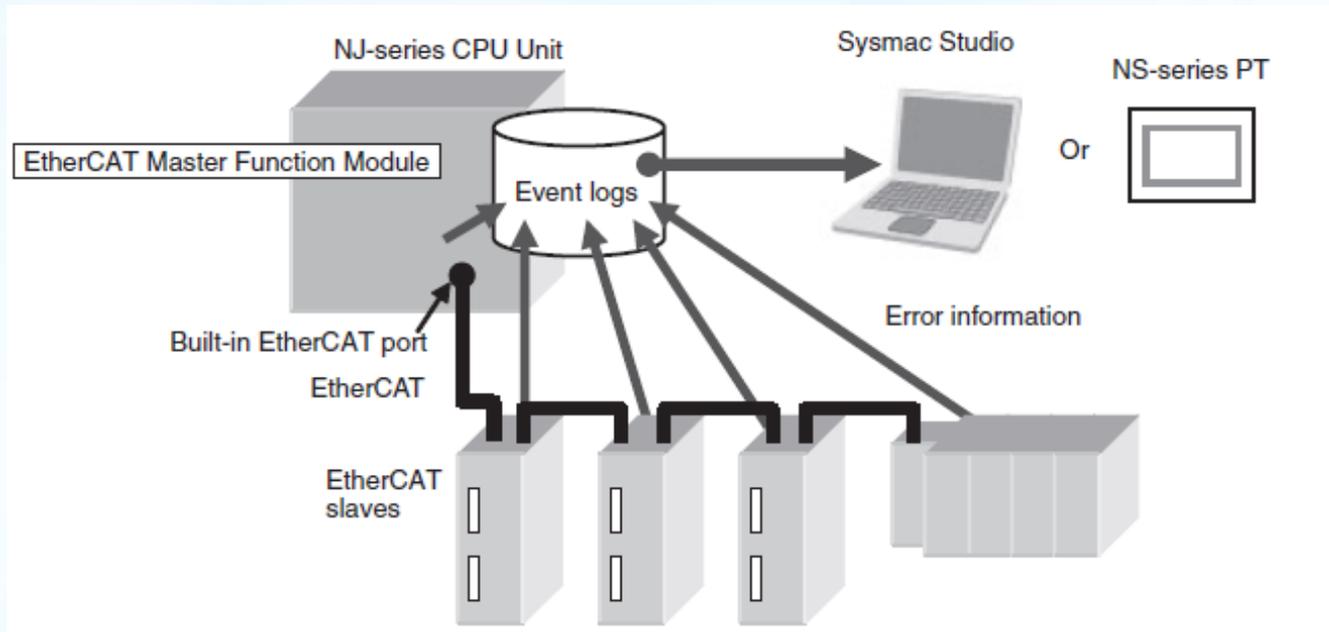
第四章 报错异常的应对

- 报错时的应对

Checking method	Description
Indicators	You can check the status of the Controller, the built-in EtherCAT port, and the built-in EtherNet/IP port.
Troubleshooter of the Sysmac Studio	You can use the Troubleshooting Dialog Box of the Sysmac Studio to check for current Controller errors and to check the error logs for past Controller errors.
Troubleshooter of an NS-series PT	You can check for Controller errors with the Troubleshooter of an NS-series PT that supports the NJ-series Controllers.
Instructions that read function module error status	You can execute instructions in the user program to read the error status of each function module to check for Controller errors.
System-defined variables	You can use communications from a host computer or other external device to read the Error Status system-defined variables to check for Controller errors.

第四章 报错异常的应对

- 网络的信息和错误详细情况会保存在事件日志(Event logs)里，通过Sysmac Studio就可以查看这些保存下来的信息



第四章 报错异常的应对

清除报错

- 有三种途径
 - (1) 软件
 - (2) NS触摸屏
 - (3) 指令 (RestECError)

Method	How
Clear the error status	Sysmac Studio
	NS-series PT
Executing the Reset EtherCAT System Error instruction	Create it in the user program. * All errors that currently occur on the EtherCAT system are cleared.

第五章 Motion导入基础

- 第一节 **MC**概述
- 第二节 任务
- 第三节 轴
- 第四节 试运行

第五章 Motion导入基础

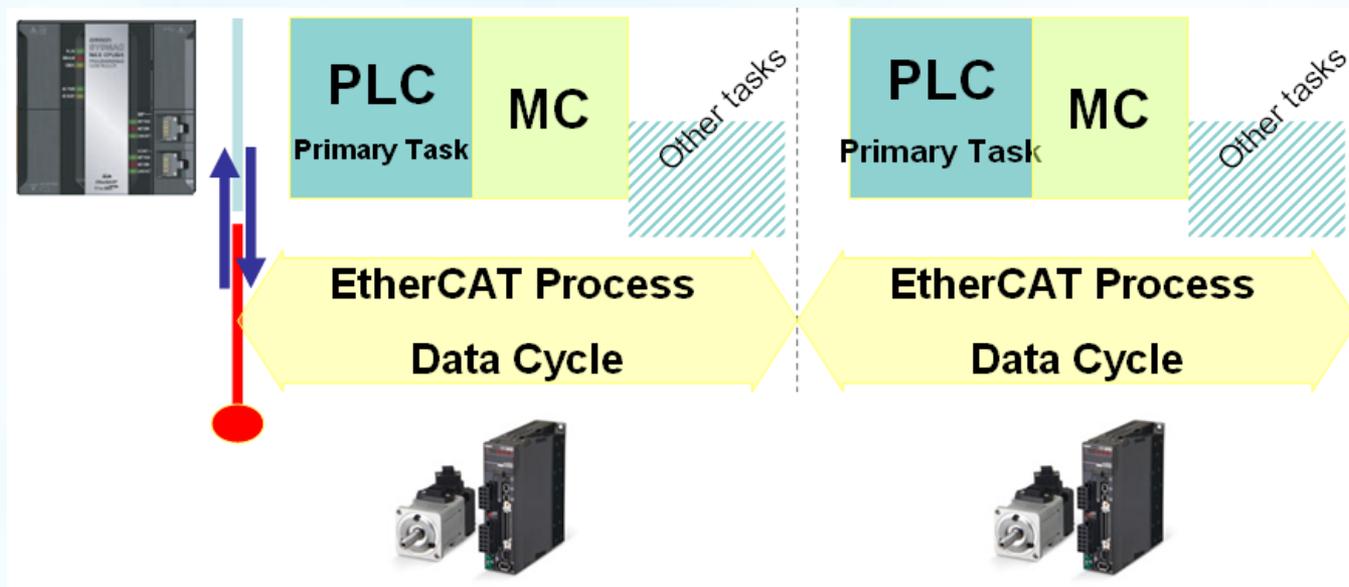
- 运动控制概述
- **MC**是指运动控制功能模块（The Motion Control Function Module）。
- 每一台NJ系列的CPU都内置了MC。
- NJ系列CPU的MC最多能控制**64**根虚轴或物理轴（EtherCAT）。
- NJ系列CPU的MC能使用独立，同步和联动（轴组）不同的控制方式

第五章 Motion导入基础

- **MC**的轴参数是通过**IEC**标准的轴变量（**Axis_REF**）数据类型来链接的
- 运动命令通过符合**IEC**标准的任务执行
- **MC**的运动控制功能通过运动命令实现
- **MC**通过符合**IEC**标准的运动程序来生成运动轨迹
- 所有轴都由**MC**来监视和控制
- **MC**与**EtherCAT**伺服同步交换数据

第五章 Motion导入基础

- 符合IEC标准的周期任务程序和MC都在一个EtherCAT循环周期内完成所有动作。
- 这意味着 NJ程序必须严格与EtherCAT驱动器同步



第五章 Motion导入基础

- 轴
- 结合G5 系列EtherCAT伺服MC可以设计出用以执行复杂的运动控制动作。
- MC能够控制不同类型的轴：伺服轴，虚轴，编码器轴。
- 轴与逻辑控制器之间通过AXIS_REF来链接

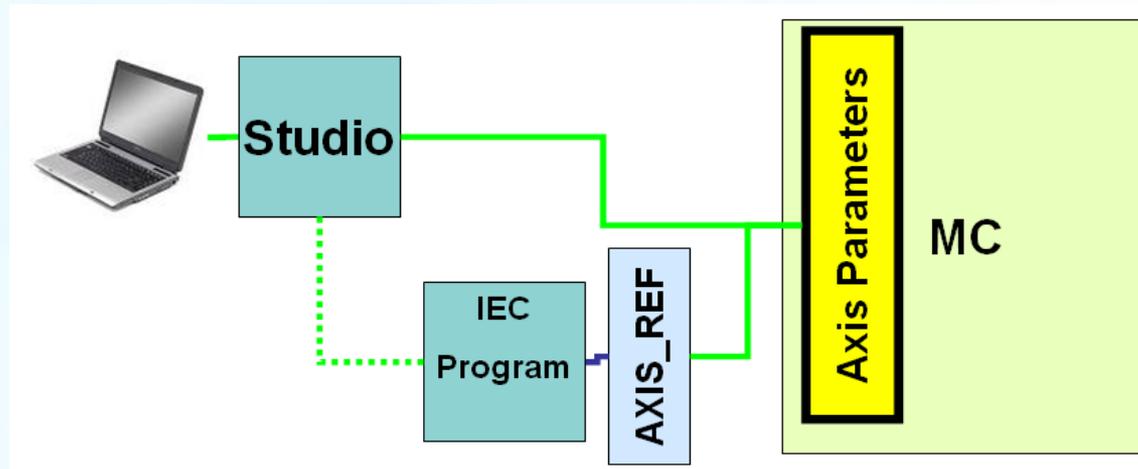
第五章 Motion导入基础

- 轴种类
- NJ系列PLC支持以下四种类型的轴

Axis Type	Description
Servo axis	These are the axes used by the EtherCAT slave Servo Drives. They are assigned to actual Servo Drives. One Servomotor is used as one axis.
Virtual servo axis	These are virtual axes that exist only inside the MC Function Module. They are not used by actual Servo Drives. <i>For example, they are used as master axes for synchronizing control.</i>
Encoder axis	An encoder axis uses an EtherCAT slave encoder input terminal. Encoder axes are assigned to actual encoder input terminals.
Virtual encoder	These axes are used virtually for encoder operation. For example, they are used for debugging when there is no encoder.

第五章 Motion导入基础

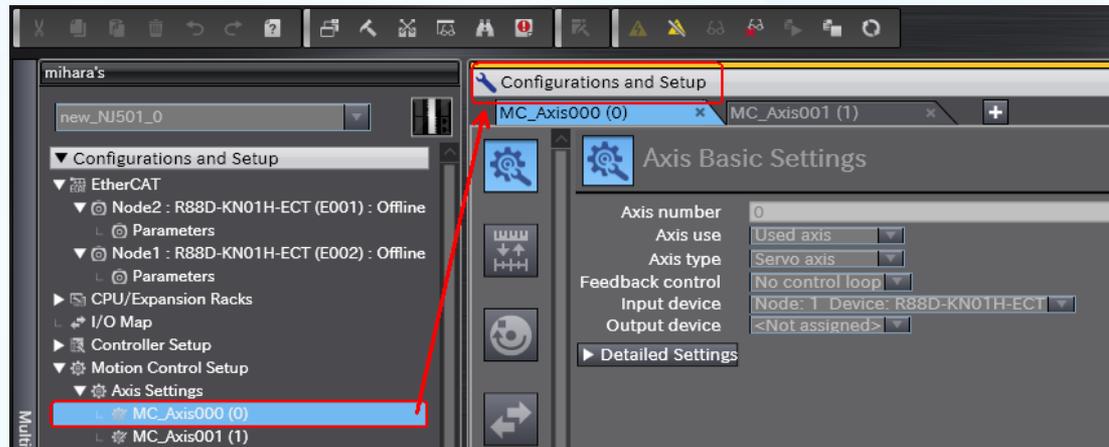
- 轴参数
- **MC**轴参数默认包含轴特性，命令信息和反馈信息
- **MC**轴参数可以通过Sysmac Studio软件来配置和监视



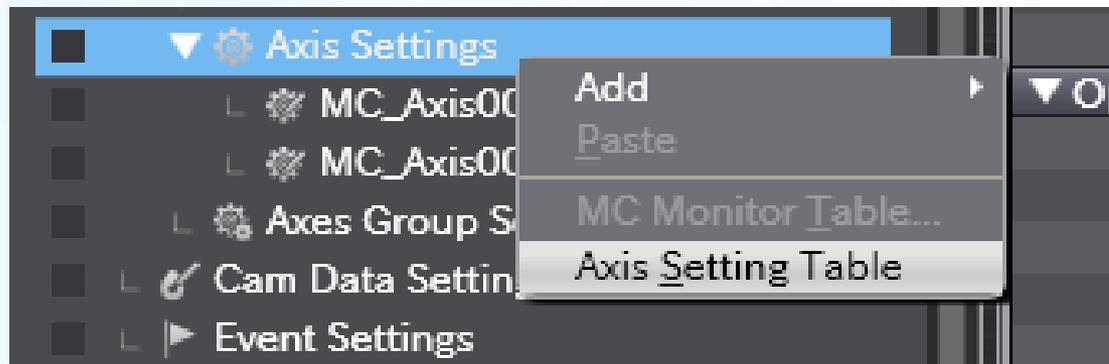
第五章 Motion导入基础

轴参数的简单显示

(1)

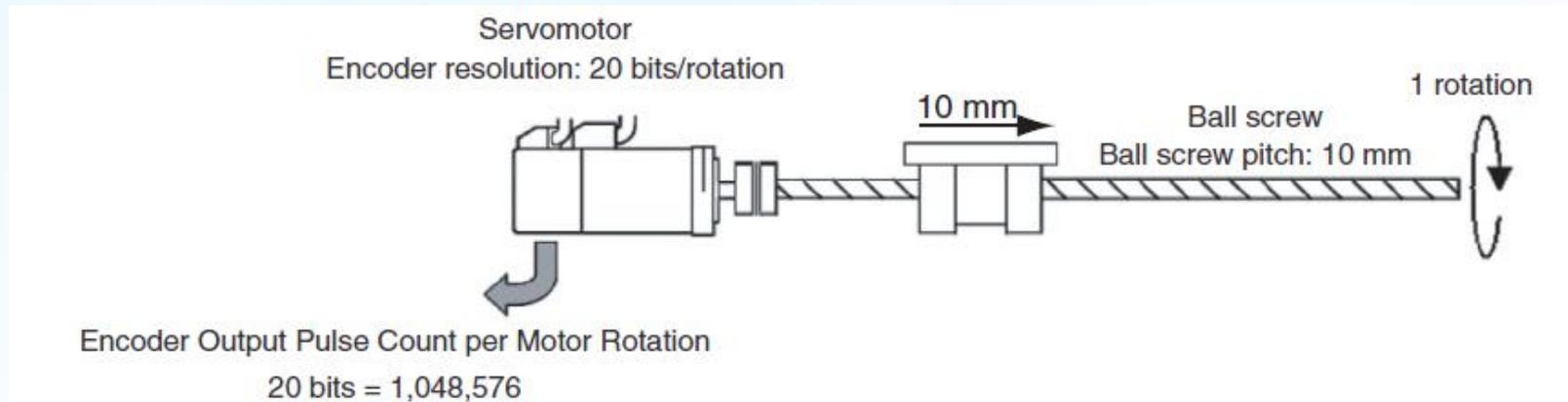


(2)



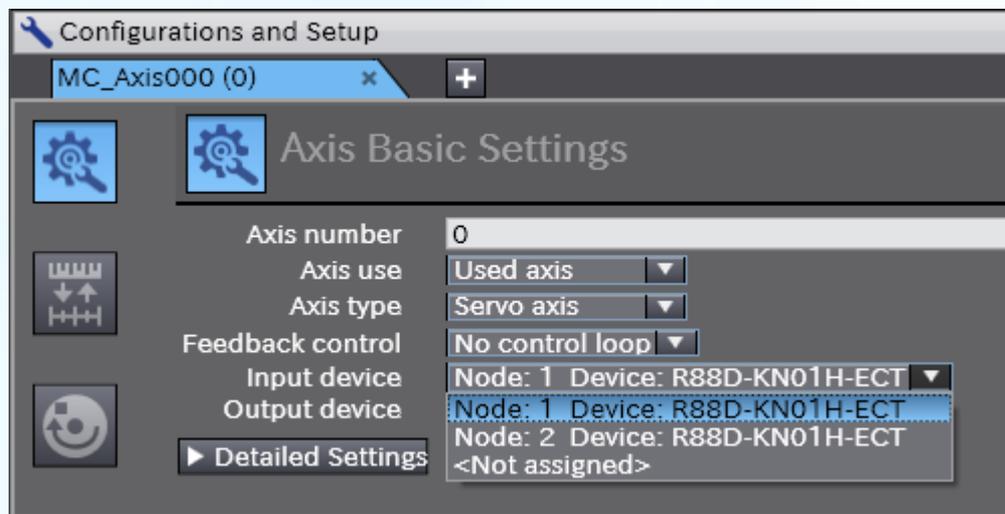
第五章 Motion导入基础

- 举例
- 现有一基本的横向丝杠结构，配置如下，电机旋转编码器分辨率为20位，等于1048576ppr,丝杠进给量为10mm。



第五章 Motion导入基础

- 选择“Axis Basic Setting”设置成使用
- 伺服轴类型
- 正确的伺服



第五章 Motion导入基础

- 选择” Unit Conversion Settings”项。
- 设置正确的每转脉冲数1048576
- 设置每转进给量10000
- 选择正确的单位 微米

Unit Conversion Settings

Unit of display pulse mm um nm degree inch

Command pulse count per motor rotation 1048576 pulse/rev

Work travel distance per motor rotation 10000 um/rev

Reference: Unit conversion formula

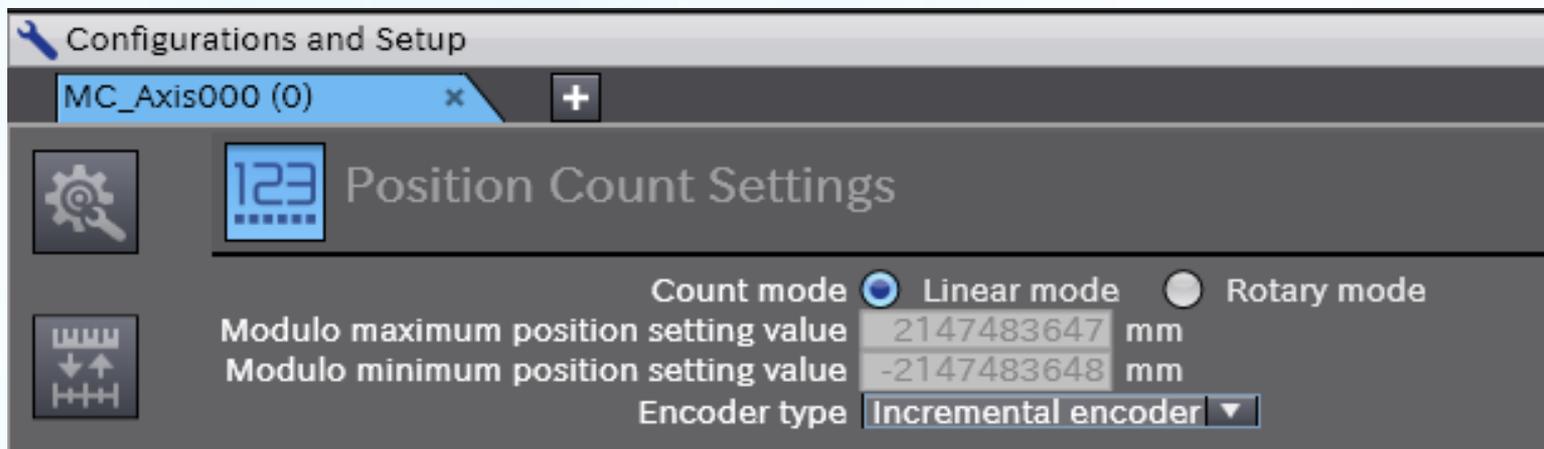
$$\text{Number of pulses [pulse]} = \frac{\text{Command pulse count per motor rotation (UDINT)}}{\text{Work travel distance per motor rotation (LREAL)}} \cdot \text{Travel distance [Unit of display]}$$

第五章 Motion导入基础

- 选择“Operation Settings”项
- 设置最大速度。
- 设置最大点动速度
- 设置最大加速度
- 设置最大减速度
- 设置速度警告范围
- 设置加速度警告范围
- 设置减速度警告范围

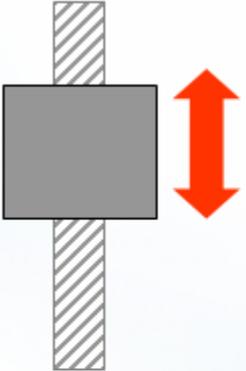
第五章 Motion导入基础

- 选择“Position Count Settings”项
- 设置线性模式或者旋转模式
- 线性模式：当计数到了极限值后就不会再变化（溢出）
- 旋转模式：当计数到了设定的最大和最小值后会进行循环



第五章 Motion导入基础

使用的推荐

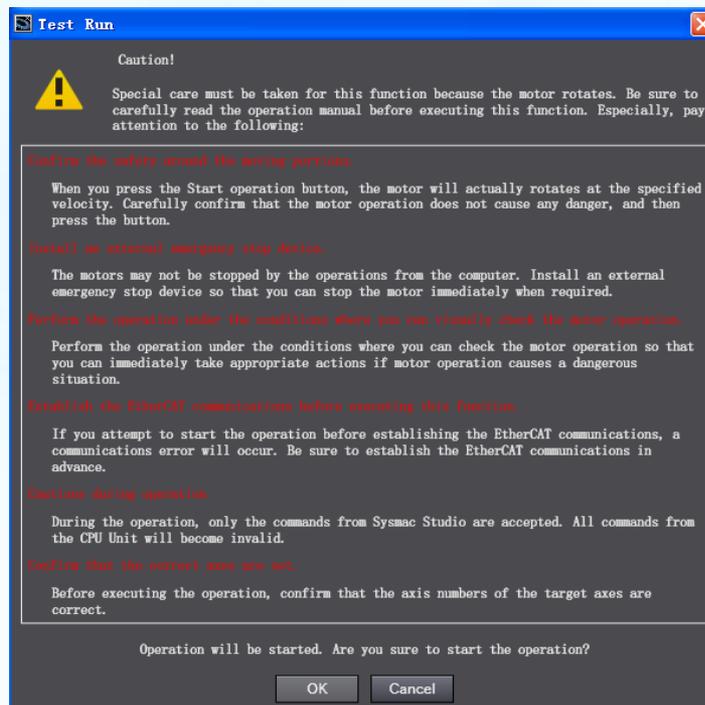
结构	计数模式
	这种位置控制的情况下比较适合用线性模式。
	这种旋转的情况适合使用旋转模式。

第五章 Motion导入基础

- 试运行（**Test Run**）是用以对电机做最基本的测试。
- 试运行的内容涵盖：
 - （1）伺服点动(**Jog**)
 - （2）伺服定位（相对和绝对）
 - （3）找原点
 - （4）**I/O**检测

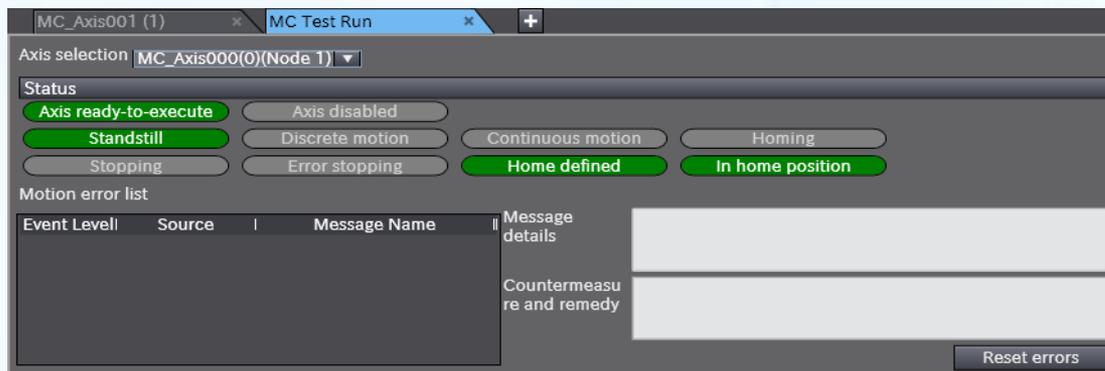
第五章 Motion导入基础

- (1) 使PLC在线。
- (2) 右键“Motion Control Setup”下的“Axis Settings”右键该选项选择“Start MC Test Run”

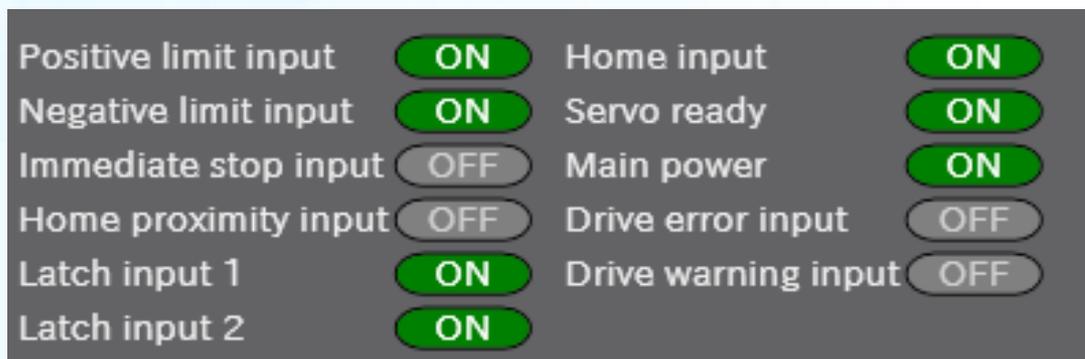


第五章 Motion导入基础

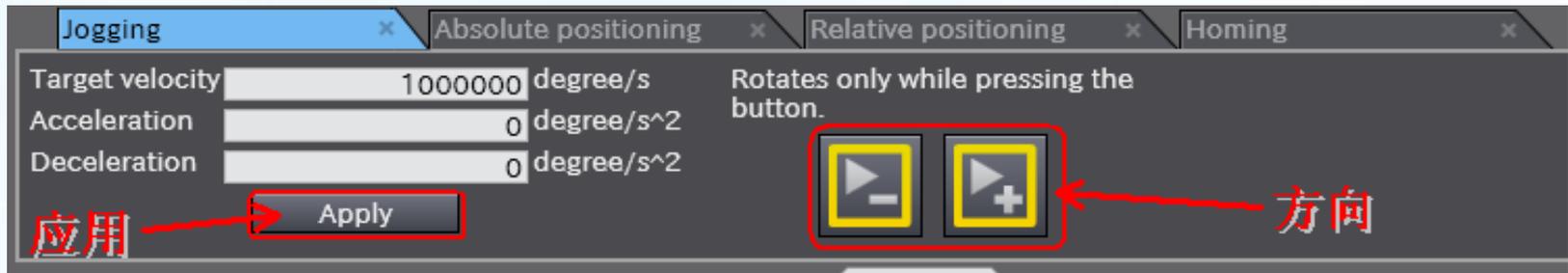
- 状态显示和操作界面



- I/O显示



- 试运行
- (1) 选择Servo ON
- (2) 设置设置目标速度，加速度和减速度后选择“Apply”



第六章 运动控制程序

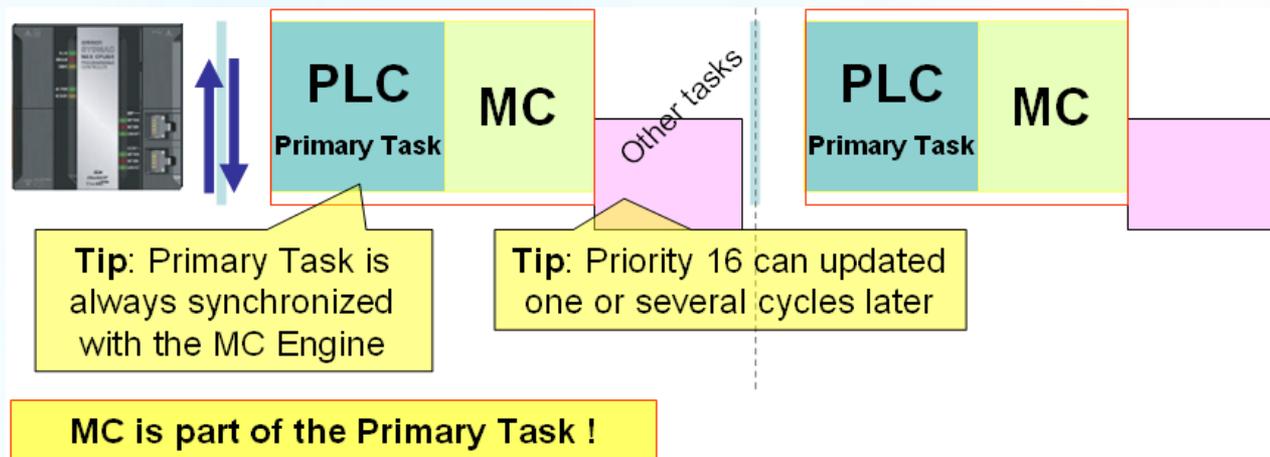
- 第一节 运动任务
- 第二节 **PLC Open**
- 第三节 轴状态
- 第四节 运动控制命令
- 第五节 运动控制功能
- 第六节 运动控制命令动作序列

第六章 运动控制程序

- 运动控制
- **NJ**运动编程是通过**IEC**标准的梯级编程或者**ST**编程来实现的。
- 符合**IEC**的程序不仅用于运动控制还可以做一些基本通用的逻辑控制

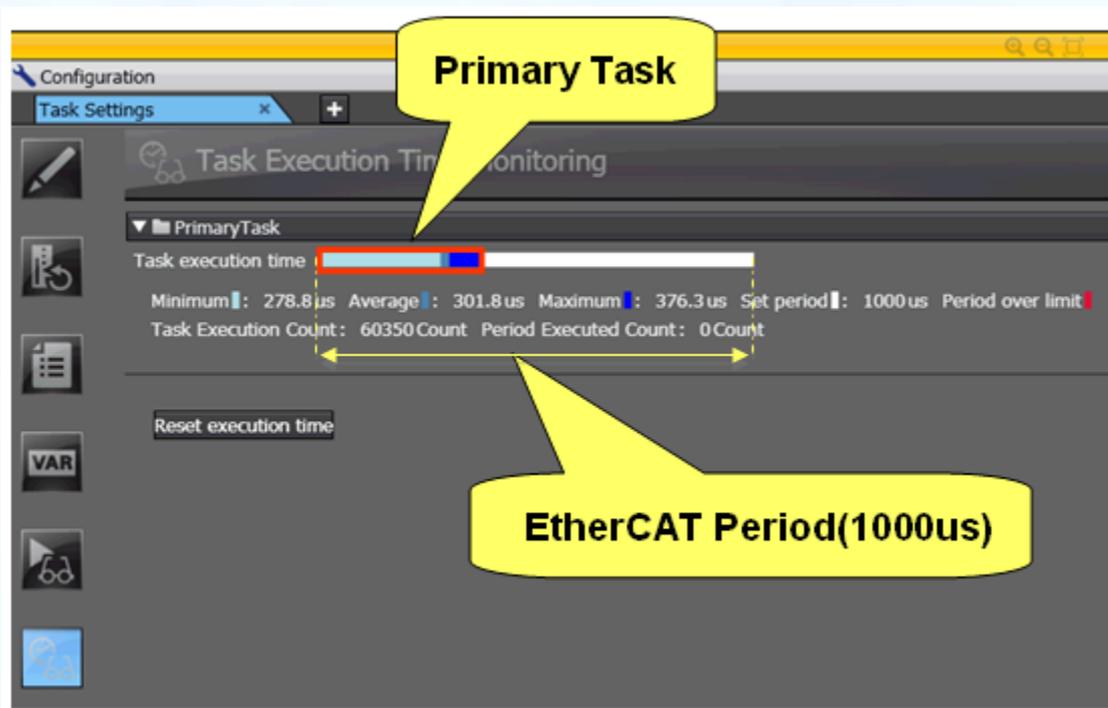
第六章 运动控制程序

- 运动控制命令必须被设置在主要周期任务或者16号周期任务里
- 但是如果某个控制命令要求立即动作则必须被设置在主要周期任务里。



第六章 运动控制程序

- 方便的任务周期时间监视



第六章 运动控制程序

- PLC Open协会
- PLCopen是一个旨在建立基于IEC61131-3标准的通用编程的国际协会组织
- PLCopen协会的技术委员会聚集了来自于大型OEM厂家，大型最终用户，运动控制技术支持厂商
- 欧姆龙公司是现PLCOpen Motion Task委员会成员之一，并在其中起到积极推动作用

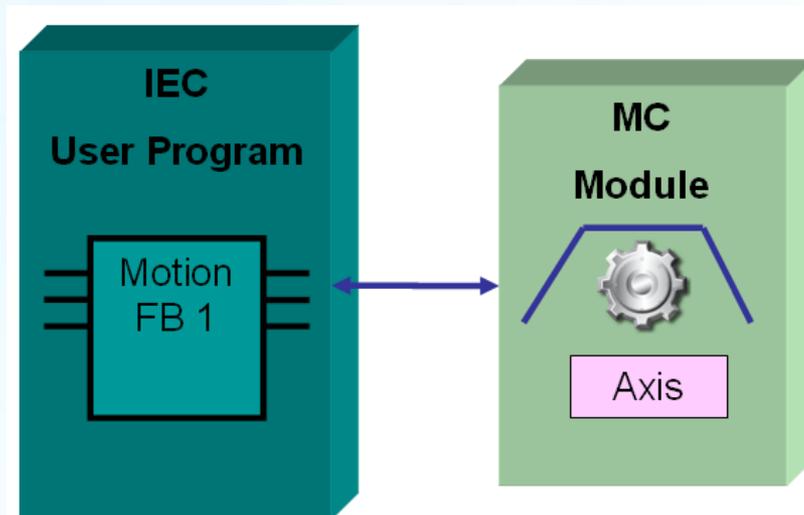


第六章 运动控制程序

- **PLCOpen**的标准是由几家核心运动生产厂家提供的重要经验来构成的。他们是**SIEMENS**（西门子）, **OMRON**（欧姆龙）, **Schneider**（施耐德）, **Beckhoff**（倍福）, **B&R**（贝加莱）, **KUKA Robotic**（库卡） 等。
- **PLCOpen**的标准是完全着眼于现场实际运用和客户最实际运用的要求。
- **NJ**系统使用**PLCopen**标准来实现运动控制。
- **NJ**系统同时也包含欧姆龙自己开发的运动控制功能（比如**MC_Movelink**）。

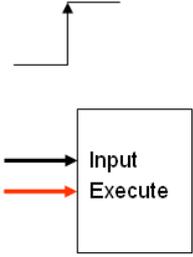
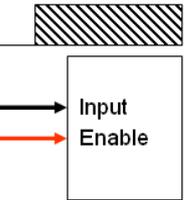
第六章 运动控制程序

- 轴状态
- 运动控制功能块（**Motion FB**）一般是用于**MC**去执行伺服运动功能以及改变轴的状态
- 轴状态包括失效状态，保持状态，停止状态，错误导致停止状态，找原点状态
- 独立运动控制状态，连续运动控制状态，同步运动控制状态



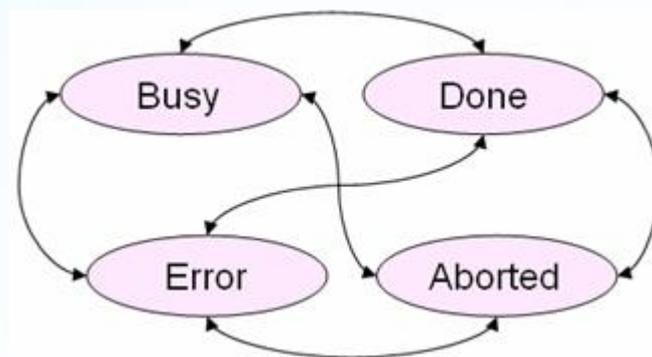
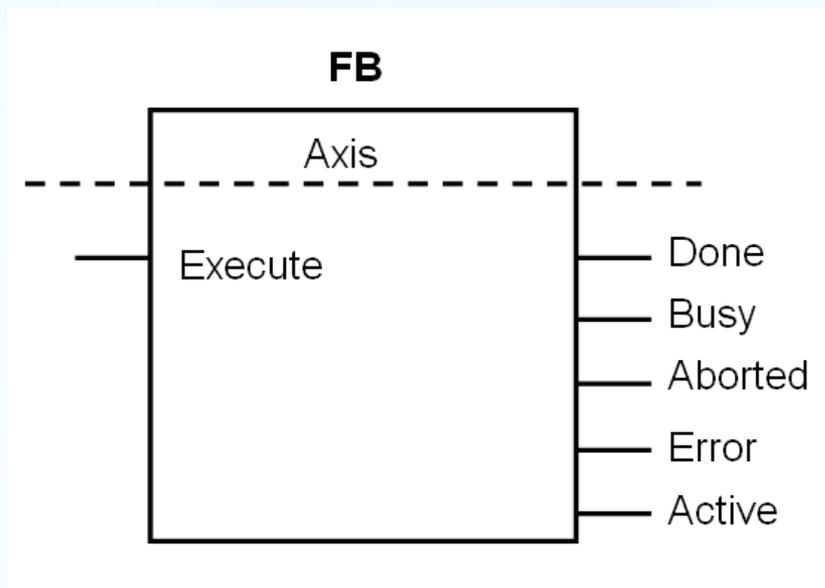
第六章 运动控制程序

- 运动控制命令
- 运动控制命令分为
- **Execute型**
- **Enable型**

<p>EXECUTE (Rising Edge BOOL)</p>  <p>The diagram shows a signal that transitions from low to high (a rising edge). Below it is a rectangular block with two input arrows: a black arrow labeled 'Input' and a red arrow labeled 'Execute'.</p>	<p>These instructions will continue execution until one of the following status occurs.</p> <ul style="list-style-type: none">• The specified operation is completed.• Another motion control instruction is executed and interrupts operation.• The instruction is restarted when Execute changes from FALSE to TRUE again. <p>Values for the other input variables are updated when Execute changes to TRUE.</p> <p>Example: MC_Move_Relative</p>
<p>ENABLE (Enable BOOL)</p>  <p>The diagram shows a rectangular pulse with diagonal hatching, representing a period of time where the signal is true. Below it is a rectangular block with two input arrows: a black arrow labeled 'Input' and a red arrow labeled 'Enable'.</p>	<ul style="list-style-type: none">• These motion control instructions execute every period while the input variable Enable to the motion control instruction is TRUE.• As long as Enable is TRUE, the other input variables are also refreshed every period. <p>Example: MC_Power</p>

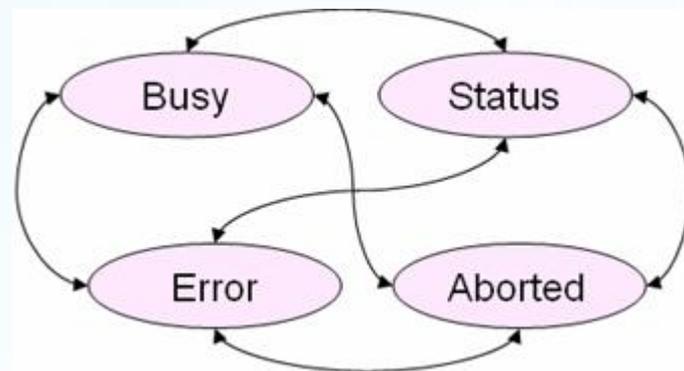
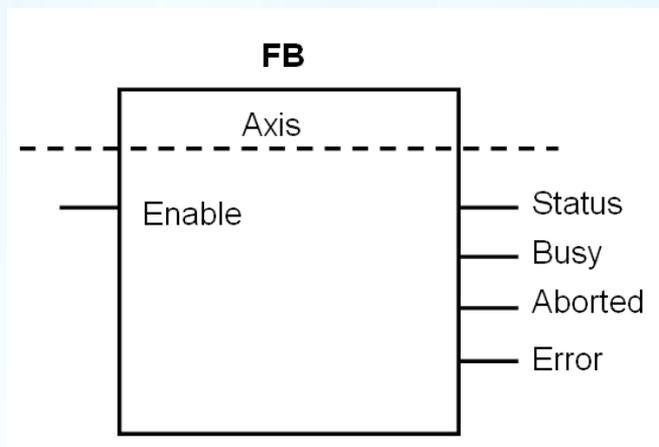
第六章 运动控制程序

- **Execute型FB**
- 基本结构与四个反映当前状态的变量



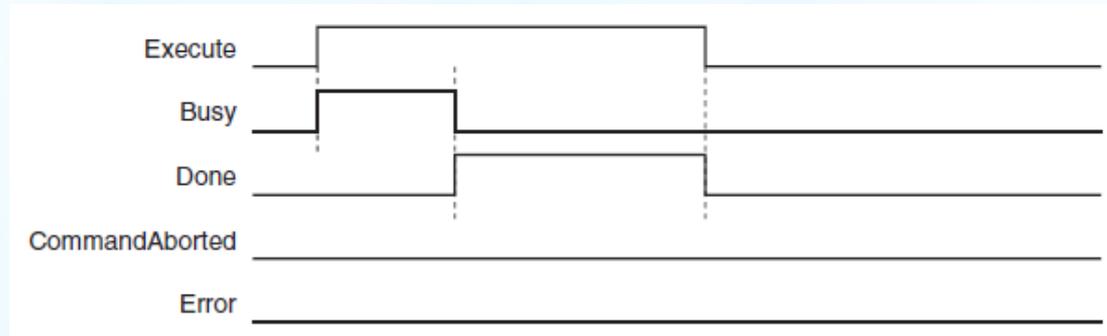
第六章 运动控制程序

- Enable型FB
- 基本结构与四个反映当前状态的变量



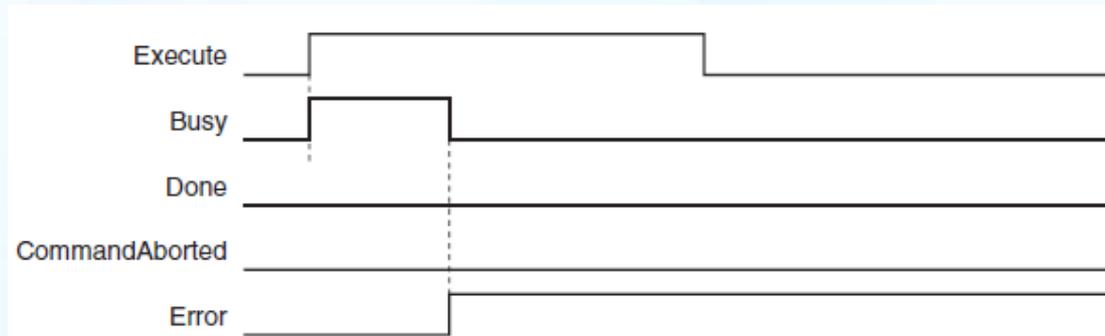
第六章 运动控制程序

- Execute型时序图
- 当输入变量Execute为True时且在Execute转为False前都没有发生Error的情况下的时序图

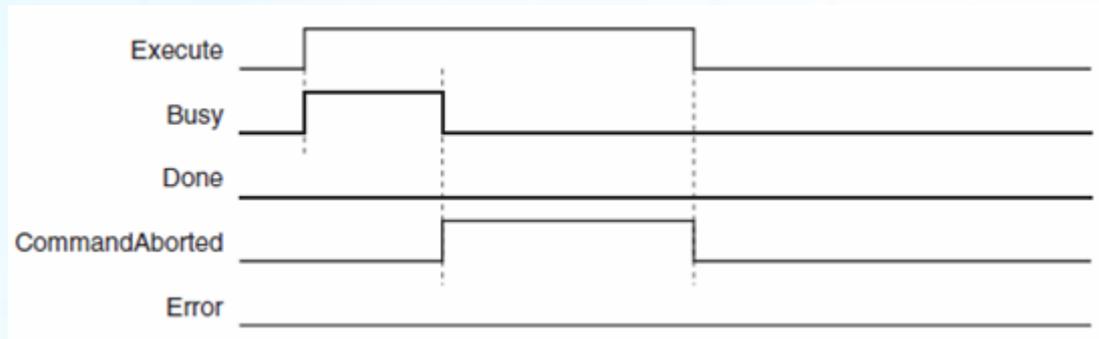


第六章 运动控制程序

- 当输入变量Execute为True期间发生了一个错误，即便当Execute转为False后，Error依然会保持True状态



- 当输入变量Execute为True的情况时，指令在动作时发生了中断



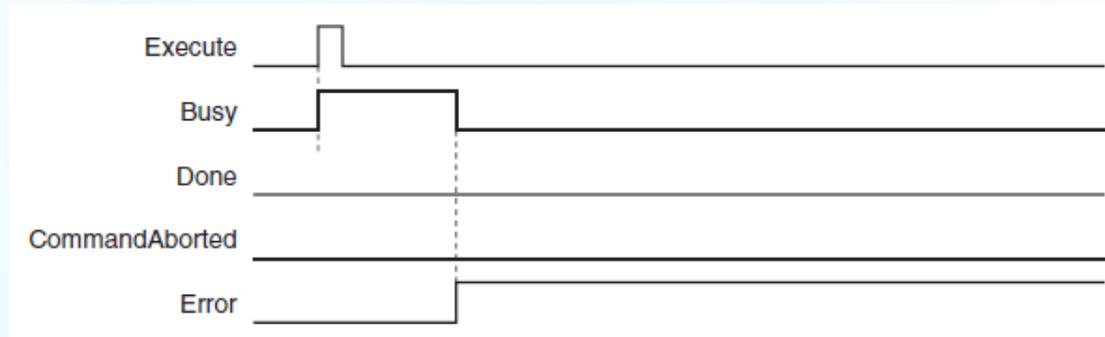
第六章 运动控制程序

- 当输入变量**Execute**只保持**True**一个周期的时间，在指令动作过程中无发生**Error**，输出变量**Done**在指令动作完成后只会转为**True**一个周期，请注意这个**Done**还是有输出的，这点很重要。



第六章 运动控制程序

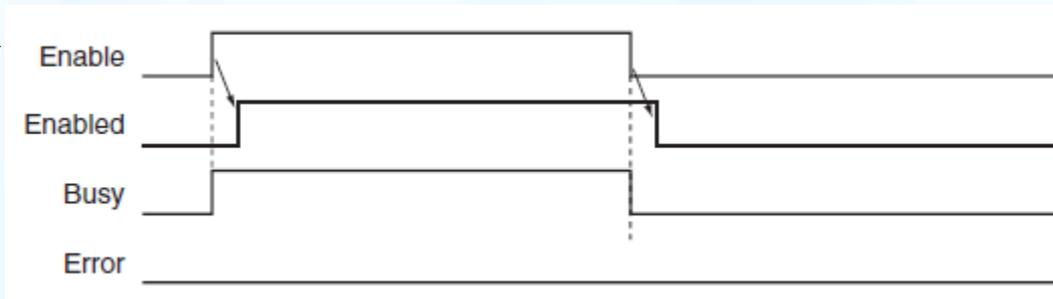
- 当输入变量**Execute**只保持**True**一个周期时间，在指令动作过程中发生**Error**，输出变量**Error**亦会一直保持为**True**



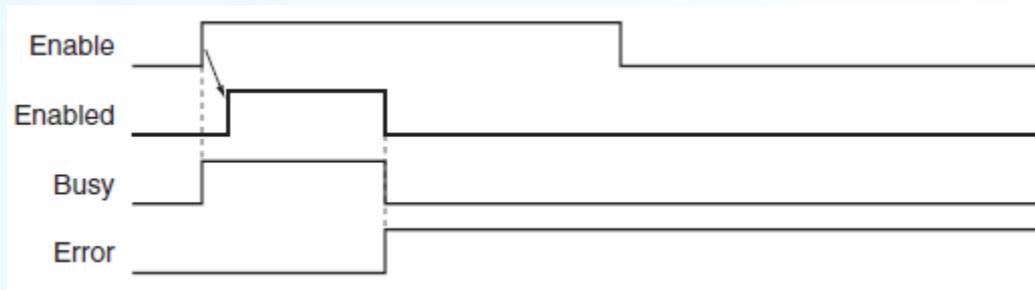
第六章 运动控制程序

- **Enable型时序图**

- 当输入变量“Enable”转为True并没有在指令运行过程中报错



- 当输入变量“Enable”转为True并在指令运行过程中报错



第六章 运动控制程序

- **MC功能**
- **MC功能是NJ系列PLC用对轴进行控制技术的部份**
- **运动控制功能块(MC FB)是指逻辑控制器内适用于MC的运动控制功能且符合IEC61131-3的那些功能块。**

第六章 运动控制程序

运动控制的分类



第六章 运动控制程序

- **MC_Power**
- **MC_Power**这条指令的目的是用以使伺服驱动器准备就绪等待，使其能被操作
- 功能介绍
- 当输入变量转为**True**后，由输入输出变量**Axis**指定的那根轴就会变为准备就绪。
- 在轴处于准备就绪后用户可以对该轴实行控制。
- 当输入变量**Enable**转为**False**后，指定轴的准备状态立刻消除。

第六章 运动控制程序

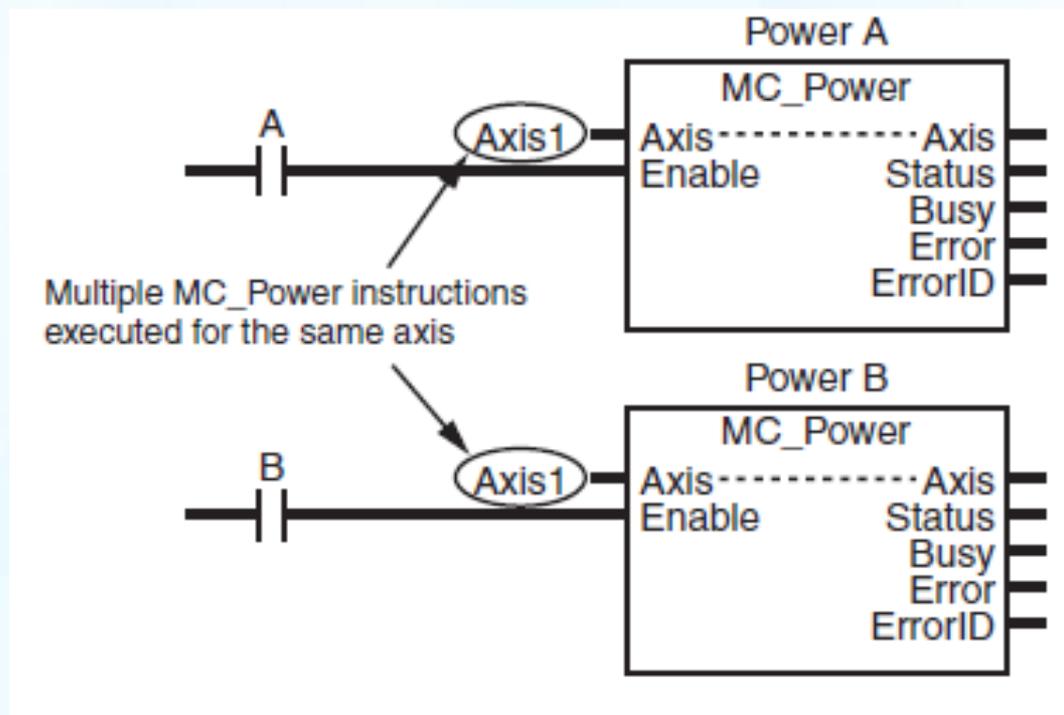
- 当准备状态被消除后用户无法对轴进行控制，这是因为在此状态下轴不会承认相关的动作命令。动作命令下达后会回馈为一个错误。在轴失效状态下用户只能只用**MC_Power**和**MC_Reset**指令
- 可以通过这个指令来使得动作中的轴停止动作

第六章 运动控制程序

- 当**Commandaborted**转为**True**，动作指令的输出会相应停止并且轴不再处于准备状态。
- 该指令适用于伺服轴和伺服虚轴，如果用于编码器轴或者编码器虚轴会产生一个错误。
- 该指令没有**Buffer**模式，但事实上还是和**Buff**动作里的中断（**Aborting**）类似。

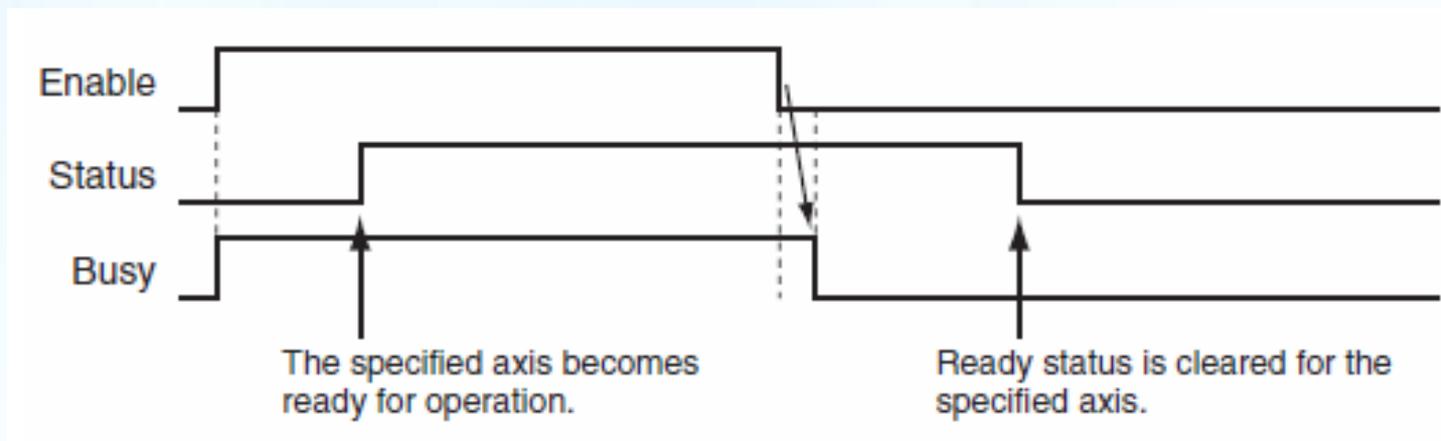
第六章 运动控制程序

- 如有多条MC_Power命令对同一对象轴进行操作，后者有效



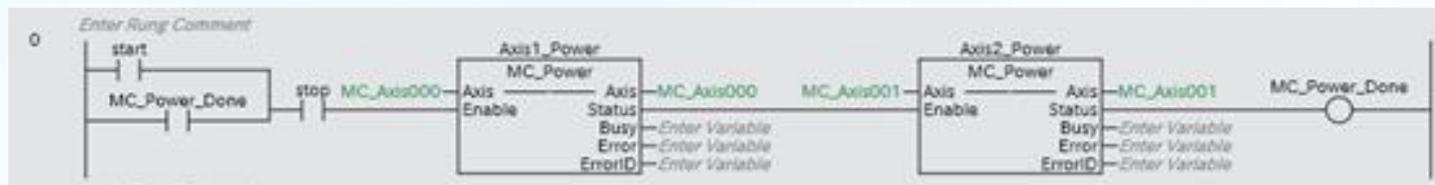
第六章 运动控制程序

· MC_Power时序图



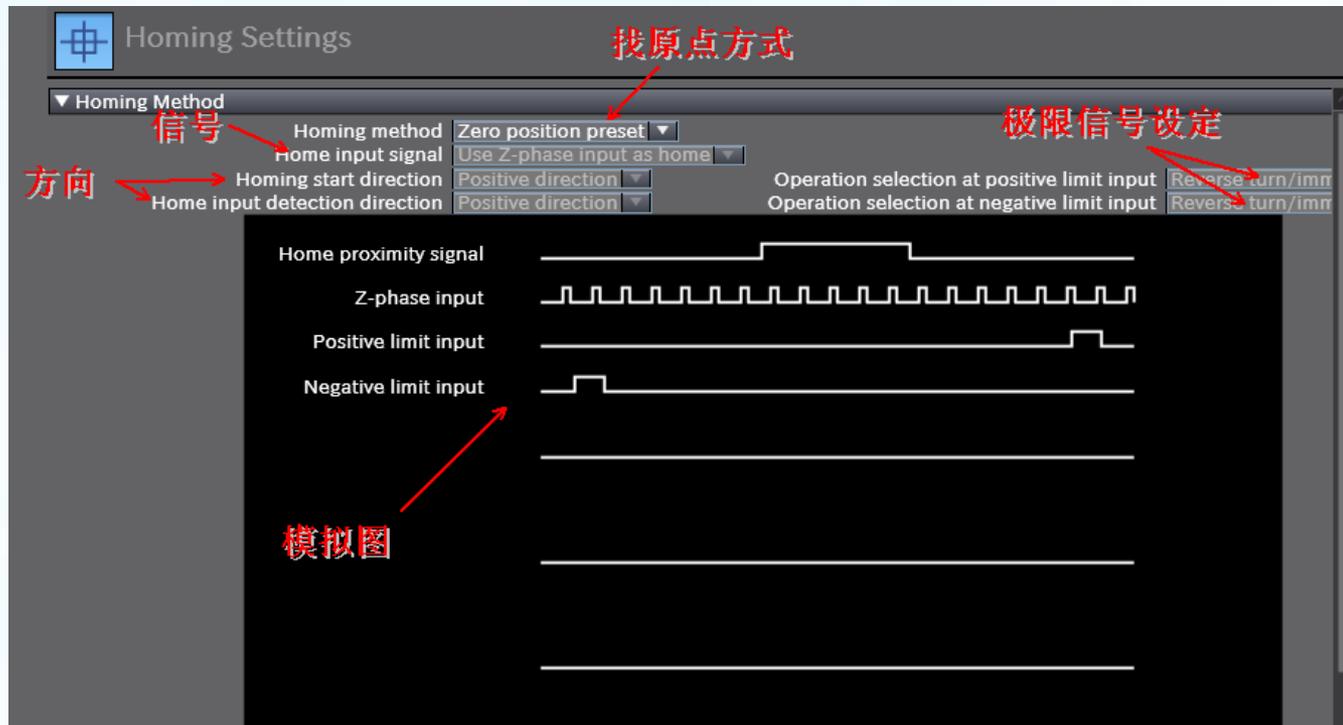
第六章 运动控制程序

- 练习
- 通过控制一个变量” **Start**”来实现两个轴（对应**G5**伺服）的伺服**ON**动作。（伺服锁上）



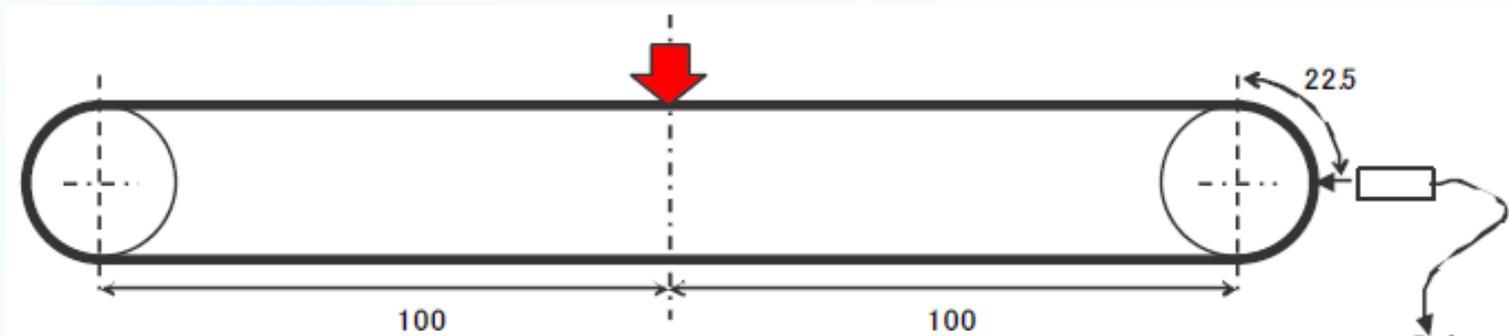
第六章 运动控制程序

- MC_Home指令
- MC_Home指令是用以驱动轴去确定原点的。其一般会用到极限信号，原点接近信号，原点信号
- Sysmac Studio 提供了方便的方法用以找原点



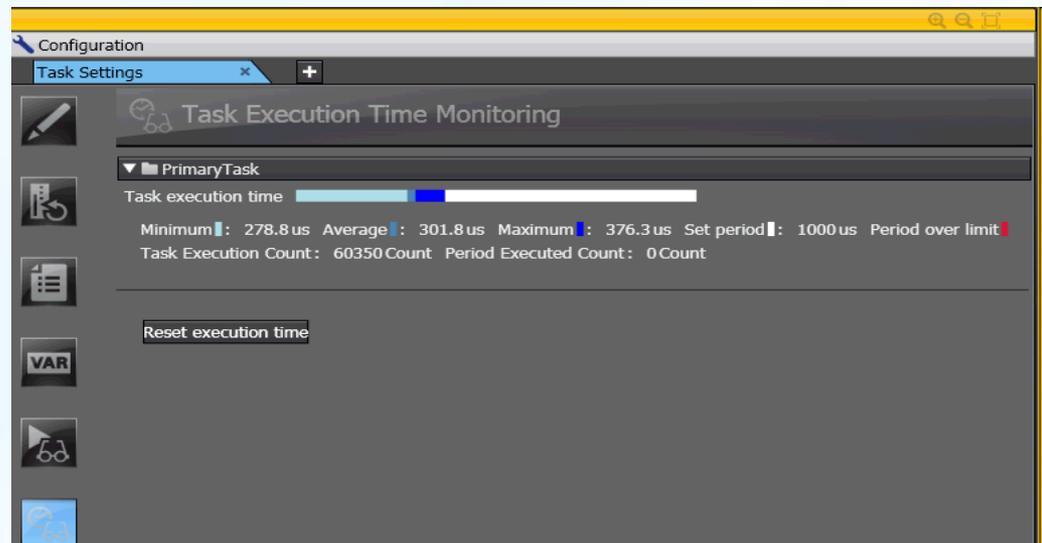
第六章 运动控制程序

- 举例：
- 环境：一套传送皮带轮。在皮带上有一处可以被光电开关检测到的标识线。
- 要求：检测线被光电开关检测出后。皮带倒退，一直退回到标志线出现在中央位置（图中红色箭头所指的位置）皮带总长490mm。此时认为原点已到。



第六章 运动控制程序

- 设置
- 注意 “Homing compensation value” 的正确设置



Velocity/Acceleration/Deceleration					
Homing velocity	100	mm/s	Homing approach velocity	100	mm/s
Homing acceleration	1000	mm/s ²	Homing deceleration	0	mm/s ²
Homing jerk	10000	mm/s ³			
Others					
Home input mask distance	10000	mm	Home offset	0	mm
Homing holding time	100	ms			
Homing compensation value	-122.5	mm	Homing compensation velocity	50	mm/s

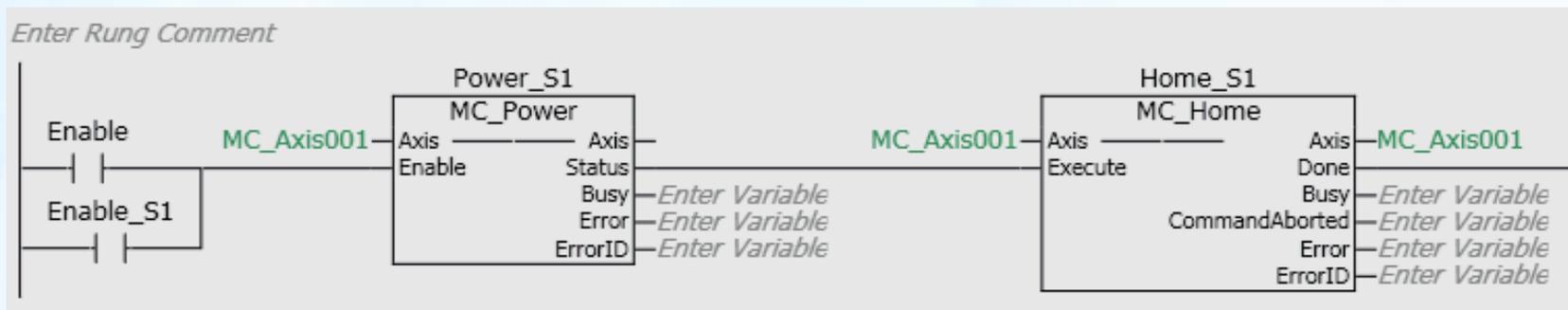
第六章 运动控制程序

伺服的设置（参考）

●	Pn400.0	3400.00	Input Signal Selection 1 - Position control/full closing...	94: E-STOP (Compulsion alarm input) - Contact NC
●	Pn400.1	3400.00	Input Signal Selection 1 - Speed control	94: E-STOP (Compulsion alarm input) - Contact NC
●	Pn400.2	3400.00	Input Signal Selection 1 - Torque control	94: E-STOP (Compulsion alarm input) - Contact NC
●	Pn401.0	3401.00	Input Signal Selection 2 - Position control/full closing...	81: POT (Forward drive prohibition input) - Contact NC
●	Pn401.1	3401.00	Input Signal Selection 2 - Speed control	81: POT (Forward drive prohibition input) - Contact NC
●	Pn401.2	3401.00	Input Signal Selection 2 - Torque control	81: POT (Forward drive prohibition input) - Contact NC
●	Pn402.0	3402.00	Input Signal Selection 3 - Position control/full closing...	82: NOT (Reverse drive prohibition input) - Contact NC
●	Pn402.1	3402.00	Input Signal Selection 3 - Speed control	82: NOT (Reverse drive prohibition input) - Contact NC
●	Pn402.2	3402.00	Input Signal Selection 3 - Torque control	82: NOT (Reverse drive prohibition input) - Contact NC
●	Pn403.0	3403.00	Input Signal Selection 4 - Position control/full closing...	22: DEC (Starting point neighborhood input) - Contact NO
●	Pn403.1	3403.00	Input Signal Selection 4 - Speed control	22: DEC (Starting point neighborhood input) - Contact NO
●	Pn403.2	3403.00	Input Signal Selection 4 - Torque control	22: DEC (Starting point neighborhood input) - Contact NO
●	Pn404.0	3404.00	Input Signal Selection 5 - Position control/full closing...	2B: EXT3 (External latch input 3) - Contact NO
●	Pn404.1	3404.00	Input Signal Selection 5 - Speed control	2B: EXT3 (External latch input 3) - Contact NO
●	Pn404.2	3404.00	Input Signal Selection 5 - Torque control	2B: EXT3 (External latch input 3) - Contact NO
●	Pn405.0	3405.00	Input Signal Selection 6 - Position control/full closing...	21: EXT2 (External latch input 2) - Contact NO

第六章 运动控制程序

- 程序

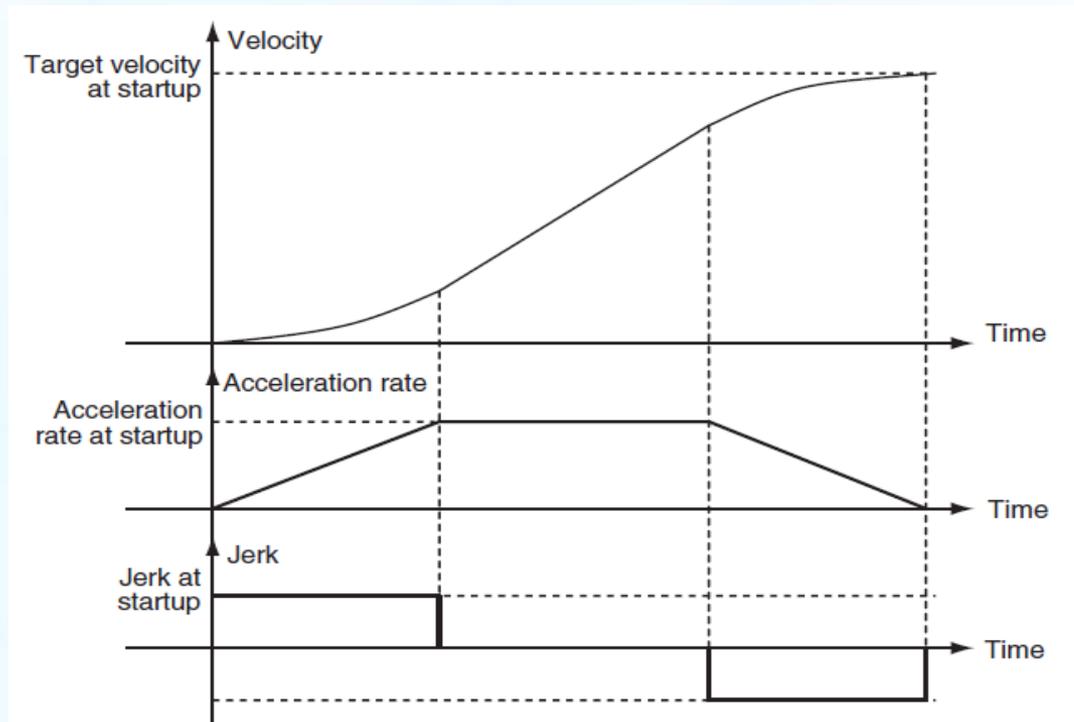


第六章 运动控制程序

- **MC_Move**指令是用于驱动轴进行相对或者绝对定位的指令。（注意，在NJ系统中相对和绝对可以使用同一条指令）
- 指令说明
- **MC_Move**在使用时，主要需要设定速度，加减速，**Jerk**,相对或者绝对位置控制方式

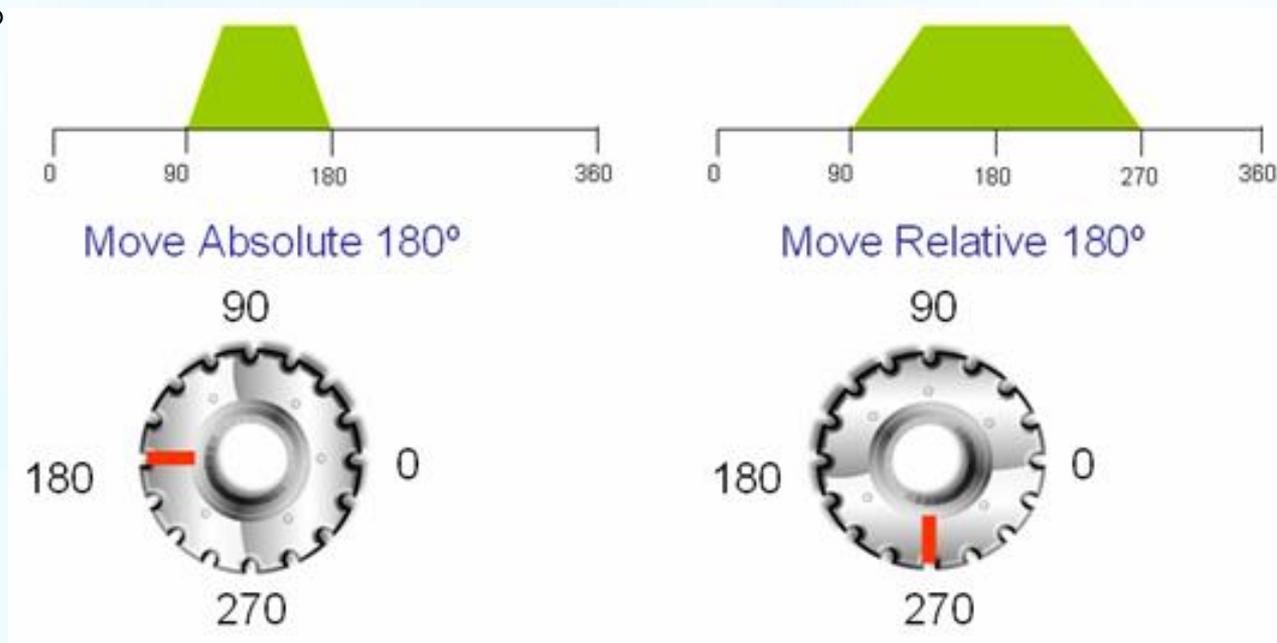
第六章 运动控制程序

- 当Jerk大于0时会生成一个梯型的加速度，这样就能直线加速至目标速度的过程变为一个曲线的加速



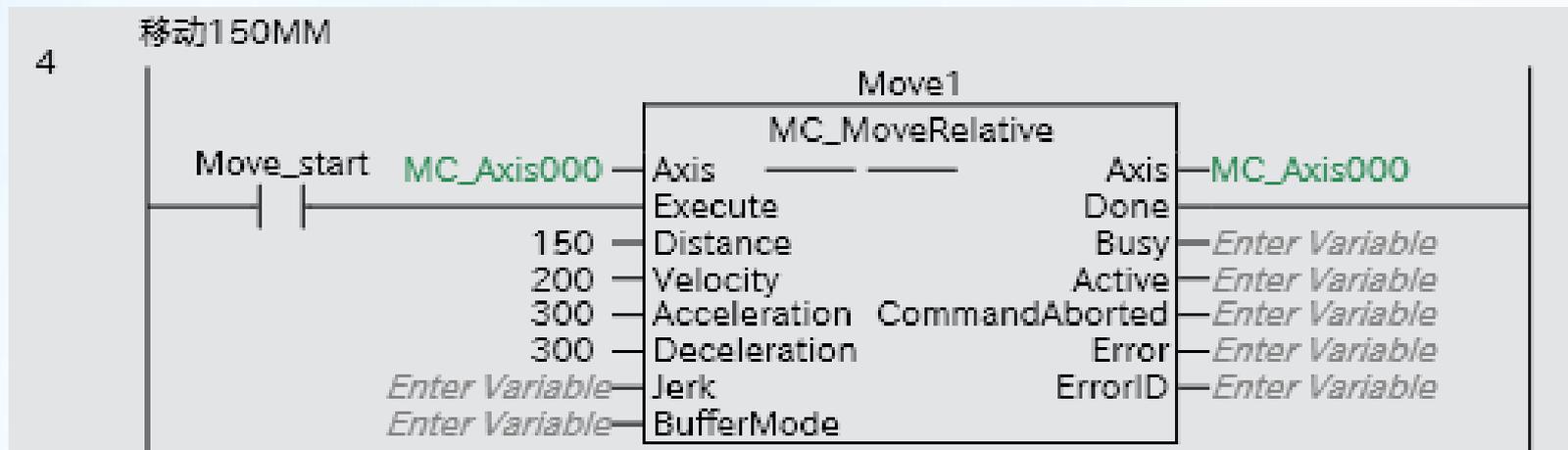
第六章 运动控制程序

- 相对与绝对的概念
- 对于绝对来说，180度是目标值，因此从90度（原停止位置）出发转动到180度的位置
- 对于相对来说，180度是移动量，因此从90度（原停止位置）出发转过了180度后，停在了270度的位置上。



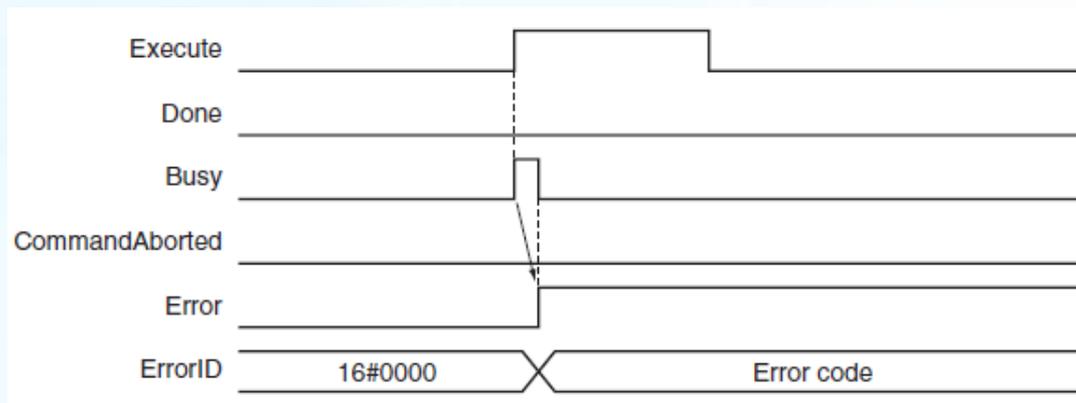
第六章 运动控制程序

- 通过MC_Move的功能块使得轴1和轴2执行相对位置移动。
- 要求使用MC_MoveRelative来实现。
- 要求触发BS2后轴1移动量为150mm。



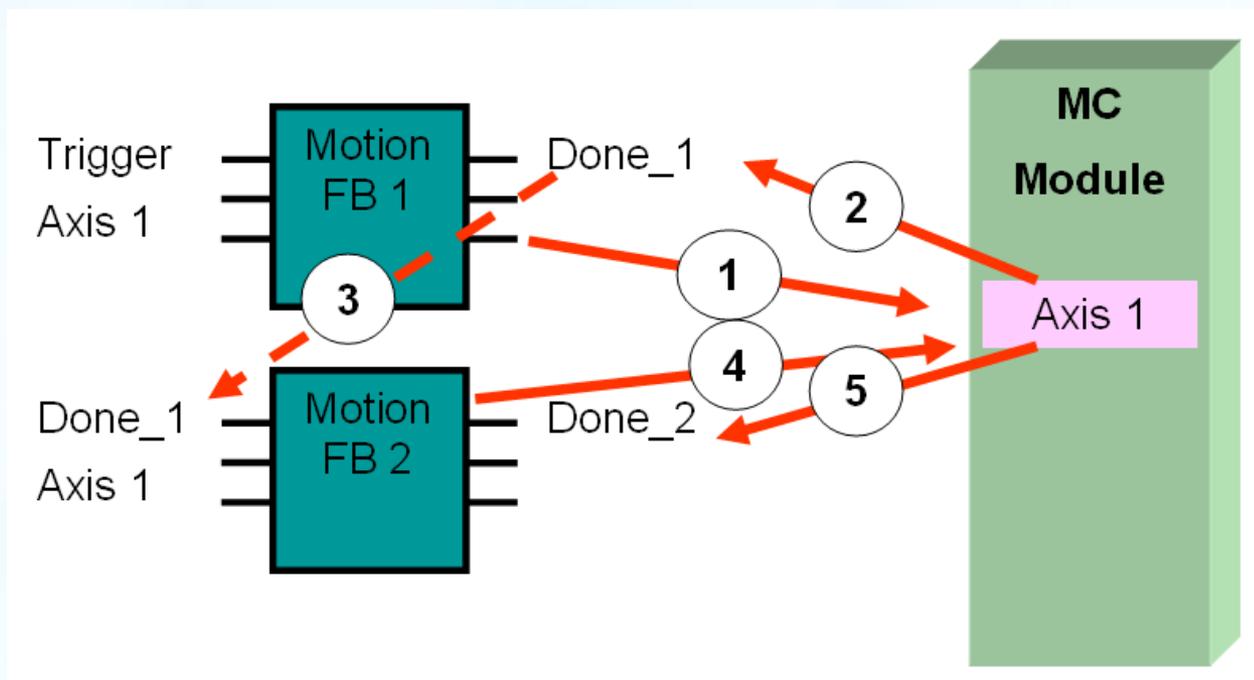
第六章 运动控制程序

- MC_Stop
- MC_Stop用以轴减速停止
- MC_Stop指令说明
- 当输入变量Execute转为True时 停止动作开始执行。
- 当轴状态有报错的情况下，MC_Stop指令无法正常执行，指令将被中断。这类场合需要使用MC_ImmediateStop指令。（运行中报错也会执行完毕）



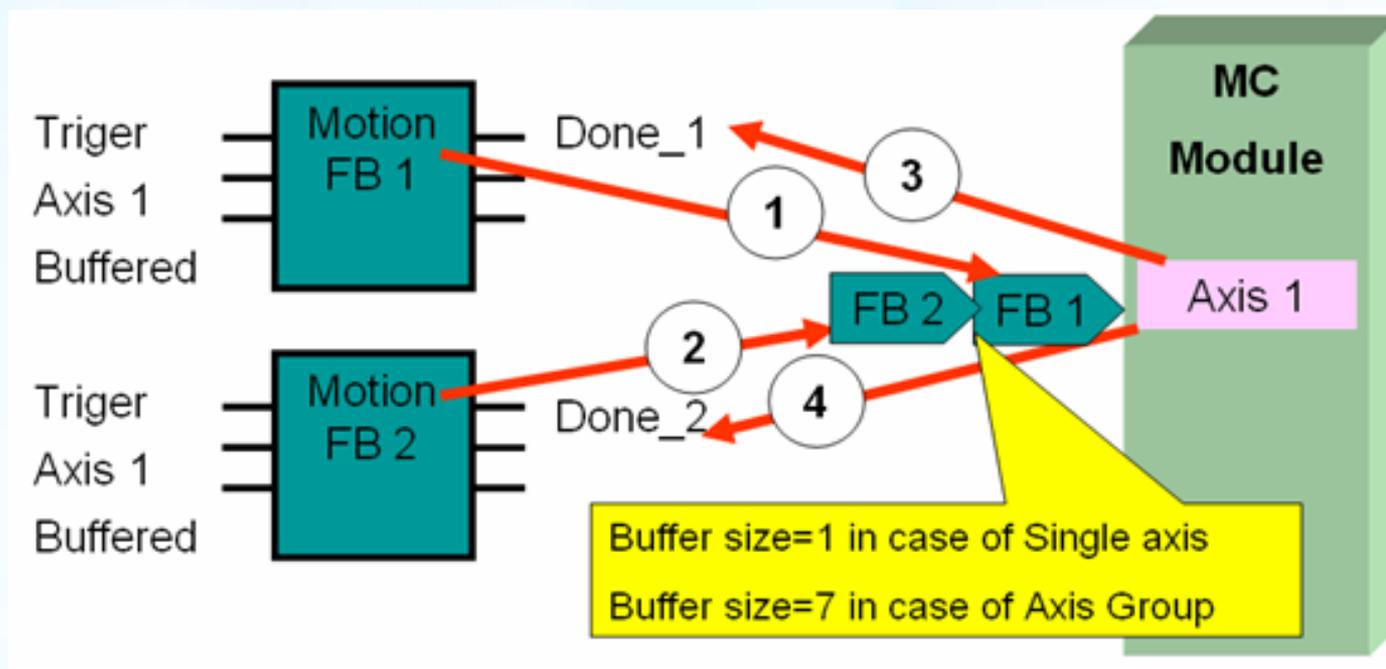
第六章 运动控制程序

- 运动控制序列
- 通过指令向MC发出控制命令



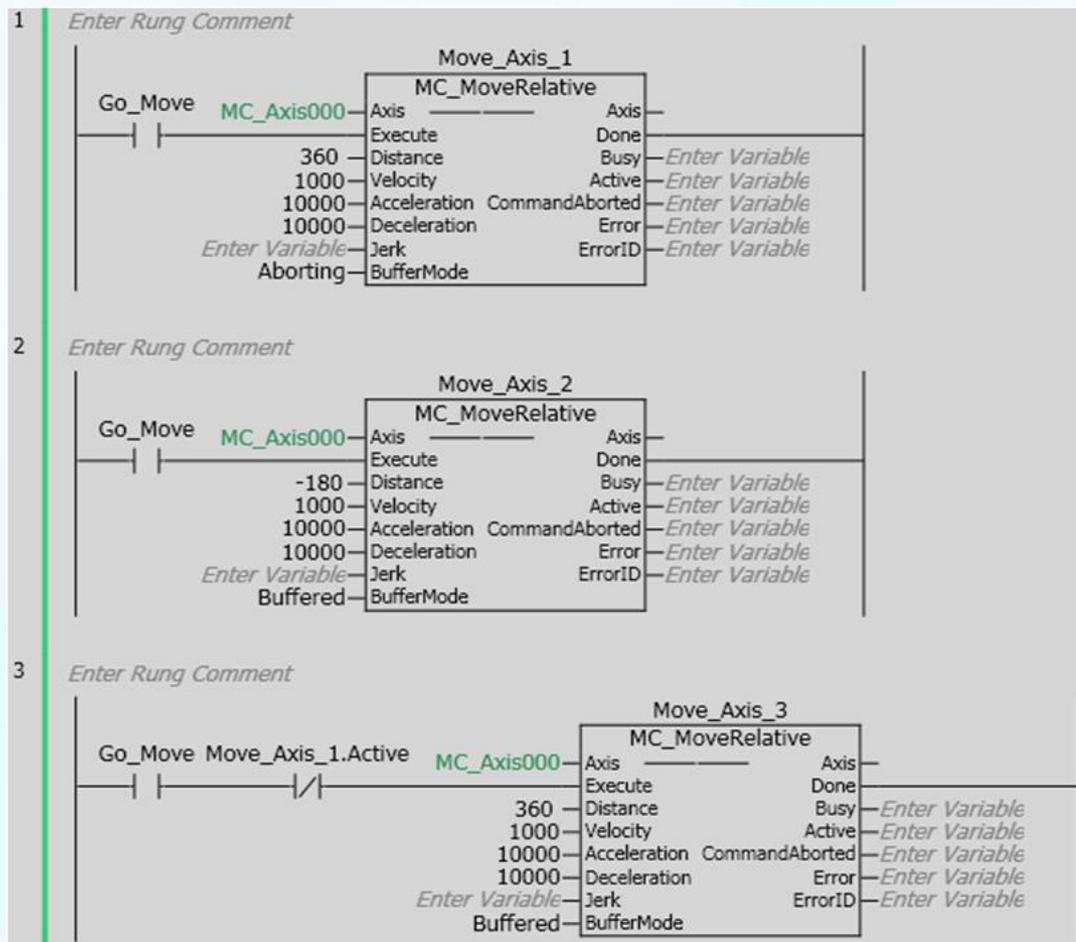
第六章 运动控制程序

- 功能块设置输入变量BufferMode后顺序序列会有所变化



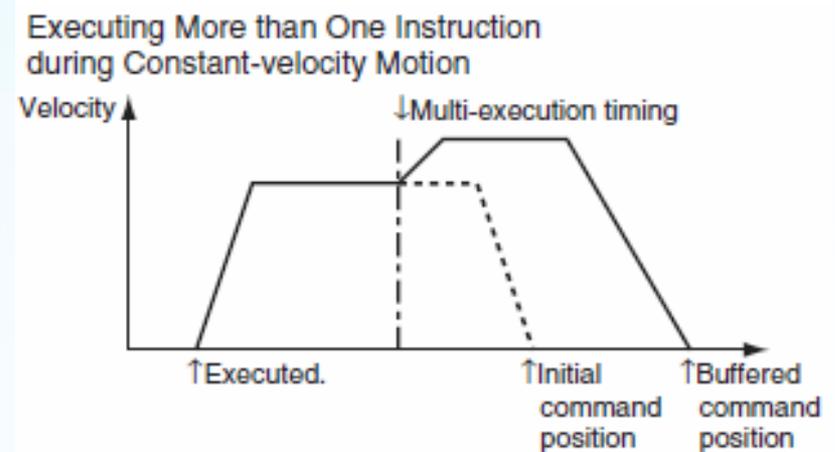
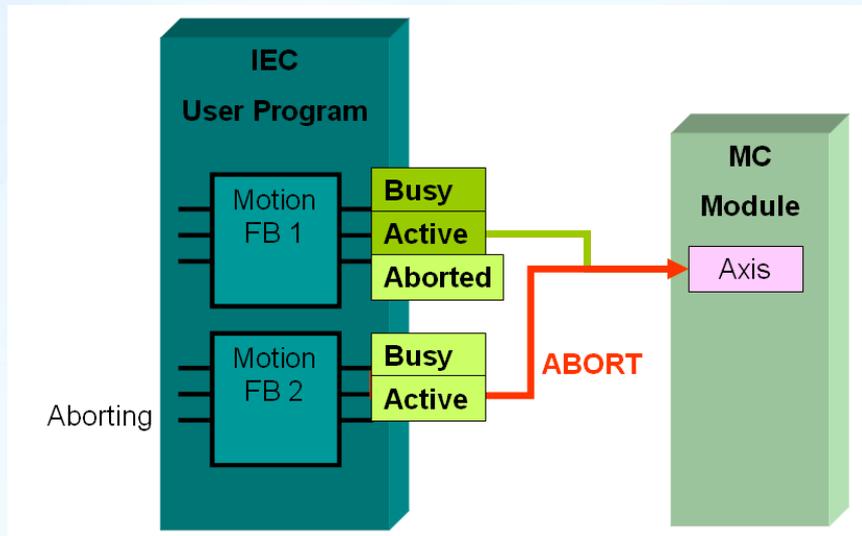
- Buffer举例
- 梯级1中的FB“Move_Axis_1”与梯级2中的FB“Mov_Axis_2”以及梯级3中的FB“Move_Axis_3”都对轴000进行操作（变量MC_Axis000）。

第六章 运动控制程序



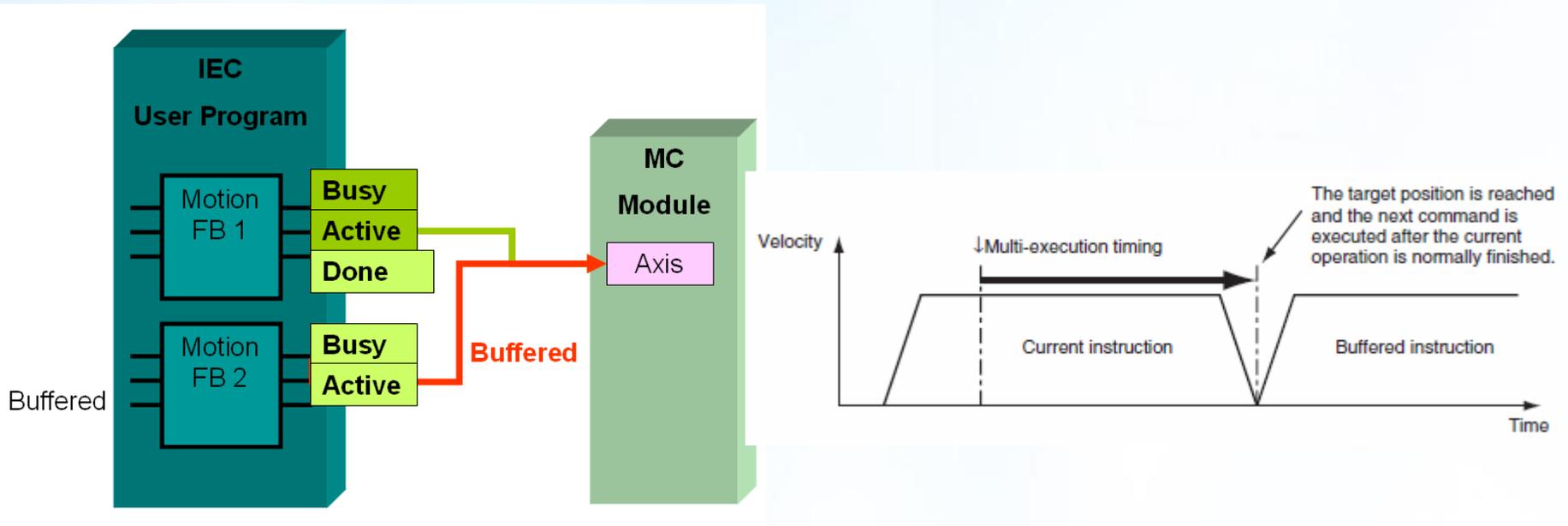
第六章 运动控制程序

- **Aborting 中断模式**
- 当选择**BufferMode**方式为**Aborting**时，后一个功能块的动作会立即中断当前功能块的动作



第六章 运动控制程序

- **Buffered**（待机模式）
- 当选择BUfferMode方式为Buffered时，后一个功能块的动作会等待前一个功能块动作完成后再动作



第六章 运动控制程序

- **Blending**（混合模式）
- **Blending**确定了一个运动控制指令在到达目标位置后到下一个运动控制指令之间速度是如何变化的。也就是说和**Aborting**不同的是当有多重动作的时候并不是立刻中断当前运动而是依然会运动到当前运动设定的位置，与**Buffered**不同的是到达位置后并不是将速度终止到0后再自动运行下一个运动控制指令，而是连续运动。

第七章 运动试运行功能

第一节 数据跟踪

第二节 监视窗口

第三节 仿真功能

第七章 运动试运行功能

- 数据跟踪
- 数据跟踪功能允许用户对相关的变量进行指定，指定后能够保存该变量的值而不需要任何额外的编程。
- 用户可以选择两种跟踪方式
 - (1) 触发跟踪
 - (2) 连续跟踪。

第七章 运动试运行功能

- 触发跟踪：用户先和数据记录动作的之前时刻以及结束时刻设置触发条件。
- 连续跟踪：整个过程中不需要任何触发条件进行连续采样，结果会保存在计算机的相关路径下。

第七章 运动试运行功能

选择数据跟踪



添加和删除各类显示窗口

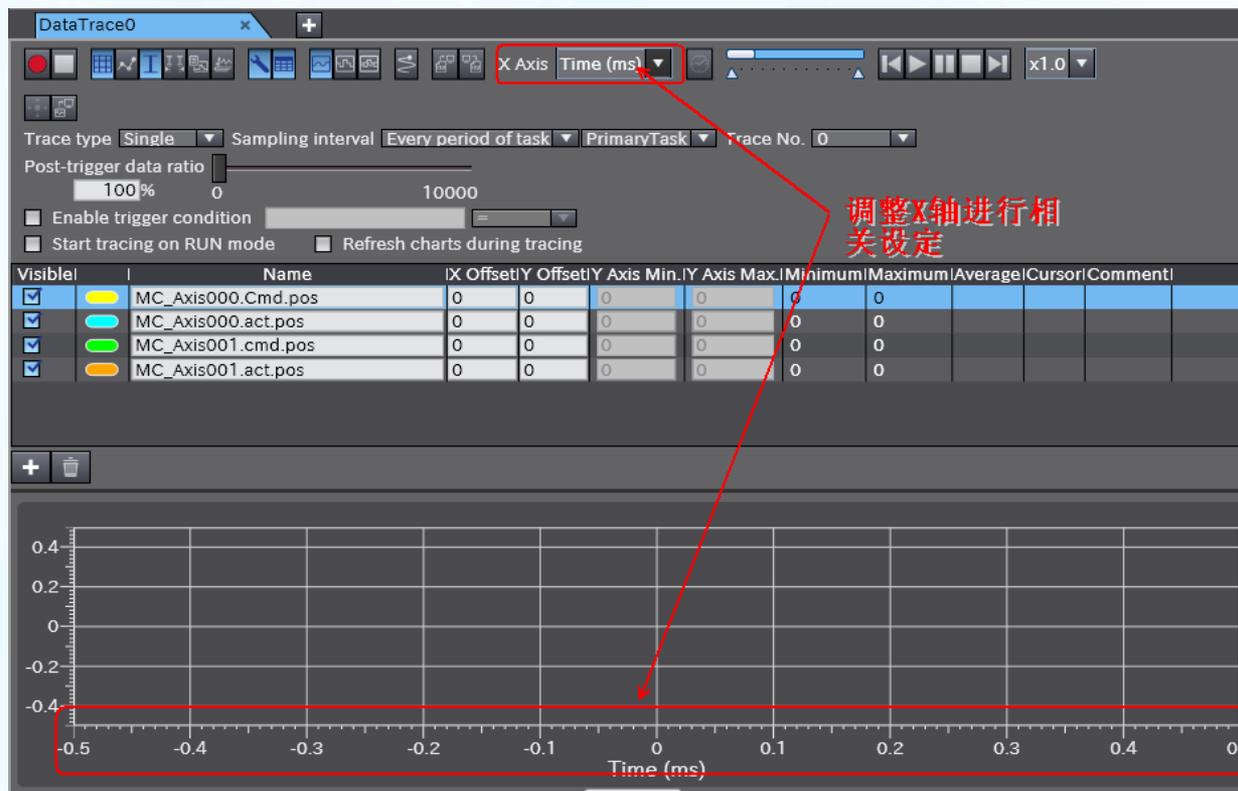
启动数据跟踪

运行已经保存的数据过程

Visible	Name	X Offset	Y Offset	Y Axis Min.	Y Axis Max.	Minimum	Maximum	Average	Cursor	Comment
<input checked="" type="checkbox"/>	MC_Axis000.Cmd.pos	0	0	0	0	0	0			
<input checked="" type="checkbox"/>	MC_Axis000.act.pos	0	0	0	0	0	0			
<input checked="" type="checkbox"/>	MC_Axis001.cmd.pos	0	0	0	0	0	0			
<input checked="" type="checkbox"/>	MC_Axis001.act.pos	0	0	0	0	0	0			

添加和删除跟踪的变量对象

第七章 运动试运行功能



第七章 运动试运行功能

- 触发跟踪的设定

- (1) 采样间隔 (sampling interval)

- (2) 触发后数据比(Post-trigger data ratio)

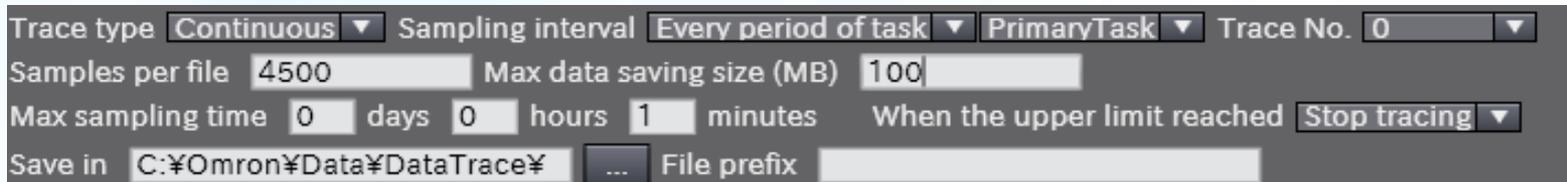
- (3) 触发条件:

- 位变量

- 数字变量

第七章 运动试运行功能

- 连续触发的设定
- 连续触发停止条件
 - (1) 点击**Stop Tracing**按钮
 - (2) 连续跟踪到最大时间或者最大指定数据容量



The screenshot shows a dialog box for configuring tracing parameters. The settings are as follows:

Trace type	Continuous	Sampling interval	Every period of task	PrimaryTask	Trace No.	0
Samples per file	4500	Max data saving size (MB)	100			
Max sampling time	0 days	0 hours	1 minutes	When the upper limit reached	Stop tracing	
Save in	C:\¥Omron¥Data¥DataTrace¥		...	File prefix		

第七章 运动试运行功能

- 监视窗口
- 选择“View”下的“Watch Tab Page”



Name	Online value	Modify	Data type	AT	Data format
EXT1_1	False	TRUE FALSE	BOOL		Boolean
EXT1_2	False	TRUE FALSE	BOOL		Boolean
<i>Input Name...</i>					

第七章 运动试运行功能

编辑窗口内可编辑的参数

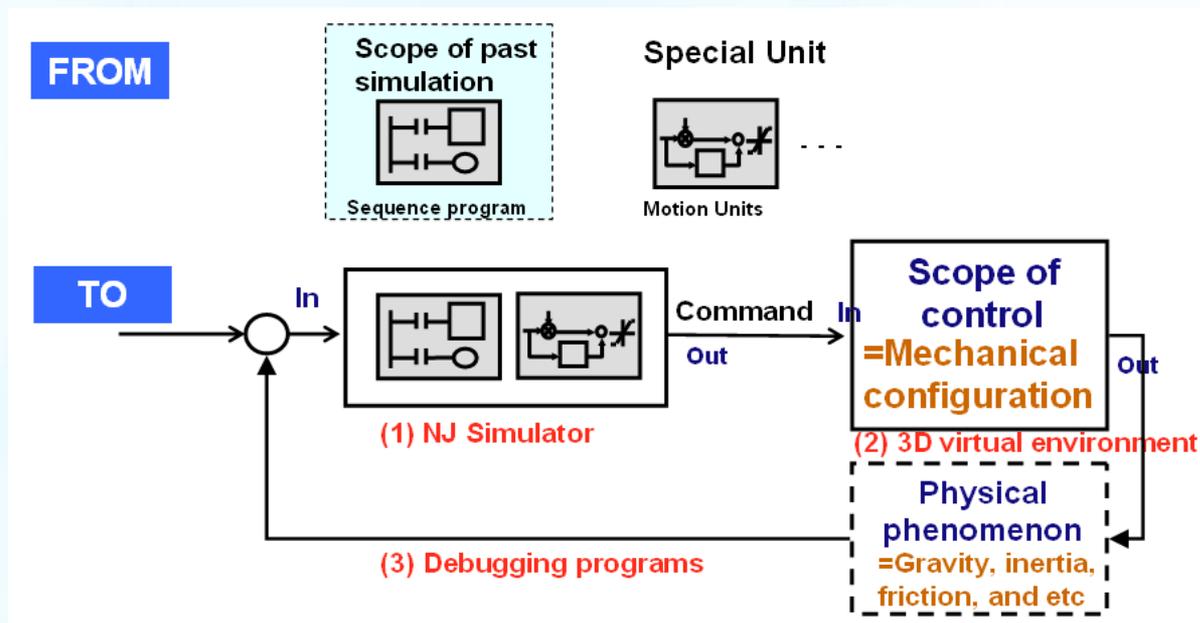
表7-1

Item	Description	Editing
Tab Page Name	The name of the Watch Tab Page is displayed.	Supported
Name	The variable is displayed.	Supported
Online value	In the monitor status, the present value in the Controller is displayed.	---
Modify	The new value is displayed.	Supported
Data type	The data type is displayed.	---
Data format	The display format (decimal, hexadecimal, etc.) of the present value or modify value is displayed.	Supported
Comment	The comment for the variable is displayed.	---
Group	The variable type (e.g., VAR_INPUT) is displayed.	---
Address	The allocated address is displayed for variables with AT specification.	---

备注：
在仿真时，外部输入信号的ON/OFF状态也可以在监视窗口里修改。

第七章 运动试运行功能

- 仿真功能
 - (1) 仿真现在适用于所有的序列和运动控制。
 - (2) 可以通过**2D**或者**3D**仿真抛物线来监视运动控制动作

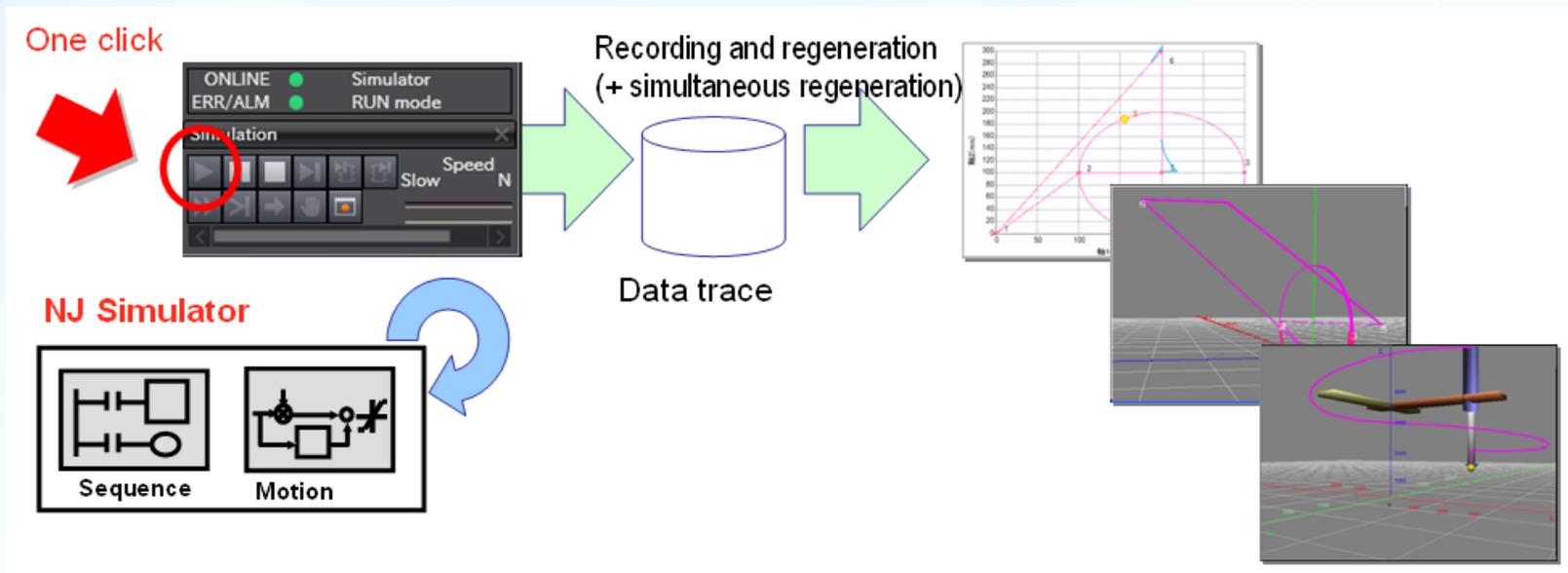


第七章 运动试运行功能

- 仿真功能主要具有以下四个具体价值。
 - 1 可以同时显示仿真序列和运动控制两方面
 - 2 可以比较运动控制设定值和实际值
 - 3 通过调试功能，当用户想看到某些值的时候可以立刻快速停止工作
 - 4 可以仿真时间

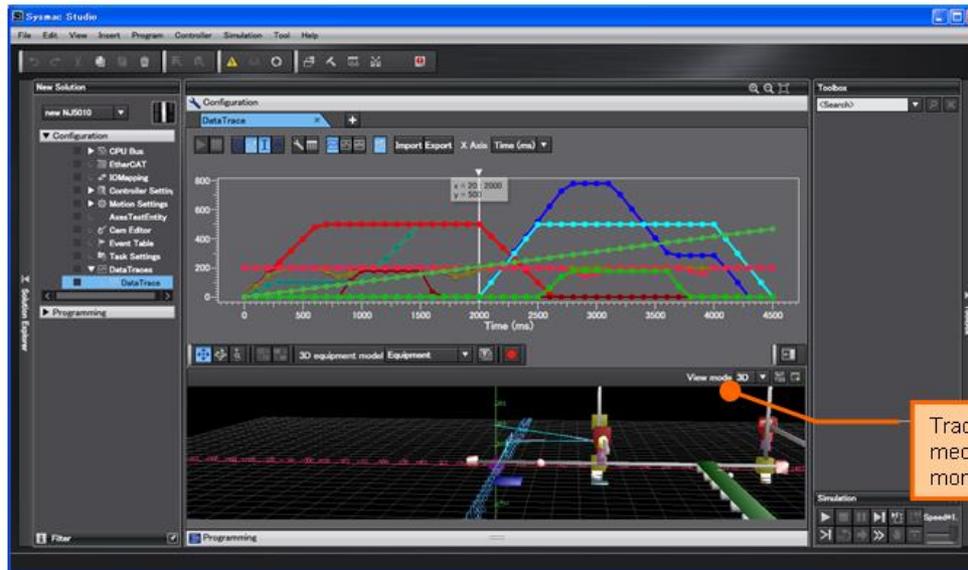
第七章 运动试运行功能

- 在使用” Click”后，就可以对序列和运动进行仿真。
- 仿真的结果可以被多种显示模式来监控。



第七章 运动试运行功能

- 同步显示仿真序列和运动控制
- 真过程中运动控制值，实际值，和仿真序列所形成的图形是可以同时显示在一个屏幕上，即可方便调试



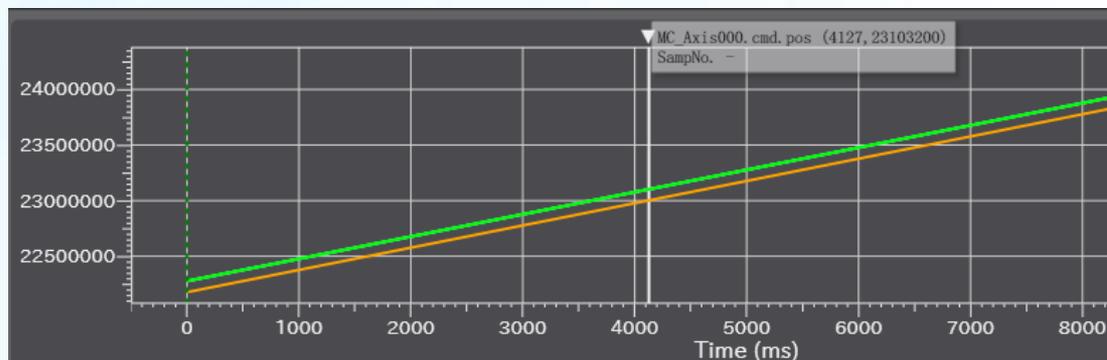
Trace data and virtual mechanical operation can be monitored at the same time.

第七章 运动试运行功能

- 可以比较运动控制设定和实际值
- 运动状态包括伺服的反馈值都能从监视对象中选择到

Visible		Name	X Offset	Y Offset	Y Axis Min.	Y Axis Max.
<input checked="" type="checkbox"/>		MC_Axis002.Act.Pos	0	0	0	0
<input checked="" type="checkbox"/>		MC_Axis003.act.pos	0	0	0	0
<input checked="" type="checkbox"/>		MC_Axis004.act.pos	0	0	0	0
<input checked="" type="checkbox"/>		MC_Axis002.cmd.pos	0	0	0	0

+ 



第七章 运动试运行功能

- 调试功能
- 打断（Break）
- 单步动作(Step exection)

第七章 运动试运行功能

- 打断/打断点（Break, Breakpoint）
- 设置打断点的目的是禁止该步程序执行

```
1  varA := 13;  
2  varB := 49;  
3  IF varA < DINT#1 THEN  
4      varA := DINT#100;  
5  END_IF;  
6  IF varA > DINT#12 THEN  
7      varA := DINT#200;  
8  END_IF;  
9  FOR varB := 0 TO 2 BY 1 DO  
10     varA := varB;  
11 END_FOR;
```

第七章 运动试运行功能

- 1) 在仿真已经运行的情况下将仿真状态切换选择到“Pause”
- (2) 选中需要打断的程序前，再选择“Simulation”下的“Set/Clear Breakpoint”



- 单步
- NJ系列的程序可以使用单步来进行调试，实现更好的观察效果

(1) 选择“Simulation”下的“Step exeution”

(2) 选择对应标记



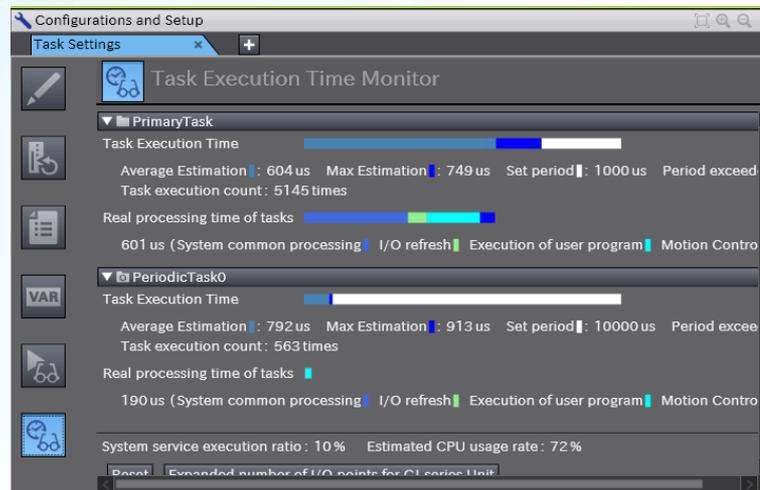
Name	Online value
Program0.t	2012-01-31-09:51:34.81
<i>Input Name...</i>	

第七章 运动试运行功能

- 动作时间的仿真
- 仿真时可以计算得到一个近似每个任务开始到结束动作时间的值。
- 根据计算机配置来近似计算周期时间所花费时间会有所不同。
- 推算出来的时间可以作为决定任务周期时间的一个判断条件。

第七章 运动试运行功能

- (1) 选择“Simulation”下的“Calibration”
- (2) 根据提示进行下一步动作，注意：需关闭所有内容
- (3) 重新打开软件并打开刚才的项目文件后。选择“Simulation”下的“Run in Execution Time Estimation Mode”
- 4) 进入“Task Settings”选择“Task Execution Time Monitor”



END

欧姆龙客户服务中心

WWW.fa.omron.com.cn

免费声讯：800-820-4535