

DENSO机械手

垂直多关节型
V* 系列

水平多关节型
H* 系列

直角坐标型
XYC 系列

内置型
XR 系列

调试启动手册

Copyright © 2008-2011 DENSO WAVE INCORPORATED
All rights reserved.

本使用说明书的著作权属于 DENSO WAVE INCORPORATED。

本说明书所登载的公司名称和产品，均属各公司的商标或注册商标。

规格如有变更，恕不另行通知。

用于本说明书中的图片与实际操作时显示的画面会有所不同。

前言

承蒙惠购DENSO机械手，深表谢意。

该产品是汇集了本公司先进技术的高速度、高精度、高功能的 "组装用机械手"。

在使用之前，请详细阅读理解本说明书，以便安全高效地使用机械手。

要求

在使用之前，请务必阅读 "安全注意事项"，以便能够正确安全地使用DENSO机械手。

本书的构成

本书的构成如下所示。

第1部 安装准备

就机械手系统、RC7M型控制器的概要、界面的概要、电缆类的连接方法及专用信号的接线进行说明。

第2部 机械手操作

就坐标系、多功能教导器的操作方法、教导方法进行说明。

第3部 简易编程

就程序的基础知识、多功能教导器和利用WINCAPSIII进行程序创建的方法及练习题进行说明。

第4部 程序的验证

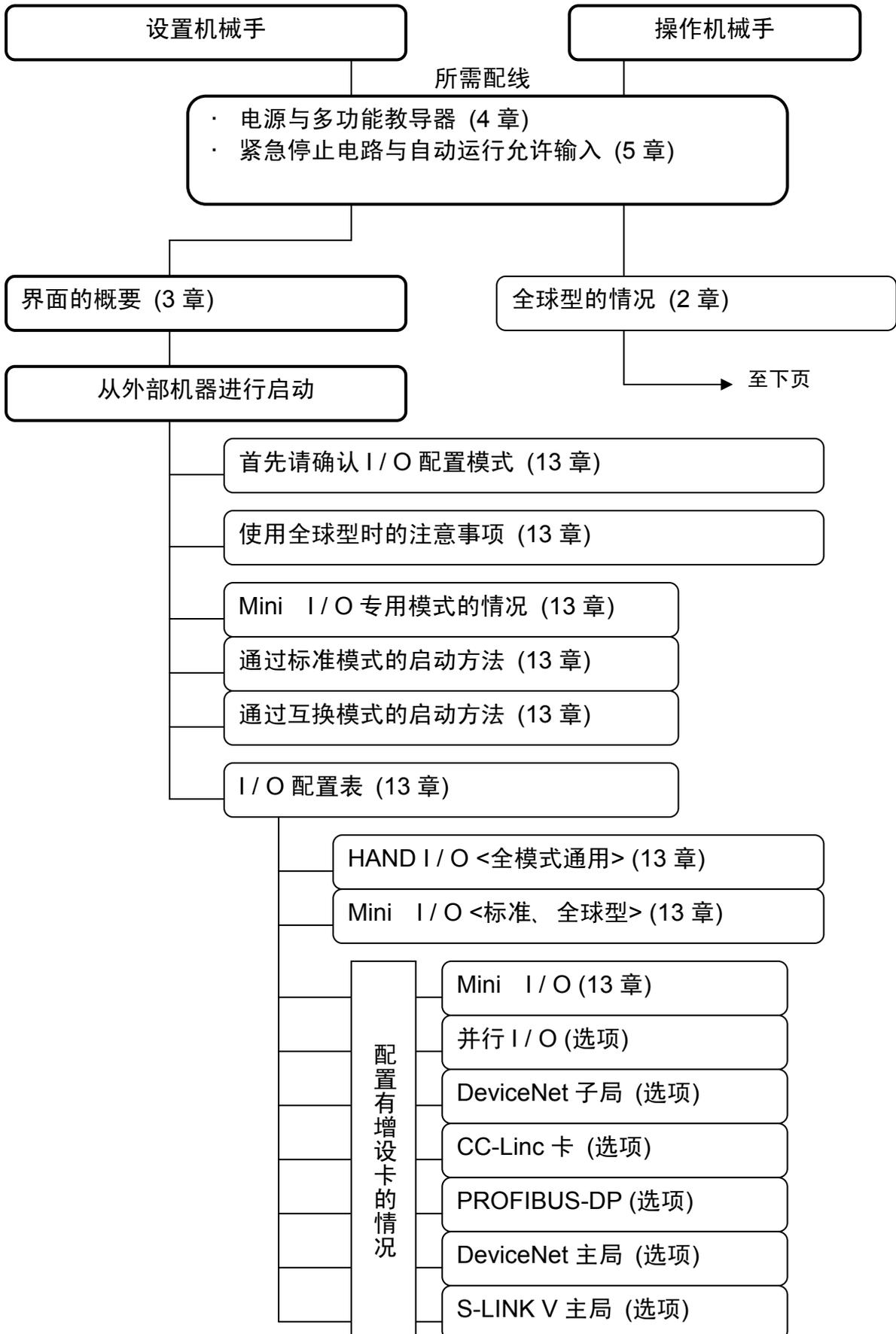
作为确认创建的程序的方法，就WINCAPSIII的仿真、多功能教导器的动作确认，通过外部机器的运行方法进行说明。同时，就IO或变量的监视方法进行说明。

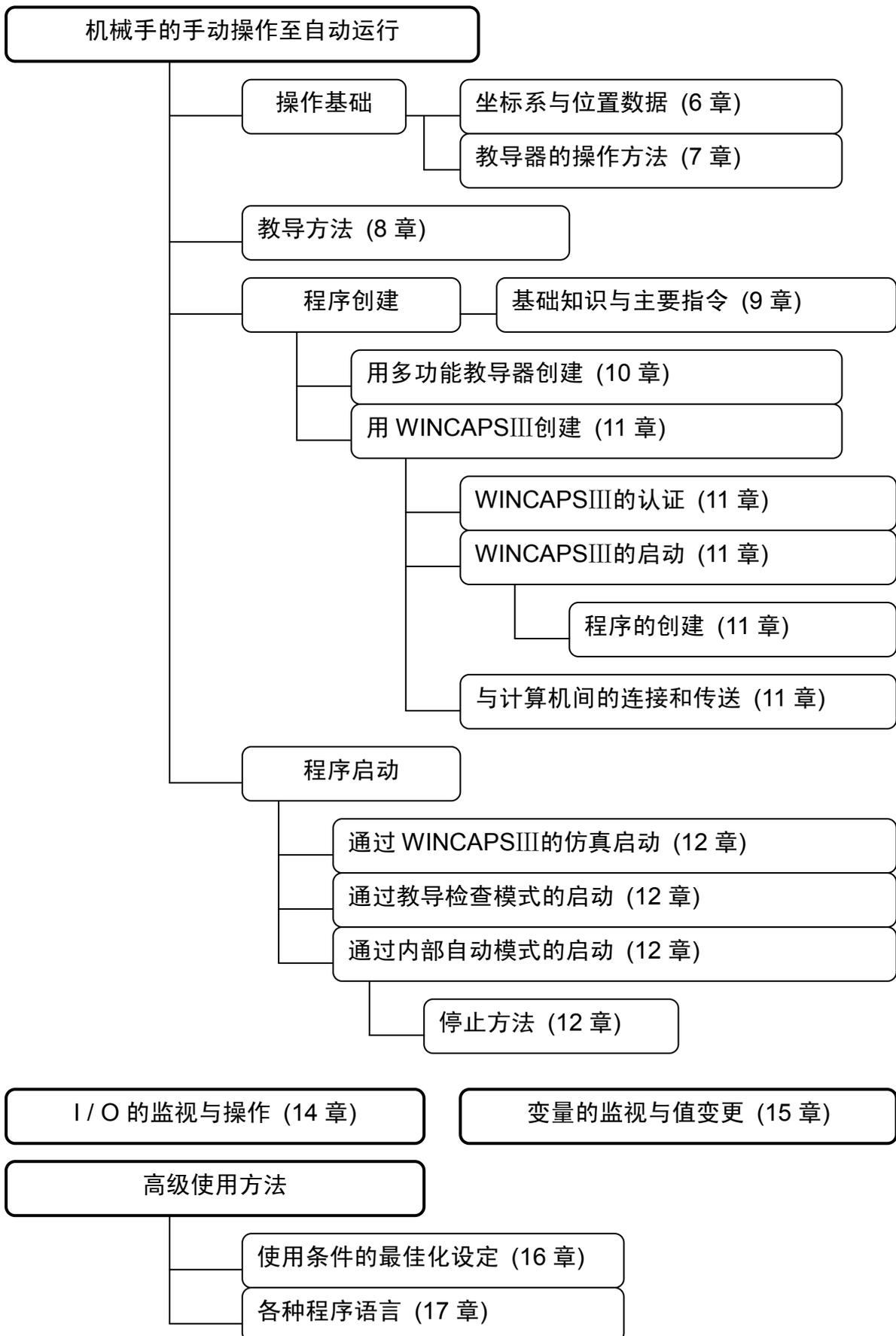
第5部 高级使用方法

提供使用条件的最佳方法和经常使用的程序命令等高级使用方法的讯息。

附录

调试启动手册 综合指南





目录

调试启动手册 综合指南

第 1 部 安装准备

第 1 章 机械手系统配置	
1.1 机械手系统配置设备	1-1
1.2 标准构件	1-2
1.3 选件品	1-3
第 2 章 RC7M 型控制器的概要	
2.1 控制器型号的确认方法	2-1
2.1.1 VS-***系列用的 RC7M 机械手控制器的型号	2-2
2.2 有关全球型控制器的注意事项	2-3
2.2.1 关于双重安全开关（允许开关）的功能	2-3
2.2.2 有关单一位置控制功能 (Single Point of control)	2-4
第 3 章 控制器的通信界面概要	
3.1 Mini I/O 输入输出信号的种类及其概要	3-1
3.1.1 Mini I/O 输入输出信号的种类（使用标准型控制器时）	3-1
3.1.2 Mini I/O 输入输出信号的种类（使用全球型控制器时）	3-2
3.2 I/O 增设卡的概要	3-3
3.2.1 I/O 增设卡的种类	3-3
3.3 I/O 增设卡的组合与配置模式	3-4
3.3.1 各配置模式 I/O 配置表	3-5
3.3.2 各配置模式的功能概要	3-5
3.4 选择互换、标准、全通用模式时的 Mini I/O 的功能	3-6
3.5 界面的必要设定	3-6
3.5.1 I/O 配置模式的设定	3-6
3.5.2 I/O 用电源 (+24V DC) 的设定	3-6
3.6 I/O 配置模式的设定方法	3-7
3.6.1 通过多功能教导器进行设定的方法	3-7
3.6.2 通过 WINCAPSIII 的设定方法	3-7
3.7 Mini I/O 用电源的设定方法	3-10
3.8 并行 I/O 用电源的设定方法	3-11
3.9 I/O 端口管理表与配置	3-12
第 4 章 电缆的连接方法	
4.1 控制器与电源、机械手本体间的电缆连接	4-1
4.2 教导器的电缆连接	4-1
4.3 断路器的选定（推荐产品）	4-2
4.4 一次侧电源的配线方法	4-3

第 5 章 专用输入信号的接线	
5.1 调整机械手时必要的接线	5-1
5.1.1 紧急停止电路的构成（使用标准型控制器时）	5-1
5.1.2 紧急停止电路的构成（全球型控制器）	5-1
5.2 电机变为 ON 状态所需的接线	5-2
5.2.1 机能	5-2
5.2.2 标准型控制器	5-2
5.2.3 全球型控制器	5-2
5.3 自动运行所需要的接线	5-2
5.3.1 机能	5-2
5.3.2 标准型控制器	5-2
5.3.3 全球型控制器	5-2

第 2 部 机械手操作

第 6 章 坐标系	
6.1 4 轴机械手坐标系的种类	6-1
6.2 4 轴机械手基准坐标系	6-1
6.3 4 轴机械手工件坐标系	6-1
6.4 4 轴机械手工件坐标系	6-2
6.5 4 轴机械手工具坐标系的优点	6-2
6.6 在 4 轴机械手上使用的位置数据	6-3
6.6.1 有关 4 轴的形态	6-3
6.7 6 轴机械手坐标系的种类	6-4
6.8 6 轴机械手基准坐标系	6-4
6.9 6 轴机械手工件坐标系	6-4
6.10 6 轴机械手工件坐标系	6-5
6.11 6 轴机械手工具坐标系的特性	6-6
6.12 在 6 轴机械手上使用的位置数据	6-7
6.12.1 有关 6 轴的形态	6-8
第 7 章 教导之前的准备	
7.1 多功能教导器的使用	7-1
7.1.1 多功能教导器的手持方法与双重安全开关	7-1
7.1.2 多功能教导器的按键开关的名称	7-2
7.2 动作模式的种类	7-4
7.2.1 手动模式	7-4
7.2.2 教导检查模式	7-4
7.2.3 自动模式	7-4
7.3 动作模式的切换方法	7-5
7.3.1 动作模式的切换方法	7-5
7.3.2 动作模式和自动允许输入信号的相关性	7-5
7.4 手动模式的种类	7-6
7.4.1 各轴、X-Y、不同工具模式下的机械手操作	7-6
7.4.2 手动模式的切换方法	7-7
7.5 以手动运行模式运行机械手	7-9

第 8 章 教导

8.1 教导定义	8-1
8.2 可以进行教导的全局变量	8-1
8.3 对位置变量的教导方法	8-2
8.4 直接移向需要教导的位置变量的方法	8-7
8.5 向包含接近长度在内的目标位置移动 (Ver.2.61 之后版本)	8-8

第 3 部 简易编程

第 9 章 程序创建的基础知识

9.1 PAC 语言	9-1
9.2 语句与行	9-1
9.3 姓名	9-1
9.4 可以登录的最大程序数	9-2
9.5 程序构成的概要	9-2
9.6 程序的主要指令	9-3
9.6.1 程序示例	9-3
9.6.2 关于用指令格式处理的特殊符号	9-3
9.6.3 程序名的定义 (PROGRAM 指令)	9-4
9.6.4 获取臂控制权 (TAKARM 指令)	9-4
9.6.5 程序的停止 (END 指令)	9-4
9.6.6 臂移动速度的指定 (SPEED 指令)	9-4
9.6.7 备注的叙述 (REM 指令)	9-4
9.6.8 向指定坐标的移动 (MOVE 指令)	9-5
9.7 向 Z 轴方向移动 (APPROACH, DEPART)	9-8
9.7.1 向夹治具方向的接近运行 (APPROACH 指令)	9-8
9.7.2 向夹治具方向的回避移动 (DEPART 指令)	9-9
9.8 变量的概要与性质	9-10
9.8.1 全局变量	9-11
9.8.2 局部 (Local) 变量	9-12
9.9 可以从外部机器启动的程序名	9-13

第 10 章 程序的创建 (多功能教导器篇)

10.1 创建的示例程序	10-1
10.2 程序的创建	10-2
10.2.1 新建程序名称得输入	10-2
10.2.2 程序的输入	10-3
10.2.3 程序的编译	10-7
10.2.4 程序的加载	10-9

第 11 章 程序的创建 (WINCAPSIII篇)

11.1 创建程序的准备	11-1
11.1.1 WINCAPSIII的种类	11-1
11.1.2 CD 外观 (CD 标签)	11-1
11.1.3 与产品版同箱装入的许可证 (用户 ID)	11-2
11.1.4 在画面上确认 WINCAPSIII的产品种类	11-2
11.1.5 版本升级时的注意事项	11-3
11.1.6 用户 ID 的输入方法	11-3

11.2 创建的示例程序	11-4
11.3 创建程序	11-5
11.3.1 WINCAPSIII的启动.....	11-5
11.3.2 新建项目	11-5
11.3.3 程序的创建	11-8
11.3.4 程序的输入与保存.....	11-10
11.3.5 程序的编译	11-11
11.4 WINCAPSIII与控制器的通信电缆的连接方法.....	11-13
11.4.1 RS-232C 连接.....	11-13
11.4.2 EtherNet 连接	11-13
11.5 与机械手控制器的通信连接准备.....	11-14
11.5.1 RS-232C 连接时.....	11-14
11.5.2 EtherNet 连接时.....	11-19
11.6 由 WINCAPSIII进行的数据传输.....	11-26
11.6.1 控制器一侧的准备（数据传输时的注意事项）	11-26
11.6.2 将程序数据发送至机械手控制器.....	11-27

第 4 部 程序的验证

第 12 章 程序启动

12.1 利用 WINCAPSIII所进行的仿真运行	12-1
12.1.1 打开臂 3D 视图	12-1
12.1.2 将 WINCAPSIII切换至监视器模式	12-1
12.1.3 置于机械锁定状态	12-2
12.1.4 进行程序启动	12-2
12.2 在教导检查模式下进行的程序启动	12-3
12.2.1 置于教导检查模式	12-3
12.2.2 选择程序	12-4
12.2.3 单步运行检查启动	12-4
12.2.4 循环检查启动	12-6
12.3 在内部自动模式下进行的程序启动	12-8
12.3.1 置于自动模式	12-8
12.3.2 选择程序	12-8
12.3.3 步骤启动	12-9
12.3.4 循环启动	12-10
12.3.5 循环启动	12-11
12.4 机械手的停止操作	12-12
12.4.1 循环停止 [F3]	12-12
12.4.2 步骤停止 [F2]	12-12
12.4.3 瞬时停止 [F1], [STOP]	12-12
12.4.4 紧急停止（机械手停止）	12-13

第 13 章 从外围机器启动机械手的方法	
13.1 I/O 配置模式的确认方法	13-1
13.2 使用全球型控制器时的注意事项	13-1
13.3 通过 Mini I/O 专用模式的启动方法	13-2
13.3.1 Mini I/O 指令专用输入信号的种类与功能	13-2
13.3.2 Mini I/O 指令处理方法	13-3
13.3.3 Mini I/O 专用输出信号的种类与功能	13-5
13.4 用标准模式进行启动的方法	13-6
13.4.1 标准 I/O 指令专用输入信号的种类与功能	13-6
13.4.2 标准 I/O 指令处理方法	13-7
13.4.3 标准 I/O 专用输出信号的种类与功能	13-9
13.5 用标准模式进行启动的方法	13-10
13.5.1 互换模式专用输出信号的种类与功能	13-10
13.5.2 互换模式的处理方法	13-11
13.5.3 互换 I/O 专用输出信号的种类与功能	13-13
13.6 I/O 配置表	13-14
13.6.1 HAND I/O (CN9): 全部模式通用的配置	13-14
13.6.2 Mini I/O (标准型控制器 CN5): 专用配置	13-15
13.6.3 Mini I/O (全球型控制器 CN5): 专用配置	13-16
13.6.4 Mini I/O (标准型控制器 CN5): 互换 / 标准 / 全通用配置	13-17
13.6.5 Mini I/O (全球型控制器 CN5): 互换 / 标准 / 全通用配置	13-18
第 14 章 I/O 的监视与操作	
14.1 由多功能教导器进行的操作	14-1
14.1.1 I/O 的监视	14-1
14.1.2 模拟 I/O 的操作方法	14-2
14.2 利用 WINCAPSI 的操作	14-4
14.2.1 I/O 的监视	14-4
14.2.2 使用模拟 I/O	14-5
第 15 章 变量的监视和值的变更	
15.1 由多功能教导器进行的操作	15-1
15.1.1 全局变量的监视和值的变更	15-1
15.1.2 局部 (Local) 变量的监视与值的变更	15-2
15.1.3 变更全局变量的使用个数	15-5
15.2 通过 WINCAPSI 的操作	15-7
15.2.1 全局变量的监视和值的变更	15-7
15.2.2 局部 (Local) 变量的监视与值的变更	15-8
15.2.3 变量使用个数的变更	15-9

第 5 部 高级使用方法

第 16 章 使用条件的最佳设定

16.1 机械手安装条件（地面设置、悬挂）的设定方法 [6 轴机械手]	16-1
16.1.1 机械手安装条件（地面设置、悬挂）设定的目的	16-1
16.1.2 由多功能教导器进行的设定	16-1
16.1.3 利用 WINCAPSIII 进行的设定	16-2
16.2 最佳可搬运质量设定功能	16-3
16.2.1 模式 0	16-3
16.2.2 模式 1	16-3
16.2.3 模式 2	16-4
16.2.4 模式 3	16-4
16.3 最佳可搬运质量功能的设定方法	16-5
16.3.1 由多功能教导器进行的设定	16-5
16.3.2 利用 WINCAPSIII 的设定	16-6
16.4 最佳可搬运质量初始化设定的设定方法 [V1.4 以后版本]	16-7
16.4.1 由多功能教导器进行的设定	16-7
16.4.2 利用 WINCAPSIII 的设定	16-8

第 17 章 程序语言（机械手控制语句）

17.1 机械手动作的种类	17-1
17.1.1 绝对动作与相对动作	17-1
17.1.2 插补控制	17-1
17.2 机械手控制指令	17-3
17.2.1 DRIVEA	17-3
17.2.2 DRIVE	17-4
17.2.3 DRAW	17-5
17.3 练习题 机械手控制指令	17-6

第 18 章 编程语言（流程控制语句）

18.1 流程控制语句的种类	18-1
18.2 调出的指令	18-2
18.2.1 CALL	18-2
18.2.2 GOSUB	18-3
18.3 无条件分支指令	18-4
18.3.1 GOTO	18-4
18.4 条件分支指令	18-5
18.4.1 IF ~ ENDIF	18-5
18.4.2 SELECT CASE	18-6
18.5 反复指令	18-7
18.5.1 FOR ~ NEXT	18-7
18.5.2 DO ~ LOOP	18-8
18.6 练习题 流程控制指令	18-10

第 19 章 程序语言（输入输出语句）	
19.1 时间控制指令	19-1
19.1.1 DELAY	19-1
19.1.2 WAIT	19-1
19.2 I/O 端口控制	19-2
19.2.1 SET	19-2
19.2.2 RESET	19-2
19.3 练习题 I/O 端口控制指令	19-3
第 20 章 程序语言（共用程序库 (Library) 篇）	
20.1 共用程序库 (Library) 程序的运用	20-1
20.1.1 共用程序库 (Library) 的概念	20-1
20.1.2 程序库 (Program Bank) 功能	20-1
20.1.3 共用程序库 (Library) 的分类	20-1
20.1.4 共用程序库 (Library) 的追加（引入）方法	20-2
20.2 码垛共用程序库 (Library) 的运用	20-4
20.2.1 码垛堆积的说明	20-4
20.2.2 简易码垛堆积共用程序库 (Library)	20-4
20.2.3 码垛堆积的参数	20-5
20.2.4 简易码垛堆积程序 "PRO1"	20-7

附录

练习题答案示例

按功能单位排列的指令一览

多功能教导器的功能菜单

示例程序集

用语

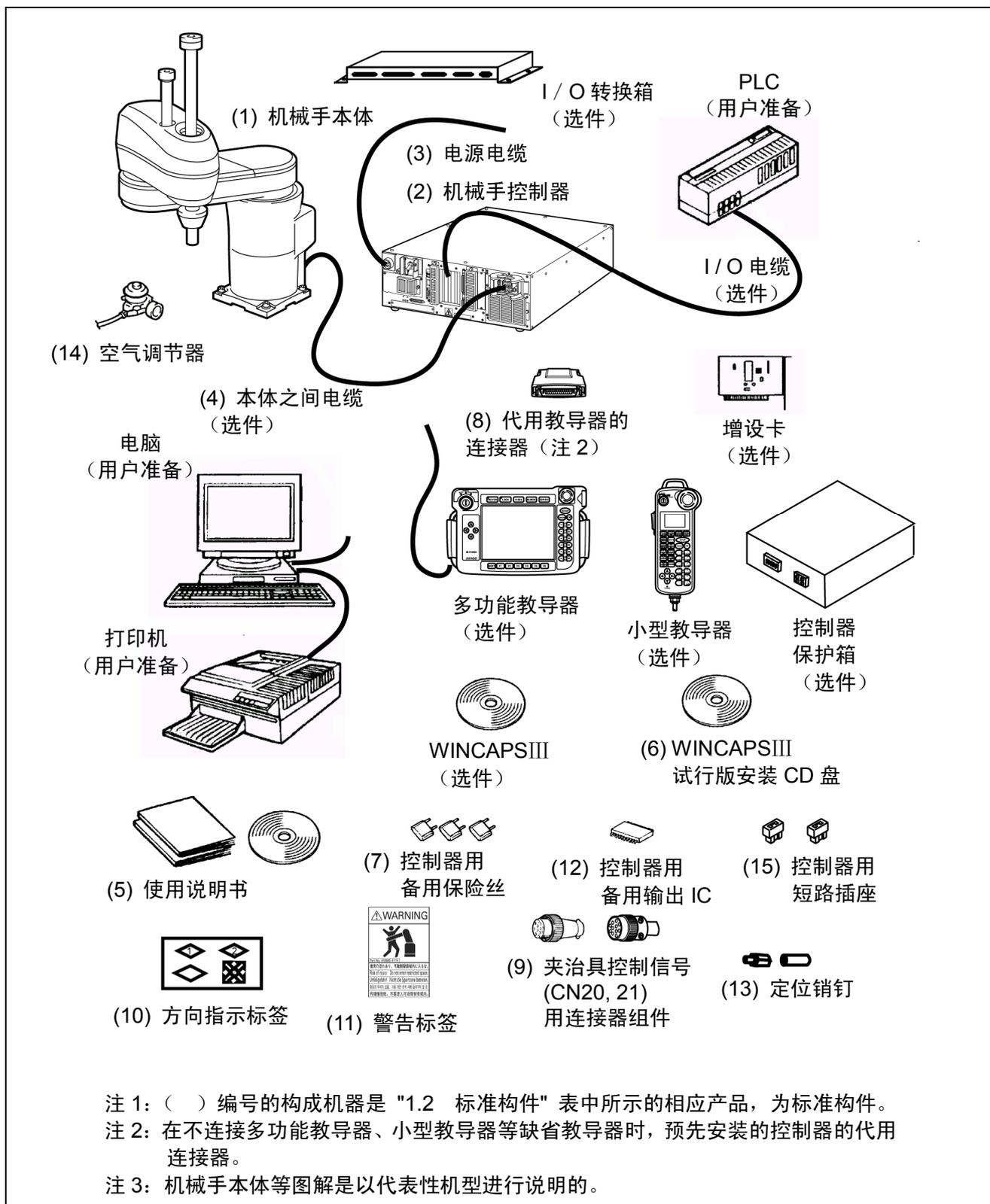
第1部

安装准备

第 1 章 机械手系统配置

1.1 机械手系统配置设备

机械手系统的整体构成如下图所示。



机械手系统配置设备

1.2 标准构件

贵公司所购买的产品，由下表中所列的内容构成。

标准构件

No	品名	数量	有无构成							
			HS系列	HM系列	XYC系列	VP系列	VS系列	VH系列	XR系列	VS-***系列
①	机械手本体	1台	○	○	○	○	○	○	○	○
②	机械手控制器	1台	○	○	○	○	○	○	○	○
③	电源电缆 (5m)	1根	○	○	○	○	○	○	○	○
④	本体之间电缆 (注1) (选件品)	1套	○	○	○	○	○	○	○	○
⑤	使用说明书 (手册文件包CD与注意事项说明书)	1套	○	○	○	○	○	○	○	○
⑥	WINCAPSIII试行版安装CD	1张	○	○	○	○	○	○	○	○
⑦	机械手控制器用备用保险丝 (1.3A × 2个, 3.2A × 1个)	3个	○	○	○	○	○	○	○	○
⑧	代用教导器的连接器 (UL规格除外)	1个	○	○	○	○	○	○	○	○
⑨	夹治具控制信号用的连接器装置 (CN20、CN21用)	1套	○	○	○	○	○	○	○	选件
⑩	方向指示标签 (注2)	1张	○	○	○	○	○	○	○	○
⑪	警告标签 (注3)	1张	○	○	○	○	○	○	○	○
⑫	机械手控制器备用输出用IC	1个	○	○	○	○	○	○	○	○
⑬	定位销钉 (带螺丝帽的定位销钉&菱形销钉)	1套	○	○	×	○	○	×	×	○
⑭	空压调节器 (注4)	1个	×	○	○	×	×	×	×	×
⑮	机械手控制器用的短路插座	2个	○	○	○	○	○	○	○	○

注1: 本体之间电缆 (电机、编码器电缆) 请与机械手一起从下表中选择订购。

本体之间电缆20m (标准/防溅) 不能在带有附加轴选项的控制器及UL规格的机械手上使用。

本体之间电缆的弯曲半径R (内径) 要在200mm以上。过度的弯曲会造成断线。

不包括XYC系列、VS-***系列的机械手的情况

品名	编号	备注
标准电缆 2m	410141-4400	用于标准规格 机械手
标准电缆 4m	410141-3611	
标准电缆 6m	410141-3621	
标准电缆 12m	410141-3631	
标准电缆 20m	410141-4440	
防溅电缆 2m	410141-4420	用于防尘、防溅规格 的机械手
防溅电缆 4m	410141-3681	
防溅电缆 6m	410141-3691	
防溅电缆 12m	410141-3701	
防溅电缆 20m	410141-4460	

XYC系列的情况

品名	编号
标准电缆 4m	410149-0960
标准电缆 6m	410149-0970

VS-***系列的情况 (注5)

品名	编号
本体之间电缆RC7-B 2m	410141-4560
本体之间电缆RC7-B 4m	410141-4570
本体之间电缆RC7-B 6m	410141-4580
本体之间电缆RC7-B 12m	410141-4590
本体之间电缆RC7-B 20m	410141-4600

注2: 在安装结束之后, 请将方向指示标签贴在本体上容易看到的位置。

注3: 请将警告标签贴在机械手的安全防护栏等容易看到的位置。根据需要, 请准备粘贴用的粘贴板。

注4: 防尘、防溅的机械手因没有Z轴平衡气缸, 所以未配置空压调节器。

注5: 当VS-***系列本体之间电缆的弯曲R (内径) 在电缆固定状态下超过33.8mm时, 请确保在可动状态下电缆达到225mm以上。过度弯曲会导致断线。此外, 本体之间电缆RC7-B 20m不能用于带附加轴选件的控制器。

1.3 选件品

本公司准备了下表中所列出的选件，请根据需要订购。

选件品 (1)

分类	No.	品名		备注	品号
I/O 电缆	1	标准I/O电缆组件		(8m)包括No.1-1和No.1-2	410149-0940
				(15m)包括No.1-1和No.1-2	410149-0950
	1-1	Mini I/O电缆 (68销钉)		(8m)	410141-2700
				(15m)	410141-2710
	1-2	HAND I/O电缆 (20销钉)		(8m)	410141-1740
(15m)				410141-1750	
2	增设并行I/O用电缆 (96销钉)		(8m)	410141-3050	
			(15m)	410141-3060	
3	安全I/O电缆 (36销钉) (全球型控制器用)		(8m)	410141-3580	
			(15m)	410141-3590	
操作用机器	4	多功能教导器		带有 (4m) 电缆	410100-1572
				带有 (8m) 电缆	410100-1582
				带有 (12m) 电缆	410100-1592
	5	小型教导器组件 (包括电缆、WINCAPSIII Light)		(4m) 日语标注	410109-0392
				英语标注	410109-0402
				(8m) 日语标注	410109-0412
英语标注	410109-0422				
(12m) 日语标注	410109-0432				
英语标注	410109-0442				
6	教导器延长电缆		(4m) TP、MP用	410141-3710	
			(8m) TP、MP用	410141-3720	
编程支持工具	7	WINCAPSIII		CD-ROM版 支持语言 (日语、英语、德语、 韩语、中文)	410090-0980
RC7M 用 增设卡	8	并行I/O卡	内置于控制器之后 出厂	NPN型I/O	410010-3320
				PNP型I/O	410010-3330
			卡单件出厂 (补充用)	NPN型I/O	410010-3340
				PNP型I/O	410010-3350
	9	DeviceNet 卡	内置于控制器之后 出厂	子局卡	410010-3370
				主局卡	410010-3380
				主局卡&子局卡	410010-3390
			卡单件出厂 (补充用)	子局卡	410010-3400
				主局卡	410010-3410
				主局卡&子局卡	410010-3480
	10	CC-Link卡		内置于控制器之后出厂	410010-3430
卡单件出厂 (补充用)				410010-3440	
11	传送跟踪卡		内置于控制器之后出厂	410010-3460	
			卡单件出厂 (补充用)	410010-3470	

选件品 (2)

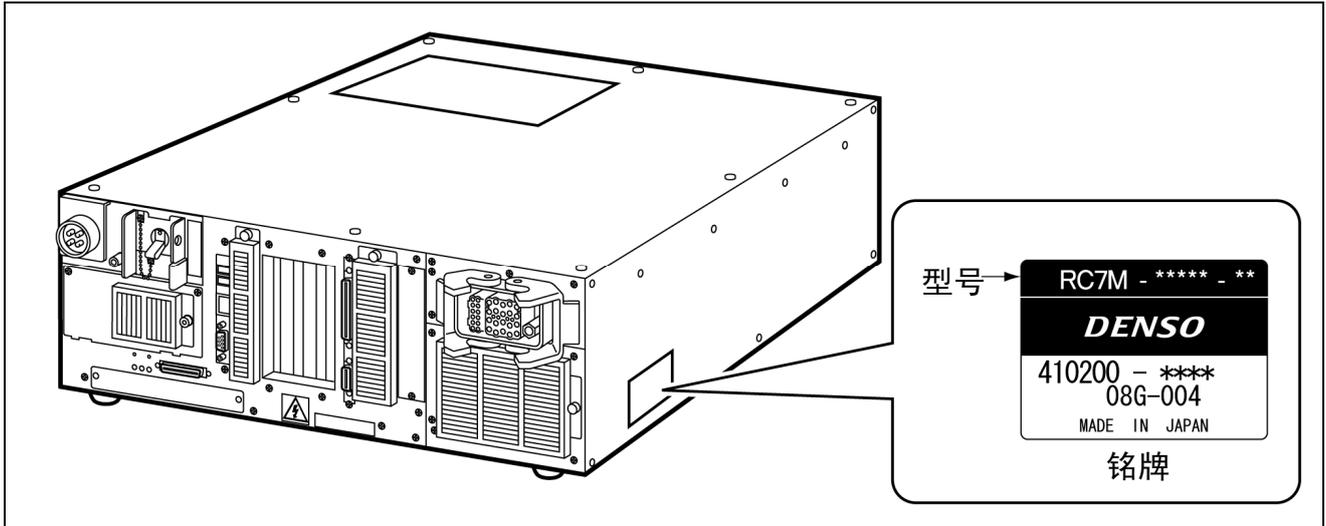
分类	No.	品名	备注	品号
选件功能 (需要用户自备的增设卡等)	12	RS232C 增设功能 卡制造商 / 型号: (株) CONTEC / COM-2P (PCI) H	控制器出厂时追加功能	410006-0260
			单件购进卡时追加功能	410006-0270
	13	S-LINK V 功能 卡制造商 / 型号: SUNX (株) / SL-VPCI	控制器出厂时追加功能	410006-0280
			单件购进卡时追加功能	410006-0290
	14	PROFIBUS-DP 子局功能 卡制造商 / 型号: Hilscher GmbH / CIF50-DPS \ DENSO	控制器出厂时追加功能	410006-0300
单件购进卡时追加功能			410006-0310	
15	EtherNet/IP 功能 卡制造商 / 型号: Hilscher GmbH / CIFX50-RE \ DENSO	控制器出厂时追加功能	410006-0800	
		单件购进板时追加功能	410006-0810	
	16	增设存储器	仅在控制器出厂时 可以进行 3.25MB→5.5MB 的增设 (只在程序空间增加)	410006-0320
选件箱	17	控制器保护箱		410181-0091
	18	RC5 I/O 转换箱	用于与 RC5 型控制器的互换	410181-0100
使用说明书 (CD版)	19	手册文件包 CD	(与机械手标准同捆)	410002-2661
与本体相关的 内容	22-a	法兰组件 (HS 用)	HS-G 用	410329-0060
			10kg 可搬运	410329-0070
	22-b	法兰组件 (HM 用)	20kg 可搬运	410329-0080
夹治具配管、 配线用套件	22-g	阀门组件 (XR-G 专用)	单件出厂 (补充用) 四通阀	410640-0230
	23-g	阀门组件 (XR-G 专用)	内置于机械手之后出厂 四通阀	410640-0330
	24-g	夹治具控制用电缆套件 (XR-G 专用)		410879-0470
	25-g	夹治具控制用电缆 (XR-G 专用)	2m	410870-3350
任选支架	26	满程支架 (XR-G 专用)		411759-0010
	27	半程支架 (XR-G 专用)		411759-0020

第 2 章 RC7M 型控制器的概要

RC7M 型控制器的细部规格会因所配置的机械手型号而有所不同。在本章中，就 RC7M 型控制器的概要进行说明。

2.1 控制器型号确认方法

控制器型号记载于控制器背面的铭牌上。在下表中列出了控制器型号的确认方法。



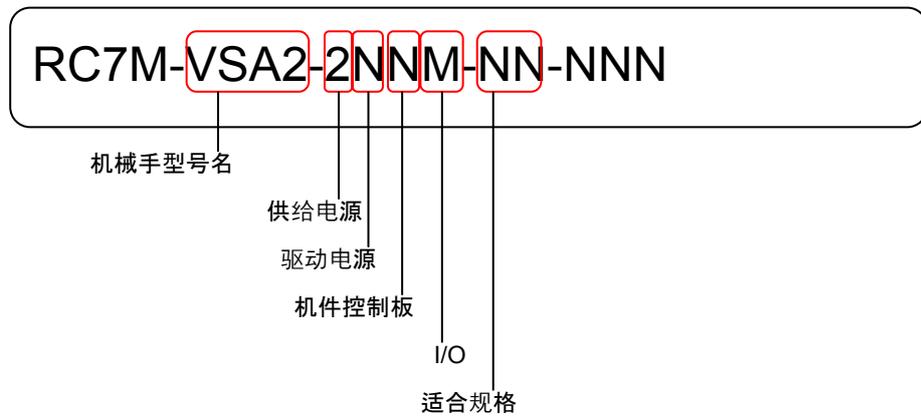
RC7M控制器型号的确认方法（G型机械手）

RC7M - <u>VSG</u> <u>6</u> <u>B</u> <u>A</u> _ - <u>B</u> <u>P</u>			
记号	例	意思	分类
(a)	VSG	机械手型号名称	VMG: VM-G用, VSG: VS-G用, VPG: VP-G用, HMG: HM-G用, HSG: HS-G用, XYCG: XYC-G用, XRG: XR-G用
(b)	6	轴数	4: 4轴, 5 / 6: 5轴或6轴, 6: 6轴
(c)	B	设计记号 1	A: 编码器A用 B: 编码器B用 C: 编码器C用
(d)	A	设计记号 2	A: 24V制动器
(e)		设计记号 3	空白: 200VAC电源 A: 100VAC电源
(f)	B	控制器的类型 (注)	空白: 标准型 (适用于日本的标准) B: 全球型 (带安全板) C: 全球型 (带安全箱) D: 全球型 UL规格 (带安全板) E: 全球型 UL规格 (带安全箱)
(g)	P	I/O的类型	空白或N: NPN型I/O (适用于日本的标准) P: PNP型I/O

注: 关于全球型的控制器, 请参照 "2.2 有关全球型控制器的注意事项"。

2.1.1 VS-***系列用的 RC7M 机械手控制器的型号

机械手控制器的型号如下所示。



机械手型号名

VSA2	VS-068/VS-087
VSA1	VS-050/VS-060

供给电源

2	200V
---	------

驱动电源

N	标准
---	----

机件控制板

N	标准
---	----

I/O

M	共负极 (NPN)
P	共正极 (PNP)

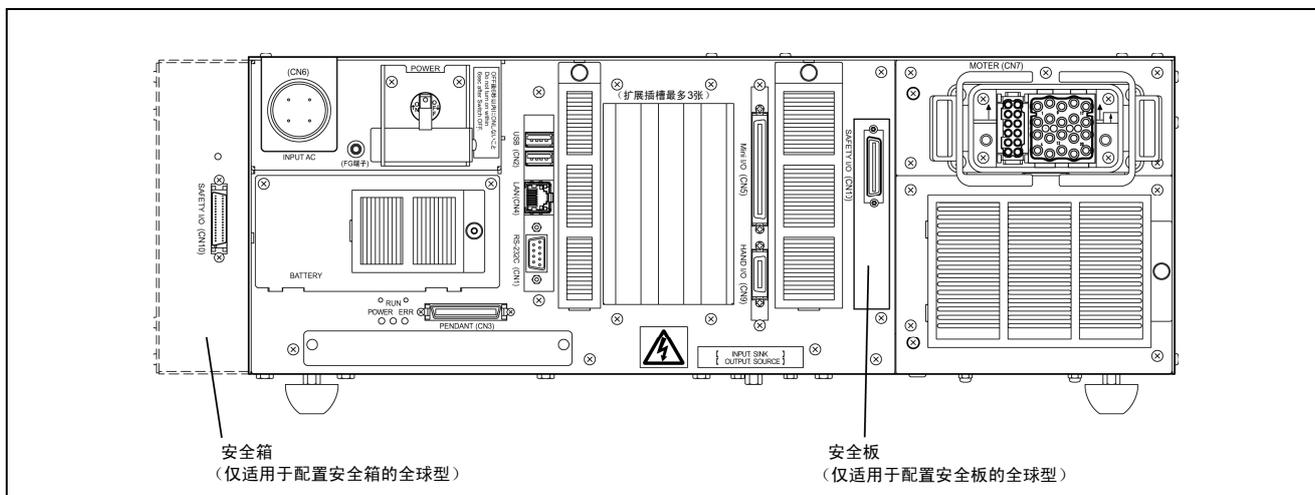
适合规格

NN	-
NB	CE (安全范畴: 3、带安全板) 注1
NC	CE (安全范畴: 4、带安全箱) 注1
UB	CE、UL (安全范畴: 3、带安全板) 注1
UC	CE、UL (安全范畴: 4、带安全箱) 注1

注1: 全球型控制器

2.2 有关全球型控制器的注意事项

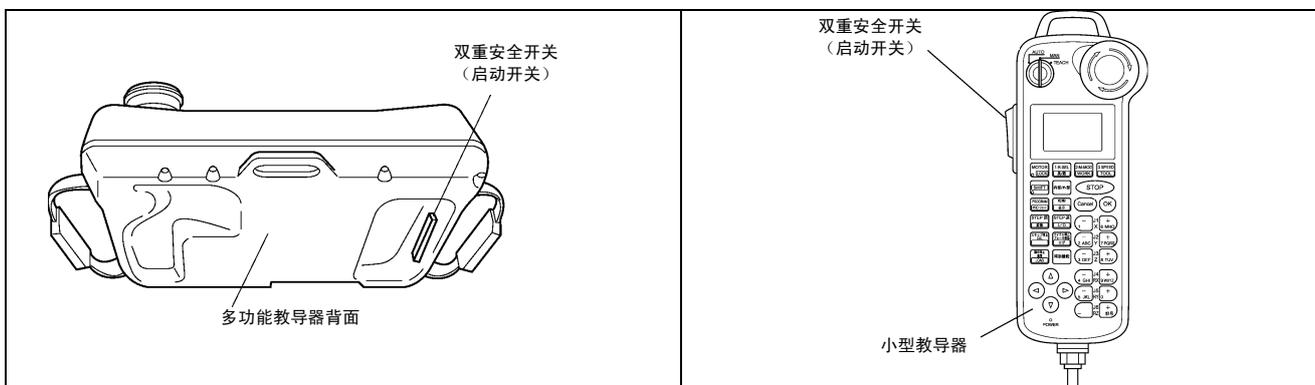
在全球型控制器中，如下图所示，有安全板或带安全箱的规格。在此对其特有的与标准控制器不同的功能进行说明。



2.2.1 关于双重安全开关（允许开关）的功能

“控制器是全球型”的情况，操作用教导器的双重安全开关的功能与使用说明书中所记述的内容有部分差异。全球型控制器的情况，使用时请阅读说明书中的相应部分。

(1) 需要的选件机器： 多功能教导器、小型教导器



(2) 双重安全开关功能的不同点

利用“手动模式”、“教导检查模式”操作多功能教导器、小型教导器时的双重安全开关功能的不同点列于下表。

全球型的情况	标准使用说明书所记述的内容
① 如果不按压“双重安全开关”，则不能操作机械手。 同时，“电机电源”也不能ON。	① 如果不按压“双重安全开关”，则不能操作机械手。 但是“电机电源”可以ON。
② 机械手操作过程中若释放“双重安全开关”，则机械手停止。 同时，“电机电源”也OFF。	② 机械手操作过程中若释放“双重安全开关”，则机械手停止。 但是，“电机电源”不OFF。 (伺服锁定状态)

2.2.2 有关单一位置控制功能 (Single Point of control)

全球型控制器上追加了单一位置控制功能 (Single Point of control)。全球型以外的机械手没有此功能。

(1) 什么叫单一位置控制功能

所谓单一位置控制功能是安全操作机械手的功能之一，是将操作机械手的机器限定为一个的功能。具体来说，就是自动模式被参数限定为 "内部自动" 或 "外部自动"。

■ "内部自动" 限定模式

自动模式被限定为 "内部自动" 模式。程序启动可以从教导器开始，但不能从外部机器开始。

■ "外部自动" 限定模式

自动模式被限定为 "外部自动" 模式。程序启动可以从外部机器开始，但不能从教导器开始。

注意：在 "外部自动" 限定模式的情况，用外部自动不能使用 "操作盘功能"。

(2) 模式的设定方法 (设定参数)

从多功能教导器上进行以下设定。

注1：出厂时，被设定为 "外部自动" 限定模式。

注2：全球型的控制器，在多功能教导器画面上有 "A" 的显示。

<操作路径>

[基本画面] → [F4: I/O] → [F6: 辅助功能] → [F1: 硬件设定] → [F3: 编号范围] → "31" 或在【Ver. 2.3 以下】的情况，

[基本画面] → [F4: I/O] → [F6: 辅助功能] → [F4: 单一控制]



第3章 控制器的通信界面概要

3.1 Mini I/O 输入输出信号的种类及其概要

本章就机械手控制器的输出信号进行说明。

输入输出信号包括通用输入输出信号和专用输入输出信号。

不用选件品的 I/O 增设卡时，为 Mini I/O 专用配置模式。使用 Mini I/O (CN5) 和 HAND I/O (CN9) 的连接器。

3.1.1 Mini I/O 输入输出信号的种类（使用标准型控制器时）

将程序启动等的命令作为 I/O 指令，使用指令执行用输入 7 点进行指示。专用输入输出信号的种类，如下表所示。

输出信号的种类（标准型控制器）

系统固定		
种类	点数	功能
专用输入	4点	外部紧急停止输入1、外部紧急停止输入2、自动启动允许输入、步骤停止（所有任务）
专用输出	13点 (注1)	自动模式、机械手初始化完成、机械手运行过程中、CPU正常、机械手异常、运行准备完成、电池耗尽警告、紧急停止输出1、紧急停止输出2、双重安全开关输出1（允许开关输出1）、双重安全开关输出2（允许开关输出2）、教导器紧急停止输出1、教导器紧急停止输出2、连续开始启动（用I/O硬件设定进行设定时）（注1）
指令执行用输入	7点	指令（3比特）、数据空间（3比特）、选通信号
指令执行用输出	1点	指令处理完成
利用用户程序控制		
种类	点数	功能
通用输入	8点	用IN指令、IO[] 变量读入外部I/O状态所进行的输入 用于分析条件判断、条件处理等待、从外部进行数据输入等
通用输出	8点 (注1)	在程序运行过程中，用SET、RESET指令等对外部机器发出信号时的输出
夹治具输入	8点	用IN指令、IO[] 变量读入外部I/O状态所进行的输入 用于夹治具的校验状态确认等。
夹治具输出	8点	用SET、RESET指令等，对外部机器发出信号时的输出 用于夹治具的开闭控制等

注1：在CN5端子No.53（端口编号24）上，出厂时配置了通用输出，但可以用I/O硬件设定设定为专用输出（连续开始允许）。

3.1.2 Mini I/O 输入输出信号的种类（使用全球型控制器时）

全球型控制器的情况，紧急停止有关的专用输入输出被集约在安全 I/O (CN10)。因此在使用全球型控制器时，不使用与 Mini I/O (CN5) 紧急停止相关的专用输入输出。

（参照 RC7M 型控制器说明书的 4.1.3~4.1.4 项以及 5.1.3~5.1.4）

另外，与标准型控制器相同，将程序启动等命令作为 I/O 指令，使用 "指令执行用输入 7 点" 来指示。

专用输入输出信号的种类，如下表所示。

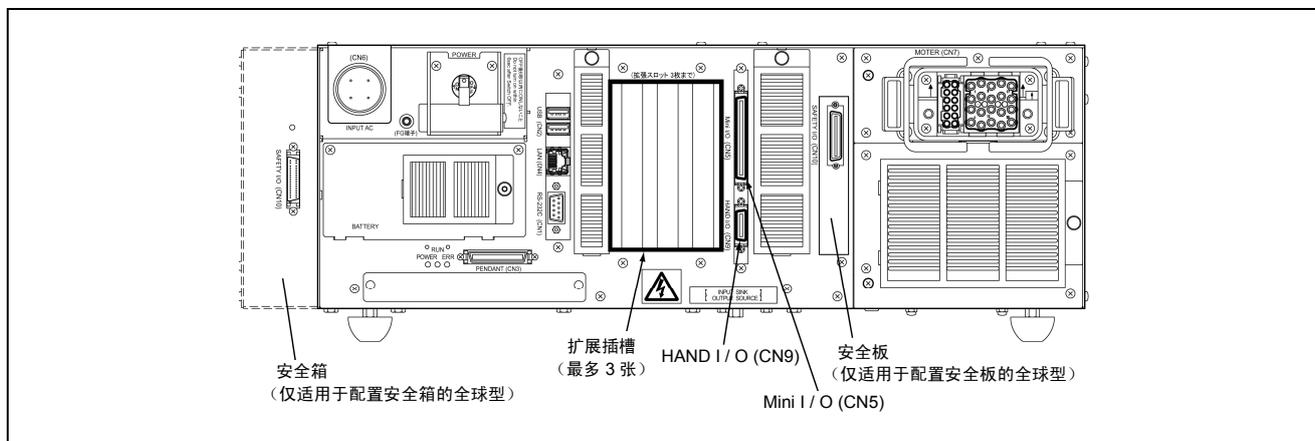
输入输出信号的种类（全球型控制器）

系统固定		
种类	点数	功能
专用输入	7 点	外部紧急停止输入1、外部紧急停止输入2、自动允许输入1、自动允许输入2、步骤停止（所有任务）、保护停止输入1、保护停止输入2
专用输出	12 点 (注 1)	自动模式、机械手初始化完成、机械手运行过程中、CPU 正常、机械手异常、运行准备完成、电池耗尽警告、教导器紧急停止输出1、教导器紧急停止输出2、双重安全开关输出1（允许开关输出1）、双重安全开关输出2（允许开关输出2）、接触器接点监视器输出连续开始启动（用 I/O 硬件进行设定时）（注1）
指令执行用输入	7 点	指令（3比特）、数据空间（3比特）、选通信号
指令执行用输出	1 点	指令处理完成
利用用户程序控制		
种类	点数	功能
通用输入	8 点	用 IN 指令、IO [] 变量读入外部 I/O 状态所进行的输入用于分析条件判断、条件处理等待、从外部进行数据输入等
通用输出	7 点 (注 1)	在程序运行过程中，用 SET、RESET 指令等对外部机器发出信号时的输出
夹治具输入	8 点	用 IN 指令、IO [] 变量读入外部 I/O 状态所进行的输入用于夹治具的校验状态确认等。
夹治具输出	8 点	用 SET、RESET 指令等，对外部机器发出信号时的输出用于夹治具的开闭控制等

注1: 在 CN5 端子 No.53（端口编号 24）上，出厂时配置了通用输出，但可以用 I/O 硬件设定设定为专用输出（连续开始许可）。

3.2 I/O 增设卡的概要

在 RC7M 型控制器中，仅限 Mini I/O (CN5) 和 HAND I/O (CN9) 的 I/O 点数不足时，或想用各种现场网络控制机械手时，使用 I/O 增设卡。在 RC7M 型控制器上，备有 3 个扩展插槽，最多可使用 2 张 I/O 增设卡。



3.2.1 I/O 增设卡的种类

I/O 增设卡有本公司作为选件品准备的 I/O 任选卡和 I/O 推荐任选卡。I/O 增设卡的种类如下表所示。

注： 重复制造使用 RC5 控制器的系统时，若使用选件品的 "I/O 转换箱" 则很方便。
(参照选件机器说明书 "I/O 转换箱")

(1) I/O 任选卡的种类

卡 采购分类	I/O 任选卡名	I/O 任选卡名的型号	
		内置于控制器上出厂	卡单独出厂
选件品 (向本公司订购)	并行 I/O 卡 (NPN 型)	410010-3320	410010-3340
	并行 I/O 卡 (PNP 型)	410010-3330	410010-3350
	DeviceNet 子局卡	410010-3370	410010-3400
	DeviceNet 主局卡	410010-3380	410010-3410
	DeviceNet 主局、子局卡	410010-3390	410010-3480
	CC-Link 卡	410010-3430	410010-3440

(2) I/O 推荐任选卡的种类

卡 采购分类	I/O 推荐卡名	制造商 (型号)	可向控制器追加功能的许可证的编号	
			追加功能完毕出厂	用户进行追加功能的操作
推荐 选件品 (用户自备)	PROFIBUS-DP 子局卡	Hilscher GmbH 制造 (CIF50-DPS \ DENSO)	410006-0300	410006-0310
	S-LINK V 卡	SUNX (株) 制造 (SL-VPCI)	410006-0280	410006-0290
	RS232C 增设卡	(株) CONTEC 制造 (COM-2P (PCI) H)	410006-0260	410006-0270
	EtherNet/IP Adapter 卡	Hilscher GmbH 制造 (CIFX50-RE \ DENSO)	410006-0800	410006-0810

3.3 I/O 增设卡的组合与配置模式

最多可以安装 2 块 I/O 增设卡。安装的插槽号码、顺序上没有限制。

可以安装的 I/O 增设卡的组合和可以选择的配置模式如下表所示。

I/O 增设卡的组合表

组合 No	I/O 增设卡 (最大 2 块)			配置模式			
	增设 (1)	增设 (2)	增设 (3)	Mini I/O 专用	配置在增设 (1) 上		全通用
					互换	标准	
0	—	—	—	○			
1	—	S-Link V	—	○			○
2	—	DeviceNet 主局	—				○
3	—	DeviceNet 主局	并行 I/O				○
4	—	DeviceNet 主局	S-Link V				○
5	并行 I/O	—	—	○	○	○	
6	并行 I/O	并行 I/O	—	○	○	○	
7	并行 I/O	S-Link V	—	○	○	○	○
8	DeviceNet 子局	—	—		○	○	
9	DeviceNet 子局	并行 I/O	—		○	○	
10	DeviceNet 子局	S-Link V	—		○	○	○
11	DeviceNet 主局 & 子局	—	—		○	○	○
12	DeviceNet 主局 & 子局	并行 I/O	—		○	○	○
13	DeviceNet 主局 & 子局	S-Link V	—		○	○	○
14	CC-Link	—	—		○	○	
15	CC-Link	并行 I/O	—		○	○	
16	CC-Link	DeviceNet 主局	—		○	○	○
17	CC-Link	S-Link V	—		○	○	○
18	PROFIBUS-DP 子局	—	—		○	○	
19	PROFIBUS-DP 子局	并行 I/O	—		○	○	
20	PROFIBUS-DP 子局	DeviceNet 主局	—		○	○	○
21	PROFIBUS-DP 子局	S-Link V	—		○	○	○
22	EtherNet/IP Adapter	—	—		○	○	
23	EtherNet/IP Adapter	并行 I/O	—		○	○	
24	EtherNet/IP Adapter	DeviceNet 主局	—		○	○	○
25	EtherNet/IP Adapter	S-Link V	—		○	○	○

注1: 配置模式栏中所显示的○标记中, 可以选择任何1个模式。

注2: I/O 增设卡可以增设到2张。安装的插槽位置、顺序上没有限制。

注3: 增设2张并行 I/O 卡时, 扩展插槽左侧的扩展插槽作为第1张卡识别。

在第1张和第2张上, 配置的 I/O 端口编号不同。

3.3.1 各配置模式 I/O 配置表

下表是 "不同配置模式的各增设卡的配置" 一览。关于实际的配置表，请参照 "13.6 I/O 配置表"。

注: "DeviceNet 主局&子局" 卡的配置请参照 "DeviceNet 子局" 以及 "DeviceNet 主局" 的配置表。

不同配置模式的各增设卡的配置

配置模式	CN5 和增设卡的配置表	
	对象 I/O	配置表
"Mini I/O 专用"	CN5	"Mini I/O 专用"
	增设(1) (2) (3)	各增设卡的 "全通用"
互换	CN5	"Mini I/O 互换 / 标准 / 全通用"
	增设(1)	各增设卡的 "互换"
	增设(2) (3)	各增设卡的 "全通用"
标准	CN5	"Mini I/O 互换 / 标准 / 全通用"
	增设(1)	各增设卡的 "标准"
	增设(2) (3)	各增设卡的 "全通用"
全通用	CN5	"Mini I/O 互换 / 标准 / 全通用"
	增设 (1) (2) (3)	各增设卡的 "全通用"

注: 增设 (1) (2) (3) 是前一页的 "I/O 增设卡的组合表" 中所记述的。

3.3.2 各配置模式的功能概要

在不同配置模式下，输入输出信号的功能有所不同。下表是 "各配置模式的功能概要" 一览。

各配置模式的功能概要

配置模式	概要
"Mini I/O 专用"	用比特组合指示动作，但相对于标准配置，功能有一定程度的削减。 Mini I/O 的空间配置 Mini I/O 专用配置。装配有 I/O 增设卡时，I/O 增设卡的空间全部都是通用信号。
互换	程序启动等功能在每个比特上都有意义，通过启用某比特对动作进行指示。 在 I/O 增设卡的空间配置有专用配置的 "互换"。Mini I/O 的空间就是所有端口通用信号（除 CPU 正常之外）。
标准	用比特的组合（I/O 指令）指示程序启动等。功能是最多的配置。 在 I/O 增设卡的空间配置有专用配置的 "标准"。Mini I/O 的空间就是所有端口通用信号（除 CPU 正常之外）。
全通用	I/O 增设卡的空间就是所有通用信号，Mini I/O 的空间就是所有端口的通用信号（除 CPU 正常之外）。

3.4 选择互换、标准、全通用模式时的 Mini I/O 的功能

选择 Mini I/O 专用模式以外的 I/O 配置模式（互换、标准、全通用）时，Mini I/O 的专用输出输入占用的端口全部变为通用输入输出端口（CPU 正常除去外）。

- 专用输入端口编号 0~7（CN5 端子 No.11~18）→通用输入
- 专用输出端口编号 17~23（CN5 端子 No.46~52）→通用输出

注意:专用输出信号 "CPU正常" 留存在端口编号16（CN5端子No.45）中。

3.5 界面的必要设定

3.5.1 I/O 配置模式的设定

将配置模式向 "Mini IO 专用"、"互换"、"标准"、"全通用" 进行变更时，需要使用多功能教导器或 WINCAPSIII对参数进行变更。

关于变更方法，请参照 "3.6 I/O 配置模式的设定方法"。

注意：以 "控制器上组装出厂" 的方式购进的可以选择配置模式为 "互换" 或 "标准" 的 I/O 增设卡，在出厂时的配置模式为 "标准" 模式。

3.5.2 I/O 用电源 (+24V DC) 的设定

在 Mini I/O 卡 (CN5) 及并行 I/O 卡 (选件) 中，对于将 I/O 用电源 (+24V DC) 作为外部电源规格或作为内部电源规格可进行选择。
变更时，需要通过跳线进行变更。

有关变更方法，请参照 "3.7 Mini I/O 用电源的设定方法" 及 "3.8 并行 I/O 用电源的设定方法"。

注意：产品出厂时，设定为外部电源。

3.6 I/O 配置模式的设定方法

3.6.1 通过多功能教导器进行设定的方法

操作路径：[F4 I/O] — [F6 辅助功能] — [F2 I/O 配置]

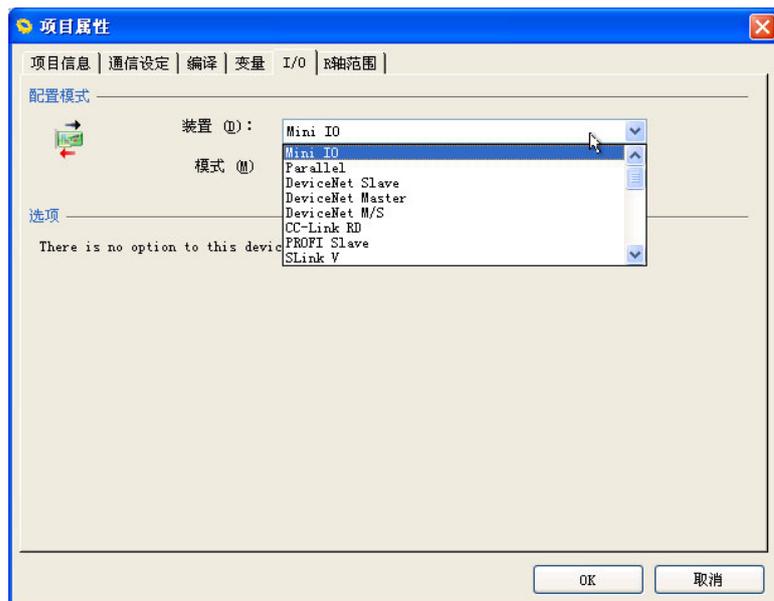
进行上述操作后，用十字键或漫步旋钮将光标对准任意的配置，然后按 [OK]。
电源重新投入后设定为有效。



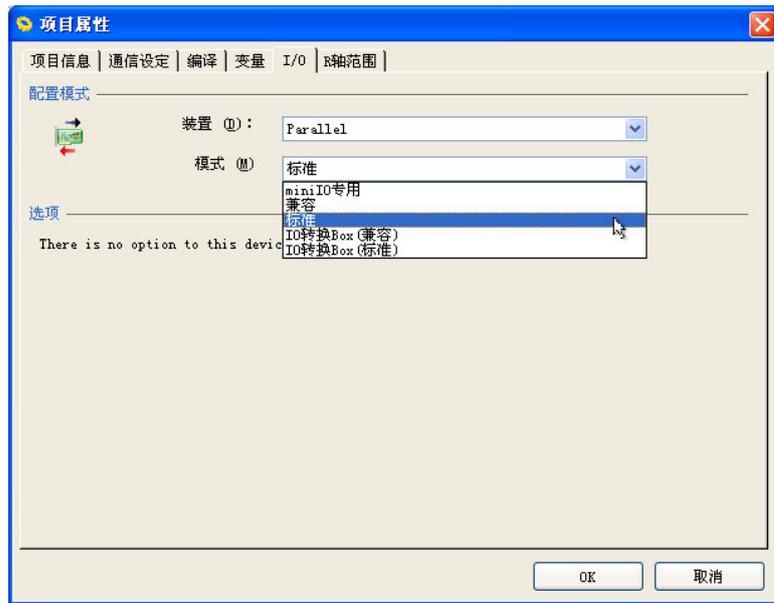
3.6.2 通过 WINCAPSIII 的设定方法

(1) 选择 [项目 (P)] — [属性 (P)] 菜单，[项目属性] 对话框就会显示出来。
选择 [I/O] 标记。

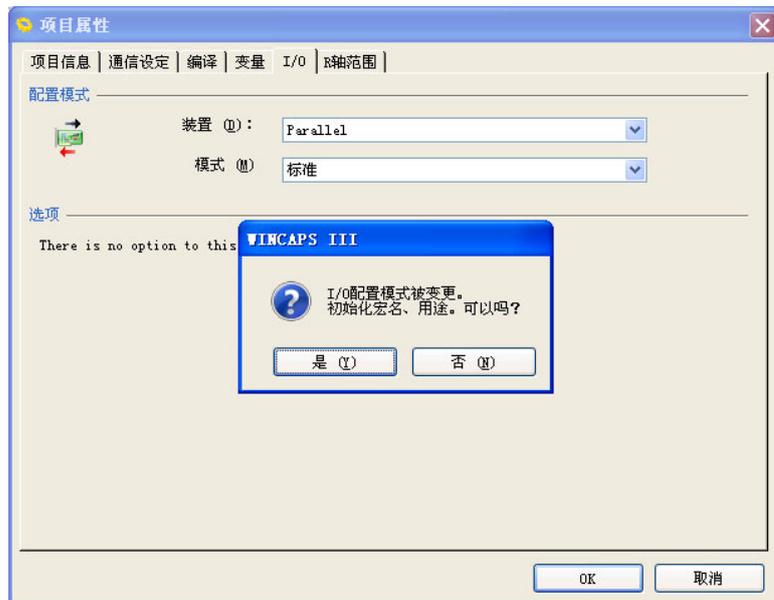
(2) 选择 [配置模式] 栏的 [装置 (D)]，再从下拉菜单中选择 I/O 卡的种类。



(3) 请选择 [模式 (M)]，并从下拉菜单选择配置。



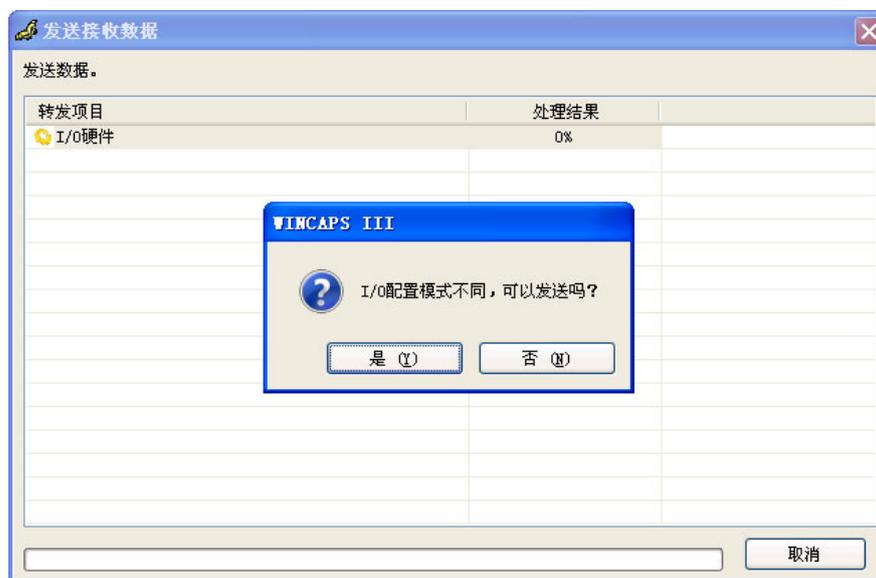
(4) 若按压 [OK]，则显示下列讯息。如果 <用途> 以及 <宏名> 可以被初始化，则请按压 [是 (Y)]。如果不能被初始化，则请按压 [否 (N)]。按压任何一个，配置都会被变更。



- (5) 选择 [通信 (N)] — [发送接收数据 (T)]，通过 WINCAPSIII 向控制器发送 I/O 配置设定。
请选择 [参数] 中的 [I/O 参数] 发送。



- (6) 控制器内数据已更新的确认讯息与 I/O 配置模式不同的确认讯息会显示出来。在这两个确认讯息处都选择 [是 (Y)]，即可向控制器发送。
重启控制器，接收到的数据就会显示出来。



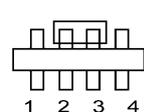
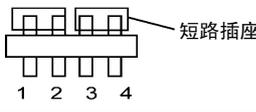
注意： 请不要选择没有装配的 I/O 任选卡并将配置数据传输给控制器。
传输之后重新接通电源，会发生错误。

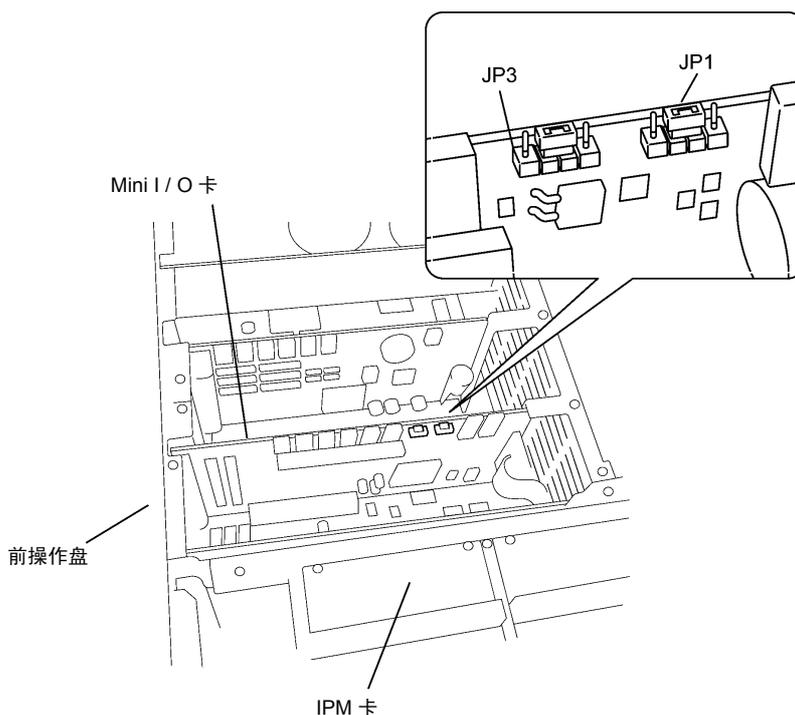
3.7 Mini I/O 用电源的设定方法

用控制器的跳线设定Mini I/O用电源(+24V DC)的来源，外部电源或内部电源。
出厂时的设定为外部电源。

Mini I/O用电源的设定

I/O用电源的设定	JP1、JP3设定	设定方法
外部电源	将2-3短路	出厂时设定的状态。
内部电源	将1-2以及3-4短路	卸下控制器的上盖，使用机械手附件短路插座，变更下图的JP1和JP3的跳线设定。

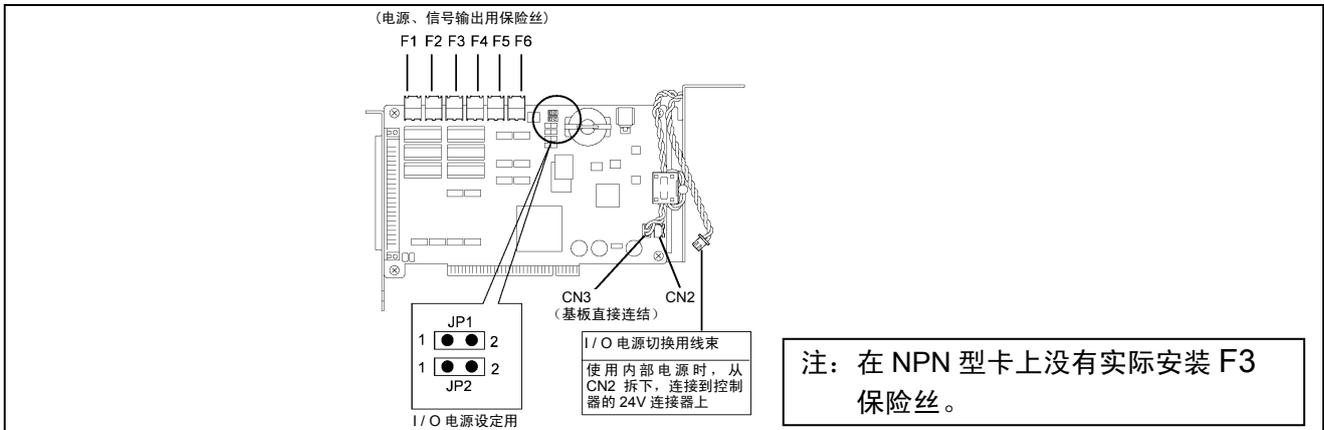
JP1、JP3 的状态	
外部电源时	短路（出厂设定） 
内部电源时	1-2 及 3-4 短路 



注： 如果将I/O用电源的设定从外部电源变更为内部电源，则CN5端子No.32~34以及端子No.66~68从外部DC电源输入变更为内部DC电源输出。

3.8 并行 I/O 用电源的设定方法

并行的 I/O 板可将 I/O 用电源 (+24V DC) 设置为外部电源或设置为内部电源。
 出厂时，为外部电源设定。
 将并行的 I/O 卡各个部分的名称列于下图。



I/O 用电源的设定	JP1、JP2 设定	设定方法
外部电源	 JP1, JP2 (打开)	出厂时设定的状态 (JP1、JP2 均打开)。
内部电源	 JP1, JP2 (短路)	(1) 用短路插座将 "JP1 的 1-2" 以及 "JP2 的 1-2" 进行短路。 (2) 卸下与并行 I/O 卡的 I/O 电源切换用配线束的 CN2 连接器的连接、请与控制器本体的 24V 连接器 (CNP101) 相连接。 与控制器的 24V 连接器 (CNP101) 连接 前操作盘侧 安装在控制器上的并行 I/O 卡 (3) 装配 2 张并行 I/O 卡使用内部电源时，请将第 2 张卡的 I/O 电源切换，用配线束与第 1 张卡的 CN2 连接。 第 1 张卡 CN3 CN2 I/O 电源切换用线束 CNP102 (与控制器的 24V 连接器) 第 2 张卡 CN3 CN2

注意：请务必在控制器电源为 OFF 的状态下进行设定。

3.9 I/O 端口管理表与配置

不使用 I/O 增设卡时，I/O 端口编号（通过 PAC 程序或 I/O 指令进行 I/O 处理时指定的编号）最大为 511，但是，若使用 I/O 增设卡（选件品）时，I/O 端口编号可追加在 512 以后。

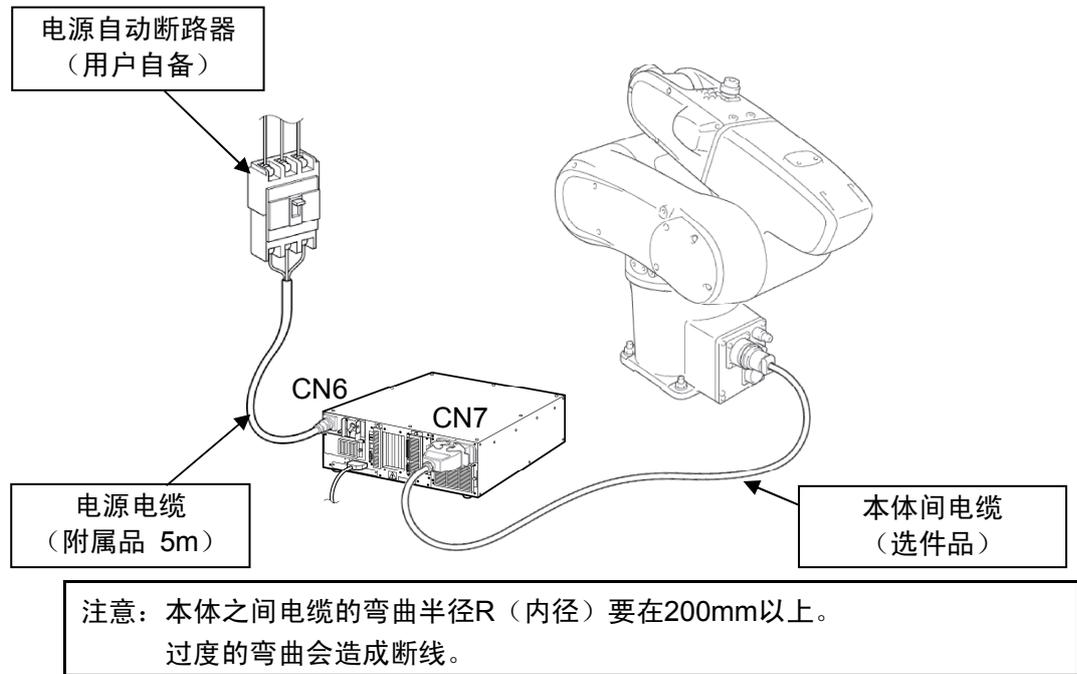
I/O 端口管理表与配置

I/O 端口编号	配置	空间
0 ~ 15	Mini I/O 输入	标准空间
16 ~ 31	Mini I/O 输出	
32 ~ 47	未使用	
48 ~ 55	夹治具输入	
56 ~ 63	未使用	
64 ~ 71	夹治具输出	
72 ~ 127	未使用	
128 ~ 511	内部 I/O	
512 ~ 767	DeviceNet 子局 输入 CC-Link 输入 PROFIBUS-DP 子局 输入 EtherNet/IP Adapter 输入	I/O 增设卡（选件）空间
768 ~ 1023	DeviceNet 子局 输出 CC-Link 输出 PROFIBUS-DP 子局 输出 EtherNet/IP Adapter 输出	
1024 ~ 2047	DeviceNet 主局 输入	
2048 ~ 3071	DeviceNet 主局 输出	
3072 ~ 3327	S-LINK V 输入	
3328 ~ 3583	S-LINK V 输出	
3584 ~ 3623	并行卡（第1张）输入	
3624 ~ 3663	并行卡（第2张）输入	
3664 ~ 3839	未使用	
3840 ~ 3887	并行卡（第1张）输出	
3888 ~ 3935	并行卡（第2张）输出	
3936 ~ 4095	未使用	
4096 ~ 4351	CC-Link 远程寄存器（RWw）输入	
4352 ~ 4607	未使用	
4608 ~ 4863	CC-Link 远程寄存器（RWr）输出	
4096 ~ 7871	EtherNet/IP Adapter 输入	
7872 ~ 11647	EtherNet/IP Adapter 输出	

第 4 章 电缆的连接方法

4.1 控制器与电源、机械手本体间的电缆连接

控制器与控制器供电电源之间请使用附属的电源电缆（5m 规格）。
在机械手本体和控制器之间，请选用选件本体间电缆（2m、4m、6m、12m、20m）进行连接。

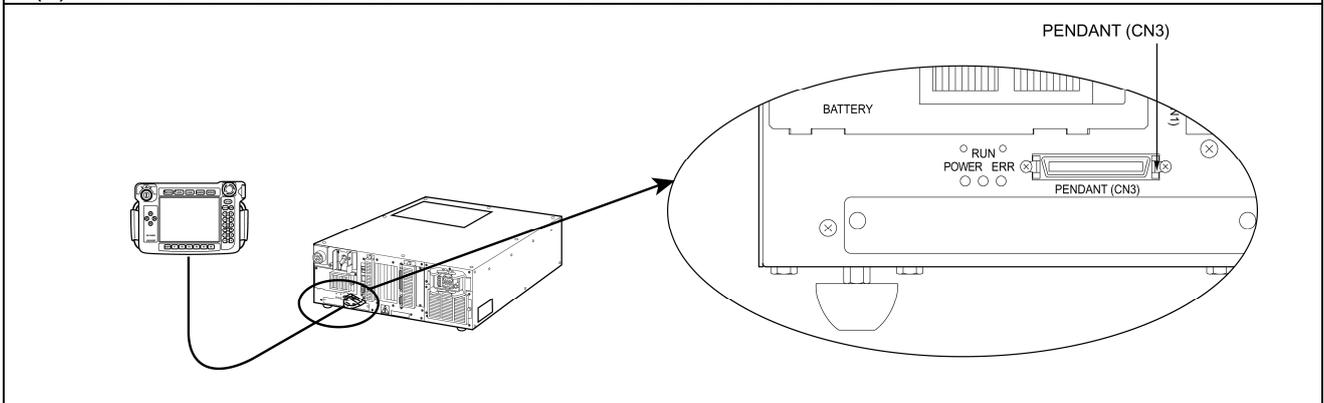


4.2 教导器的电缆连接

多功能教导器如下图所示，请与机械手控制器的 PENDANT 连接器 (CN3) 连接。

教导器电缆与控制器连接时的注意事项

- (1) 电缆连接之后，请不要在连接器部（下图的○部位）的上下或者左右方向施加外力。会成为连接器破损引起通信异常的原因。
- (2) 拆卸电缆时，请解除连接器的锁定，直着拔出连接器而不要撬连接器。



多功能教导器的连接

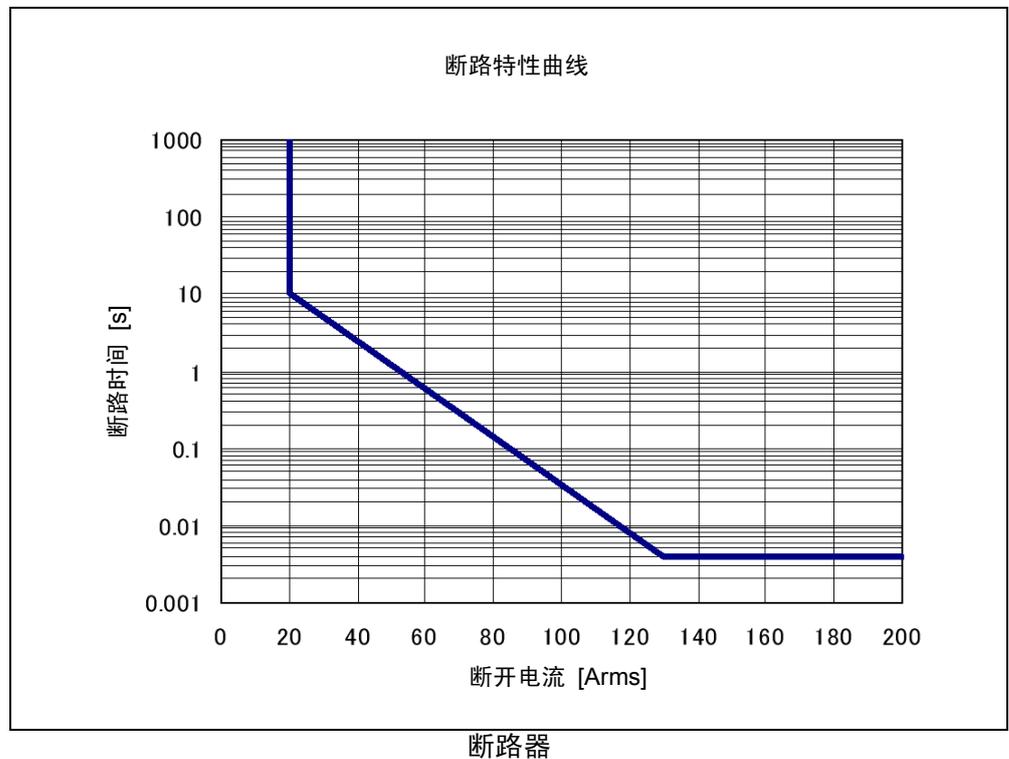
4.3 断路器的选定（推荐产品）

机械手控制器的一次侧电源的配线时，请注意以下情况。

- (1) 机械手所用的电源与焊接电源必须以不同电源配线。
- (2) 机械手用电源电缆的地线（保护接地）（绿 / 黄），请切实接地。
- (3) 机械手控制器的接地端子（功能接地）要用 1.25mm^2 以上的配线进行接地。
- (4) 机械手用电源的接地（保护接地），请使用 D 种接地（接地电阻 100Ω 以下）。
- (5) 机械手控制器的供电电源侧使用漏电断路器时，作为变换器，请使用已采取高频对策的产品。
- (6) 在 AC 输入电源上连接断路器时，请选定断路特性在下图以上的断路器。

推荐断路器示例：CP33V / 20（富士电机机器制御株式会社）

注意：连接的断路器如果其断路特性在下图以下，则有时会因机械手的动作而使断路器切断。

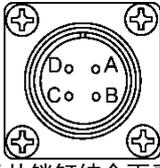
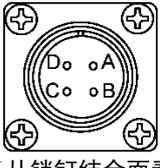


4.4 一次侧电源的配线方法

机械手控制器的一次侧电源的配线时，请注意以下情况。

- (1) 机械手所用的电源与焊接电源必须以不同电源配线。
- (2) 机械手用电源电缆的地线（保护接地）（绿/黄），请切实接地。
- (3) 机械手控制器的接地端子（功能接地）要用 1.25mm² 以上的配线进行接地。
- (4) 机械手用电源的接地（保护接地），请使用 D 种接地（接地电阻 100Ω 以下）。
- (5) 机械手控制器的供电电源侧使用漏电断路器时，作为变换器，请使用已采取高频对策的产品。
- (6) 电源电缆请参照下表准备适当容量的电缆。

机械手控制器的电源规格

项目		规格	电源连接器 (CN6) 的销钉排列	
电源电压	3 相 AC200V 规格	3相AC200V-15%~AC230V+10%、50 / 60Hz	 <p>A: AC200V R相 B: AC200V S相 C: AC200V T相 D: 地线（保护接地）</p>	
	电源容量	VMG6BA: 3.3kVA		VSG6BA: 1.85kVA
		VPG5 / 6CA: 1kVA		HMG4BA: 2.45kVA
		HSG4BA: 1.8kVA		XYCG4AA: 1.15kVA
		XRG4BA: 1.8kVA	VS-050/060: 1.15kVA	
			VS-068/087: 2.78kVA	
	单相 AC200V 规格	单相AC230V-10%~AC230V+10%、50 / 60Hz	 <p>A: AC200V R相 B: AC200V S相 D: 地线（保护接地）</p>	
	电源容量			VSG6BA: 1.85kVA
		VPG5 / 6CA: 1kVA		HMG4BA: 2.45kVA
HSG4BA: 1.8kVA		XYCG4AA: 1.15kVA		
XRG4BA: 1.8kVA		VS-050/060: 1.15kVA		
		VS-068/087: 2.78kVA		
单相 AC100V 规格	单相AC100V-10%~AC110V+10%、50 / 60Hz	 <p>A: AC100V R相 B: AC100V S相 D: 地线（保护接地）</p>		
电源容量	VPG5 / 6CAA: 1kVA			
接通电源时的最大瞬间电流		40A（1 / 50秒或1 / 60秒）		

注意：机械手动作时，如果发生错误6102（电源电压降低）的状况，一次侧电源的容量不足可能是一个重要原因。

- (7) 请不要将教导器电缆、输入输出电缆、机电缆等电缆与电源电缆、外围设备等强电线捆绑在一起，也不要将机电缆安装在强电机（电机、焊接机、零件供给器等）的附近。
- (8) 在机械手本体内不要增设夹治具用的电缆，气体配管，以避免发生机械手用机电线及编码器线的断线故障。
- (9) 请务必与适合控制器的电源规格（AC200V 或 AC100V 规格）的电源进行连接。

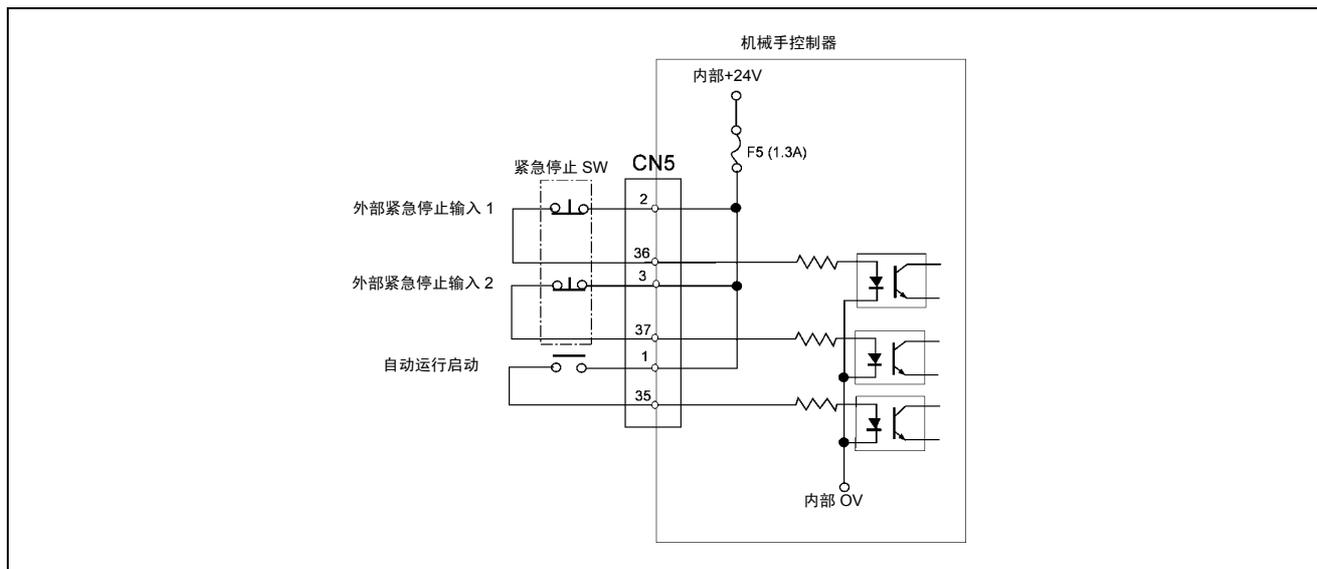
第 5 章 专用输入信号的接线

5.1 调整机械手时必要的接线

本章节对安装调试机械手时，为了进行机械手裸机的电机 ON、自动运行、手动运行操作所需的最低限度的接线进行说明。

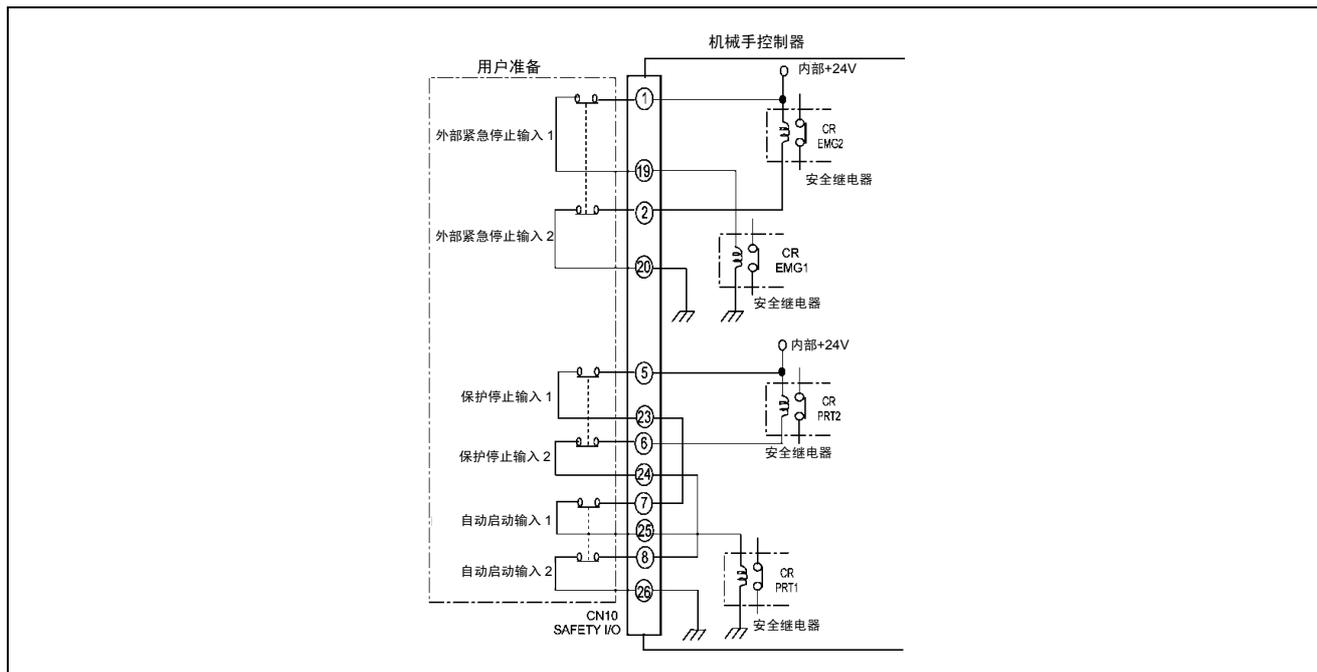
5.1.1 紧急停止电路的构成（使用标准型控制器时）

外部紧急停止和自动允许的信号，是保证安全的重要信号。如下图所示，必须用有接点的电路构成。



5.1.2 紧急停止电路的构成（全球型控制器）

安全回路的输入信号是保证安全的重要信号。注意以下几点，必须用有接点的电路构成。



注：关于安全回路的整体结构，请参照 RC7M 型控制器说明书的 "4.2.5.2 安全回路"。

5.2 电机变为 ON 状态所需的接线

5.2.1 机能

在紧急停止输入的 2 个系统均处于短路状态时，电机才能 ON。

5.2.2 标准型控制器

信号名称	端子编号
外部紧急停止输入1	连接器CN5的No.2和36
外部紧急停止输入2	连接器CN5的No.3和37

[注意] 紧急停止2系统的状态有1秒钟以上的异常时，发生错误279E（机械手停止输入的1系统不能输入）。

5.2.3 全球型控制器

信号名称	端子编号
外部紧急停止输入1	连接器CN10的No.1和19
外部紧急停止输入2	连接器CN10的No.2和20

[注意] 请用各自的接点控制2个外部紧急停止输入。采用1个接点的 "2电路并联连接" 或 "1电路常短路" 判断为外部电路异常，紧急停止不能解除。

5.3 自动运行所需要的接线

5.3.1 机能

- ① 在短路状态下，可以将机械手切换至自动模式。
- ② 在释放状态下，可以将机械手切换至手动模式、教导检查模式。

5.3.2 标准型控制器

信号名称	端子编号
自动允许输入	连接器CN5的No.1和35

5.3.3 全球型控制器

信号名称	端子编号
自动允许输入1	连接器CN10的No.7和25
自动允许输入2	连接器CN10的No.8和26
保护停止输入1	连接器CN10的No.5和23
保护停止输入2	连接器CN10的No.6和24

- [注意] ① 请用各自的连接控制2个自动允许输入和2个防护降低输入。1个接点的 "2电路并联连接" 或 "1电路常时短路" 判断为外部电路异常，不能动作。
- ② 自动允许输入和保护停止输入在机械手控制器内串联连接。作为自动模式切换许可信号，可以进行2种输入。
- ③ 不需要保护停止时，跳线连接连接器CN10的No.5和23、No.6和24，即使常时短路也能使用。

第2部

机械手操作

第 6 章 坐标系

6.1 4 轴机械手坐标系的种类

操作机械手时的坐标系有以下 3 种。

- 基准坐标系
- 工件坐标系
- 工具坐标系

6.2 4 轴机械手基准坐标系

所谓基准坐标系即通用坐标系，意为三维直角坐标。其原点位于机械手的底座中心。基准坐标系的 X 轴表示为 X_b 、Y 轴为 Y_b 、Z 轴为 Z_b 。

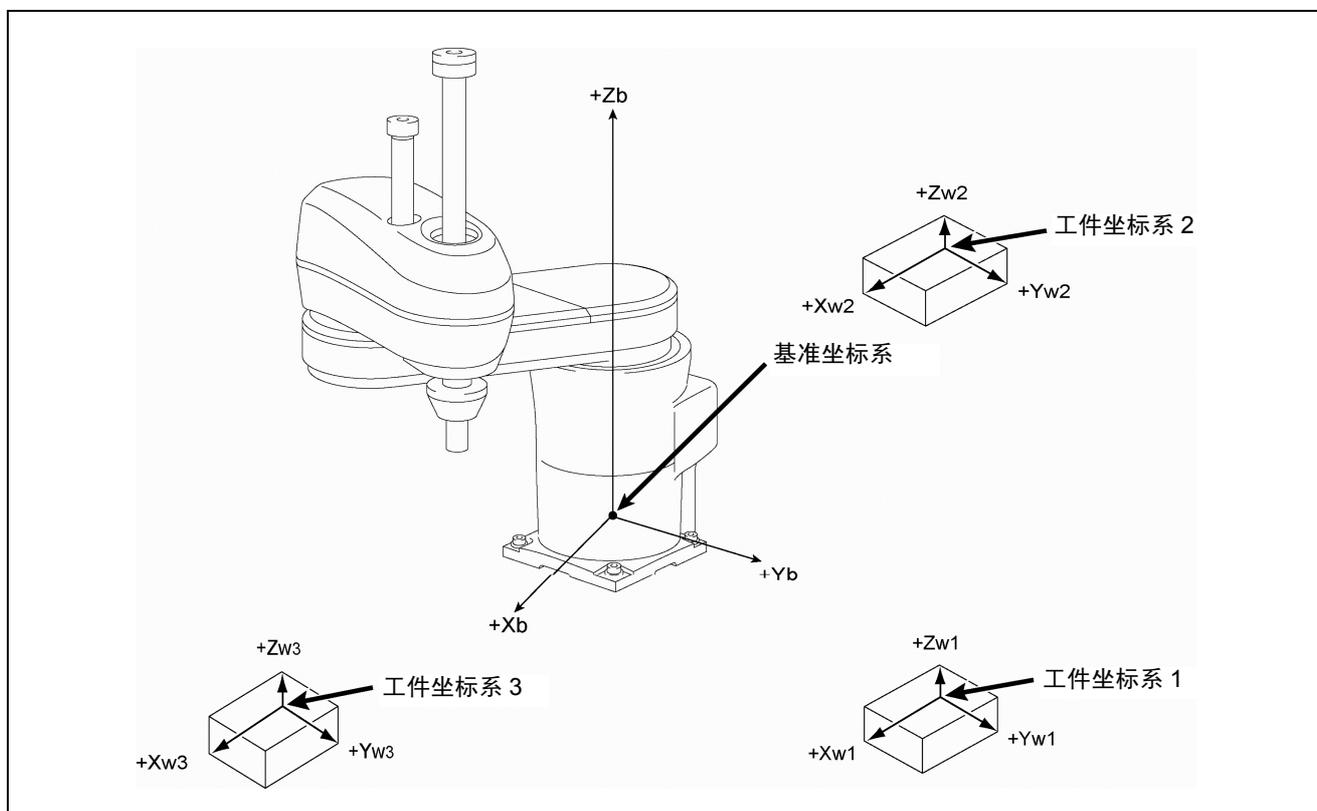
6.3 4 轴机械手工件坐标系

工件坐标系，被定义为各个轴与基准坐标系的各个轴平行的三维直角坐标系。工件坐标系，是在机械手欲要操作的目标对象工件的每个作业空间上被定义的三维直角坐标。如下图所示，其原点为目标对象工件的包络长方体的一个顶点。工件坐标是用相对于基准坐标的坐标原点位置 (X, Y, Z) 与围绕基准坐标的 X 轴、Y 轴、Z 轴的旋转角度 (R_x, R_y, R_z) 来表示。

最大可以登录 7 种工件坐标，赋予其 1~7 号的工件坐标编号。

工件坐标未定义的情况下，就是基准坐标。原点定义可以设定在任意位置，任意数量。

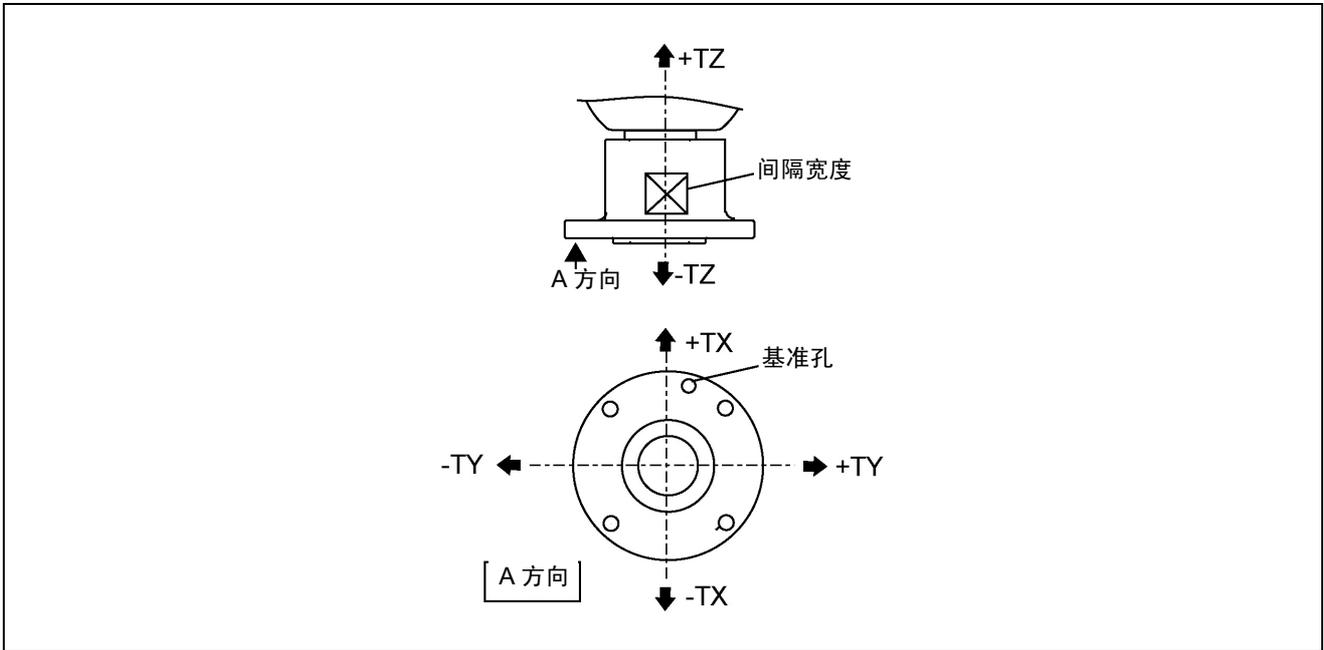
注：为了利用工件坐标系，需要预先设定工件坐标。详细内容请参照操作指南 "4.2.1 [1.3] 工件坐标系的设定"。



工件坐标系和基准坐标系

6.4 4轴机械手工件坐标系

工具坐标系是以如下图所示的机械界面坐标的原点为基准，定义偏移量和旋转角度的三维直角坐标。最大可以登录 63 种工具坐标，赋予其 1~63 号的工具坐标编号。

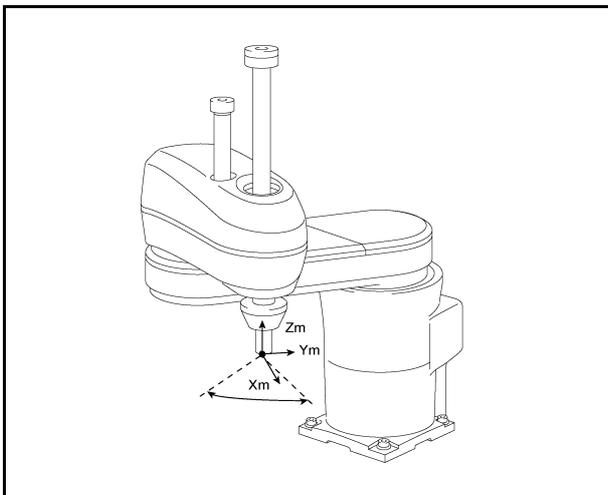


机械界面坐标系

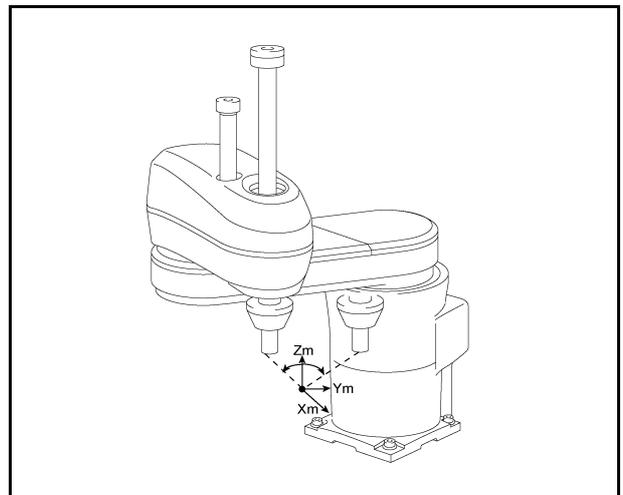
注：为了利用工具坐标系，需要预先设定工具坐标。详细内容请参照操作指南 "4.2.1 [2.2] 工具定义的输入方法"。

6.5 4轴机械手工具坐标系的优点

在手动模式下动作的情况下，如果使用工具坐标系，则通过工具定义可以以偏移点为中心运行机械手工具的前端。



不使用工具定义而在X-Y模式下手动旋转第4轴时

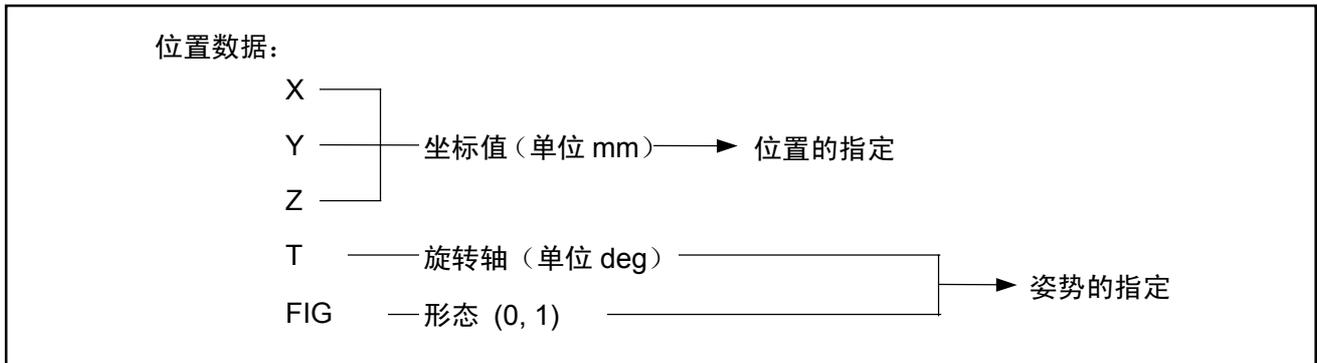


使用工具定义，在X-Y模式下手动旋转第4轴时

6.6 在 4 轴机械手上使用的位置数据

位置数据是指,机械手的法兰中心（工具定义有效时工具前端）的位置与表示此时的机械手的姿势的 5 要素（4 轴机械手的情况下）的基准坐标系的数据。

表示机械手的当前位置及移动目标位置的时候，使用包括机械手的姿势在内的该位置数据。



位置数据的构成

所谓用 X、Y、Z 表示的坐标值，是用基准坐标系 (Xb、Yb、Zb) 表示机械手的法兰中心（工具定义有效时为工具的前端）的位置的值，单位是 [mm]。

所谓用 T 所示的旋转角度，就是机械手工具 ϕ 坐标系（工具定义有效时为工具坐标系）的 X 轴与基准坐标系的 Xb 轴的夹角，单位是 [deg]。

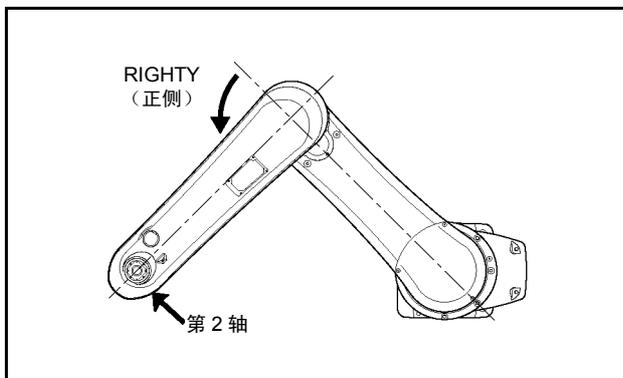
所谓用 FIG 表示的形态是指机械手臂的关节所取得的形态。

6.6.1 有关 4 轴的形态

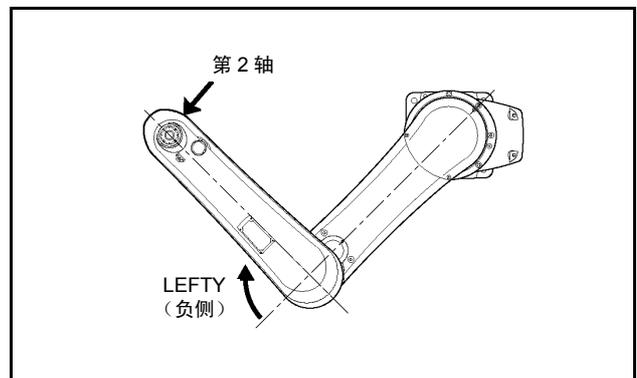
在对 4 轴机械手进行固定时，可以按下图所示采用 2 种姿势。

机械手形态

值	形态
0	RIGHTY
1	LEFTY



右手系 (RIGHTY)



左手系 (LEFTY)

第 2 轴位于正侧的状态称为右手系，第 2 轴位于负侧的状态称为左手系。

6.7 6 轴机械手坐标系的种类

操作机械手时的坐标系有以下 3 种。

- 基准坐标系
- 工件坐标系
- 工具坐标系

6.8 6 轴机械手基准坐标系

所谓基准坐标系即通用坐标系，意为三维直角坐标。其原点位于机械手的底座中心。基准坐标系的 X 轴表示为 X_b 、Y 轴为 Y_b 、Z 轴为 Z_b 。

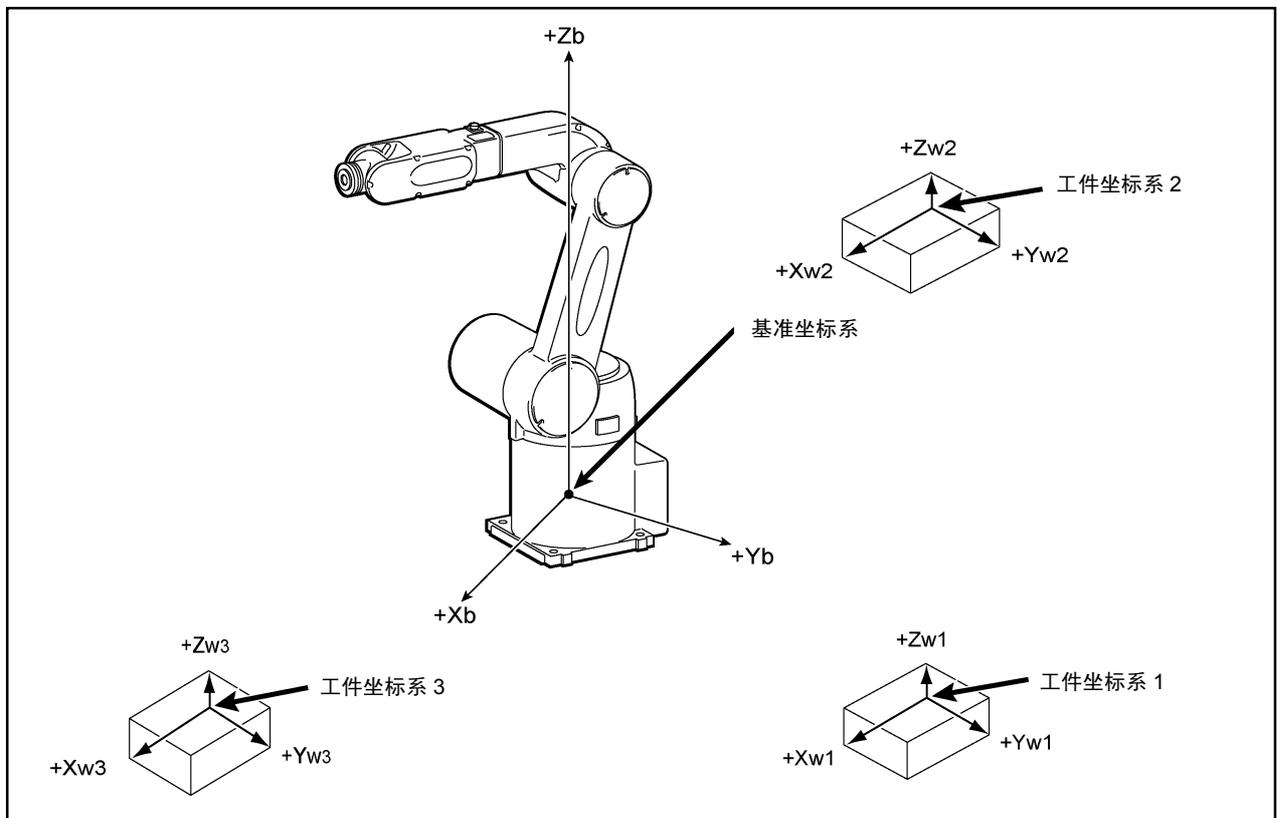
6.9 6 轴机械手工件坐标系

工件坐标系，被定义为各个轴与基准坐标系的各个轴平行的三维直角坐标系。工件坐标系，是在机械手欲要操作的目标对象工件的每个作业空间上被定义的三维直角坐标。如下图所示，其原点为包围目标对象工件的长方体的一个顶点。工件坐标是用相对于基准坐标的坐标原点位置 (X, Y, Z) 与围绕基准坐标的 X 轴、Y 轴、Z 轴的旋转角度 (R_x, R_y, R_z) 来表示。

最大可以登录 7 种工件坐标，赋予其 1~7 号的工件坐标编号。

工件坐标未定义的情况下，就是基准坐标。原点在任意的位置可以定义任意的数。

注：为了利用工件坐标系，需要预先设定工件坐标。详细内容请参照操作指南 "4.2.1 [1.3] 工件坐标系的设定"。

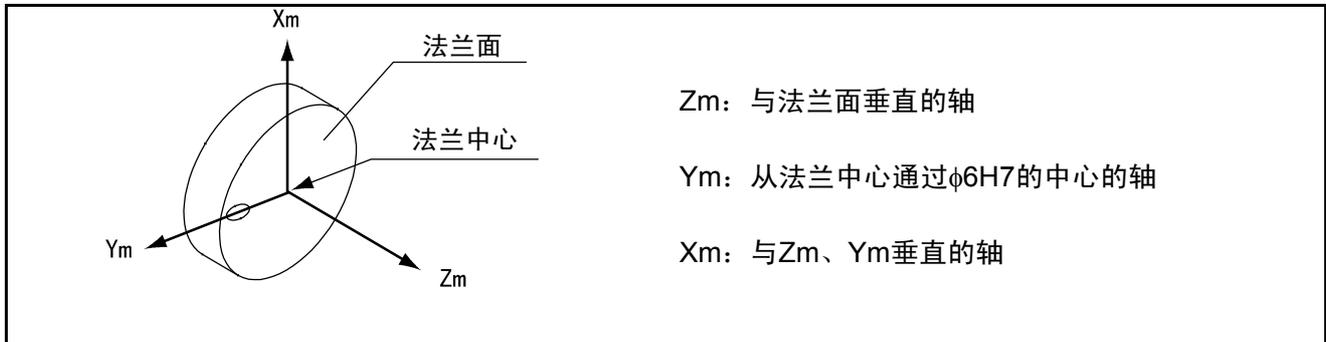


工件坐标系和基准坐标系

6.10 6轴机械手工件坐标系

工具坐标系是以如下图所示的机械界面坐标的原点为基准，定义偏移量和旋转角度的三维直角坐标。

最大可以登录 63 种工具坐标，赋予其 1~63 号的工具坐标编号。



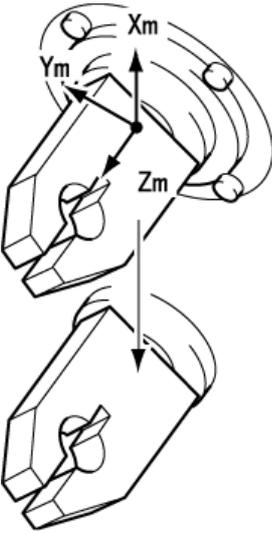
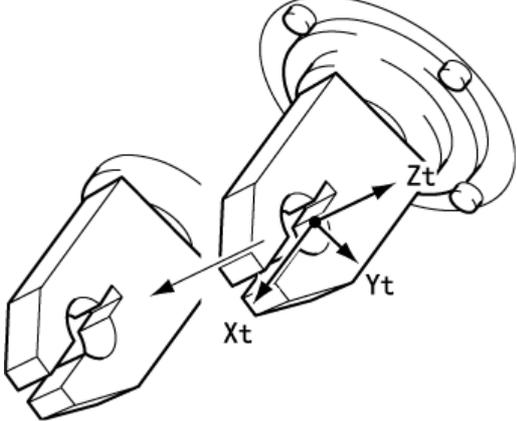
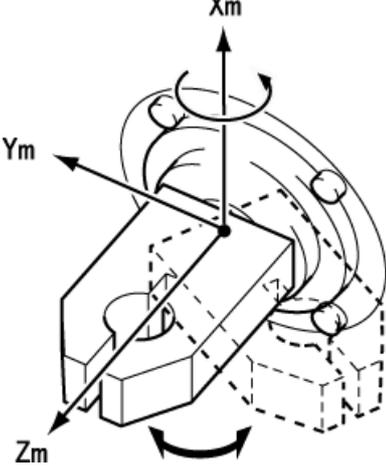
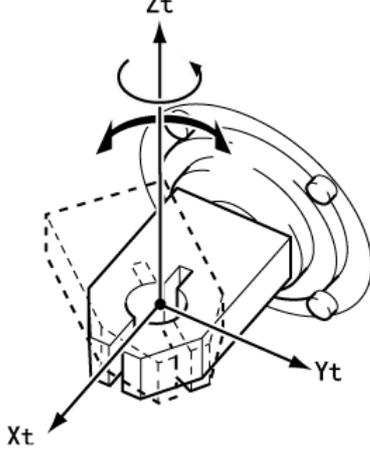
机械界面坐标系

注：为了利用工具坐标，需要预先设定工具坐标。详细内容请参照操作指南
"4.1.1 [2.4] 工具坐标的形成方法"。

6.11 6轴机械手工具坐标系的特性

用夹治具模式使其动作的情况下，如果使用工具坐标系，可以将夹治具面向任意方向运行，容易进行教导。

与机械界面坐标系适用时的动作比较如下图所示。

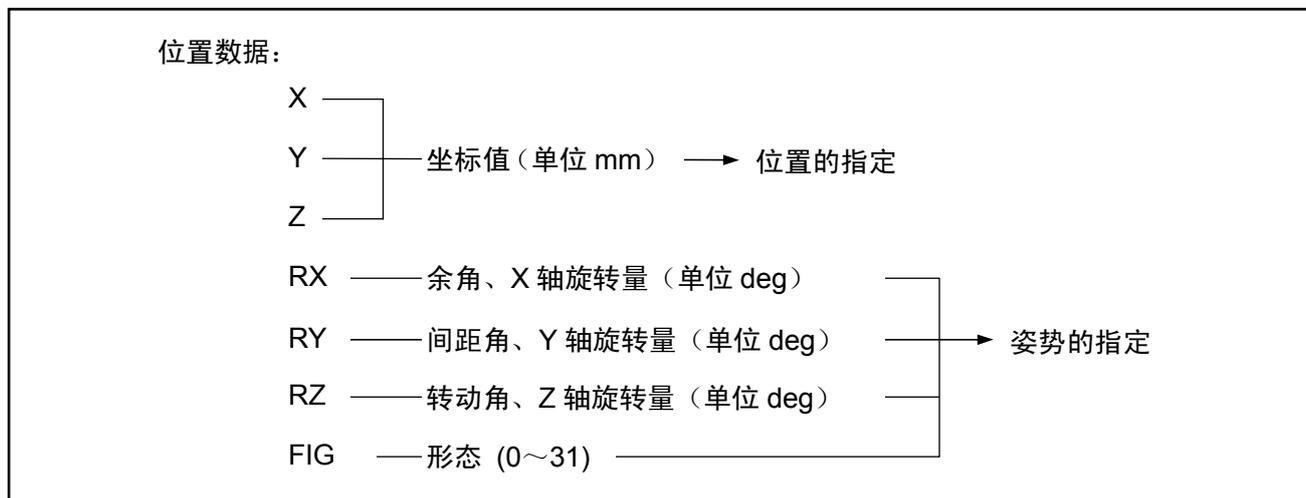
机械界面坐标系 (TOOLO)	工具坐标系
<p data-bbox="153 542 392 577">X- 按压的情况下</p> 	<p data-bbox="810 542 1050 577">Z- 按压的情况下</p>  <p data-bbox="810 1070 1362 1106">可以对将夹治具面向任意方向移动进行教导。</p>
<p data-bbox="153 1182 421 1218">RX+ 按压的情况下</p> 	<p data-bbox="810 1182 1078 1218">RZ+ 按压的情况下</p>  <p data-bbox="810 1706 1331 1742">可进行以夹治具为中心的旋转动作的教导。</p>

使用工具坐标的手动操作

6.12 在 6 轴机械手上使用的位置数据

位置数据是指机械手的法兰中心（工具定义有效时为工具前端）的位置与表示此时的机械手的姿势的 7 要素（6 轴机械手的情况下）的基准坐标系的数据。

表示机械手的当前位置及移动目标位置时，包括机械手的姿势也使用该位置数据。



位置数据的构成

用 X、Y、Z 表示的坐标值是用基准坐标系 (Xb、Yb、Zb) 表示机械手的法兰中心（工具定义有效时为工具的前端）的位置的值，单位是 [mm]。

用 RX、RY、RZ 表示的余角、导程角、倾滚角，是以围绕法兰中心为原点的坐标系（机械界面坐标系）的各自轴的旋转角度，单位是 [deg]。该旋转是按 RZ、RY、RX 的顺序进行旋转。如果改变旋转的顺序，则即使是相同的旋转角度也会成为不同的姿势，所以顺序是被决定的。

用 FIG 表示的形态是指机械手臂的关节所取得的形态。

6.12.1 有关 6 轴的形态

6 轴机械手，对于工具前端的 1 个位置和姿势 (X、Y、Z、RX、RY、RZ)，如以下所述，在臂、肘（关节）、手腕、第 6 轴、第 4 轴上可以采取不同的形态。

在(1)~(5)中，对于臂、肘、手腕、第 6 轴、第 4 轴的各自形态的不同进行说明。

如果将这些形态进行组合，则可以发现对于 1 个位置和姿势可以采取 32 种形态。将该形态的组合列于下表。

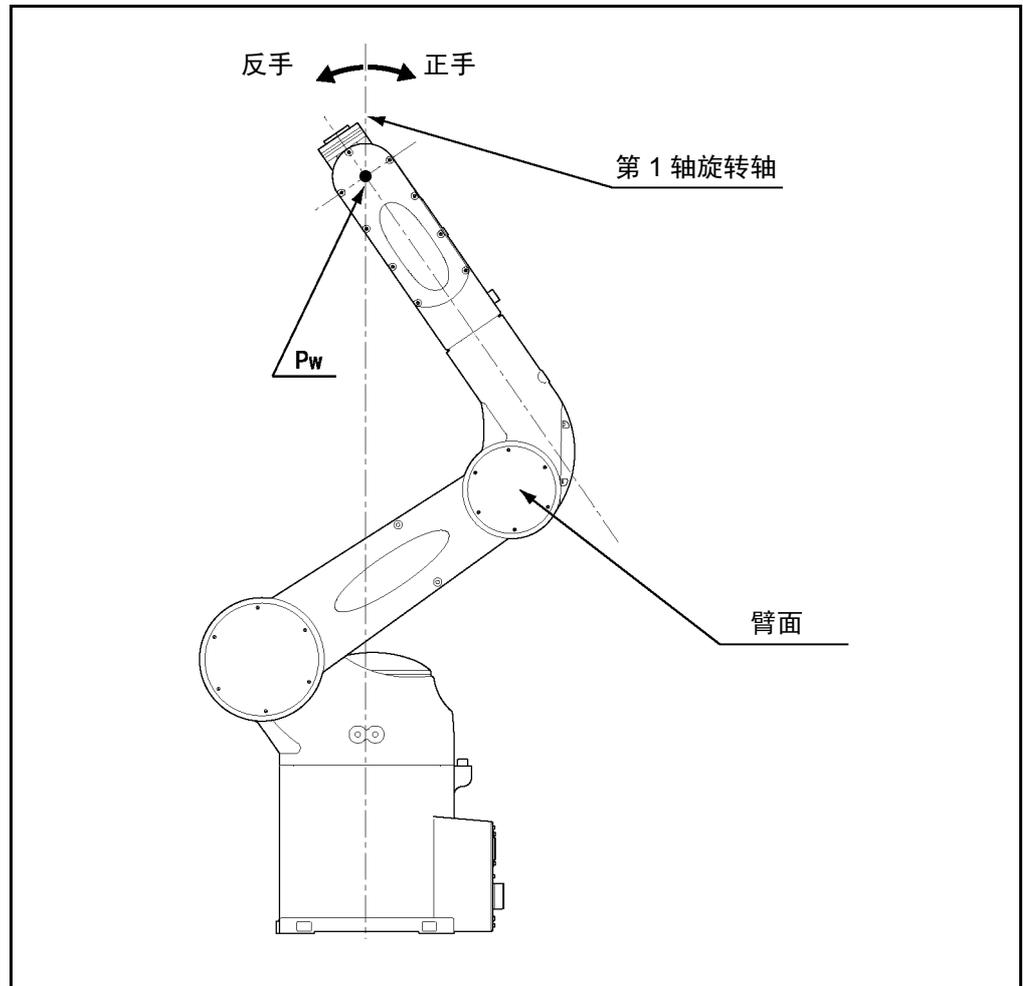
机械手形态

值	第 4 轴形态	第 6 轴形态	手腕形态	肘形态	臂形态
0	SINGLE 4	SINGLE	FLIP	ABOVE	RIGHTY
1	SINGLE 4	SINGLE	FLIP	ABOVE	LEFTY
2	SINGLE 4	SINGLE	FLIP	BELOW	RIGHTY
3	SINGLE 4	SINGLE	FLIP	BELOW	LEFTY
4	SINGLE 4	SINGLE	NONFLIP	ABOVE	RIGHTY
5	SINGLE 4	SINGLE	NONFLIP	ABOVE	LEFTY
6	SINGLE 4	SINGLE	NONFLIP	BELOW	RIGHTY
7	SINGLE 4	SINGLE	NONFLIP	BELOW	LEFTY
8	SINGLE 4	DOUBLE	FLIP	ABOVE	RIGHTY
9	SINGLE 4	DOUBLE	FLIP	ABOVE	LEFTY
10	SINGLE 4	DOUBLE	FLIP	BELOW	RIGHTY
11	SINGLE 4	DOUBLE	FLIP	BELOW	LEFTY
12	SINGLE 4	DOUBLE	NONFLIP	ABOVE	RIGHTY
13	SINGLE 4	DOUBLE	NONFLIP	ABOVE	LEFTY
14	SINGLE 4	DOUBLE	NONFLIP	BELOW	RIGHTY
15	SINGLE 4	DOUBLE	NONFLIP	BELOW	LEFTY
16	DOUBLE 4	SINGLE	FLIP	ABOVE	RIGHTY
17	DOUBLE 4	SINGLE	FLIP	ABOVE	LEFTY
18	DOUBLE 4	SINGLE	FLIP	BELOW	RIGHTY
19	DOUBLE 4	SINGLE	FLIP	BELOW	LEFTY
20	DOUBLE 4	SINGLE	NONFLIP	ABOVE	RIGHTY
21	DOUBLE 4	SINGLE	NONFLIP	ABOVE	LEFTY
22	DOUBLE 4	SINGLE	NONFLIP	BELOW	RIGHTY
23	DOUBLE 4	SINGLE	NONFLIP	BELOW	LEFTY
24	DOUBLE 4	DOUBLE	FLIP	ABOVE	RIGHTY
25	DOUBLE 4	DOUBLE	FLIP	ABOVE	LEFTY
26	DOUBLE 4	DOUBLE	FLIP	BELOW	RIGHTY
27	DOUBLE 4	DOUBLE	FLIP	BELOW	LEFTY
28	DOUBLE 4	DOUBLE	NONFLIP	ABOVE	RIGHTY
29	DOUBLE 4	DOUBLE	NONFLIP	ABOVE	LEFTY
30	DOUBLE 4	DOUBLE	NONFLIP	BELOW	RIGHTY
31	DOUBLE 4	DOUBLE	NONFLIP	BELOW	LEFTY

(1) 臂形态

对于臂面从垂直方向看到的 "Pw" 的位置是在第 1 轴的旋转轴的左边时为反手 (LEFTY)形态可移动范围、在右边时为正手 (RIGHTY) 形态可移动范围。

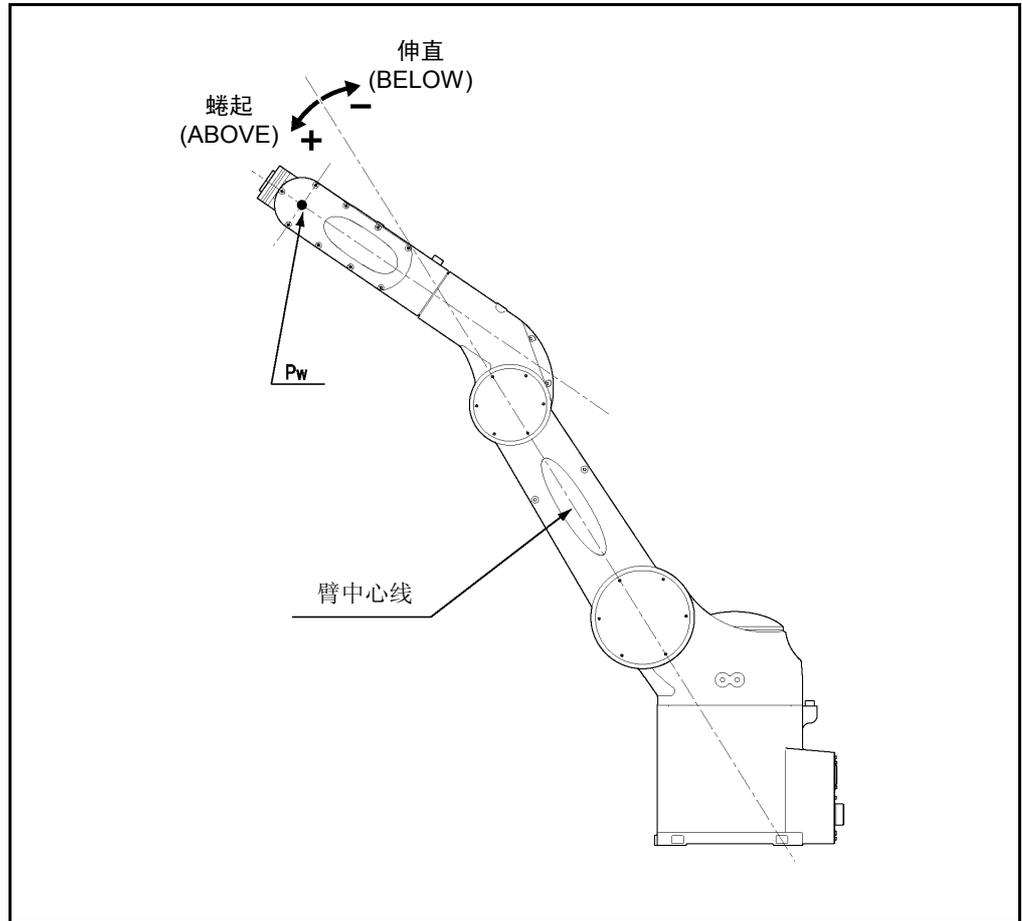
注意：第5轴的支轴中心 "Pw" 的位置在第1轴的旋转轴的上方时，则成为在2个形态的边界上，这个位置叫做特异点。



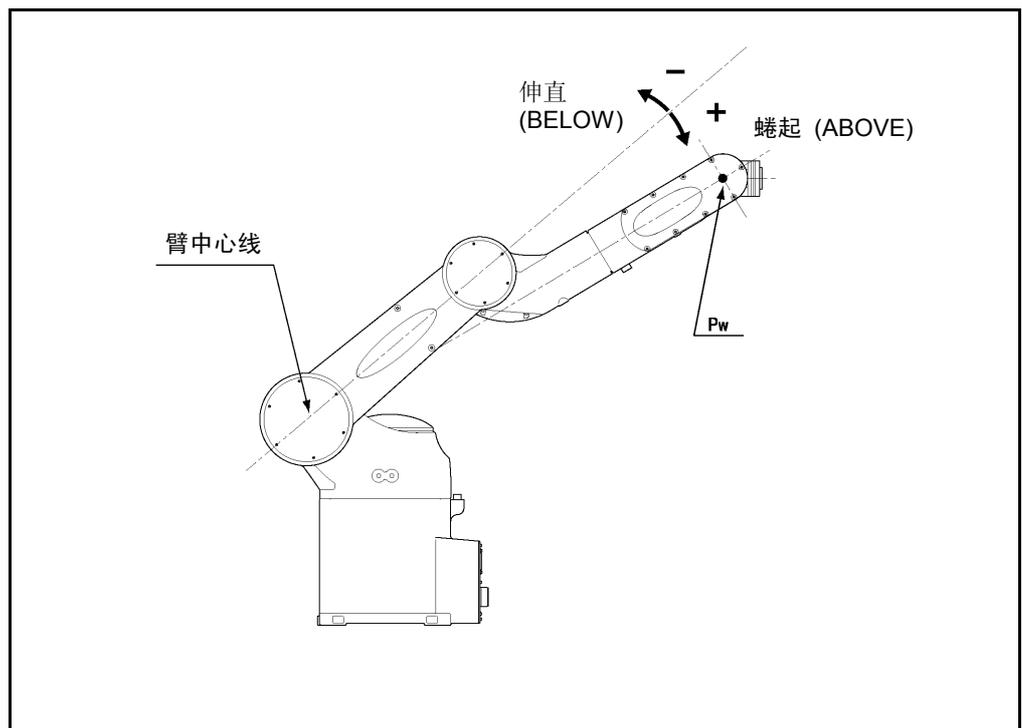
反手、正手的界限

(2) 肘的形态

对于臂中心线，在 "Pw" 的位置位于 +侧时是蜷起 (ABOVE) 形态可移动的范围。
位于 -侧时是伸直 (BELOW) 形态可移动的范围。



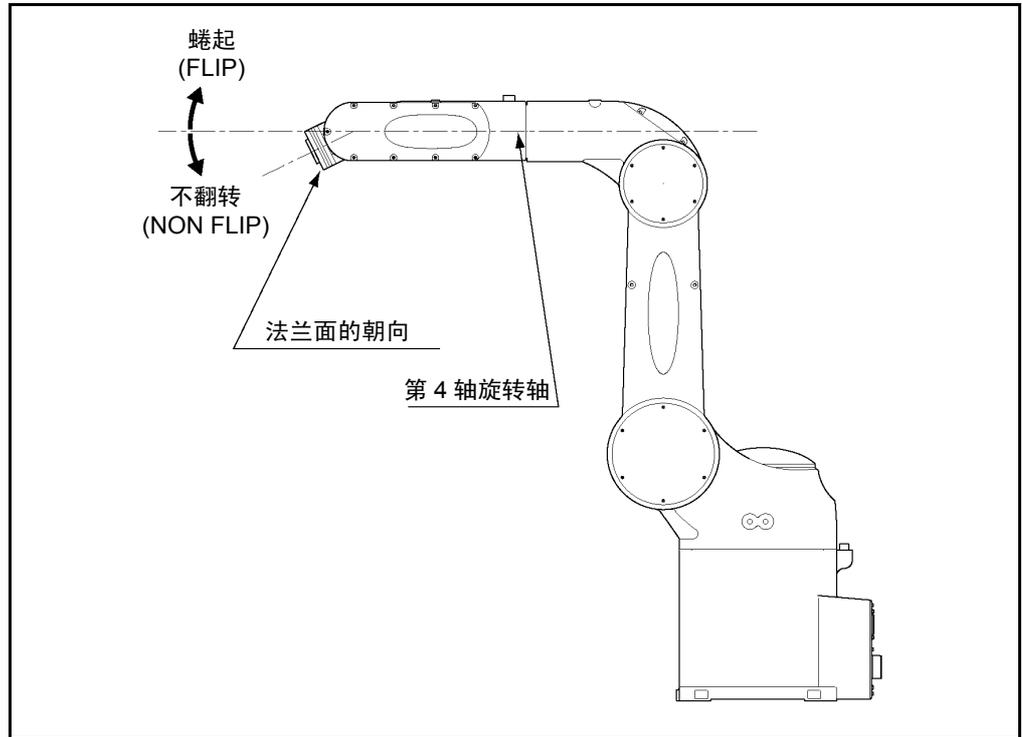
蜷起、伸直的界限（反手的情况）



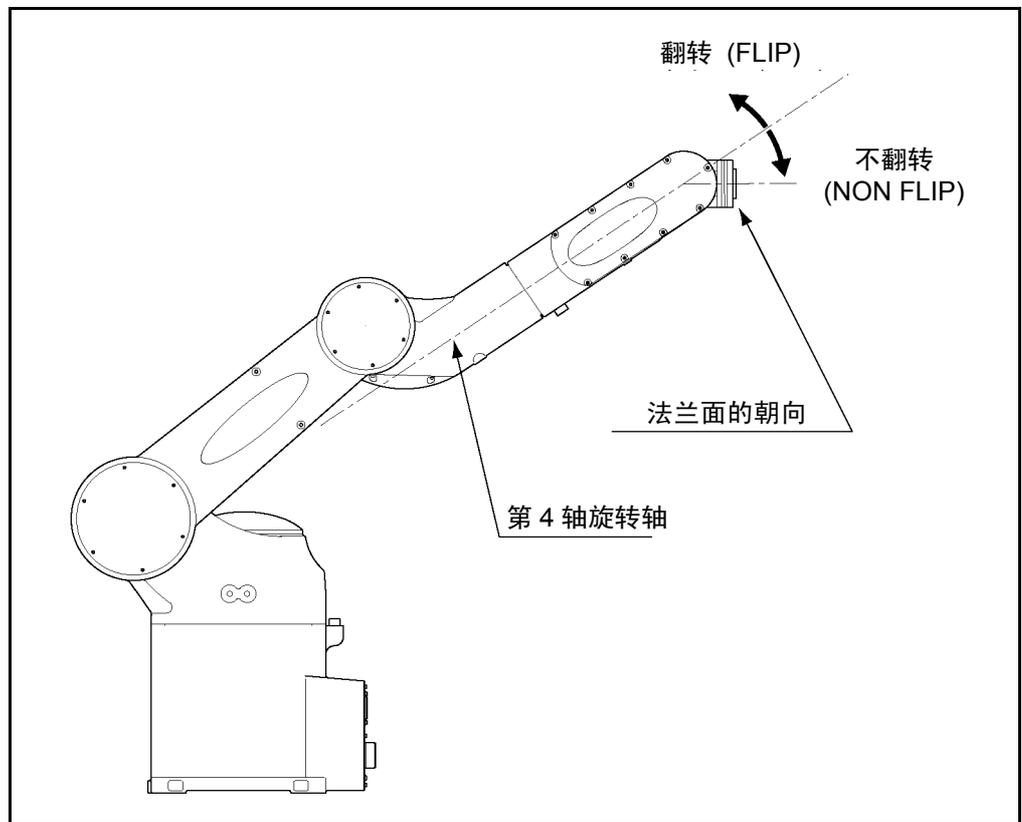
蜷起、伸直的界限（正手的情况下）

(3) 手腕形态

以第 4 轴的旋转轴中心线为基准，法兰面的朝向位于上侧时是翻动 (FLIP)，而位于下侧时是不翻动 (NONFLIP)。(图 4-38、图 4-39 位于下侧)



翻转、不翻转的界限（反手的情况）



翻转、不翻转的界限（正手的情况）

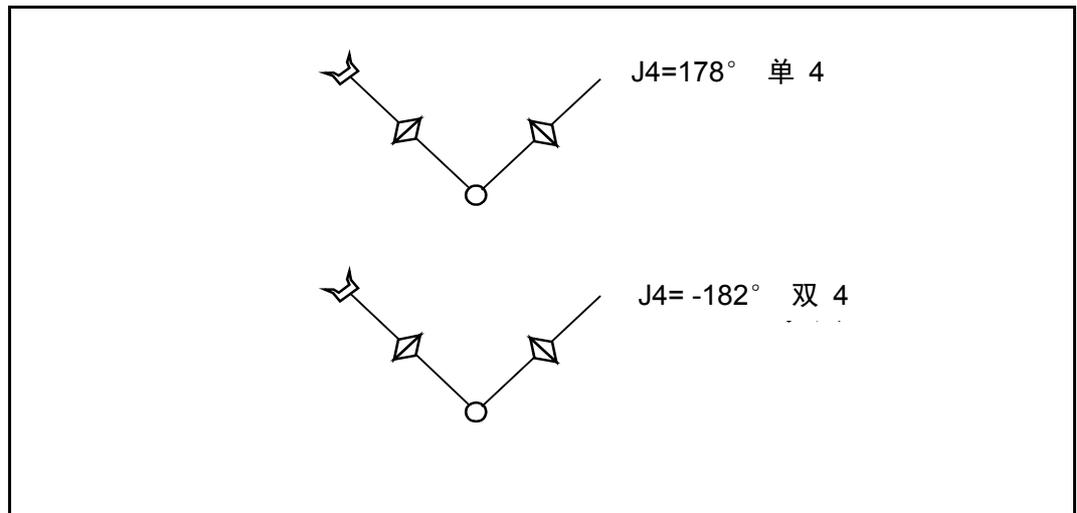
(4) 第 4 轴形态

第 4 轴形态、由第 4 轴的值决定。

第 4 轴的形态可以采取单 4 (SINGLE 4) 和双 4 (DOUBLE) 的 2 种类型。

第 4 轴的旋转角度 θ_4 为 $(-180 < \theta_4 \leq 180)$ 时是单 4，为 $(180 < \theta_4 \leq 185$ 或者 $-185 < \theta_4 \leq -180)$ 时是双 4。

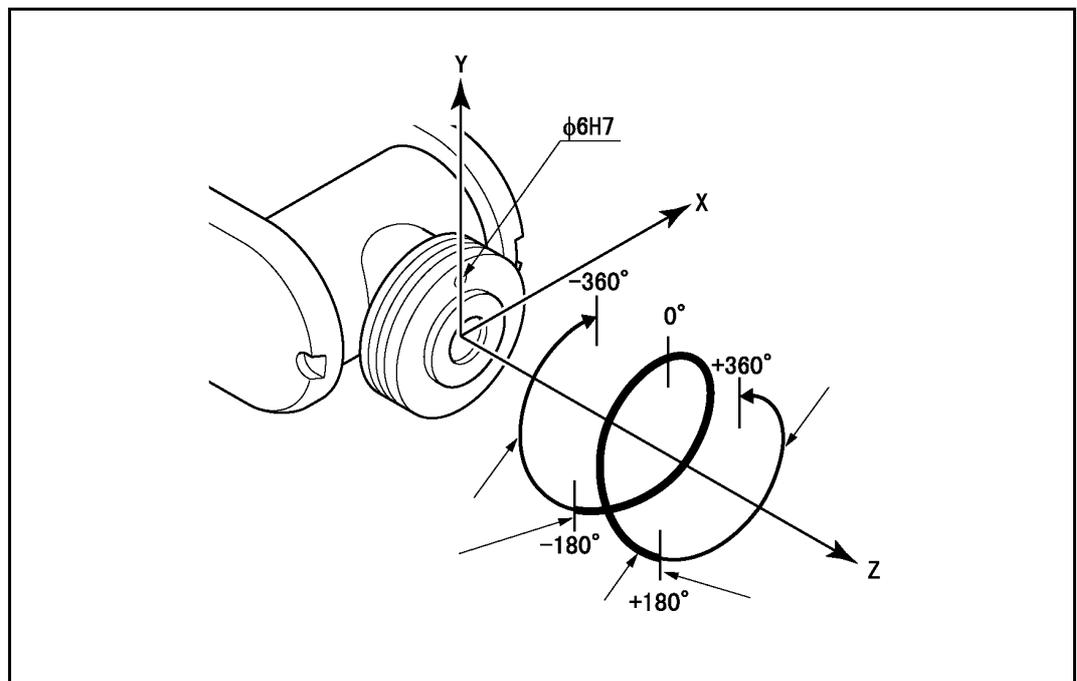
在 $\theta_4 = 180^\circ$ 和 $\theta_4 = 181^\circ$ 时第 6 轴形态改变。将位置数据进行数值变更时，请注意第 6 轴形态的设定。(打算修正为 $\theta_4 = 181$ ，但第 6 轴形态不被修正时则成为 $\theta_4 = -179$ 。)



第4轴形态

(5) 第 6 轴形态

第 6 轴的旋转角度 θ_6 为 $(-180 < \theta_6 \leq 180)$ 时是单 (SINGLE)，为 $(180 < \theta_6 \leq 360$ 或者 $-360 < \theta_6 \leq -180)$ 时是双 (DOUBLE)。



单轴、双轴的界限

第 7 章 教导之前的准备

7.1 多功能教导器的使用

7.1.1 多功能教导器的手持方法与双重安全开关

多功能教导器用手保持进行操作。多功能教导器上有确保安全的双重安全开关，采用如下所示的一种方法，可以用手指按住进行操作。

多功能教导器的拿法与双重安全开关

双手拿住多功能教导器进行操作。



★要点★

使用多功能教导器用手动模式操作机械手时，万一操作人员发生意识丧失、死亡等意外，双重安全开关将自动且安全地停止机械手动作。

双重安全开关根据受到的按压力度设定了开关，具体分为以下3种位置状态。发生意外的操作人员不会在“按压力度正常”范围内。

- 1) 按压力度正常 → 开关ON
- 2) 没有按压或按压力度微弱 → 开关OFF
- 3) 按压力度过大 → 开关OFF

开关为ON以外的情况下，机械手停止、无法运行。

为了确保多功能教导器的安全性，设计了例如在手动模式时、为了按压移动方向键让机械手运行、必须同时按压双重安全开关的机能。

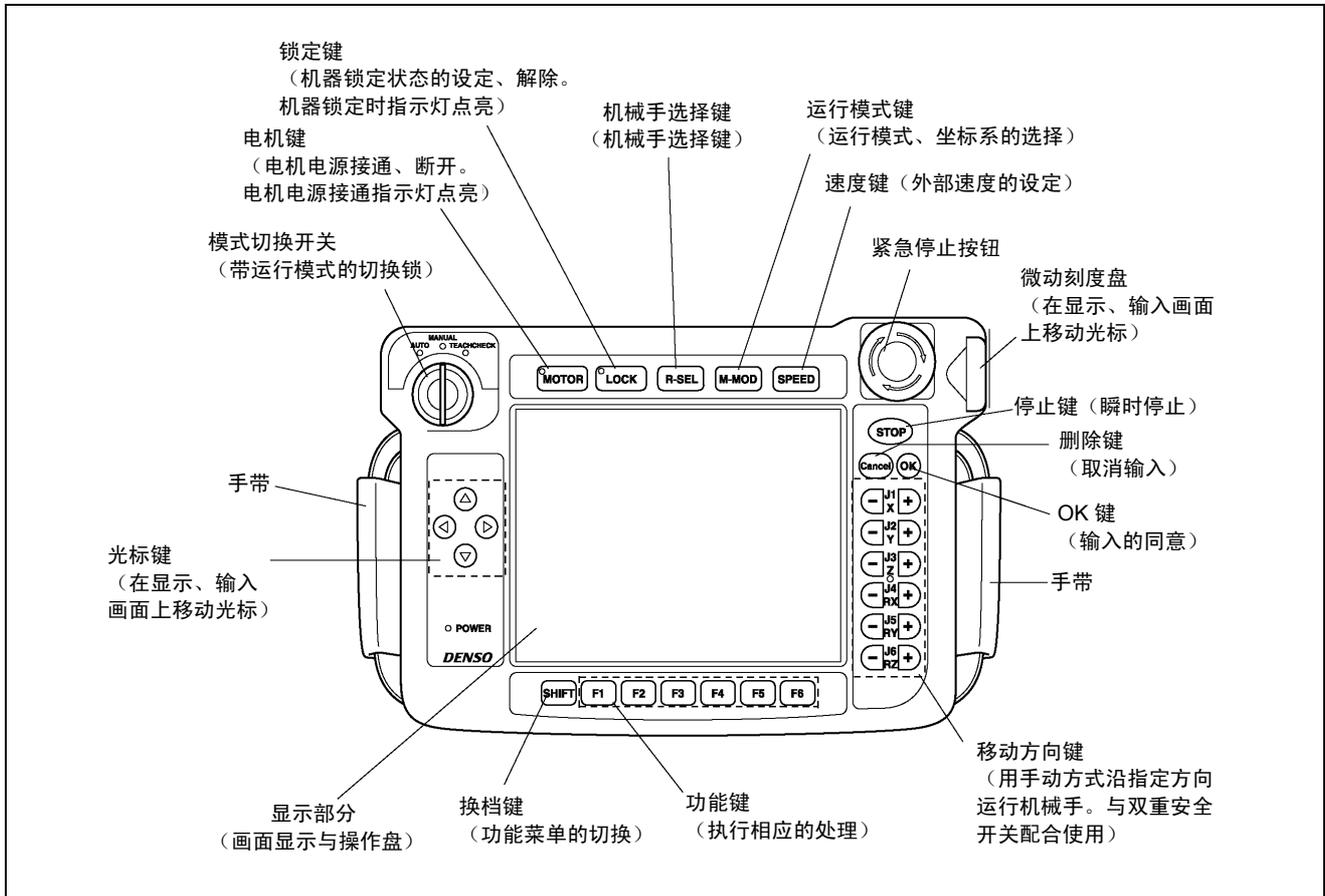
注：双重安全开关，也称为允许开关。

7.1.2 多功能教导器的按键开关的名称

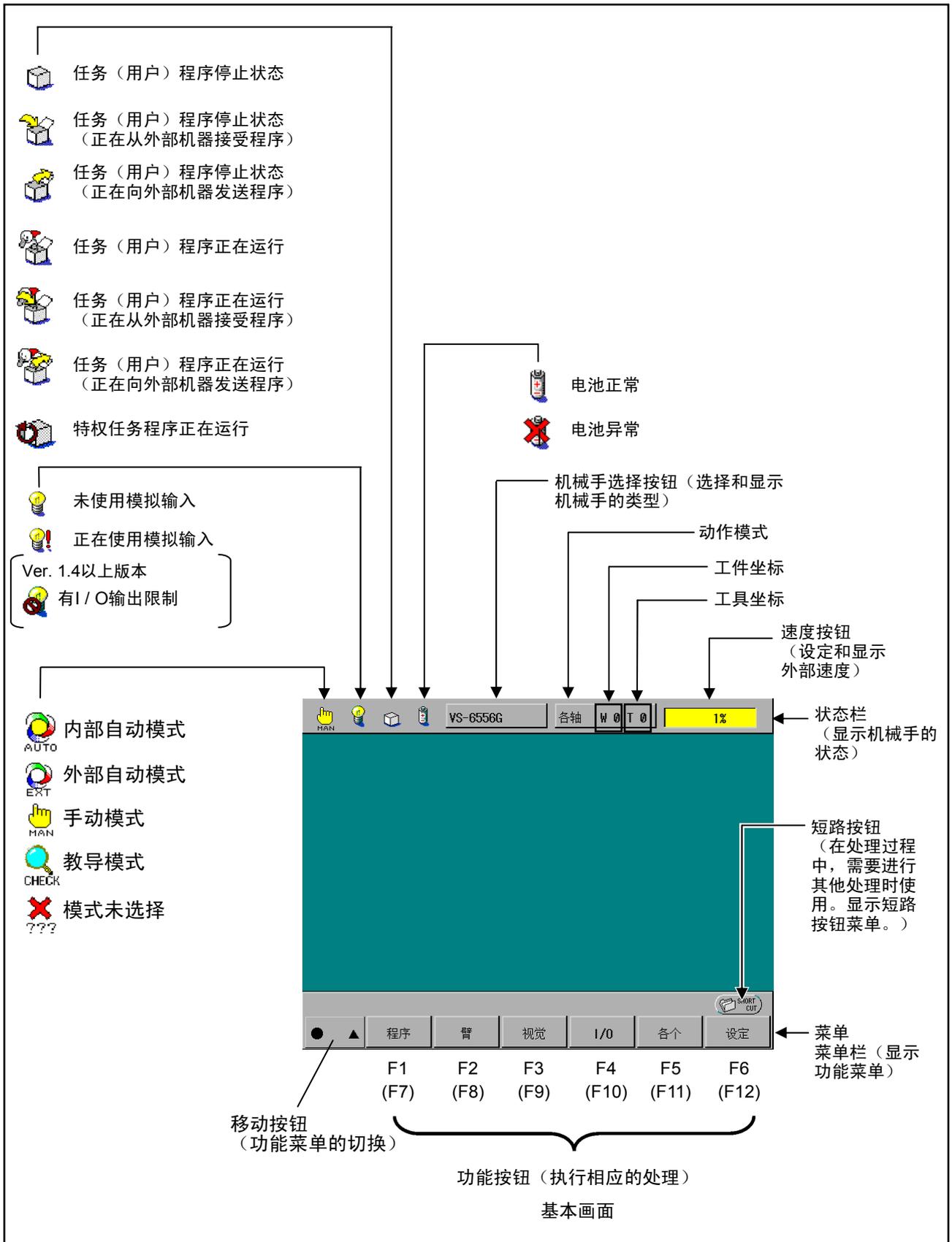
多功能教导器的按键开关等名称如下图所示。

在下一页中就多功能教导器的画面配置和功能以图示进行说明。

为了安全有效地操作机械手，请正确掌握图示的按键、开关所在的位置和大致含义。



多功能教导器的键、按钮及开关的名称



多功能教导器的键、按钮以及开关名称

7.2 动作模式的种类

机械手有 "手动模式"、"教导检查模式"、"自动模式" 的 3 种动作模式。

7.2.1 手动模式

是使用多功能教导器、小型教导器手动运行机械手的模式。

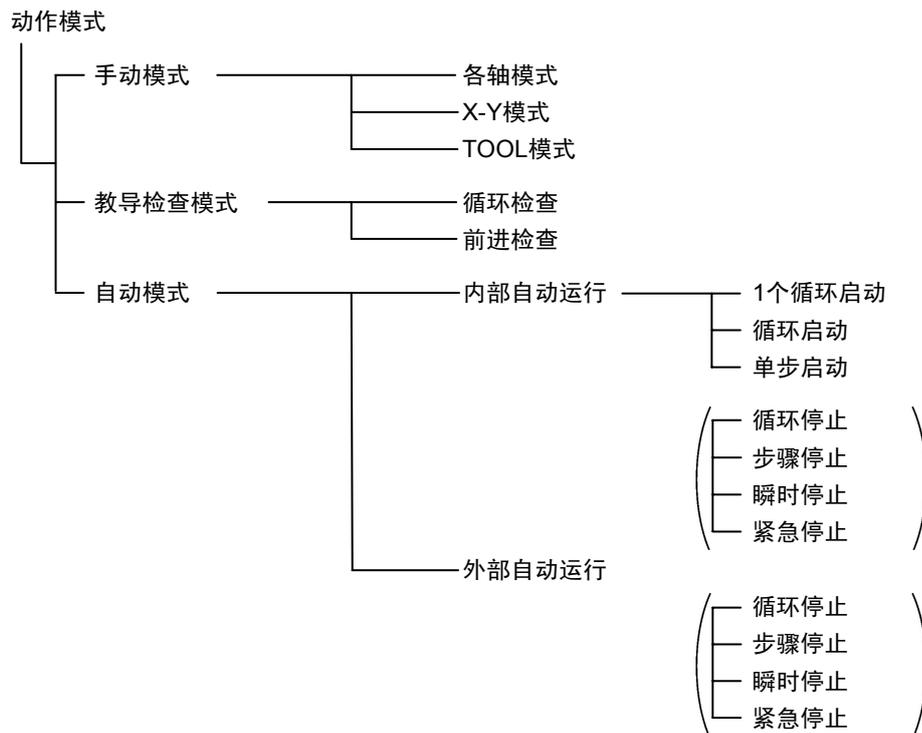
7.2.2 教导检查模式

是在教导之后使用多功能教导器为了最终检查程序中有没有错误、带限制的进行自动运行的模式。

7.2.3 自动模式

是进行自动运行的模式。

多功能教导器与小型教导器对应上述 3 种动作模式的全部。



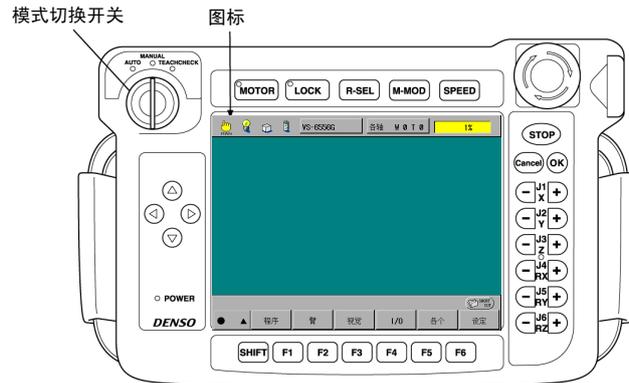
上述的 3 种动作模式，可以锁定（称为机器锁定）机械手的动作，不实际运行机械手，只是用机械手控制器进行仿真动作。此外,在机器锁定时，能够限制 I/O 的输出。关于详细内容，请参照操作指南 "5.5 I/O 信号的显示和机械手动作的仿真"。

7.3 动作模式的切换方法

在教导的情况下，需要设定为手动模式 (MANUAL)。

7.3.1 动作模式的切换方法

将模式切换开关对合所希望的模式。



表示在状态栏的左端部位选择的模式的图标会被显示。



7.3.2 动作模式和自动允许输入信号的相关性

需要对准使用的动作模式，还要变更自动允许输入信号（专用输入信号）的 ON / OFF 状态。

关于输入信号的接线，请参照 "5.4 自动运行所需要的接线" 确认。

动作模式	自动允许输入信号
手动模式	OFF (释放)
教导检查模式	OFF (释放)
自动模式	ON (短路)

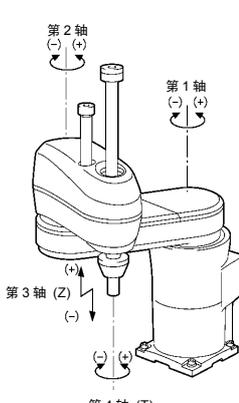
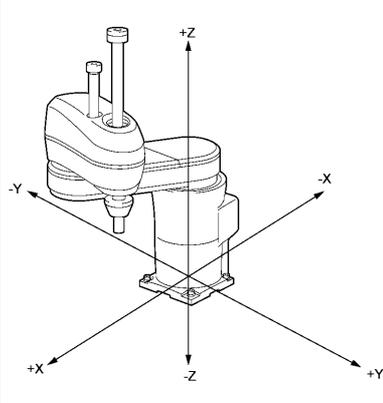
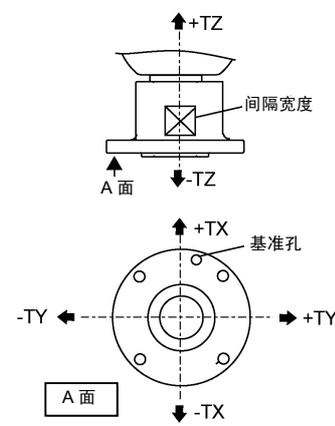
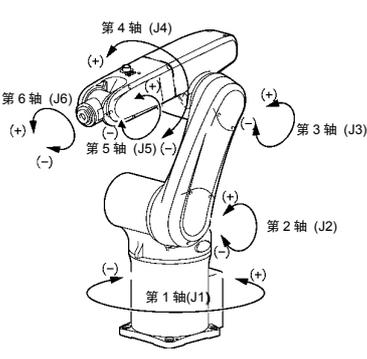
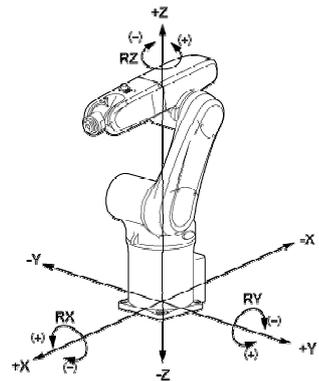
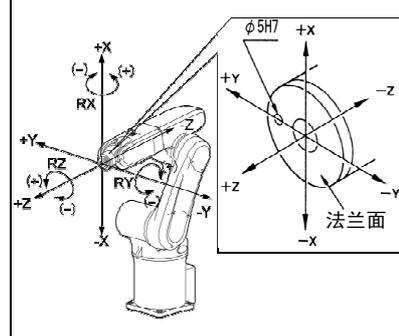
在状态不一致时，会发生 ERROR21 F2 (自动允许输入) 或 ERROR21 F3 (自动允许断开)，不能进行操作。

7.4 手动模式的种类

手动模式中，有各轴模式、X-Y 模式、TOOL 模式的 3 种。任一模式都可以利用多功能教导器或小型教导器的手动模式操作机械手。

注：为了手动操作机械手，需要专用输入的自动允许输入信号 OFF（释放）。

7.4.1 各轴、X-Y、不同工具模式下的机械手操作

	<各轴模式>	<X-Y 模式>	<工具模式>
动作	4个轴分别单独动作。	沿直角坐标直线动作。	沿第4轴的直角坐标直线动作。
4轴机械手	 <p>第2轴 (+) (-) 第1轴 (+) (-) 第3轴 (Z) (+) (-) 第4轴 (T) (+) (-)</p>	 <p>+Z -Y -X +X -Z +Y</p>	 <p>+TZ 间隔宽度 A面 -TZ +TX 基准孔 -TY +TY A面 -TX</p>
动作	6个轴分别单独动作。	沿直角坐标直线动作。	机械手凸缘面的X、Y、Z沿直角坐标直线动作。
6轴机械手	 <p>第4轴 (J4) (+) (-) 第6轴 (J6) (+) (-) 第5轴 (J5) (+) (-) 第3轴 (J3) (+) (-) 第2轴 (J2) (+) (-) 第1轴 (J1) (+) (-)</p>	 <p>+Z RZ (+) (-) -Y -X +X -Z +Y RX (+) (-) RY (+) (-) RV (+) (-)</p>	 <p>+X RX (+) (-) +Y RZ (+) (-) +Z -Y -X 法兰面 φ5H7 +X -Z -Y +Z -X</p>

7.4.2 手动模式的切换方法

△注意： 运行开始时，请将速度设定在 20%以下进行操作。如果从开始就实行高速动作，则机械手可能发生误动作与周围的设备发生碰撞。

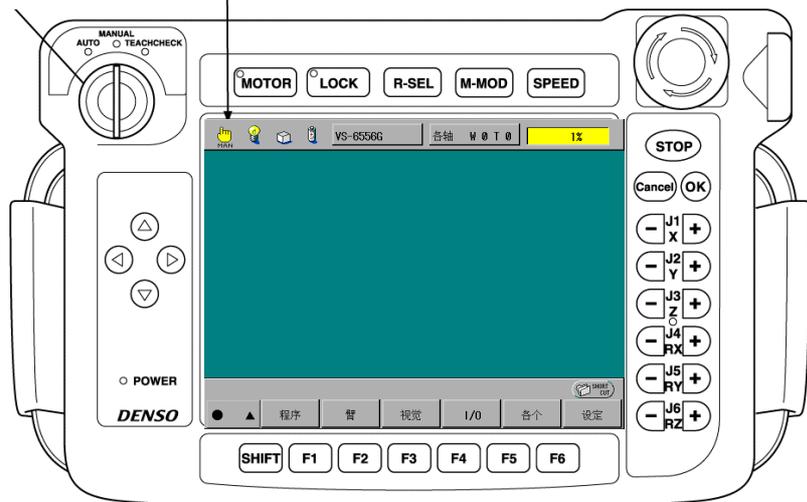
■ 使用多功能教导器的情况

步骤 1

将模式切换开关对准 [MANUAL]。

模式切换开关

图标



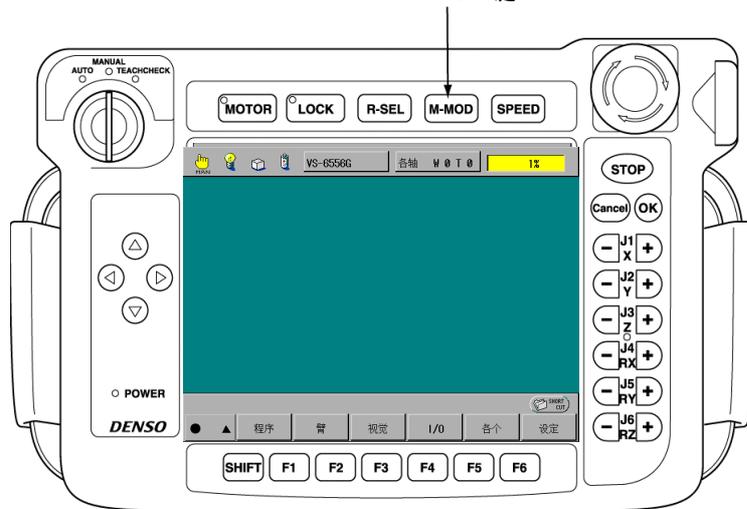
步骤 2

请按压 [MOTOR]，接通电机电源。

步骤 3

按压 [M-MOD]。

M - MOD 键



如下一步骤所示的 [选择动作模式] 视窗会被显示。

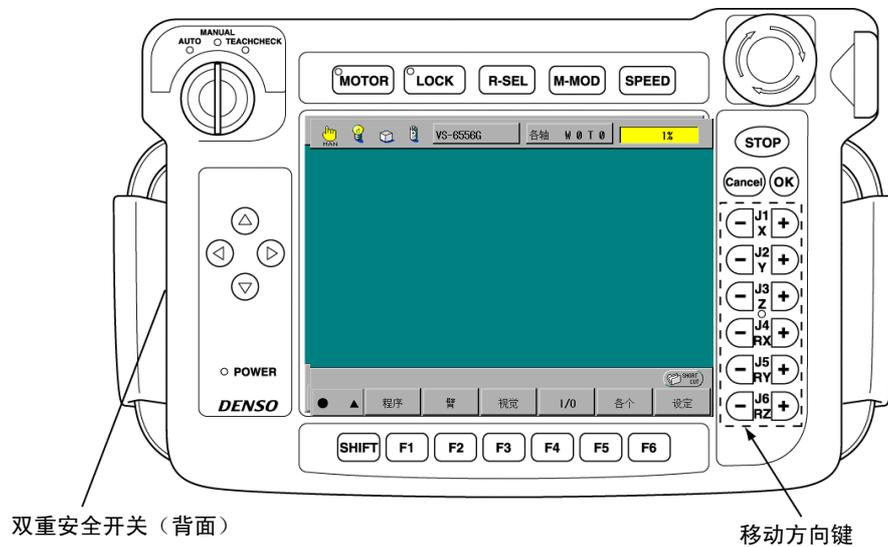
步骤 4

使用光标键、或者直接触摸画面选择模式，按压 [OK]。
在状态栏的模式显示部分选择的动作模式会被显示。



步骤 5

一边按压双重安全开关一边按压任意一个移动方向键，运行机械手臂。
有关移动方向键和轴的动作关系，请参照 7.4.1 各轴、X-Y、不同工具模式下的机械手操作。



7.5 以手动运行模式运行机械手

接通机械手控制器和电机的电源，使用多功能教导器用手动模式运行机械手。

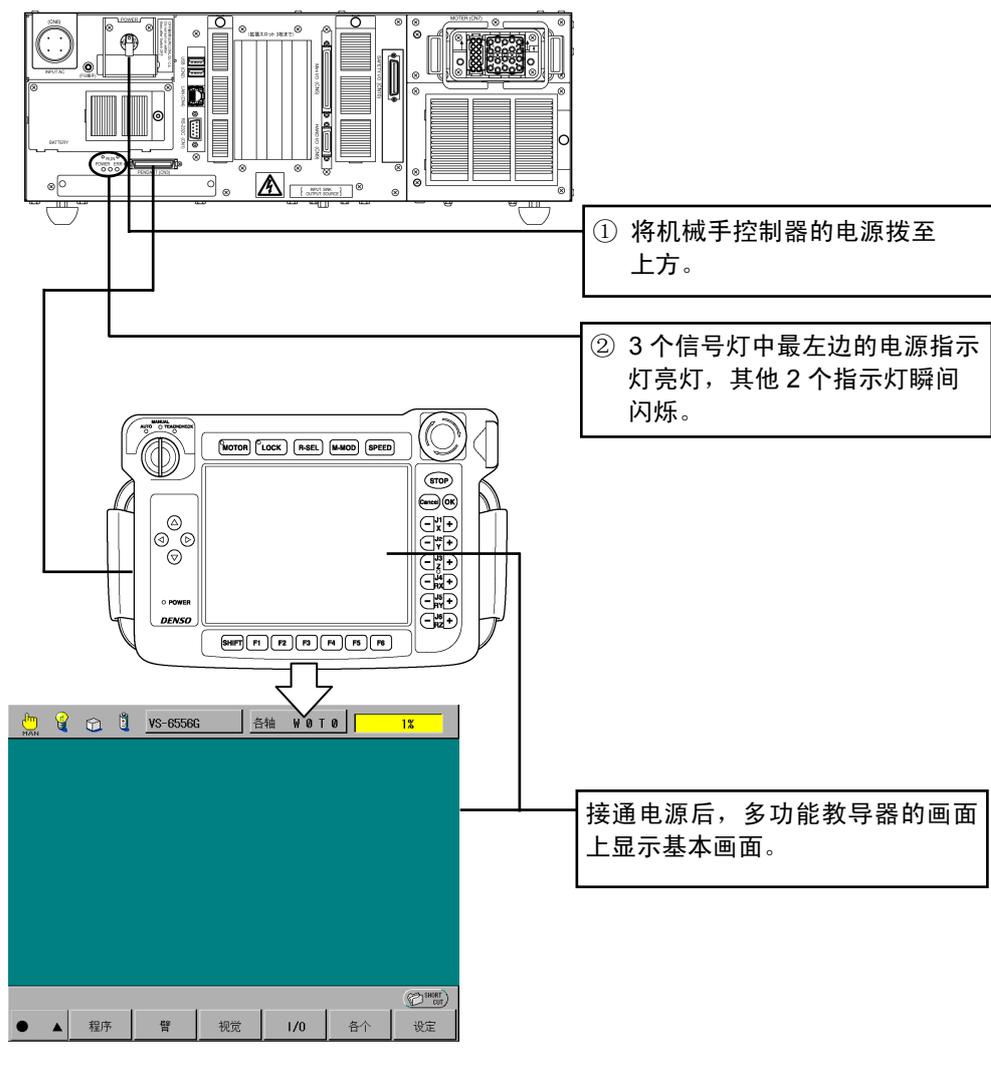
步骤 1

确认安全

- 机械手的安装状况正常。
- 确认在机械手的动作范围内没有人员。

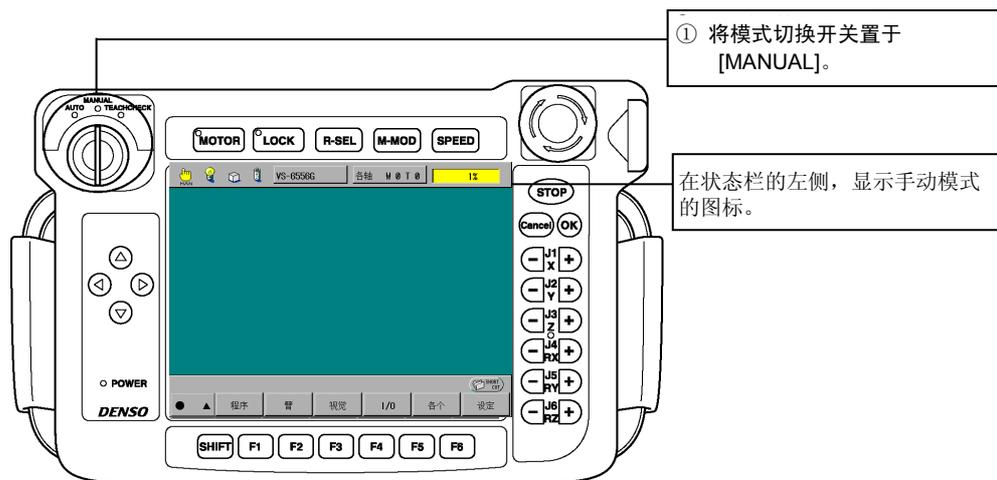
步骤 2

接通机械手控制器的电源



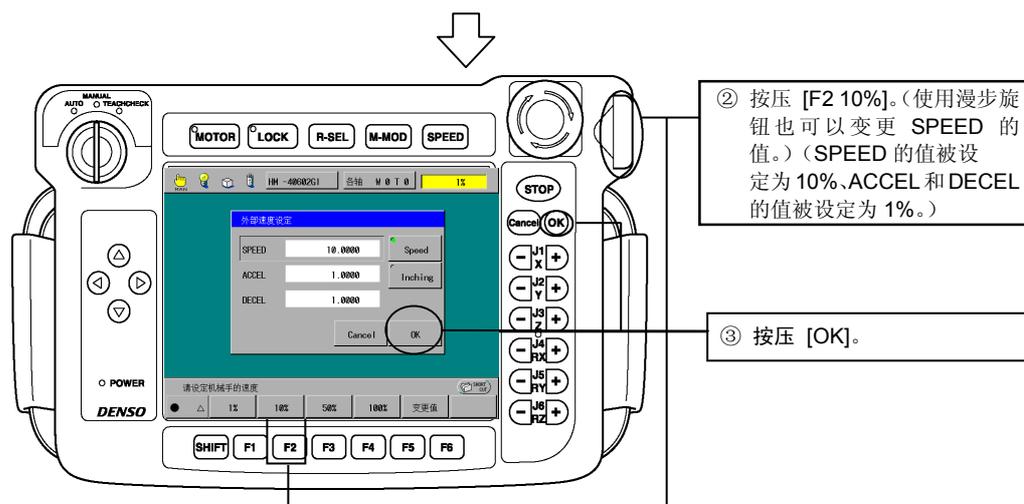
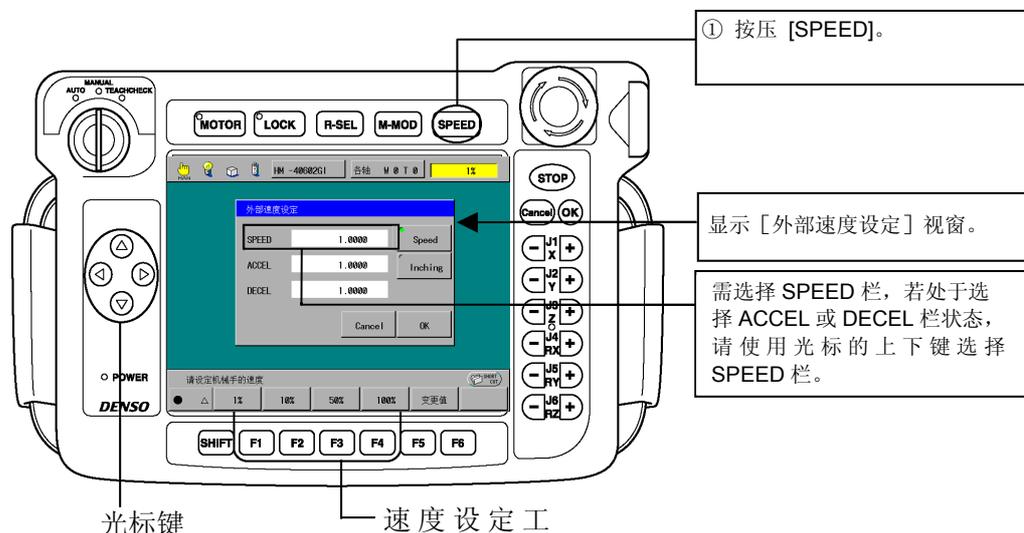
步骤 3

置于手动模式



步骤 4

设定速度、加速度



★参考★

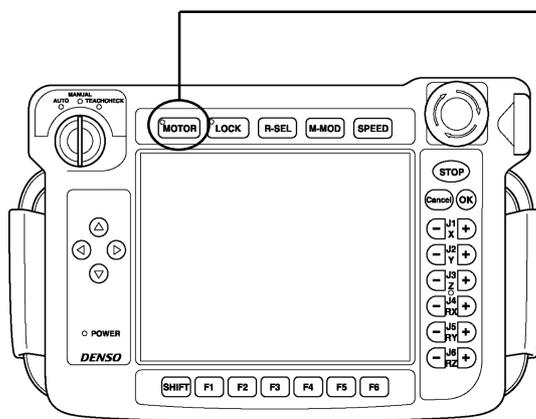
开始时，为了安全起见缓慢运行，需预先置于该位的数值。熟悉后可重新输入该值。



速度显示为 10%。

步骤 5

接通电机的电源



① 按压 [MOTOR]。
接通电机电源，[MOTOR] 指示灯亮灯。

步骤 6



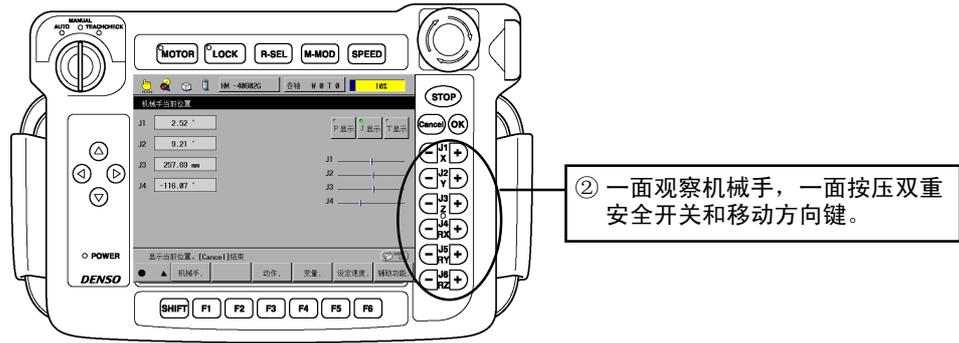
用手动模式实际动作机械手的各手臂

如果进行该操作，则机械手动作。操作人员要远离机械手的动作范围。



① 按压 [F2 臂]。





② 一面观察机械手，一面按压双重安全开关和移动方向键。

4 轴机械手和 6 轴机械手的对应轴分别根据 J1~J4 或 J1~J6 的移动方向键的机械手操作而运行。在 [机械手当前位置] 视窗，显示各轴的角度。

步骤 7

执行 CAL（校准）的（D 系列及 XYZ 型机械手）

所谓 CAL，是指接通机械手控制器的电源之后，为了对机械手的当前位置进行确认，让所有的轴进行轻微动作。

以下说明 CAL 的步骤。

★参考★

除 XYZ 系列，在-E / -F / -G 系列及 VM-6083D、VM-60B1D 机械手中，不需要步骤 7 的 "CAL 执行"。

※ 即使执行 CAL，也没有问题。

除上述以外的 D 系列机械手及 XYZ 系列机械手，必须执行 CAL。



如果进行该操作，则机械手动作。请操作人员远离机械手的动作范围。确认机械手的动作范围内没有障碍物之后再执行 CAL。

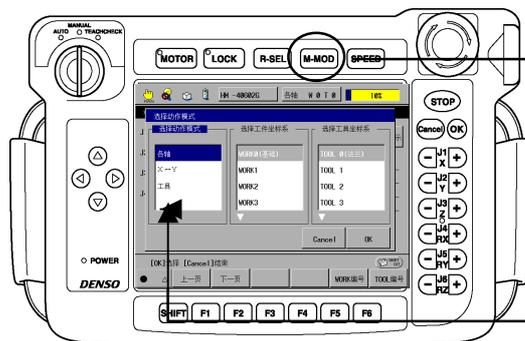


① 在显示 [机械手当前位置] 视窗的状态下，按压 [F6 辅助功能.]。



步骤 8

选择动作模式，手动运行



① 按压 [M-MOD]。

显示 [选择动作模式] 视窗。

步骤 8 (续)

此时，以 X-Y 模式试运行机械手。



② 在 [选择动作模式] 视窗上选择 "X-Y" 栏。(使用上下光标键漫步旋钮进行选择。)

③ 按压 [OK]。出现基本画面之后，按压 [F2 Arm]。



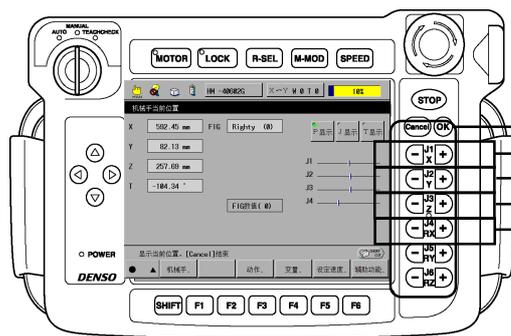
在状态栏上显示为X-Y。

④ 按压P显示。或按压SHIFT键，将F1置于P显示，按压F1键。(这是在X-Y模式下动作所必需的。)

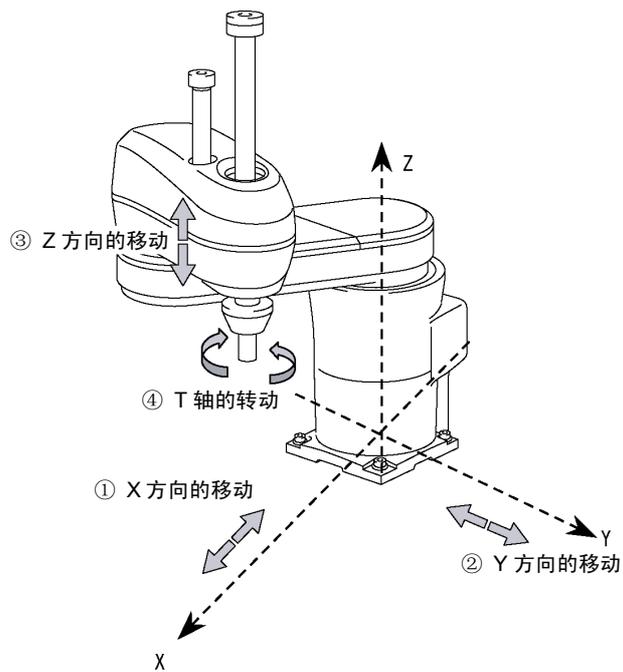


P显示亮灯，当前位置的显示形式变为 [P显示]。

⑤ 按压移动方向键和双重安全开关，让机械手运行。



- 移动方向键
- ① X方向的移动
 - ② Y方向的移动
 - ③ Z方向的移动
 - ④ T轴的旋转



第 8 章 教导

8.1 教导定义

为了实际运行机械手而指定位置的操作称为教导。在程序中也可以用常量指定位置，但为了使工件和机械手的相对位置关系正确吻合，需要在现场实际运行机械手，然后决定位置。为此，在程序中将位置作为变量记述，以教导方式指定变量的具体值。

8.2 可以进行教导的全局变量

所谓 "变量"，是将程序中使用的数据暂时存放。可以进行教导的全局变量有以下 3 种。

- P 型：位置型变量 (X, Y, Z, RX, RY, RZ, FIG) [6 轴]
(X, Y, Z, T, FIG) [4 轴]
- J 型：关节（轴）型变量 (J1, J2, J3, J4, J5, J6) [6 轴]
(J1, J2, J3, J4) [4 轴]
- T 型：齐次变换型变量 (X, Y, Z, 0x, 0y, 0z, Ax, Ay, Az, FIG) [6 轴 / 4 轴]

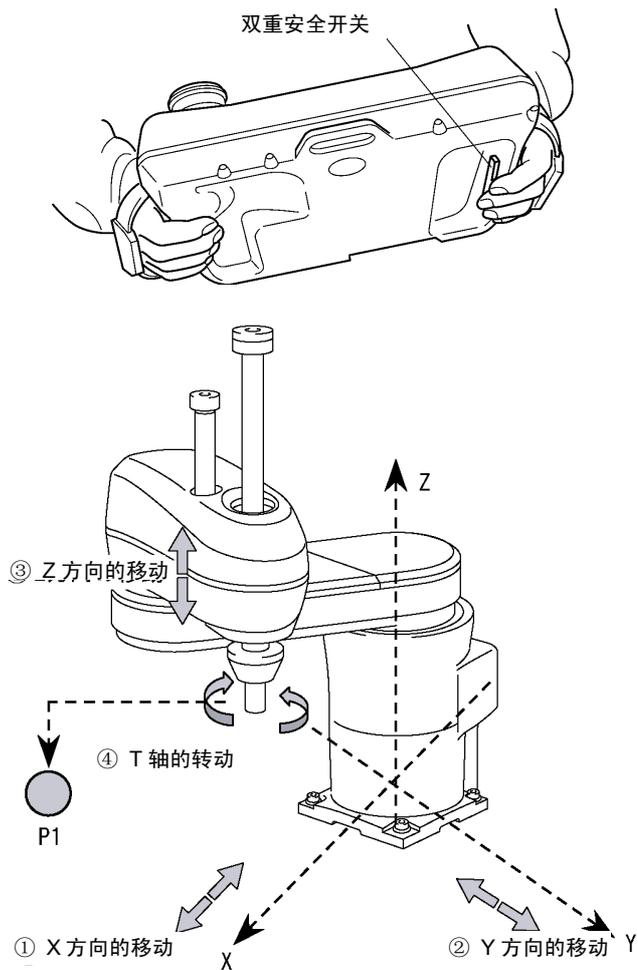
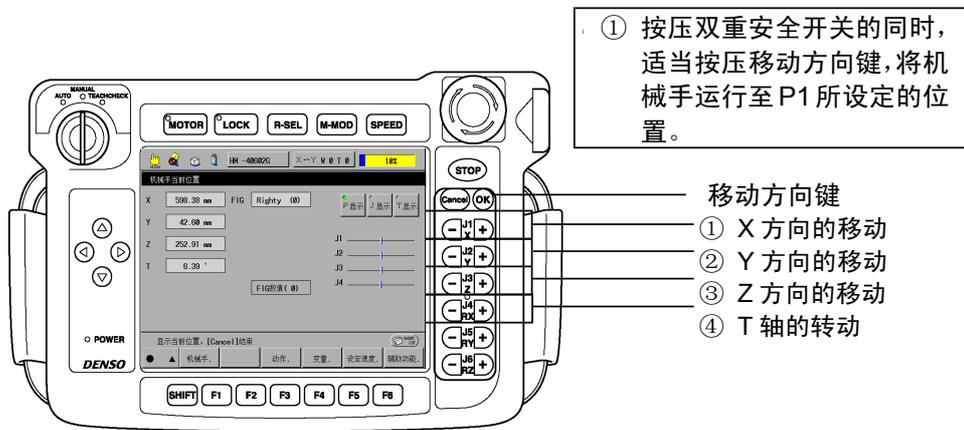
各变量最多可设定 32766 个，根据控制器的存储容量的使用状况，可设定的最大数将减少。

8.3 对位置变量的教导方法

在此，就机械手在手动模式下对位置变量 P1 和 P2 的值进行教导的方法进行说明。

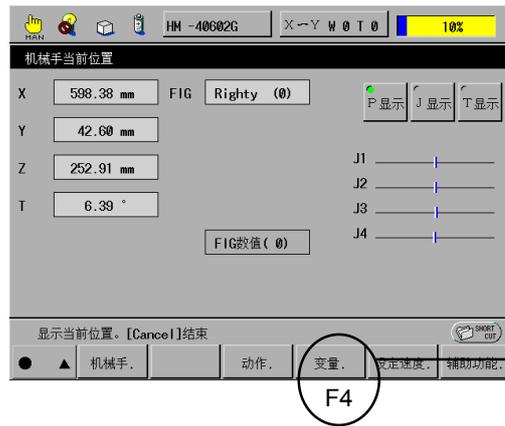
步骤 1

教导机械手的位置 (P1)



步骤 2

将教导的值保存在 [变量名 P1] 中



① 按压 [F4 变量.]。



② 在 [选择变量类型] 视窗，选择变量类型。

在此，因为要将位置保存在P变量上，所以按压 [F4 P.位置]。
(也可以触摸位于视窗中的 [P.位置]。)



显示 [位置型 (Position)] 视窗。

③ 使用光标键或漫步旋钮，选择 "变量名P1" 栏。

4轴机械手的情况，在 [位置型 (Position)] 视窗中，对于一个变量显示 5 种数据，而如果是 "变量名 P1"，则 5 种数据中某一个如果反衬显示，则已选择 "变量名 P1"。



④ 确认 "变量名P1" 已被选择。

⑤ 按压 [F6 位置输入.]。



⑥ 确认系统讯息，如果没有问题则按压 [OK]。

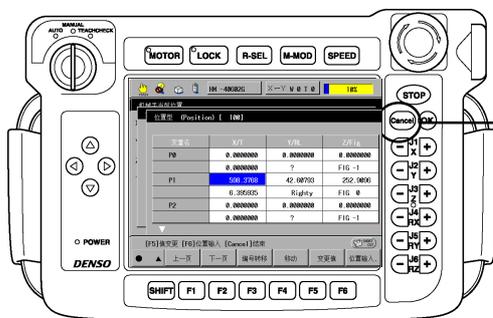




当前位置作为变量P1的值被输入。

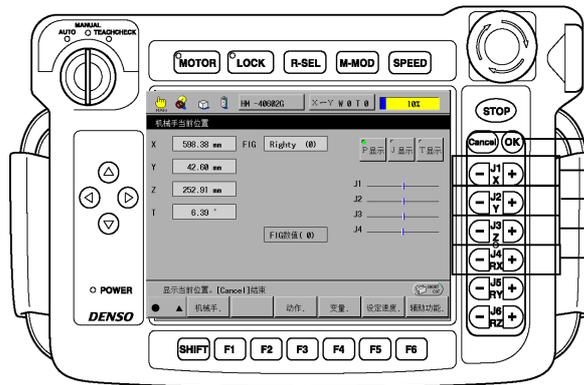
步骤 3

机械手的位置 (P2) 的教导及在 [变量名 P2] 的保存



① 2次按压 [Cancel], 返回 [机械手当前位置] 视窗。

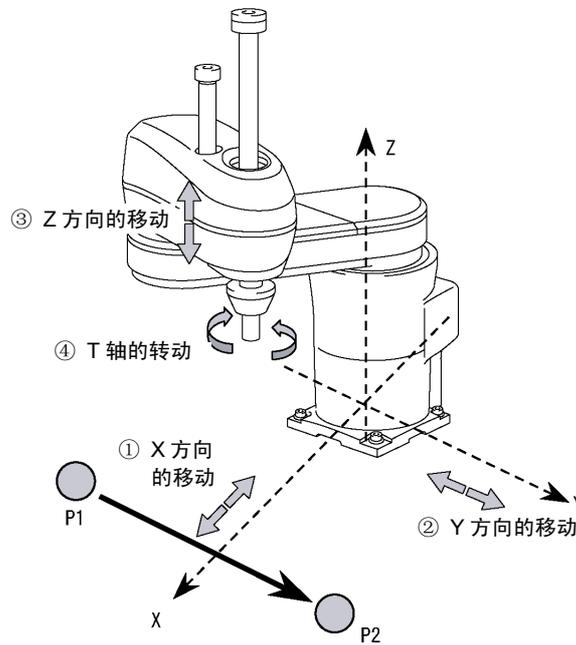




② 按压双重安全开关的同时，适当按压移动方向键，将机械手运行至 P2 所设定的位置。

- 移动方向键
- ① X 方向的移动
- ② Y 方向的移动
- ③ Z 方向的移动
- ④ T 轴的转动

双重安全开关



③ 步骤2 "将教导的值保存在 [变量名P1]"
"D步骤, 将P2的值保存在[变量名P2]"。

至此，P1 和 P2 的教导结束。

8.4 直接移向需要教导的位置变量的方法

在手动模式、教导检查模式下、可以直接移向位置变量上所登录的点。

操作路径： [F2 臂] — [F4 变量.] — [F4 P.位置]

如果按压 [F4 P.位置]，则显示以下的 [位置型 (Position)] 视窗。

将光标对准需要移动的变量编号。



若按压 [F4 移动]，"在变量○○的位置上动作" 这一系统讯息被显示。按压双重安全开关 + [OK] 的过程中，机械手臂向所指定的位置移动。此时，也可以指定 PTP（到达目标位置的动作路径取决于机械手）或 CP 运行（用直线连接到达目标位置的动作路径）。

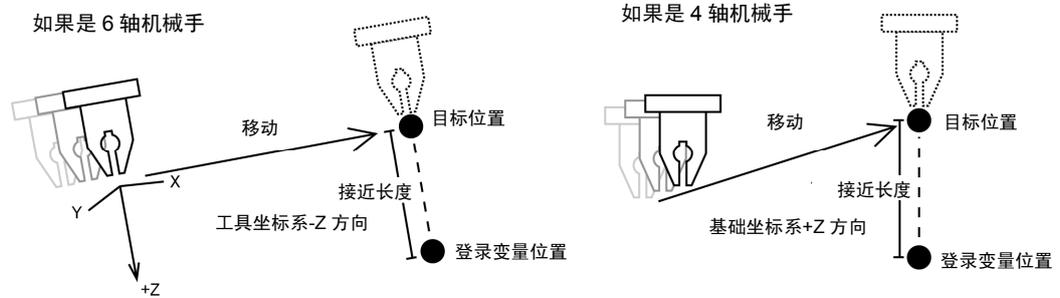
注意：机械手臂移动时，如果释放双重安全开关或 [OK] 按钮，则移动会中途停止。

移动的恢复 (Ver. 2.61 之后版本)

只要松开 [OK] 按钮，机械手即可在向所设定目标位置移动的过程中中途停止。再次按双重安全按钮与 [OK]，机械手即会再次朝目标位置移动。按取消，即会返回变量画面。

8.5 向包含接近长度在内的目标位置移动（Ver. 2.61 之后版本）

您可将目标位置，从已登录的变量，设定为偏移的位置。如果是 6 轴机械手，则向工具坐标系-Z 方向偏移，如果是 4 轴机械手，则向基础坐标系+Z 方向偏移。我们将此偏移量称为接近长度（参照编程手册 I APPROACH 指令）。通过移动至包括接近长度在内的目标位置，可使之通过手动模式轻松地进行更接近于实际程序的工作。



在指定变量移动画面按 [F6 Approach]，指定接近长度的画面即会显示出来。通过小键盘输入接近长度。



将目标位置设定为包含接近长度在内的位置。



第3部

简易编程

第 9 章 程序创建的基础知识

9.1 PAC 语言

记述机械手的动作和操作的编程语言称为机械手语言。

DENSO 机械手所使用的机械手语言，被命名为 PAC (Programming language for Assembly Cell)。PAC 是为了使机械手控制程序的开发和维护变得更为高效而新开发的机械手语言。主要的特征如下所示。

- 它是 JIS 所规定的、产业用机械手语言 SLIM 的高度互换语言。
- 由于它是结构化的编程语言，所以具有易读、易开发和易维护的优势。
- 不仅机械手，而且视觉装置的控制也能用 PAC 进行统一的记述。
- 通过多项任务功能，能够进行高效的程序处理。
- 具备中断处理功能，可以高效地记述对应错误的发生及时刻等所采取的例外处理。

9.2 语句与行

- PAC 语言的程序由若干行构成。
- 在任意的行中都可以记述一个语句。
- 1 行的长度最大为 255 字节。
- 语句是记述通过 PAC 语言处理的最小单位，由一个指令构成。
- 指令由指令的名称和给予指令的讯息（参数）构成。

9.3 姓名

PAC 语言里含有为了识别在程序中的各种要素的规则。在本章中，就该规则进行说明，并对表示指令、变量、函数、标签、程序的名称做了如下规定。

- 名称以字符（半角拉丁字母、没有大小写的区分）或者被规定的符号开头。
- 在名称中可以使用字符、数字和下划线。
- 名称的开头字符必须是字母。
- 不能使用句号、前括号、后括号、空格、冒号、分号、单引号、双引号、星号。
- 不能使用+、-、*、/、(、)等作为运算符使用的字符。
- 为了将名称和其他的语言识别开，在名称与其他语言之间插入空格字符。
- 名称所能使用的字符数最大为 64。

9.4 可以登录的最大程序数

在 1 台控制器上能够登录的程序数有限制。
此外，根据存储器的使用状况，可以登录的程序数有所减少。

区分	文件格式	最大数
PAC程序	***.pac	256个
标题文件	***.h	对准标题文件和操作盘
操作盘文件	***.pnl	
文件夹		256个

9.5 程序构成的概要

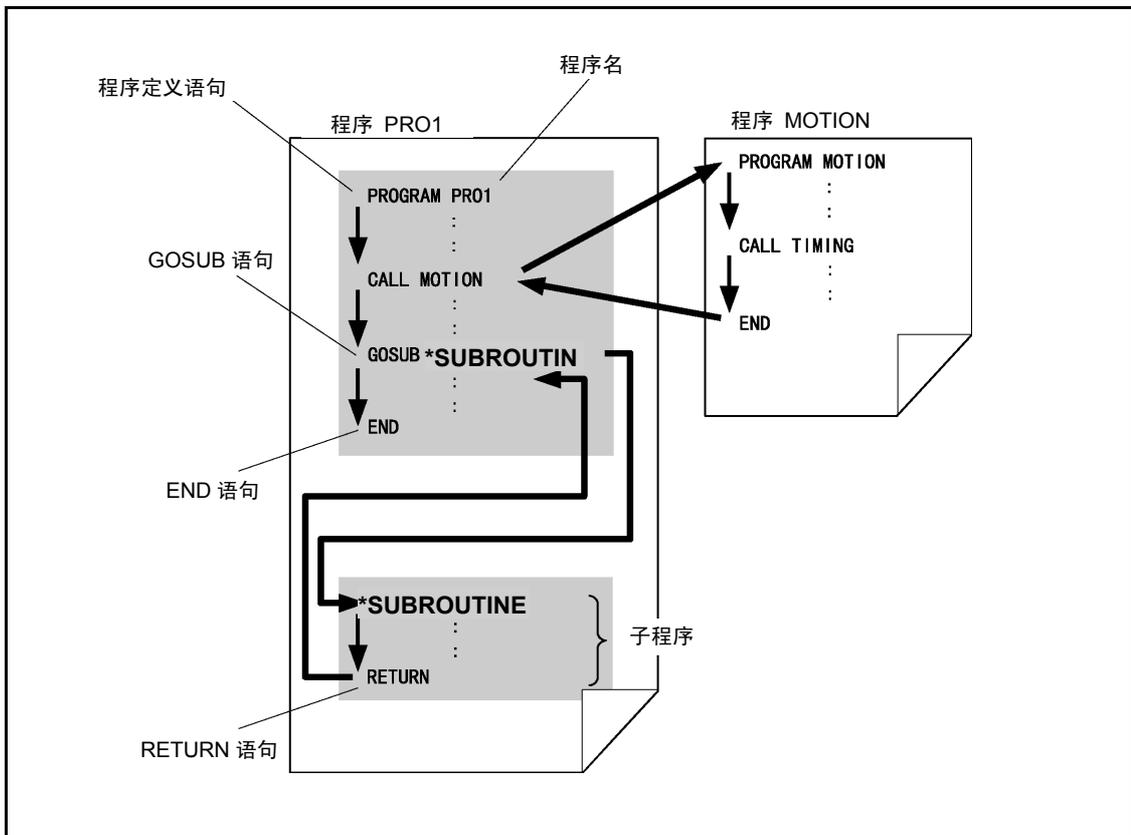
可以在程序中事先抽出循环执行的特定动作部分，视需要随时可调出。

在同一个程序文件中被放置的此部分称为子程序。独立于别的文件，事先作为另外的程序调出的行为称为程序调出。

子程序必须与调出侧的程序在相同的文件中。

调出独立的另外的文件中的程序时，某一个程序可以被多个程序调出，作为通用程序来使用。

将一系列的工作作为子程序或另外的程序总结到一处，由此可以无需多次记述相同内容的程序即可。对于错误记述、削减程序创建时间、或提高程序的可读性等都是有利的。



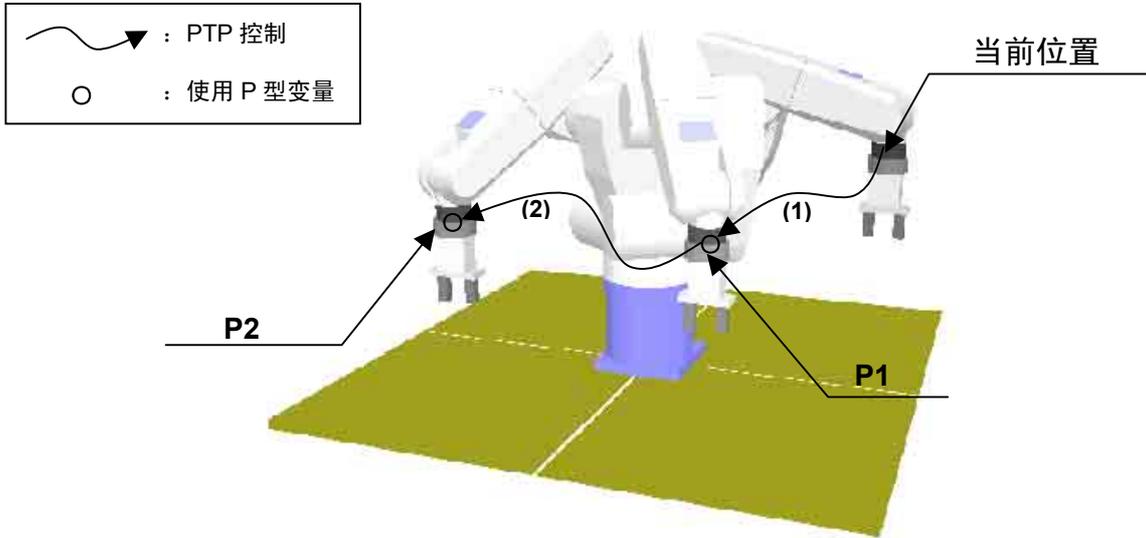
程序与子程序的调用方法的不同

9.6 程序的主要指令

以最简单的运行程序为例，就编写程序所需的最低限度的动作指令进行说明。

9.6.1 程序示例

通过 PTP 控制从当前位置依次向 P1、P2 动作臂。



程序	运行说明
'TITLE "程序PRO1"	'程序标题
PROGRAM PRO1	'程序名的定义
TAKEARM	'获取臂控制权
SPEED 100	'臂移动速度（内部速度）100%
MOVE P, P1	'(1): 移动至指定坐标P1
MOVE P, P2	'(2): 移动至指定坐标P2
END	'程序停止

9.6.2 关于用指令格式处理的特殊符号

就指令的格式解释上所使用的特殊符号 (□、< >、[]) 进行以下说明。

- 表中的 □ 符号表示空格（空白）。
- < > 符号需要叙述。
- [] 符号可以省略，能够按照需要进行叙述。
- 英文字母的大写字符、小写字符不需要分开使用。

9.6.3 程序名的定义（PROGRAM 指令）

说明

程序名称和变量等程序执行时所需要的项目，在执行之前要进行定义。特别是进行程序名称定义时，必须在程序的最初的有效行进行定义。将该语句叫做 PROGRAM 定义语句。

格式

PROGRAM □ <程序名>

注：需要从外部启动时，请将 <程序名> 设置为 PRO <编号>。

9.6.4 获取臂控制权（TAKARM 指令）

说明

在复数任务控制的多项任务控制中，需要对机械手的控制权进行提交。使用让机械手的臂动作的动作命令时，必须使用 TAKEARM 命令获取臂控制权。

格式

TAKEARM

9.6.5 程序的停止（END 指令）

说明

如果执行该指令，则结束程序所控制的动作。

格式

END

9.6.6 臂移动速度的指定（SPEED 指令）

说明

用百分数的形式 (1~100) 指定内部速度。
外部速度与内部速度的关系如下所示。

实际的臂移动速度 (%) = 外部速度 (%) × 内部速度 (%)

※ 所谓外部速度，是指从教导器和 PLC 上所设定的速度。

※ 直至执行下一个 SPEED 命令时，SPEED 命令才生效。

格式

SPEED □ <移动速度>

9.6.7 备注的叙述（REM 指令）

说明

在程序上对语句进行备注。

格式

' [<备注>] （或 REM □ [<备注>] ）

9.6.8 向指定坐标的移动（MOVE 指令）

说明 使机械手从当前位置向目标位置 <姿势 (Pose)> 进行动作。

格式 MOVE □ <插补方法>, [@<通过 (Pass) 开始位移>] □ <姿势 (Pose)> [, <动作选择>]

9.6.8.1 插补方法

在 <插补方法> 中需要选择 P、L、C 3 种。

P: PTP (Point to Point) 控制动作

- 从当前位置向下一个点进行动作时，途中的路径由机械手决定。

L: CP (Continuous Path) 控制动作…直线插补

- 将夹治具的姿势和速度保持一定状态，同时让机械手在当前位置和下一个点之间进行直线动作。

C: CP (Continuous Path) 控制动作…圆弧插补

- 机械手前端在连结 3 个点的圆弧上动作。

9.6.8.2 姿势 (Pose)

在 <姿势 (Pose)> 上，输入作为移动目标的动作位置的变量名称。可以使用变量的位置型 (P 型)、关节型 (J 型)、变压器型 (T 型)。此外，变量的构成会因机械手的轴数而异。

机械手	变量名	变量的构成						
4 轴机械手	P 型	X	Y	Z	T	FIG		
	J 型	J1	J2	J3	J4			
6 轴机械手	P 型	X	Y	Z	RX	RY	RZ	FIG
	J 型	J1	J2	J3	J4	J5	J6	

也可以输入各个坐标值的数值，但在实际中经常使用变量 P○○或变量 J○○。

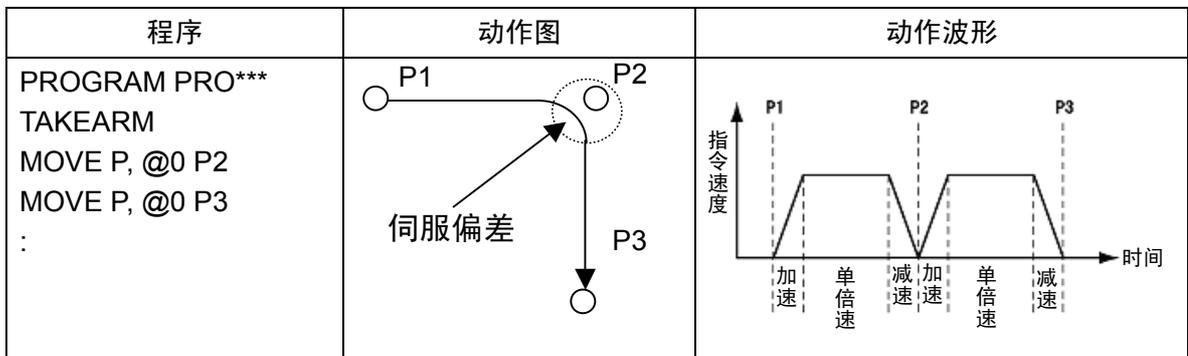
应用示例

```
PROGRAM PRO5 '程序名的定义
  TAKEARM '获取臂控制权
  SPEED 80 '内部移动速度 80%
  MOVE P, P1 '以 PTP 插补方式向变量 P1 移动
  MOVE L, P2 '以 CP 插补方式向变量 P2 移动
  MOVE L, P3 '以 CP 插补方式向变量 P3 移动
END '程序停止
```

9.6.8.3 通过 (Pass) 开始位移

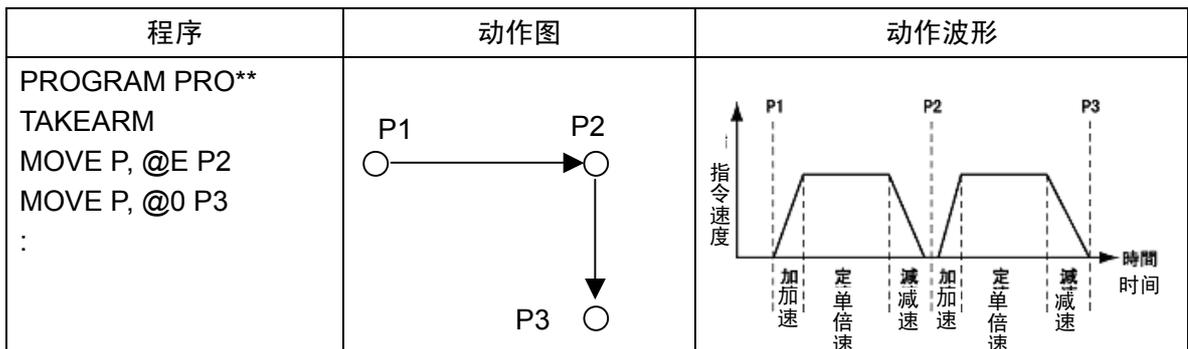
<通过 (Pass) 开始位移> 的值是以指定坐标 (姿势 (Pose)) 为中心的球的半径, 如果动作指令值进入其中, 则移向下一个控制。用 mm 单位指定。换言之, 表示动作的点的 "固定方法"。从一个动作移向下一个动作时, 有以下的方式。

末端动作 (@0 或省略时)



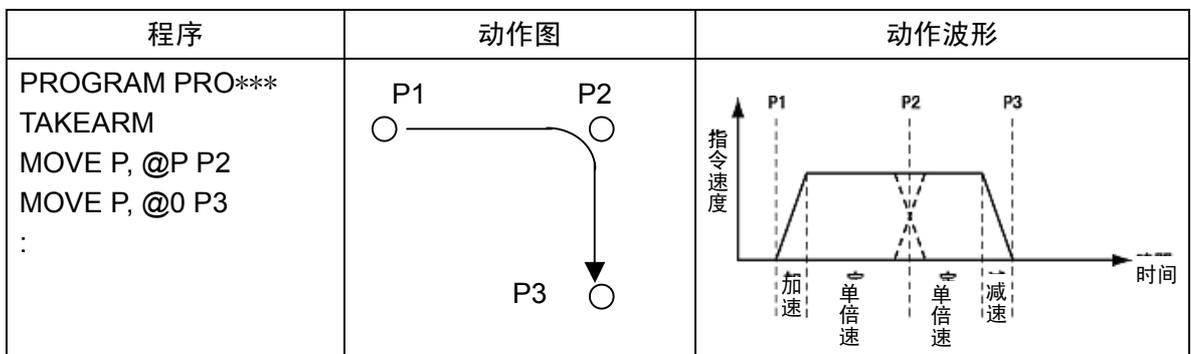
移动至教导的动作位置 (P2), 至伺服系的指令值与目标位置相一致时, 判断到达目标位置的动作被称为结束动作。此外, 还将该点称为 "End 点"。

编码器值的确认动作 (@E)



向被教导的动作位置 (P2) 移动、编码器值对于目标位置进入指定脉冲 (初始值 20) 以内时, 则判断为到达目标位置, 将该动作叫做编码器值的确认运行。停止精度得到提高, 为了消除伺服偏差, 要与末端动作进行比较, 需要一定的充裕时间。

通过 (Pass) 动作 (@P)



将通过被教导的动作位置 (P2) 附近的动作称为通过 (Pass) 动作。此外, 还将该点称为 "Pass 点"。

指定方法

种类	叙述方法	含义
结束动作	省略	作为默认值@0 处理。
	@0	在到达目标位置（指定坐标）的时刻，电机指令值移向下一个动作。是一般的末端动作。
编码器值 确认动作	@E	在到达目标位置（指定坐标）的时刻，编码器值移向下一个动作。机械手确实停止运行。
通过 (Pass) 动作	@P	在动作目标位置附近（半径由控制器自动设定）进行通过动作。是一般的通过 (Pass) 动作。
	@1~@255	从目标位置到达指定的半径 (1~255mm) 时，指令值移向下一个动作。 注意：半径值是通过 (Pass) 起始的大致目标值，而不是保修值。

9.6.8.4 动作选择

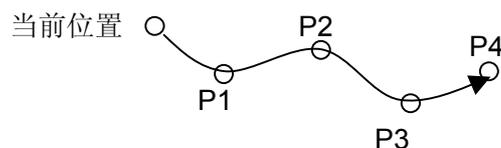
在动作选择中有 SPEED、ACCEL、DECEL。动作选择指定，只对实行的 MOVE 等动作命令有效。

动作选择	含义
SPEED (或标记为 S)	指定移动速度。
ACCEL	指定加速度。
DECEL	指定减速度。

9.6.8.5 其他输入示例

也可在 1 行内记录连续的点的动作。

MOVE P, @P P1, @P P2, @P P3, @E P4



注意点：“单步骤”是到P4的全部动作。所以，不能在P1和P2等途中的点停止。（单步运行，步骤停止时）

应用示例

MOVE L, P1, SPEED = 100	'移动至P1的坐标（CP控制，内部速度 = 100%）。
MOVE P, @30 P2, P3, S = 80	'依次移动P2 (@30)、P3的坐标（PTP控制，内部速度 = 80%）。
MOVE L, @20 P4, @50 P5, @100 P6	'依次动作P4 (@20)、P5 (@50)、P6 (@100) 的坐标（CP控制）。
MOVE L, @P P[6 TO 15], P16	'从P[6] 到P[15] 依次以通过 (Pass) 动作移动，动作至P16的'坐标（CP控制）。
MOVE C, P1, @P P2	'通过P1进行移动至P2的圆弧插补动作。通过P2进行'通过 (Pass) 动作，移向下一个控制。

9.7 向 Z 轴方向移动 (APPROACH , DEPART)

机械手对产品进行抓取和放置时，如果从任何位置直接移动，则会使产品和机械手夹治具与其他的障碍物发生碰撞。在这种情况下，最好是将机械手夹治具移动至产品的正上方、让其从此处进行直线下降和上升。在此，介绍 Z 轴方向专用动作的命令。

9.7.1 向夹治具方向的接近运行 (APPROACH 指令)

功能

靠近基准动作点的、沿Z轴上方向指定的距离点（接近点）的动作。

格式

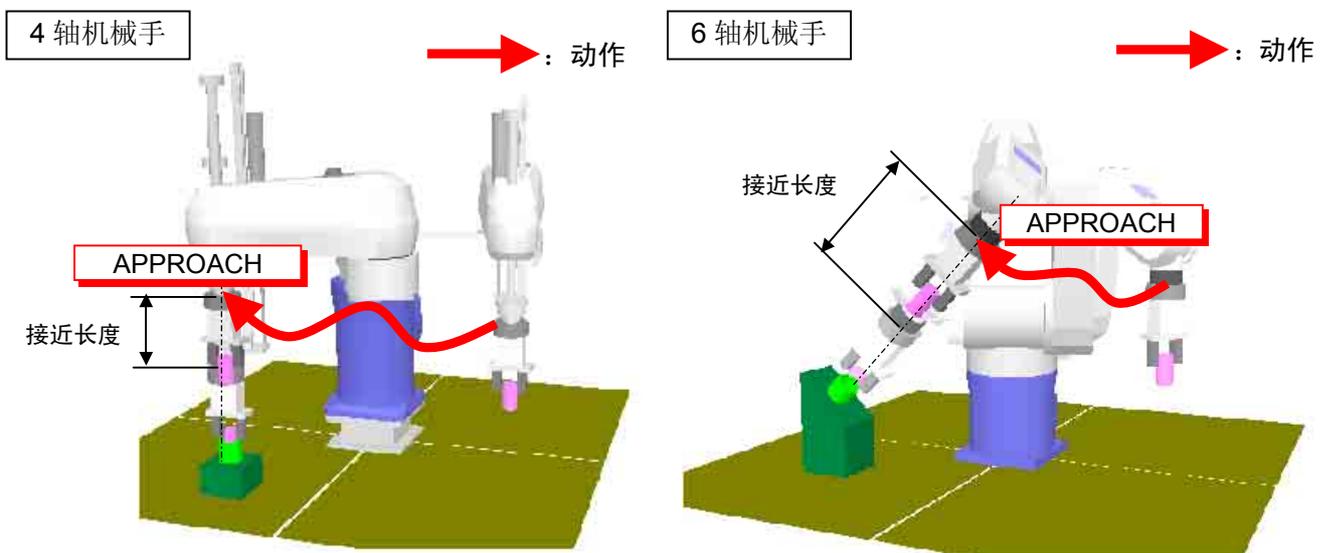
APPROACH [□] <插补方法>, <基准位置>, [@<通过 (Pass) 开始位移>]
<接近长度> [, <动作选择>]

说明

- ① 在 <插补方法> 上可以选择 P (PTP 控制)、L (CP 控制) 中的一个。
- ② 在 <基准位置> 上可以使用 P 型、J 型、T 型。
- ③ 根据机械手的型号，接近方向有以下的不同。
 - 4 轴：从 <基准位置> 沿基准坐标系的 +Z 轴方向、向 <接近长度> 离开的位置移动。
 - 6 轴：从 <基准位置> 沿工具坐标系的 -Z 轴方向、向 <接近长度> 离开的位置移动。
- ④ @ <通过 (Pass) 起始位移>、<动作选择> 的内容与 MOVE 命令相同。

应用示例

APPROACH P, P1, @P 50 '通过 (Pass) 传输动作、以 PTP 控制向变量 P1 的 50mm 上方的点移动。



9.7.2 向夹治具方向的回避移动（DEPART 指令）

功能 从当前位置向仅从Z轴上方向指定的距离离开的点（起程点）动作。

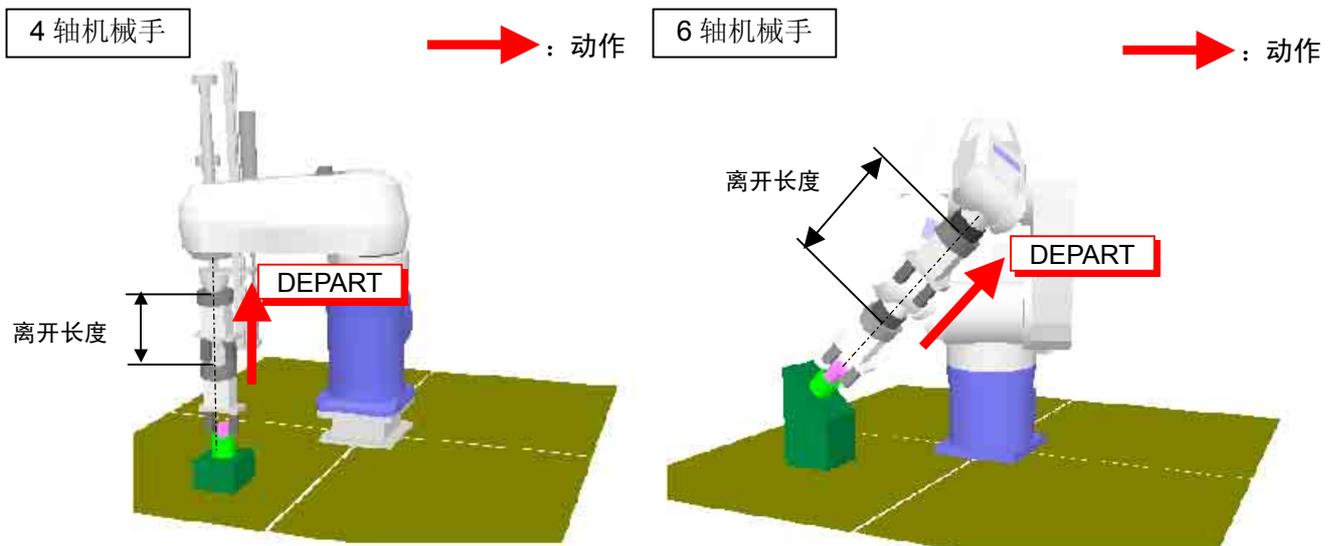
格式 DEPART [□]<插补方法>, [[□]@<通过 (Pass) 开始位移>] [□]<起程长度>
[, <动作选择>]

说明

- ① 在 <插补方法> 上可以选择P（PTP控制）、L（CP控制）中的一个。
- ② 根据机械手的型号，起程方向有以下的不同。
4轴：从当前位置向基准坐标系的 +Z 轴方向移动 <起程长度> 量。
6轴：从当前位置向工具坐标系的 -Z 轴方向移动 <起程长度> 量。
- ③ @<通过 (Pass) 起始位移>、<动作选择> 的内容与 MOVE 命令相同。

应用示例

DEPART L, @P 50 在当前位置的 50mm 上方通过 (Pass) 传输动作，以 CP 控制方式移动。

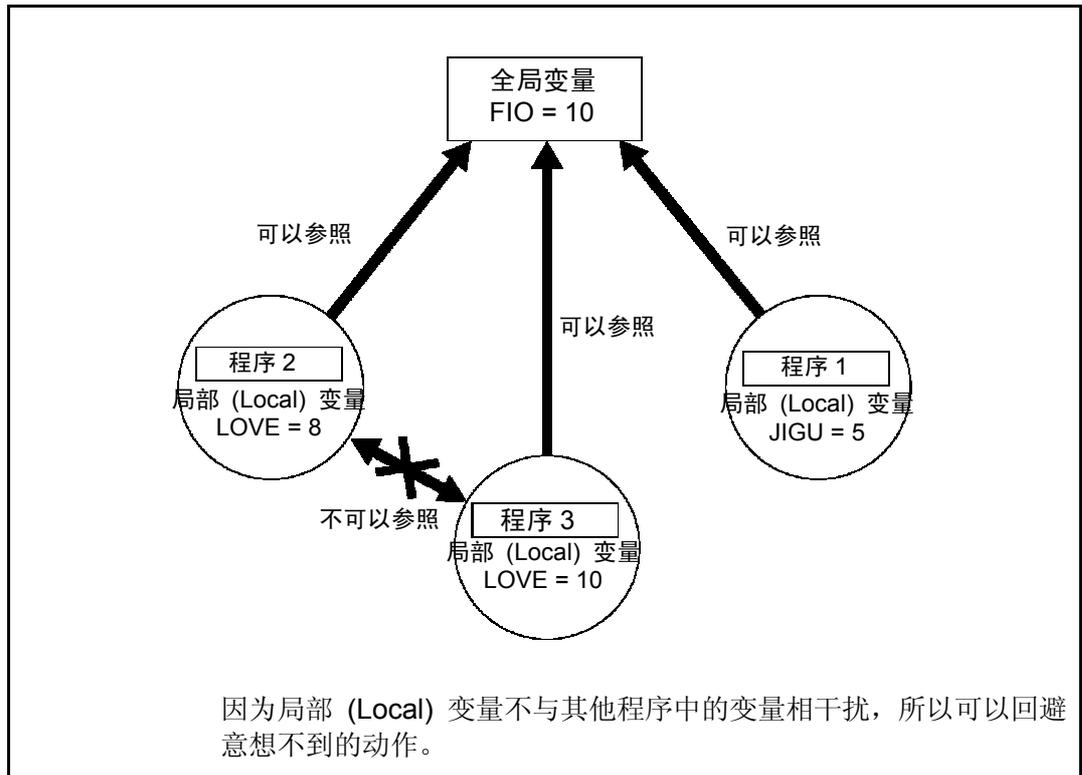


9.8 变量的概要与性质

将程序中使用的数据进行暂时存放的数据称为变量。分为全局变量和局部 (Local) 变量。

全局变量是从任何一个程序 (任务) 都能共同使用的变量。

局部 (Local) 变量是只在一个程序中有效的变量。在同时被执行的其他程序中, 即使有名称相同的局部 (Local) 变量, 也会在各自所属的程序中运行, 相互不影响。



全局变量和局部 (Local) 变量

●全局变量的性质

- ① 在所有的程序上均可以参照 (共有)
- ② 不需要类型定义
- ③ 可登录宏名

●局部 (Local) 变量的性质

- ① 不需要考虑与其他的程序发生干扰
- ② 在编译程序时, 数值被初始化
- ③ 可自由决定变量名 (最大 32 个字符)
- ④ 最大可以进行三维的排列定义 (排列的要素数最大为 32767 个)

9.8.1 全局变量

全局变量的名称，在表示类型的字母 (I、F、D、S、V、P、J、T、IO) 的后面加上整数式来表示。只有 I/O 变量为 2 个字母 (IO)。

例如，F0001、F1、F [1] 均表示相同单精度实数型变量。

变量名因在系统中被预约，所以无需定义即可使用。

在全局变量中可以使用以下所示的类型。

- I 型：整数型（范围：-2147483648~+2147483647）
例）I0001、I1、I [1]
- F 型：单精度实数型(-3.402823E + 38~3.402823E + 38)
例）F0001、F1、F [1]
- D 型：双精度实数型(-1.7976931348623157D + 308~1.7976931348623157D + 308)
例）D0001、D1、D [1]
- S 型：字符串型（最大为 243 个字符）
例）S0001、S1、S [1]
- V 型：向量型 (X, Y, Z)
例）V0001、V1、V [1]
- P 型：位置型 (X, Y, Z, RX, RY, RZ, FIG) (6 轴)
(X, Y, Z, T, FIG) (4 轴)
例）P0001、P1、P [1]
- J 型：关节型 (J1, J2, J3, J4, J5, J6) (6 轴)
(J1, J2, J3, J4) (4 轴)
例）J0001、J1、J [1]
- T 型：齐次变换型 (X, Y, Z, Ox, Oy, Oz, Ax, Ay, Az, FIG) (6 轴 / 4 轴)
例）T0001、T1、T [1]
- IO 型：I/O 型
例）IO0001、IO1、IO [1]

使用示例

I [1] = I [2] * I [3]

F [10] = 50.3

S [3] = "DENSO" + S [5]

J [5] = (10, 20, 30, 40, 50, 60)

P [1] = P [4]

9.8.2 局部 (Local) 变量

和全局变量相同，局部 (Local) 变量可以使用以下类型的变量。

- I 型：整数型（范围：-2147483648~+ 2147483647）
- F 型：单精度实数型 (-3.402823E + 38~3.402823E + 38)
- D 型：双精度实数型(-1.7976931348623157D + 308
~1.7976931348623157D + 308)
- S 型：字符串型（最大为 243 个字符）
- V 型：矢量型(X, Y, Z)
- P 型：位置型(X, Y, Z, RX, RY, RZ, FIG)（6 轴）
(X, Y, Z, T, FIG)（4 轴）
- J 型：关节型 (J1, J2, J3, J4, J5, J6)（6 轴）
(J1, J2, J3, J4)（4 轴）
- T 型：齐次变换型 (X, Y, Z, Ox, Oy, Oz, Ax, Ay, Az, FIG)（6 轴 / 4 轴）
- IO 型：I/O 型

局部 (Local) 变量使用类型定义命令，在类型定义之后才可以使用。

数值型和字符串型的局部 (Local) 变量，使用类型定义字符也能进行类型定义。

独立变量的定义方法

在独立变量的定义中，有以下的 3 种方法。

种类	定义示例 1	定义示例 2	定义示例 3
I 型	DEFINT denso	DIM denso AS INTEGER	denso%
F 型	DEFSNG denso	DIM denso AS SINGLE	denso!
D 型	DEFDBL denso	DIM denso AS DOWBLE	denso#
S 型	DEFSTR denso	DIM denso AS STRING	denso\$
V 型	DEFVEC denso	DIM denso AS VECTOR	
P 型	DEFPOS denso	DIM denso AS POSITION	
J 型	DEFJNT denso	DIM denso AS JOINT	
T 型	DEFTRN denso	DIM denso AS TRANS	
I/O 型	DEFIO denso		

使用示例

```

DEFINT Denso , Robo '定义 I 型变量 Denso 和 Robot
DEFDBL AA '定义 D 型变量 AA
DEFIO Port = BYTE,104,&B00101011 '定义 I/O 型变量 Port, 使用从输入端口 104
                                '开始 8 比特 (BYTE)

CC% = Denso * 2 '定义 I 型变量 CC, 赋值 Denso * 2 的结果
DD$ = "Denso Robot" '定义 S 型变量 DD, 赋值字符串 "Denso Robot"
AA = F[5] / 5 '在 D 型变量 AA 内赋值右侧算数结果
IN Robo = Port '将 Port 的 I/O 数据转换为 10 进制赋值 I 型变量 Robo
    
```

9.9 可以从外部机器启动的程序名

在外部自动模式的情况下，从外部 I/O 进行输入信号，能够启动程序。

能够启动的程序仅限于有 "PROGRAM PRO <编号>" 形式的程序名称的程序。

同时，根据使用的 I/O 配置模式，能启动的程序编号受到限制。

I/O 配置模式	可启动PRO <编号>
Mini I/O 模式	PRO0 ~ PRO7
标准模式	PRO0 ~ PRO32767
互换模式	PRO0 ~ PRO127

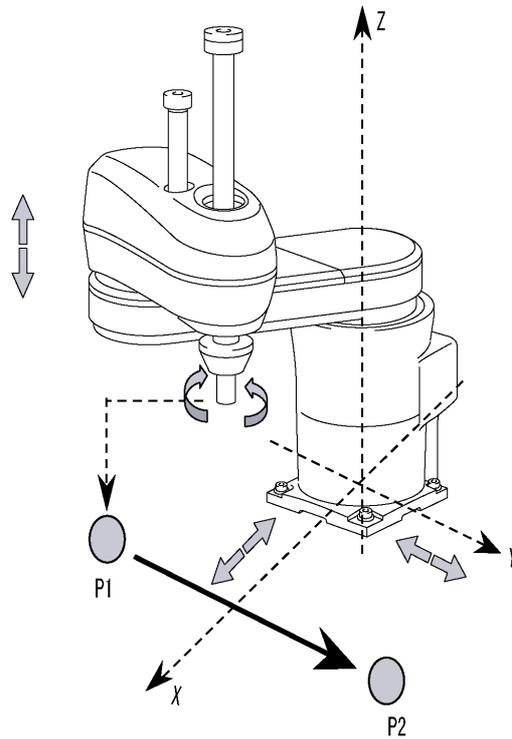
注意：从外部机器能启动的程序仅限保存在控制器的路径内的程序。如果在创建使用文件夹功能的程序时，请注意程序的保管场所。

第 10 章 程序的创建 (多功能教导器篇)

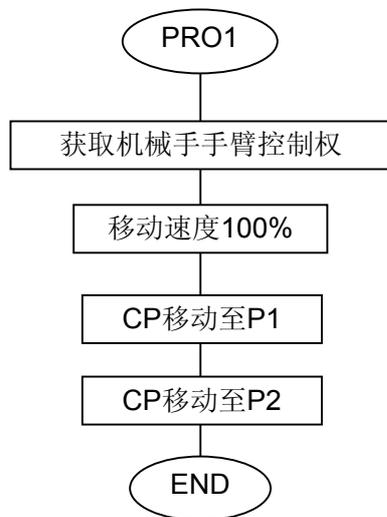
本章就使用多功能教导器的程序输入方法进行说明。

10.1 创建的示例程序

创建从当前位置向 P1 移动，然后向 P2 移动的程序。



流程图

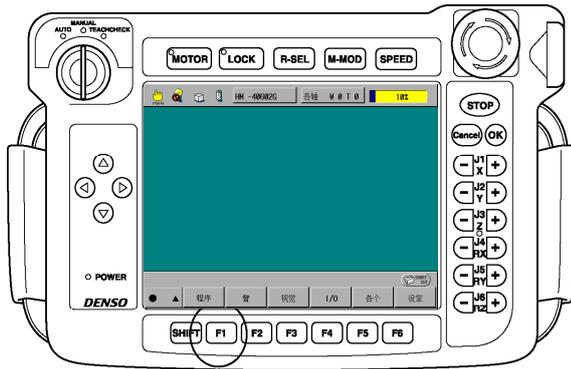


10.2 程序的创建

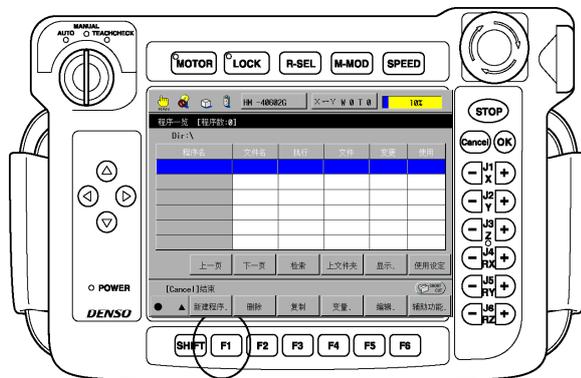
使用多功能教导器，实际输入简单的程序。
在手动模式下进行程序的编制和创建。

10.2.1 新建程序名称得输入

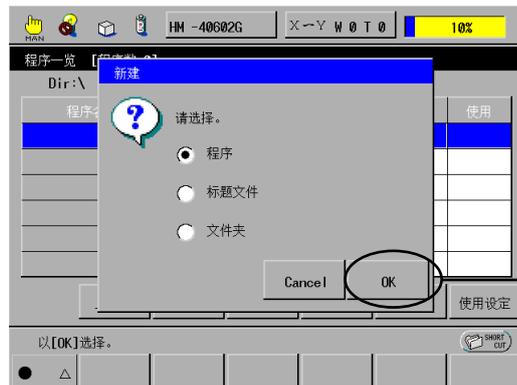
为了输入新的程序，要在多功能教导器的画面上打开程序编辑的视窗。



① 在基本画面上按压 [F1 程序]。

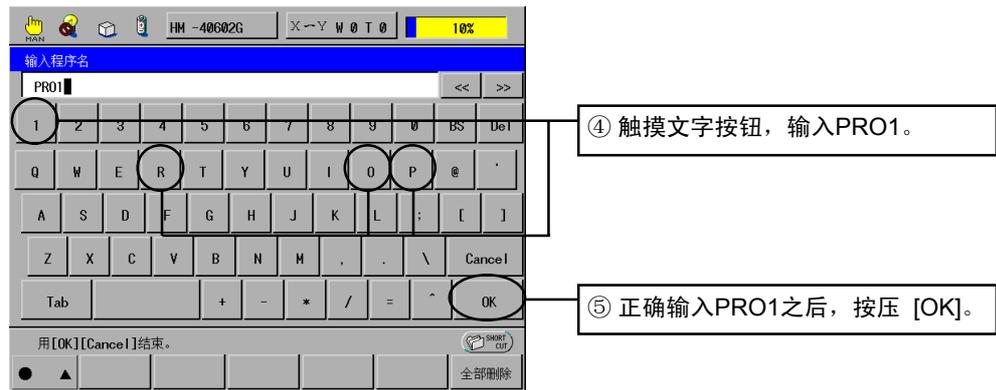


② 按压 [F1 新建程序.]。

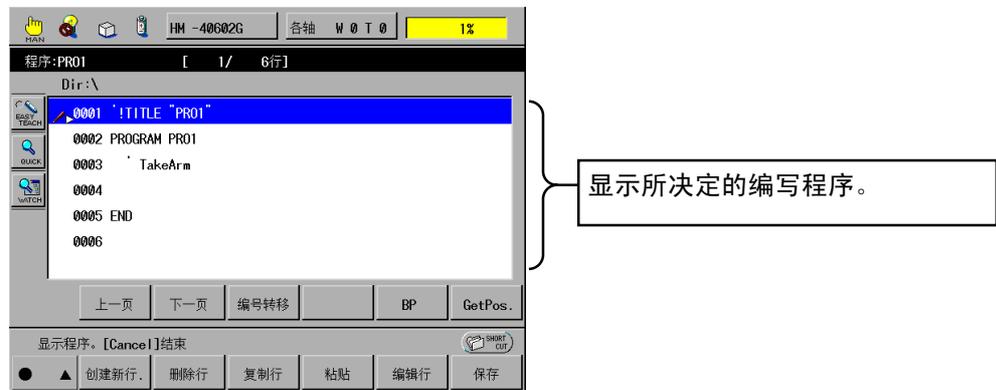


③ 按压 [OK]。

然后输入创建程序的名称（在此为PRO1）。



至此程序编辑的准备工作完成。



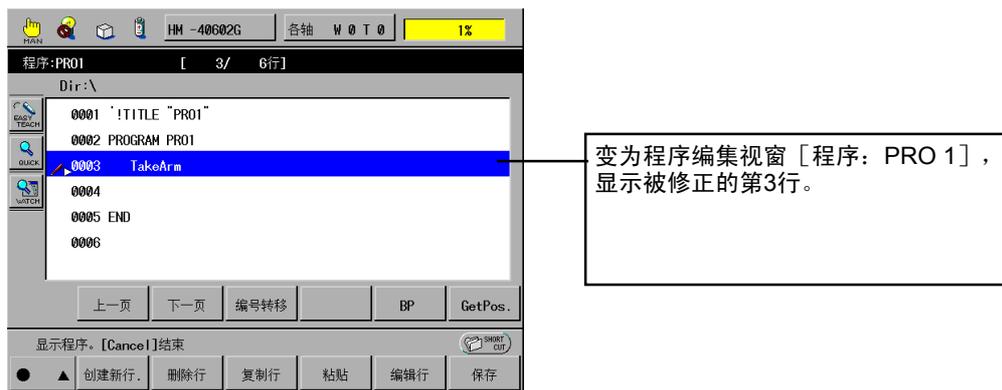
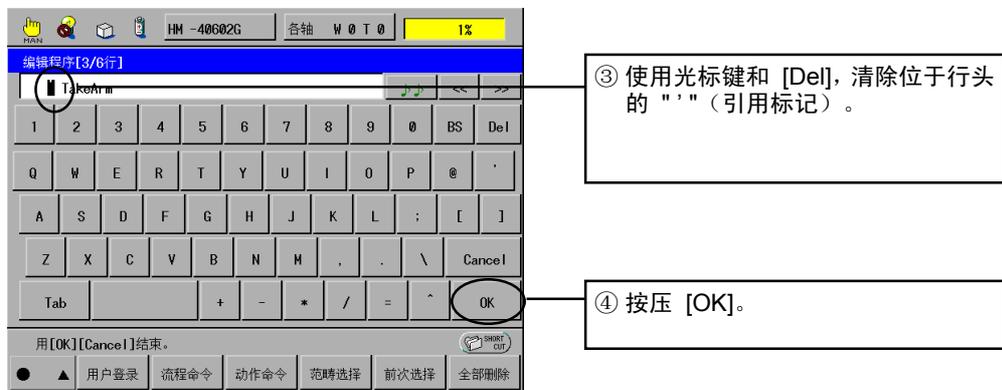
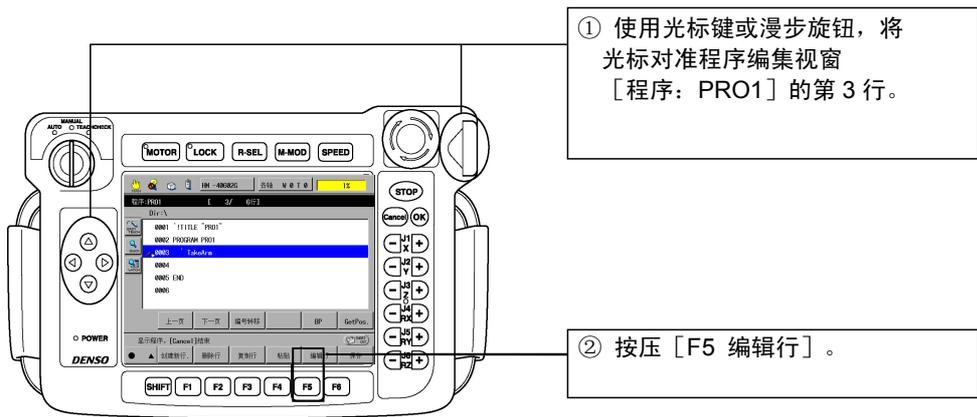
10.2.2 程序的输入

在此，创建从P1向P2移动的程序。请输入下表所示的程序编码。

程序 "PRO1"

```

PROGRAM PRO1
  TAKEARM '获取臂控制权
  SPEED 100 '指定手指端的内部移动速度
  MOVE L, P1 '向指定坐标P1移动
  MOVE L, P2 '向指定坐标P2移动
END
  
```





⑤ 将光标对准第3行，按压 [F1 创建新行.]。



⑥ 用键盘输入的 "SPEED 100" 被显示在该输入视窗上。

⑦ 按压 [OK]。



变为程序编辑视窗 [程序: PRO 1]，在第4行显示 "SPEED 100"。





⑧ 与 "SPEED 100" 的输入相同，
将p.10-3页的程序编码全部输入。

⑨ 全部输入程序之后，按压
[F6 保存]。



⑩ 按压 [OK]，将输入的程序保存。

显示返回至 [程序一览] 视窗。



★注意★

- ① 如果不需要保存变更结果，则按压 [Cancel]，不被保存而返回编辑画面。
- ② 新输入程序时，要同样进行从步骤1开始的操作。

10.2.3 程序的编译

为了让机械手控制器理解程序，程序编辑完成之后，将其变换为执行形式。将程序转换为执行形式的过程称为编译程序。

在创建的程序中如果有语法错误，在变换为执行形式的过程中将被检测。检测到语法错误时，不能加载和执行程序。



① 在[程序一览]视窗,选择 "PRO1"。
(如果操作光标键、漫步旋钮或触摸画面,则可以进行选择。)

② 按压 [F12 编辑行]。

程序名	文件名	换行	设计	变更	使用
PRO1	pro1.pac	有	有变更	未使用	



③ 选择 "只有该程序处于使用状态"。
※也可选择全部程序使用状态。

④ 按压 [OK]。

设定编译标志

仅限该程序处于使用状态
只对当前文件夹内的程序进行编译
只对当前文件夹以外的程序进行编译
全部程序处于使用状态

Cancel OK 使用设定





⑤ 按压 [OK]。
开始进行编译。

★注意★

- ① 在此，如果取代 [OK] 而按压 [Cancel]，则不经编译而返回到 [程序一览] 视窗。
- ② 将程序变换为执行形式，还有另一种操作方法。
在 [程序一览] 视窗，如果按压 [F6 辅助功能.]，则显示 [辅助功能 (程序)] 视窗。
由 [辅助功能 (程序)] 出现的 [F12 编译程序]，可以将程序变换为执行形式。



编译结束后，继续进行 (自动) 项目的加载。



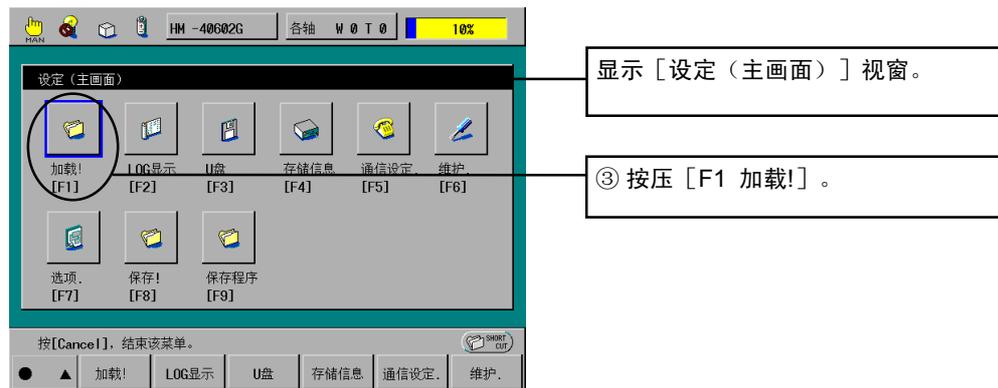
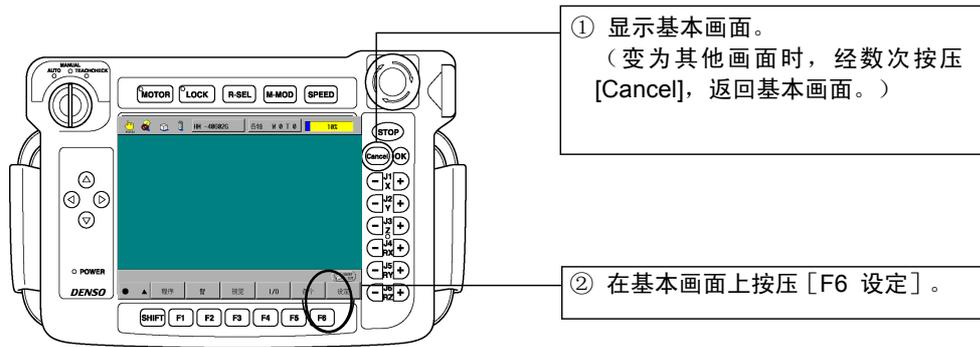
如果没有文法错误，显示 "局部 (Local) 变量已初始化"。



10.2.4 程序的加载

为了使控制器能够执行转换成执行形式后的程序，需要进行准备。该过程称之为程序的加载。

从与控制器连接的电脑传输执行形式的程序时，为了执行该程序也需要进行加载。程序数据只从电脑进行传输，机械手控制器不能使用。需要在执行的存储器空间进行加载。





显示 "请稍候...正在加载项目。" 信息。



加载完成之后，则返回 [设定 (主画面)] 视窗。

★注意★

加载使用项目不同于以前项目的局部 (Local) 变量的处理时，错误讯息显示 "局部 (Local) 变量已初始化"。

如果按压 [OK]，则向下进行。



⑤ 按压 [OK]。

经过以上的操作，程序即可执行。
要返回到基本画面，按压 [Cancel]。

至此，动作机械手的程序创建完成。

第 11 章 程序的创建

(WINCAPSIII篇)

在此，就使用 WINCAPSIII 的程序输入方法进行说明。

11.1 创建程序的准备

在此说明创建程序的准备事项。

11.1.1 WINCAPSIII 的种类

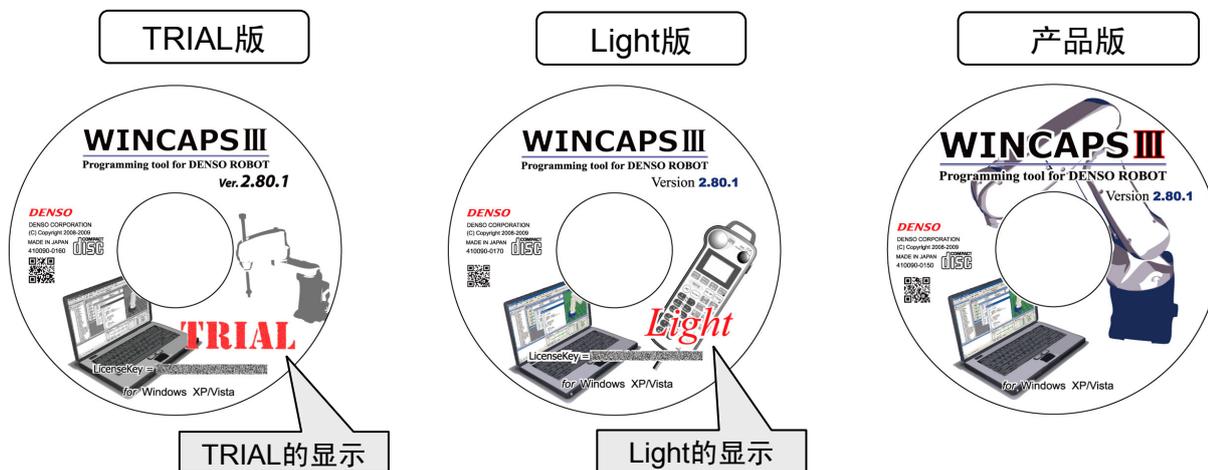
WINCAPSIII 的安装 CD 共计准备有 3 种产品。

其使用功能在不同的种类中是有限制的。

1. TRIAL 版 : 与机械手同箱包装。打印、臂动作演示 Plus、3D 数据输入、监视间隔以及程序库的部分功能无法使用。能够编辑的程序只有 "PRO1.pac"。
2. Light 版 : 与小型教导器 (选件品) 同箱包装。打印、臂动作演示 Plus、3D 数据输入、监视间隔以及程序局的部分功能无法使用。
3. 产品版 : 作为选件购买。许可证与产品版 CD 同箱包装。

【补充】如果在购买机械手时，若同时购买小型教导器和 WINCAPSIII (产品版)，就会得到 3 种 CD (TRIAL 版、Light 版、产品版)。

11.1.2 CD 外观 (CD 标签)



11.1.3 与产品版同箱装入的许可证（用户 ID）

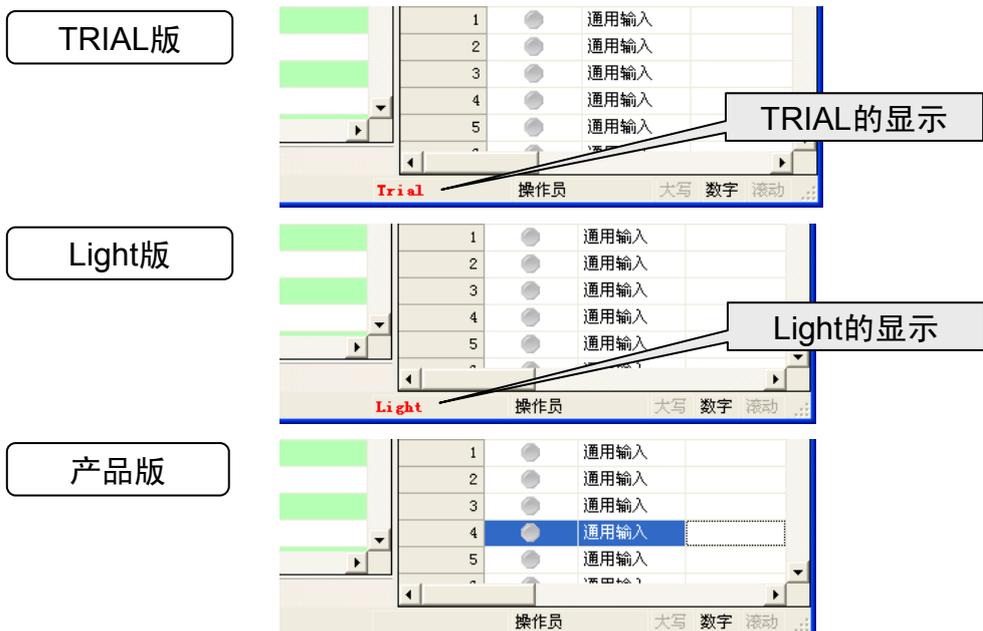
产品版 WINCAPSIII 中许可证同箱包装。在 Light 版以及 Trial 版的许可 CD 的正面印有文字。



【用户 ID】
是使 WINCAPSIII 作为产品版
发挥功能作用所必要的 ID。

11.1.4 在画面上确认 WINCAPSIII 的产品种类

可在 WINCAPSIII 画面上可以确认当前所安装软件的产品种类。



11.1.5 版本升级时的注意事项

WINCAPSIII产品的高级别顺序是 [TRIAL 版] → [Light 版] → [产品版]。
可以使用任何一种产品进行版本升级。在这种情况下，需安装的产品要高于已经安装的产品。

例如，对已经安装 [Light 版] 的计算机使用 [TRIAL 版] 进行升级后，相当于 [Light 版] 的功能。

【补充】 不只限于已经安装的产品，只要输入用户ID，就会作为产品版使用。

11.1.6 用户 ID 的输入方法

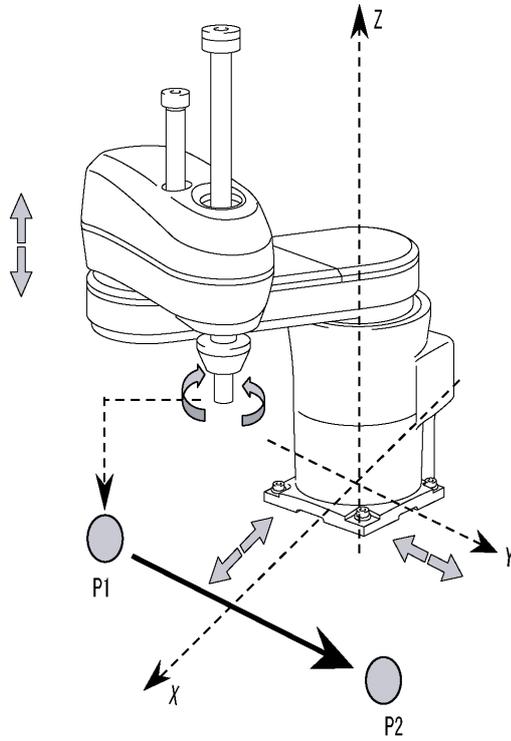
对产品版进行升级时，需要将购买时许可证上标注的用户 ID 输入到 WINCAPSIII的 "认证信息" 上。

为了显示以下的画面，请选择 [帮助 (H)] - [许可证信息 (L)]。

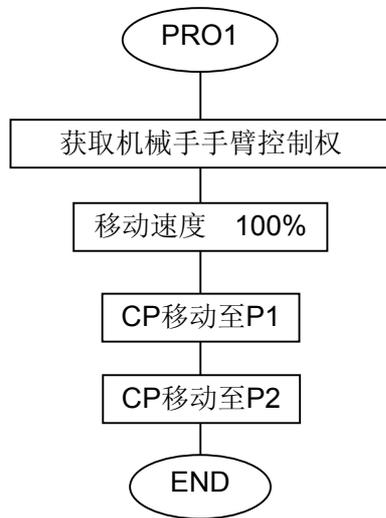


11.2 创建的示例程序

创建从当前位置向 P1 移动，然后向 P2 移动的程序。



流程图



11.3 创建程序

使用 WINCAPSIII实际创建一个简单的程序。

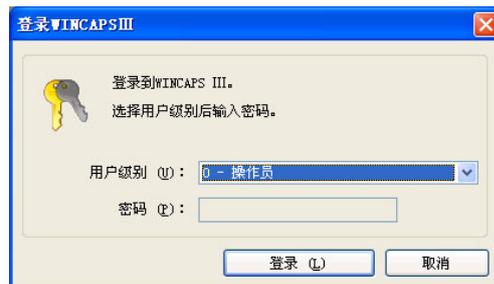
11.3.1 WINCAPSIII的启动

启动程序支持工具 "WINCAPSIII"。

步骤 1

点击 [开始] 菜单的 [所有程序]，选择 [DENSO FACTORY WARE] - [WINCAPSIII] - [WINCAPSIII]。

[登录 WINCAPSIII] 的对话框显示出来。



步骤 2

利用操作员身份登录。

在选择 [用户级别 (U)] 中选择 [0-操作员]，点击 [登录 (L)]。在进行变更设定等操作时，需要以 [1-程序员] 身份登录。利用程序员身份登录需要密码。初次以程序员身份登录，需要设置密码。

11.3.2 新建项目

在 WINCAPSIII中，可利用名为项目的单位，管理复数个程序。在项目内创建复数个程序，搭配使用，可更简单易懂地进行管理。

在创建程序时，需要新建一个项目。

步骤 1

选择 [文件 (F)] - [新建项目 (N)]。

[WINCAPSIII项目创建向导] 视窗显示出来。



点击 [下一步]。

步骤 2

输入项目名与项目的保存位置。

输入的项目名的文件夹会在所指定的保存位置被创建。



点击 [下一步]。

步骤 3

选择机械手类型。

选择所使用的机械手控制器及机械手类型，点击 [下一步]。



步骤 4

选择控制器选项。

选择所使用的控制器规格（标准/全球）与有无使用附加轴，点击 [下一步]。



步骤 5

输入控制器与 PC (WINCAPSIII) 的连接方法。

输入连接方法 (以太网/RS-232C) 与详情, 点击 [下一步]。
也可在创建项目完毕后, 对连接方法的设定进行变更。



步骤 6

输入变量的使用个数。

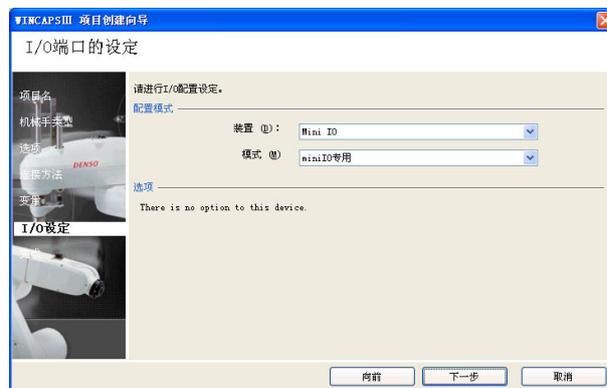
也可在创建项目完毕后, 变更变量的使用个数。



步骤 7

输入 I/O 端口的设定。

在控制器上选择所采用的装置与配置模式。根据需要, 输入装置的详细设定。
点击 [下一步]。



步骤 8

确认设定内容。

如果设定内容正确无误，点击 [完成]，关闭向导。



项目创建完成。

11.3.3 程序的创建

在项目内创建程序。

步骤 1

选择 [项目 (P)] - [程序的创建 (P)]。



步骤 2

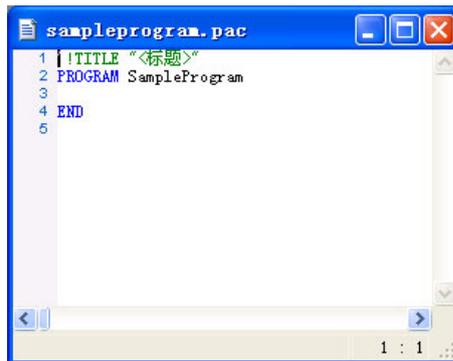
在 [种类] 中选择 [程序 (*.PAC)], 输入 [程序名] 与 [文件名]。

程序名必须是以字母开头的半角英数字, 且必须在 64 个文字以内。
输入程序名之后, 它会自动输入文件名, 如果想取不同的名字, 请重新输入文件名。
完成输入, 点击 [OK]。



步骤 3

程序创建完成, 程序输入窗口即会显示出来。



11.3.4 程序的输入与保存

在程序输入窗口中，输入以下程序示例。

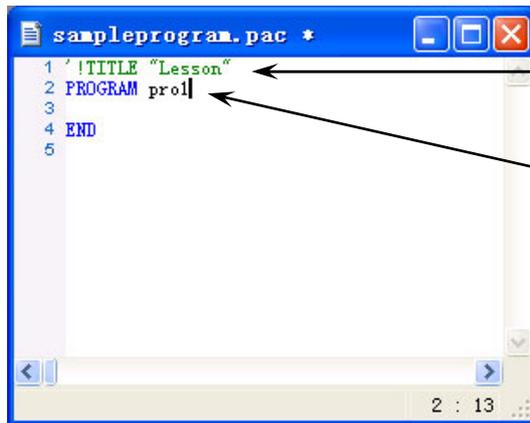
此程序示例是让机械手从登陆点 P1 向登录点 P2 移动的程序。
请输入下表所示的程序编码。

程序示例 程序 "PRO1"

PROGRAM PRO1	'程序名 "PRO1"的定义"
TAKEARM	'获取臂控制权。'
SPEED 100	'将手前端内部速度设为100 (%)。'
MOVE L, P1	'移动至登陆点P1。'
MOVE L, P2	'移动至登陆点P2。'
END	

步骤 1

输入程序标题与程序名。

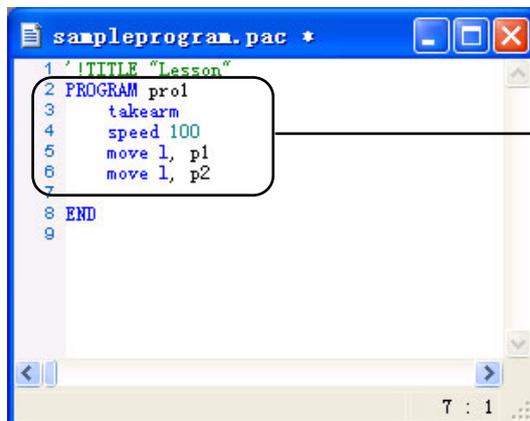


① 输入程序的标题。(在此输入 "lesson" 这个标题。)

② 输入程序名。(在此输入 "PRO1"。)(大字、小字均可)。

步骤 2

输入程序指令。



③ 输入PRO1的程序指令。

步骤 3

保存所创建的程序。

选择 [文件 (F)] - [覆盖保存 sampleprogram.pac (S)]。



程序保存完毕。

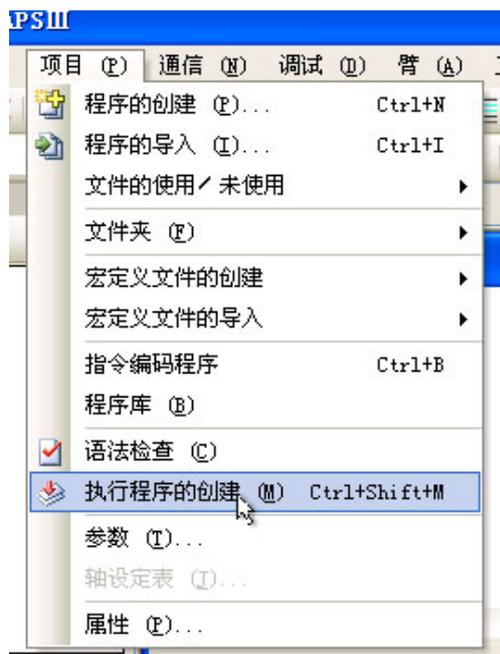
11.3.5 程序的编译

在 PAC 语言中，要想执行输入的程序，需要事先将其转换（编译）成机械手控制器能够理解的数据形式。编译好的即为“执行程序”。

步骤 1

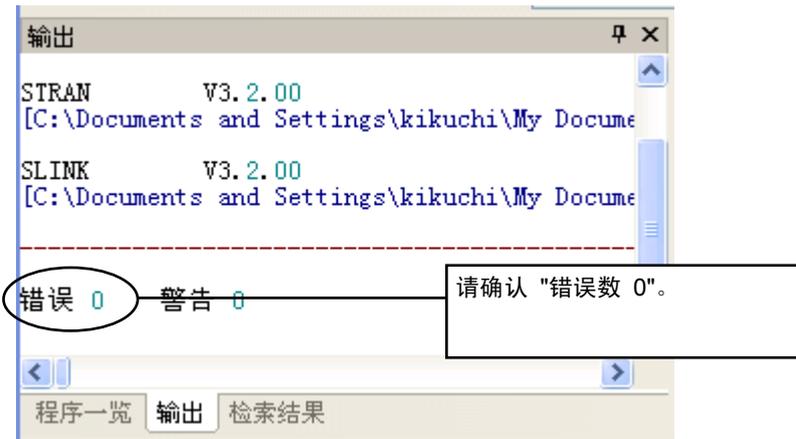
转换成执行程序

选择 [项目 (P)] - [执行程序的创建 (M)]，[程序一览] 视窗内的所有程序就会转换为执行程序。



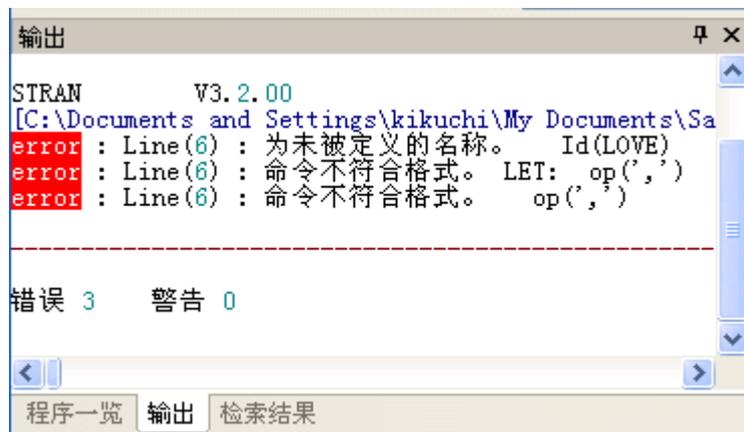
步骤 2

确认没有错误。



当出现错误时，程序指令的输入有错误。

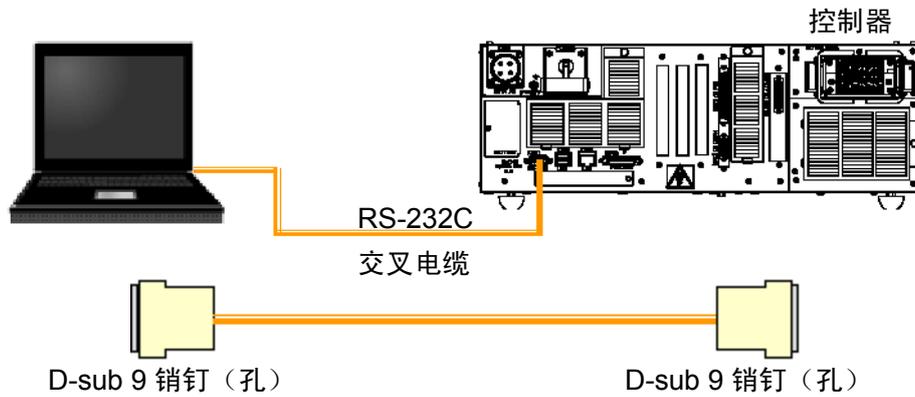
[输出] 视窗会显示出错误位置与其内容等，请参考并确认程序，然后再次输入指令。



11.4 WINCAPSIII与控制器的通信电缆的连接方法

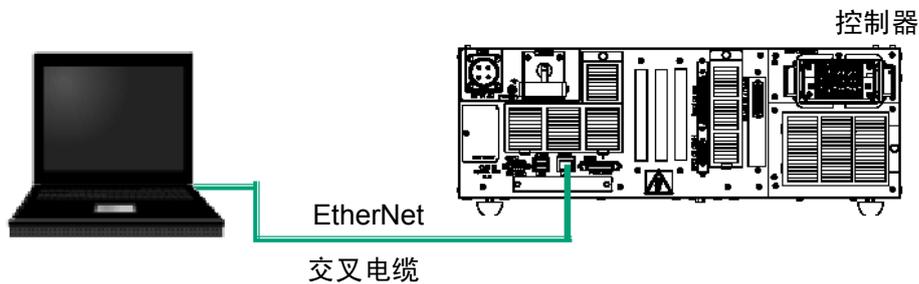
为了让 WINCAPSIII (PC) 和机械手控制器之间进行数据通信，需要用通信电缆连接。请用以下的方法进行电缆连接。

11.4.1 RS-232C 连接

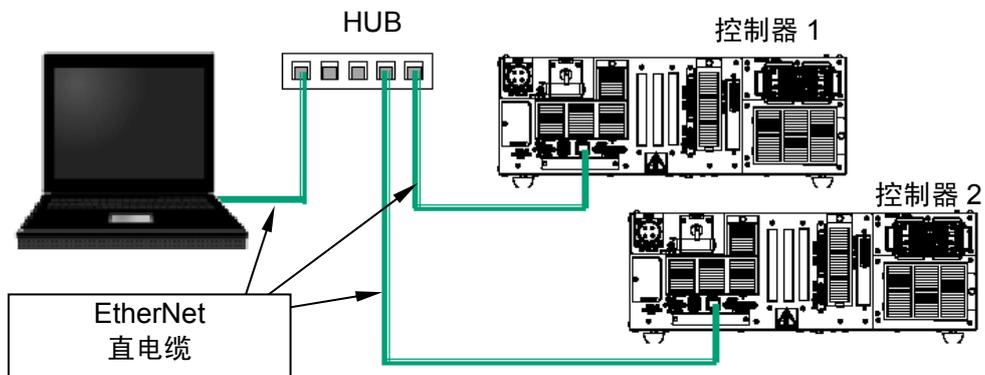


11.4.2 EtherNet 连接

- 1 对 1 连接时



- 1 对 N 连接时



11.5 与机械手控制器的通信连接准备

11.5.1 RS-232C 连接时

11.5.1.1 RS232C（机械手控制器的设定）

要想实现利用 RS-232C 卡与 WINCAPSIlll 通信，需对机械手控制器进行设定。在多功能教导器上进行该设定。

步骤 1

F6

在多功能教导器的基本画面上按压 [F6 设定]。



显示 [设定（主画面）] 视窗。



步骤 2

F4

按压 [F5 通信设定.]。

显示 [通信设定菜单] 视窗。



步骤 3

F2

按压 [F2 设定串行]。
显示 [RS232C 设定] 视窗。

步骤 4

F5

选择COM2，按压 [F5 变更值]。

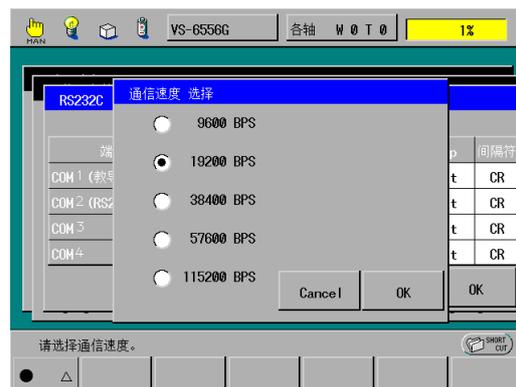


显示 [通信速度 选择] 视窗。

步骤 5

OK

选择通信速度，按压 [OK]。



显示返回到 [RS232C 设定] 视窗。

步骤 6

OK

确认显示内容，按压 [OK]。
通信速度被设定。



画面显示返回到 [通信设定菜单] 视窗。

步骤 7

F1

按压 [通信设定菜单] 视窗上的 [F1 通信权.]。



显示 [通信权 设定] 视窗。



步骤 8

F5

选择COM2, 按压 [F5 变更设定]。

显示 [通信权 选择] 视窗。

步骤 9

选择需要的通信权。



通信权的含义如下所示。

- 不能使用：不使用通信端口。
- 只能读入：在计算机一侧，可以读机械手控制器的数据。不能从计算机向机械手控制器发送数据。
- 可读可写：在计算机与机械手控制器之间，可以双向进行数据的存取。

创建程序时，选择 [可读可写]。

自动运行已经做好的程序，只在监视变量及I/O等的数值时，选择 [可读可写]。

选择后，按压 [OK]。

显示返回到 [通信权 设定] 视窗。

注意：不能将 RS232C 与 Ethernet 同时设为可读可写。

步骤 10

OK

确认显示内容，按压 [OK]。

通信权被设定。



画面显示返回到 [通信设定菜单] 视窗。



步骤 11

Cancel

按压2次 [Cancel]。

显示返回到基本画面。

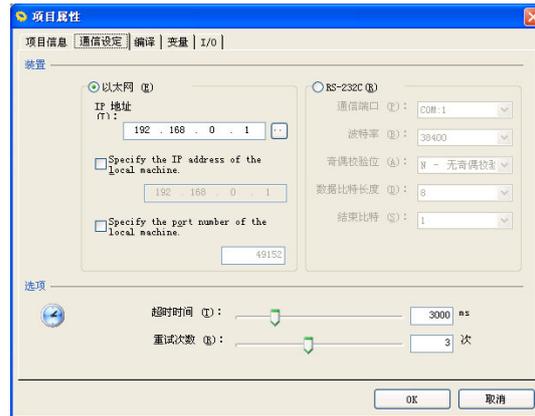
11.5.1.2 RS-232C 连接的设定（WINCAPSIII的设定）

要想实现利用 RS-232C 让 WINCAPSIII 与机械手控制器通信，需在 PC 上进行设定。在 WINCAPSIII 的项目内进行该设定。

步骤 1

WINCAPSIII 的项目通信设定会显示出来。

项目的通信设定在项目创建向导中进行，创建完成后，也可选择 [项目 (P)] - [属性 (P)] - [通信设定] 图标，也可显示和进行设定。



步骤 2

选择 RS-232C。

输入 RS-232C 的详细通信设定。



在 [通信端口] 栏选择计算机使用的通信端口。请按照使用的计算机规格进行选择。关于 [波特率], [奇偶校验位], [数据比特长度], [结束比特] 的设定, 请根据利用机械手控制器所设定的值进行设定。

步骤 3

请设定 [超时时间] 与 [重试次数]。

输入完成后, 请点击 [OK]。

利用 PC 进行的通信设定完成。

11.5.2 EtherNet 连接时

11.5.2.1 EtherNet（机械手控制器的设定）

要想利用以太网实现 PC（WINCAPSIII）与机械手控制器之间的通信，需对机械手控制器进行设定。在多功能教导器上进行该设定。

进行通信权的设定以及 IP 地址设定。

步骤 1

F6

在多功能教导器的基本画面上按压 [F6 设定]。



显示 [设定（主画面）] 视窗。



步骤 2

F4

按压 [F5 通信设定.]。

显示 [通信设定菜单] 视窗。



步骤 3

F1

按压 [F1 通信权.]。
显示 [通信权 设定] 视窗。



步骤 4

F5

选择EtherNet，按压 [F5 变更设定]。
显示 [通信权 选择] 视窗。



步骤 5

选择 [可读可写]。

通信权的含义如下所示。

使用EtherNet时需要选择 [可读可写]。

- 不能使用：不使用通信端口。
- 只能读入：在计算机一侧，可以读机械手控制器的数据。不能从计算机向机械手控制器发送数据。
- 可读可写：在计算机与机械手控制器之间，可以双向进行数据的存取。

选择后，按压 [OK]。

显示返回到 [通信权 设定] 视窗。

注意：无法同时将 [COM2(RS232C)] 与 [Ethernet] 设定为 [可以读入 / 写入]。
如果希望 [可以读入 / 写入] [Ethernet]，请先将 [COM2(RS232C)] 设定为 [不可使用] 或 [只可读入]，然后再进行设定。

步骤 6

OK

确认显示内容，按压 [OK]。
通信权被设定。



画面显示返回到 [通信设定菜单] 视窗。



步骤 7

F4

按压 [F4 设定地址]。
显示 [控制器IP地址设定] 视窗。



步骤 8

F5

按压 [F5 变更值], 设定地址。



步骤 9

Cancel

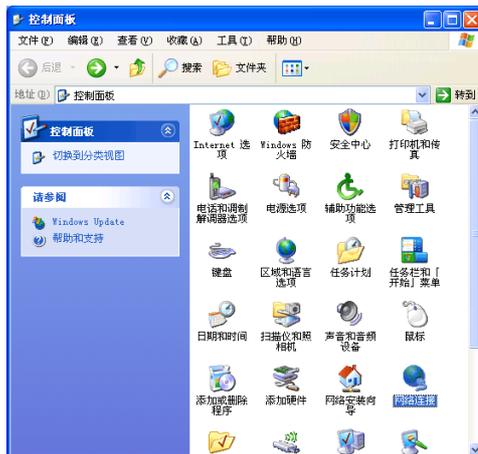
按压2次 [CANCEL]。
显示返回到基本画面。

11.5.2.2 计算机网络环境设定状态

为了进行以太网的连接，需要进行Windows的设定。在此，以安装了网卡（转接器）、网络通信协议（TCP / IP）有效为前提，说明设定网络环境的步骤。首先，确认局域网连接是否有效。然后在TCP / IP的属性上设定IP地址。

步骤 1

从Windows 的 [开始] 按顺序选择 [控制面板 (C)]。显示 [控制面板] 视窗。



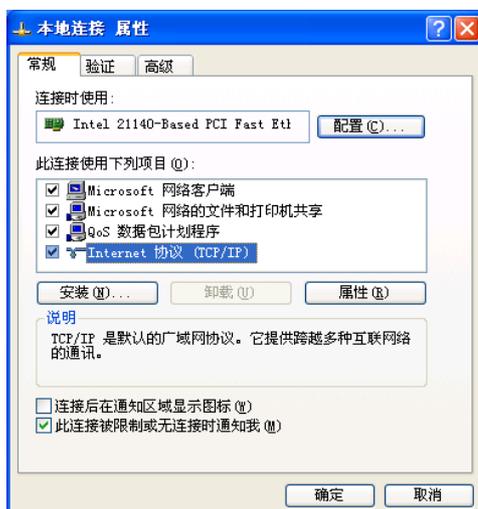
步骤 2

点击 [控制面板] 视窗中的 [网络连接]。显示 [本地连接] 的图标。该图标显示 "无效" 时，请将鼠标对准图标进行右击，选择 [设为有效 (A)]。



步骤 3

将鼠标对准 [本地连接] 图标，如果进行右击选择 [属性]，则 [本地连接 属性] 对话框就会被显示。



步骤 4

选择 [常规] 的标签, 选择 [Internet协议 (TCP/IP)], 点击 [属性 (R)]。

步骤 5

选择 [常规] 的标签, 点击 [使用下面IP地址 (S)], 然后输入 [IP地址 (I)] 和 [子网掩码 (U)]。

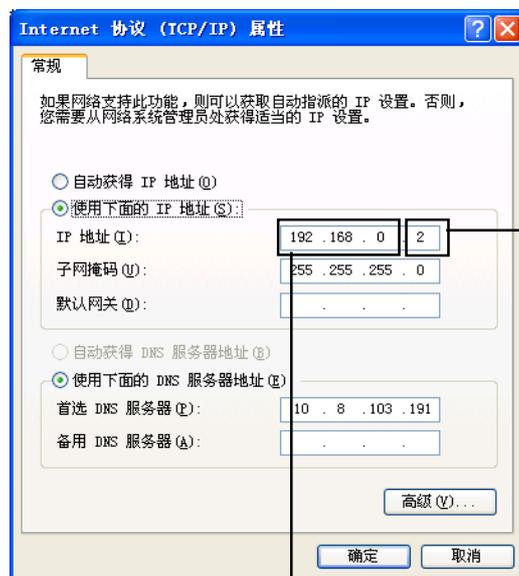
关于输入的IP地址和辅助网络地址的具体值, 请与管理该网络的网络管理员联系。如果该网络是局域网 (例如, 只有电脑和机械手控制器连接的环境), IP地址可以自由设定, 在此将IP地址设为 [192. 168. 0. 2]、辅助网络地址设为 [255. 255. 255. 0]。点击 [确定], 结束IP地址的设定。

注意①: 与广域网络 (例如公司内的网络) 相连接时, 必须与网络管理员联系, 设定IP地址、辅助网络掩码。

将局域网内使用的IP地址设定, 不作无效处理直接接入广域网 (例如公司内的网络) 的情况下, 有时会给连接网络带来混乱。

②: IP地址在同一网络内不能重复。与广泛共有的网络进行连接时, 请注意IP地址不要与其他的末端重复。一般认为以下地址是与其他末端重复概率小的IP地址。

不容易重复的IP地址: [192. 168. 0. 2]~[192. 168. 0. xxx]、(xxx为003~254)



与机械手控制器上设定的值是相同的。

设定的值与机械手控制器上的设定值不重复。

11.5.2.3 Ethernet 连接的设定（WINCAPSIII的设定）

要想利用Ethernet将PC（WINCAPSIII）与机械手控制器连接起来，需对PC（WINCAPSIII）进行通信设定。在WINCAPSIII的项目内进行该设定。

步骤 1

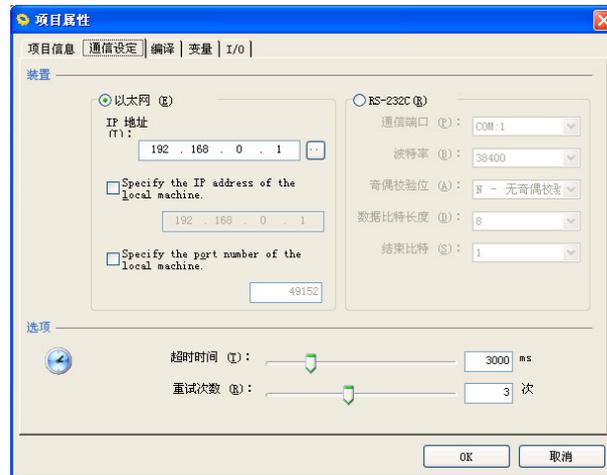
打开WINCAPSIII的项目通信设定。

项目的通信设定在创建项目向导中进行，创建完成后，也可通过选择 [项目 (P)] - [属性 (P)] - [通信设定] 图标打开或进行设定。



步骤 2

确认 [以太网] 已被选择，输入 [IP地址]。



在 [IP地址] 处，输入机械手控制器的IP地址。

步骤 3

请设定 [超时时间] 与 [重试次数]。

输入完成后，请点击 [OK]。

PC的通信设定完成。

11.6 由 WINCAPSIII进行的数据传输

在进行数据传输（发送、接收）的基础上，需要确认机械手控制器的通信权的设定和机械手控制器的状态。

有时会由于机械手控制器的运行状态会造成数据传输失败，对此请予注意。

11.6.1 控制器一侧的准备（数据传输时的注意事项）

- ① 在多功能教导器画面上，请确认无错误讯息显示。
- ② 请确认连接的通信端口（RS232C 或 EtherNet）的通信权是否处于 "可以读入 / 写入" 状态。

注意：选择 "只可读入" 时，如果发送数据，就会发生错误 "200B参数传输失败"。

- ③ 由于机械手控制器的电机 ON / OFF 状态和动作模式状态，有时不能进行数据发送。是否可以发送各状态的数据，如下表所示。

数据发送可否对应表

状态 发送数据	电机	控制器 运行模式				备注
		外部自动	内部自动	手动	教导	
PAC 程序	ON	×	×	×	×	△：不能自动保存程序
	OFF	×	△	○	△	
变量	ON	×	○	○	○	
	OFF	×	○	○	○	
I/O	ON	×	×	×	×	
	OFF	×	○	○	○	
机械臂	ON	×	△	△	△	△：只可以发送工具、 工件、空间数据
	OFF	×	○	○	○	

注意：

- 关于数据接受，与电机的ON / OFF状态及控制器运行模式无关，可以接收。
- 程序运行中的数据接收将造成程序处理迟滞，对此请予注意。

- ④ 请确认多功能教导器的画面上未显示有 "程序一览" 或 "选择变量类型"。

11.6.2 将程序数据发送至机械手控制器

步骤 1

在WINCAPSIII中选择 [通信 (N)] - [发送接收数据 (T)]。
[发送接收数据] 视窗即会显示出来。



步骤 2

打开 [WINCAPSIII] 中的 [程序] - [源文件], 所保存的程序即会显示出来。



步骤 3

在 [程序] 上打勾, 点击 [发送 (S) ->] 按钮。



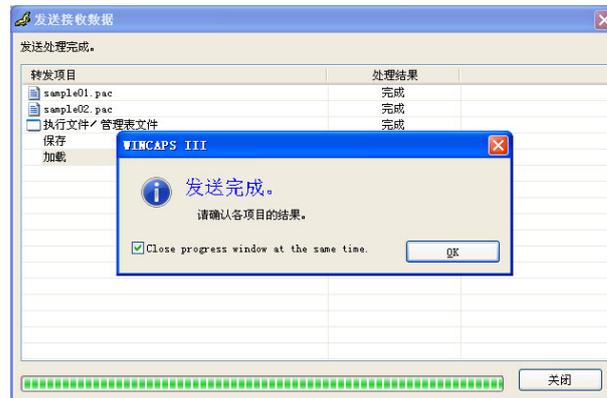
步骤 4

确认对话框打开，选择 [是 (Y)]，数据即会发送至机械手控制器。



步骤 5

请确认所发送数据的处理结果已全部 [完成]。



步骤 6

在多功能教导器上点击 [F1 程序]，确认有程序发送至 [程序一览]。

在机械手控制器上创建的程序即会发送完成。

第4部

程序的验证

第 12 章 程序启动

12.1 利用 WINCAPSIII所进行的仿真运行

可以不运行实际的机械手，而是在观察计算机的仿真映象的同时对创建生成的机械手程序的运行进行确认。

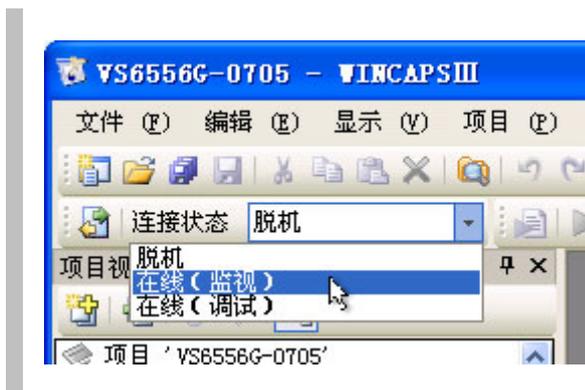
12.1.1 打开臂 3D 视图

要想打开机械手的模拟图像，需打开臂 3D 视图。
选择 [显示 (V)] - [臂 3D 视图 (A)], [臂 3D 视图] 视窗即会显示出来。



12.1.2 将 WINCAPSIII切换至监视器模式

将 WINCAPSIII的连接状态切换为 [在线 (监视)]。



12.1.3 置于机械锁定状态

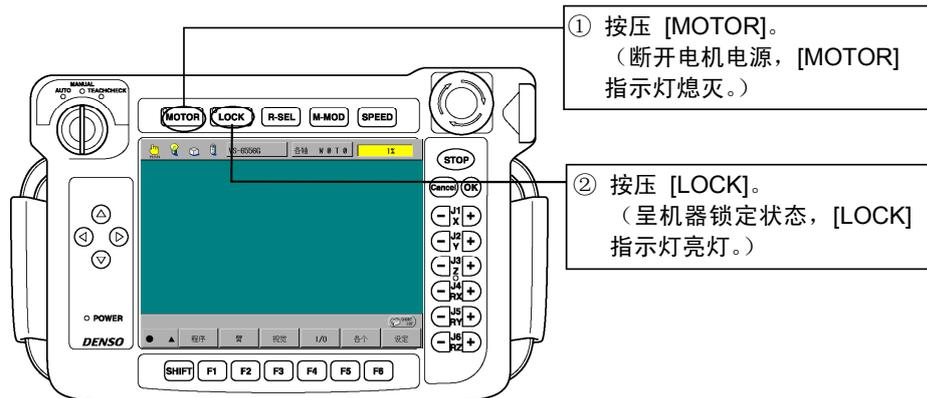
将机械手控制器置于机械锁定状态。据此不实际动作机械手而执行程序，从而实现机械手仿真运行。

步骤 1

切断电机电源

步骤 2

置于机械锁定状态



★注意★

机械锁定时，要将电机电源置于切断状态。也就是要确认 [MOTOR] 指示灯熄灭。

★参考★

[Ver. 1.4以上版本]

在机械锁定时，能够限制I/O输出。关于详细内容，请参照操作指南“5.5 I/O信号的显示和机械手动作的仿真”。

状态栏的图标根据I/O输出的状态而变化。



：没有I/O输出限制



：有I/O输出限制

12.1.4 进行程序启动

为了启动程序，请用教导检查模式或自动模式中的任何一种方法进行。

关于启动方法，请遵照“12.2 教导检查模式下进行的程序启动”或“12.3 在内部自动模式下进行的程序启动”中任何一种启动方法，在机器锁定状态下进行程序启动。

WINCAPSIII的[臂3D视图]窗口上所显示的机械手臂按照程序进行动作。

12.2 在教导检查模式下进行的程序启动

用手动方式检查教导结果的程序动作过程，称之为教导检查。教导检查在教导检查模式下进行。

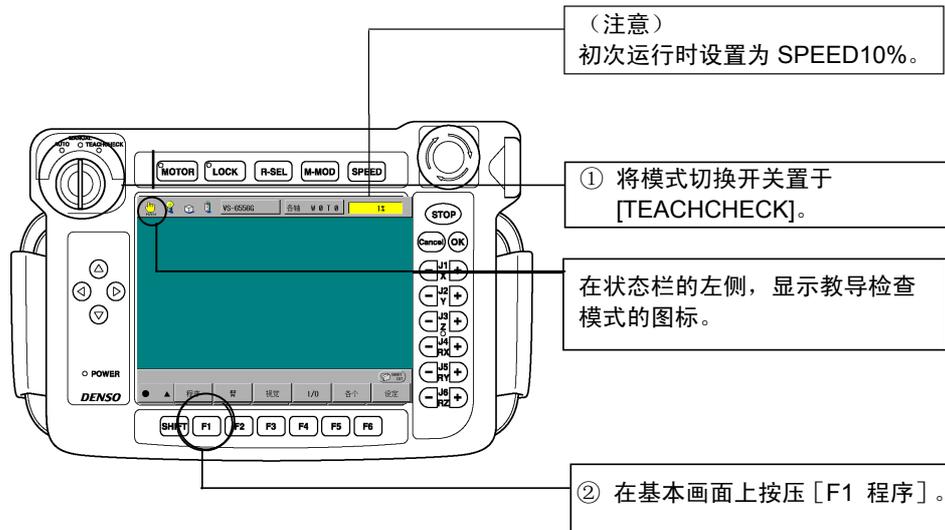
12.2.1 置于教导检查模式

步骤 1

接通电机电源

机器锁定状态下进行动作时，请不要接通电机电源。

步骤 2



★参考★

发生ERROR 21F2（自动允许接通）时，请确认 "7.3.2 动作模式和自动允许输入信号的相关性"。



显示 [程序一览] 视窗。

程序名	状态	行编号	执行时间	优先顺序	方向
AUTOEXEC	停止中	4	0.00	128	▽进
PR01	停止中	2	0.00	128	▼进

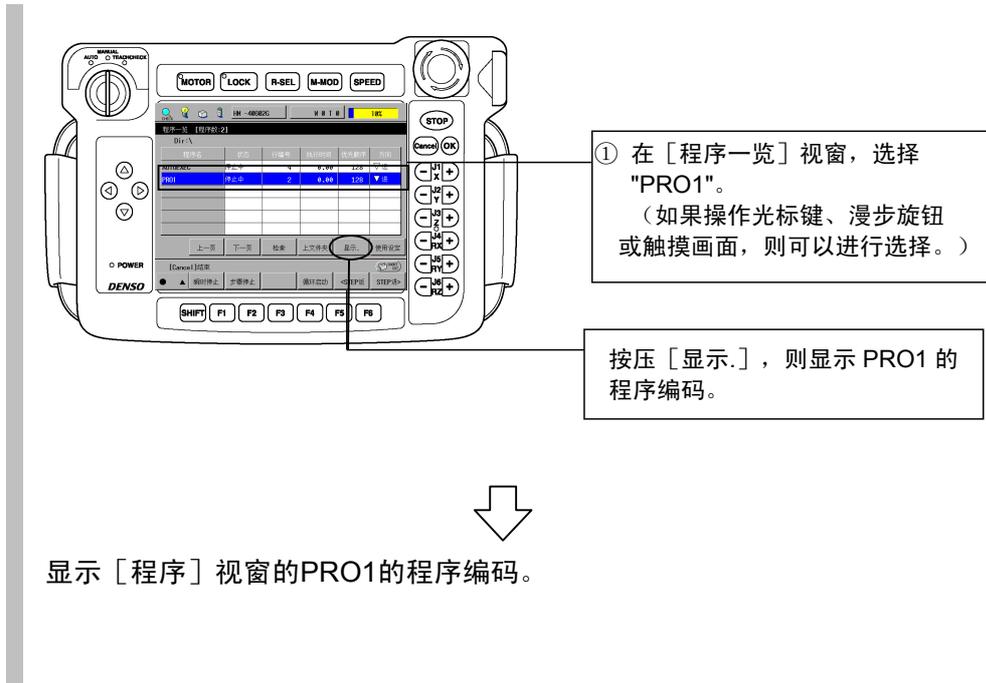
Dir:\

上一页 下一页 检索 上文件夹 显示 使用设定

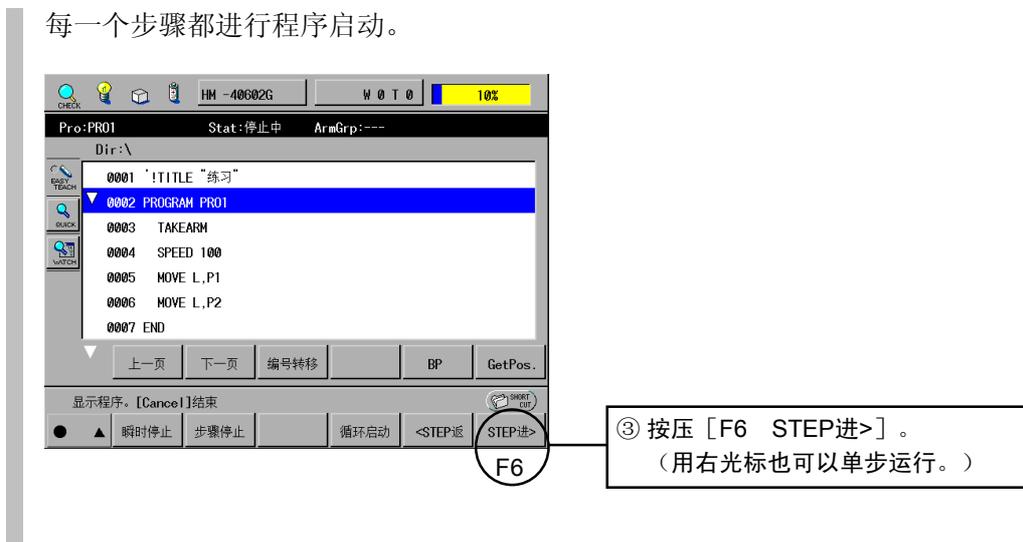
[Cancel] 结束

● ▲ 瞬时停止 步骤停止 循环启动 <STEP返 STEP进>

12.2.2 选择程序



12.2.3 单步运行检查启动





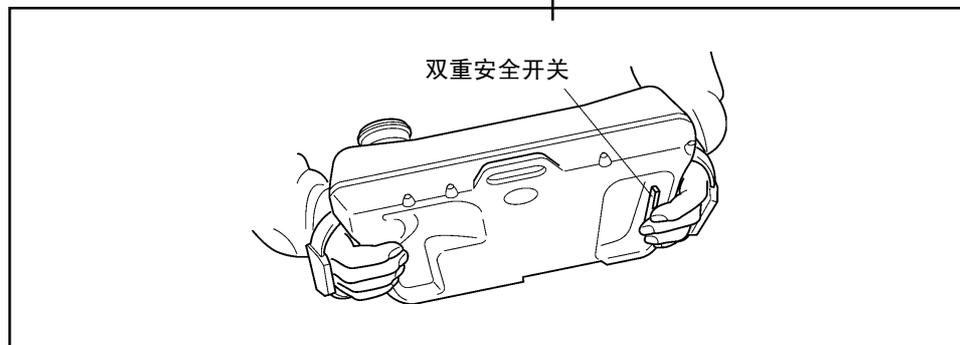
显示系统信息。

★注意★

确认程序时，请确保随时都可以按压 [机械手停止] 的状态运行。



④ 按压双重安全开关的同时，
按压 [OK]。
(需要中止步进时，
按压 [Cancel]。)



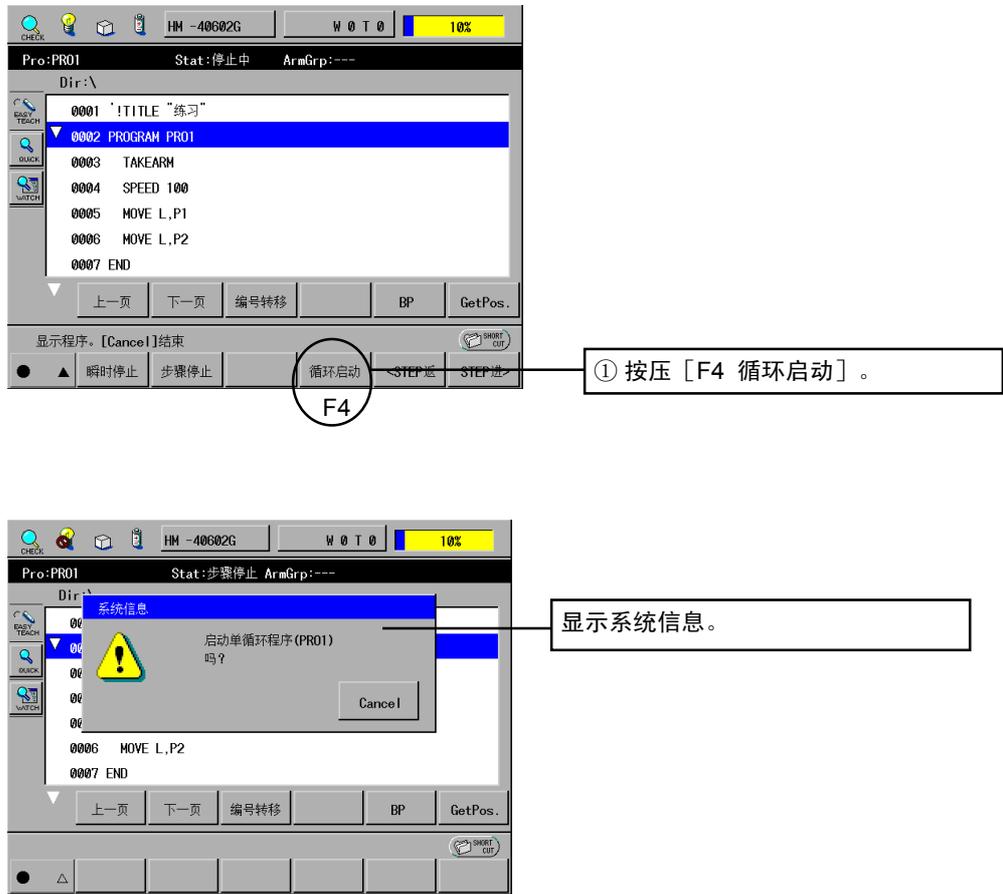
在教导检查模式下，仅限同时按压双重安全开关和 [OK] 时才能执行程序。
释放其中的任何一个按钮，机械手就会瞬时停止。



反复上述步骤的操作，一边确认动作的安全，一边执行完最终程序。

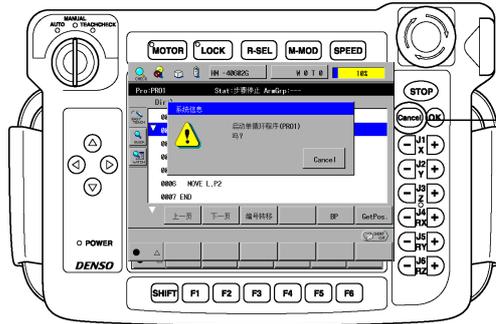
12.2.4 循环检查启动

然后，将单步运行检查确认的程序以循环动作进行确认。

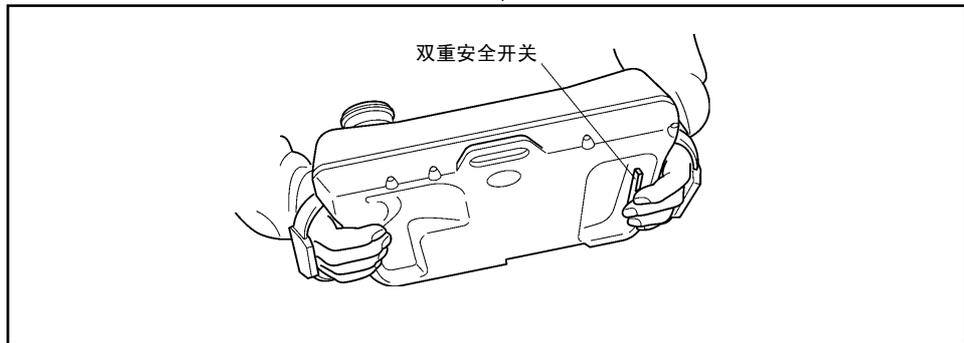


★注意★

确认程序时，请确保随时都可以按压 [STOP] 的状态运行。



② 按压双重安全开关的同时，
按压 [OK]。
(需要中止循环检查时，
按压 [Cancel]。)



在教导模式下，仅限同时按压双重安全开关和 [OK] 时才能执行程序。
释放其中的任何一个按钮，机械手就会瞬时停止。



程序在循环启动、机械手运行的同时，按照反衬显示的程序一览执行。
执行到程序结束，则停止。

12.3 在内部自动模式下进行的程序启动

以自动运行方式对教导检查完毕的程序进行试运行。

注意：第一次以自动模式运行的程序，要将速度设定为10%左右后再开始运行。
在手动模式和教导检查模式下，可以自动模式的10%的速度运行，而在自动运行时可以最高速度运行。

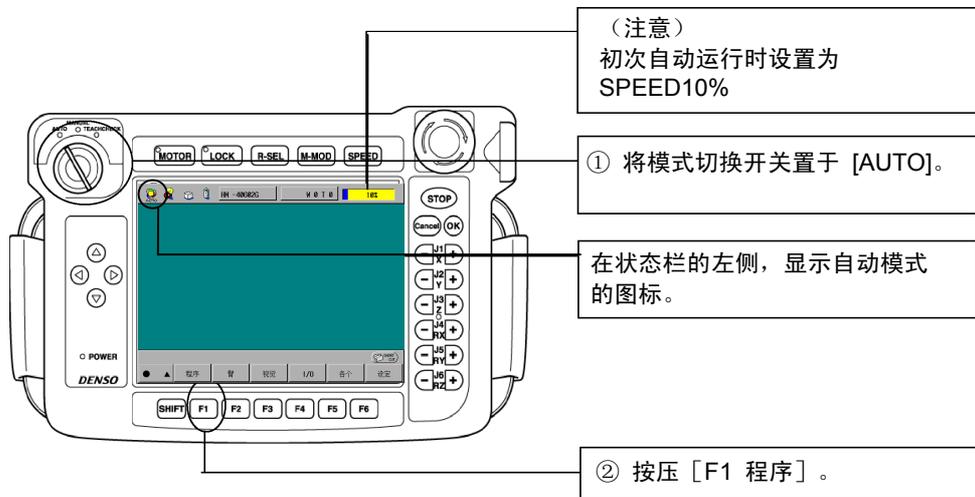
12.3.1 置于自动模式

步骤 1

接通电机电源

机械锁定状态下进行动作时，请不要接通电机电源。

步骤 2



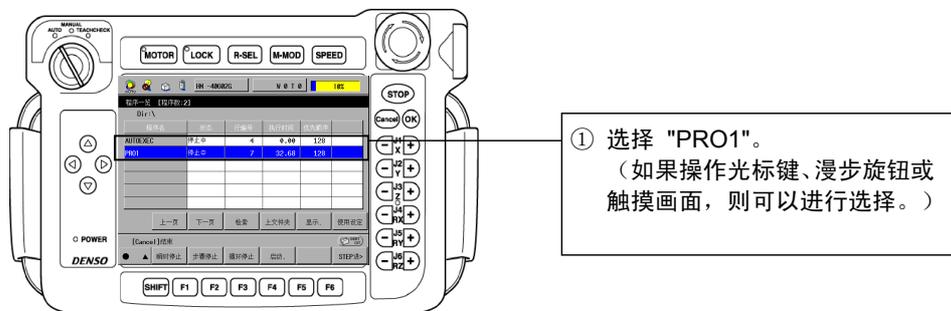
在发生ERROR 21F3（自动允许切断）时，请确认 "7.3.2 动作模式和自动允许输入信号的相关性"。

★参考★



12.3.2 选择程序

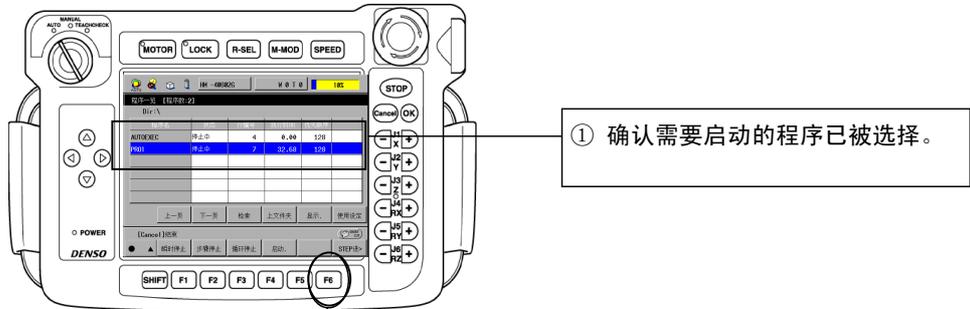
在 [程序一览] 视窗，选择可自动运行的程序。



12.3.3 步骤启动

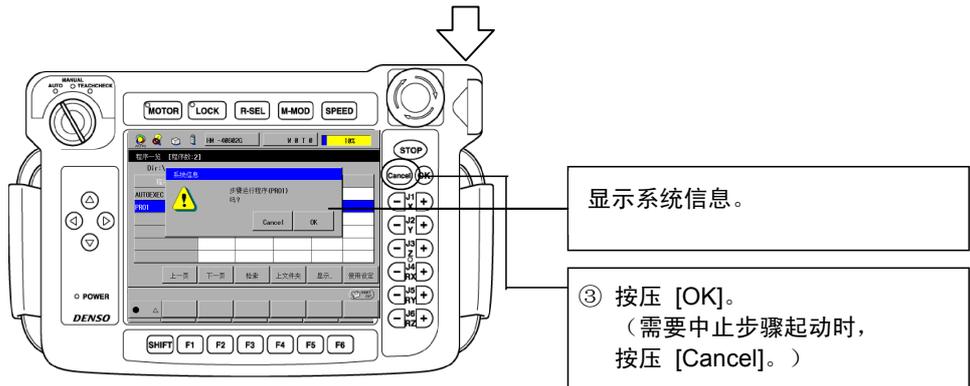
★注意★

在显示程序的同时需要步骤启动时，要在步骤运行之前按压 [F11 显示]。



① 确认需要启动的程序已被选择。

② 按压 [F6 STEP 进>]。
(用右光标也可以单步运行。)



显示系统信息。

③ 按压 [OK]。
(需要中止步骤启动时,
按压 [Cancel]。)

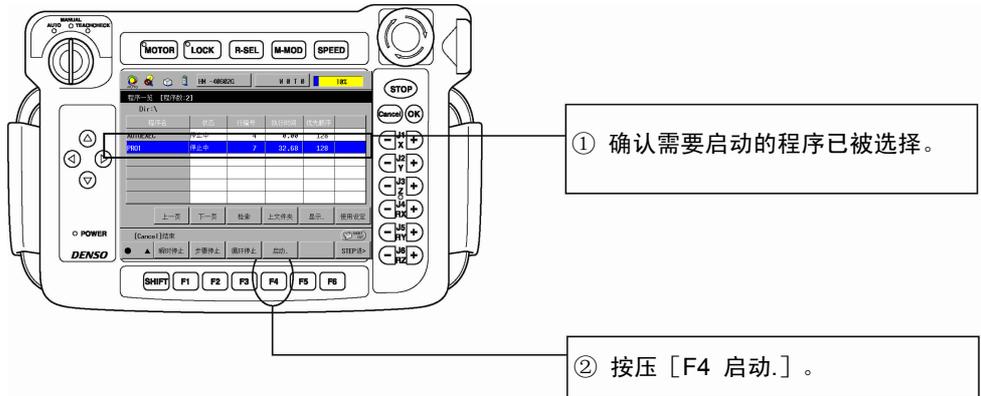
★注意★

确认程序时，请确保随时都可以按压 [STOP] 的状态运行。

PRO1的程序在自动模式下步骤启动。
反复进行上述的操作，确认动作并将程序执行到最后。

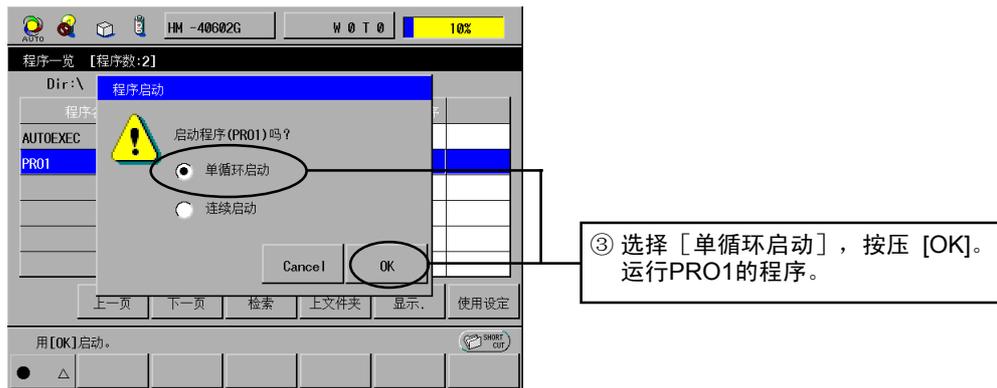
12.3.4 循环启动

确认步骤启动之后，进行循环启动。



★注意★

确认程序时，请确保随时都可以按压 [STOP] 的状态运行。



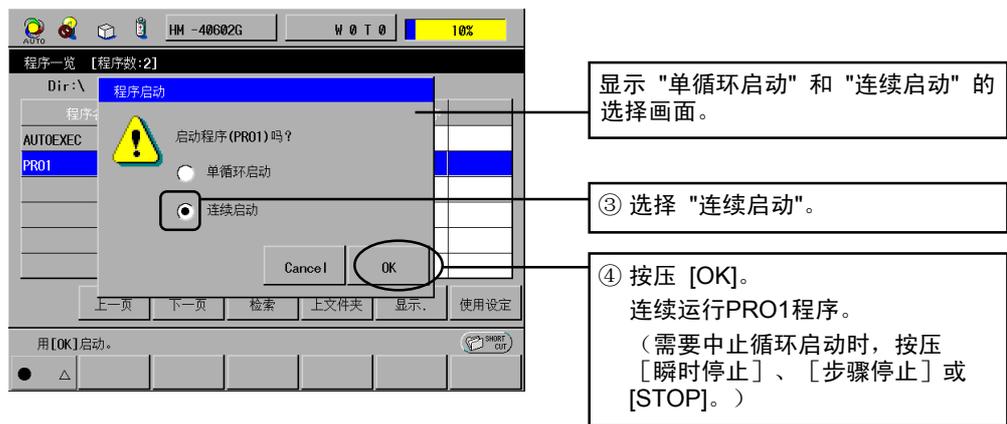
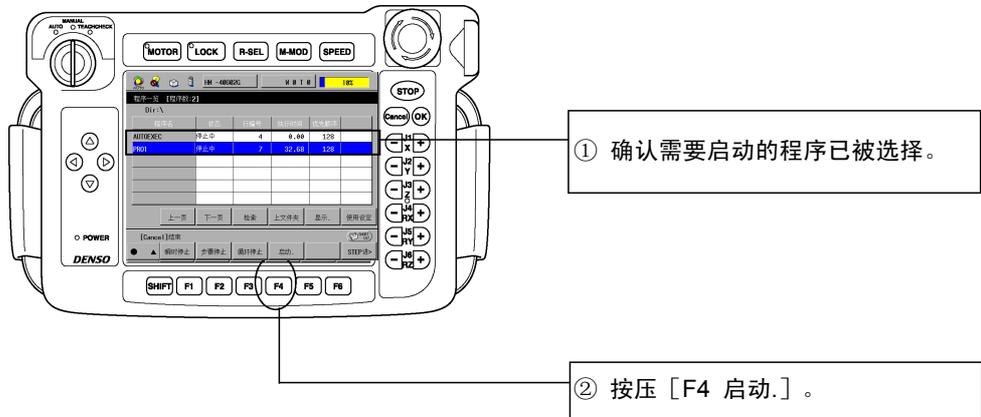
执行到程序结束，则停止。

★注意★

显示的执行时间是从程序开始到结束的时间，也包括步骤停止、瞬时停止所造成的暂时停止时间。

12.3.5 循环起动

循环起动程序。



★注意★

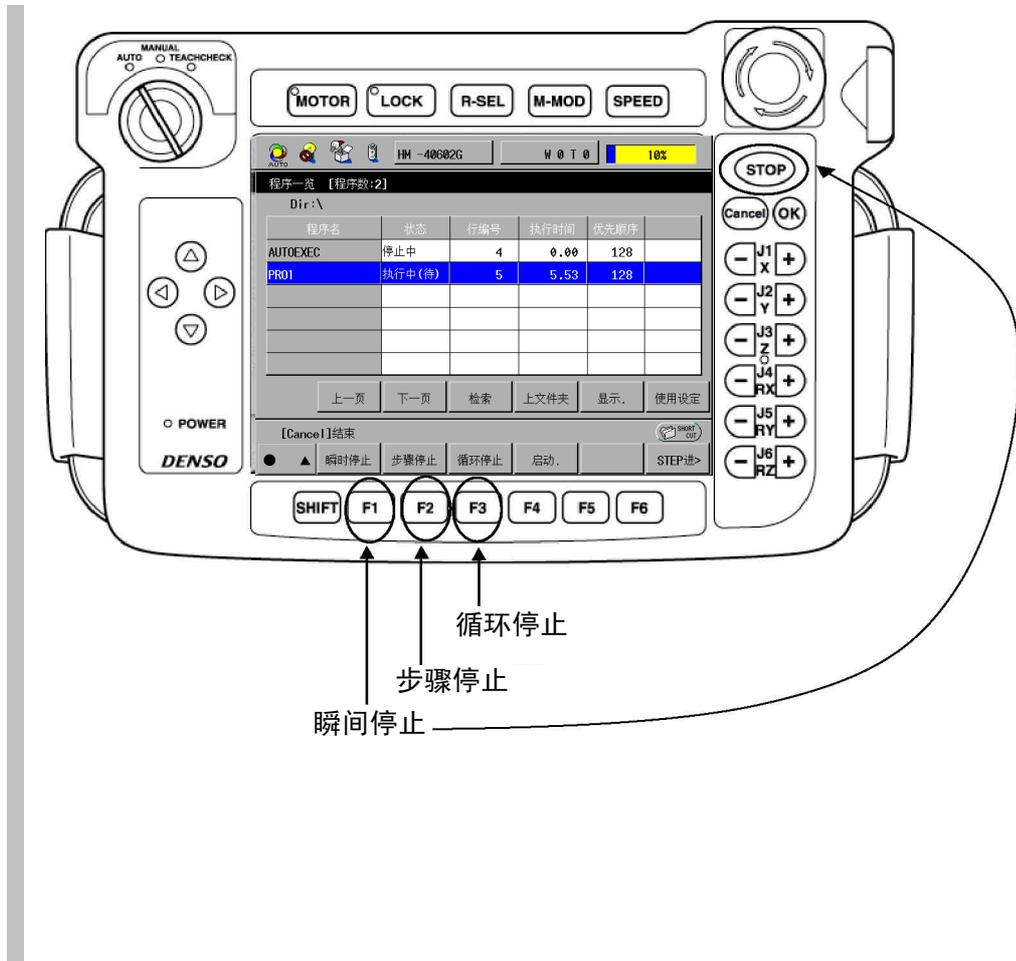
确认程序时, 请确保随时都可以按压 [STOP] 的状态运行。



以上, 使用教导器运行机械手的操作结束。

12.4 机械手的停止操作

在机械手的停止操作中，有以下介绍的4种操作。在此，介绍各自的含义和操作方法。



12.4.1 循环停止 [F3]

如果执行循环停止，则用程序的END命令停止。在以循环启动方式运行机械手时使用。但是电机电源没有关闭。

12.4.2 步骤停止 [F2]

如果执行步骤停止，则按压按钮时所正在执行的步骤结束之后停止。此时电机电源并不关闭。

12.4.3 瞬时停止 [F1], [STOP]

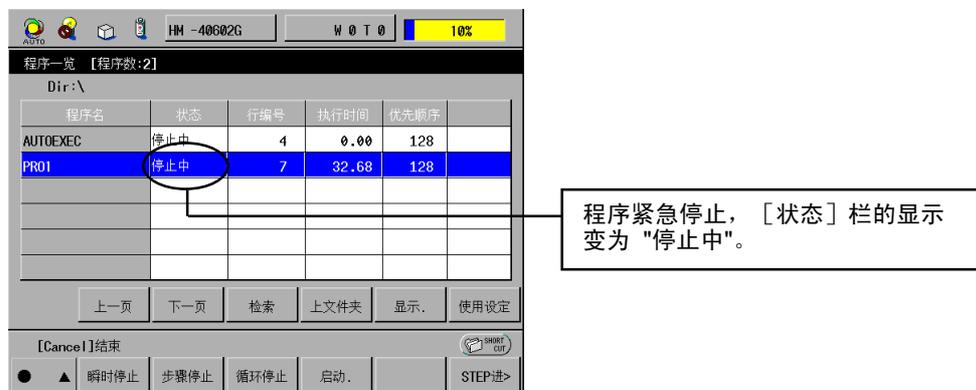
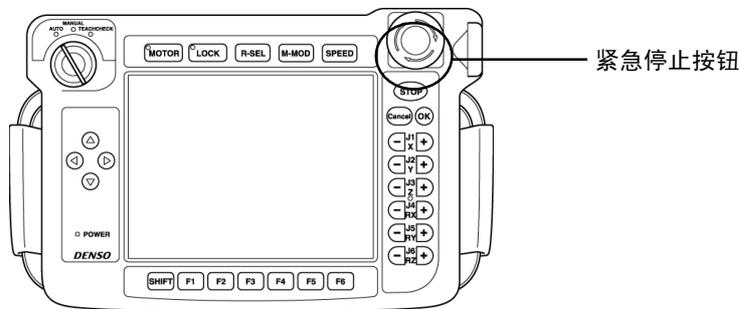
如果执行瞬时停止、则会立即停止。此时电机电源并不关闭。

12.4.4 紧急停止（机械手停止）

如果按压紧急停止，则机械手紧急停止。这种情况下，电机电源被切断。在需要紧急停止机械手运行时使用。

步骤 1

让 "机械手停止"



如果机械手停止之后就那样进行再启动操作，则从被选择的程序的开头开始执行。

重新启动是将电机置于 ON，由 "步骤启动"、"循环启动"、"连续启动" 中的任何一个来决定。

第 13 章 从外围机器启动机械手的方法

13.1 I/O 配置模式的确认方法

当前设定的配置模式可以通过多功能教导器的 I / O 进行确认。由于从外部机器（PLC 等）进行启动的方法会因 I / O 配置模式而改变，所以需要在事先对设定状态进行确认。

操作路径：[F4 I/O]

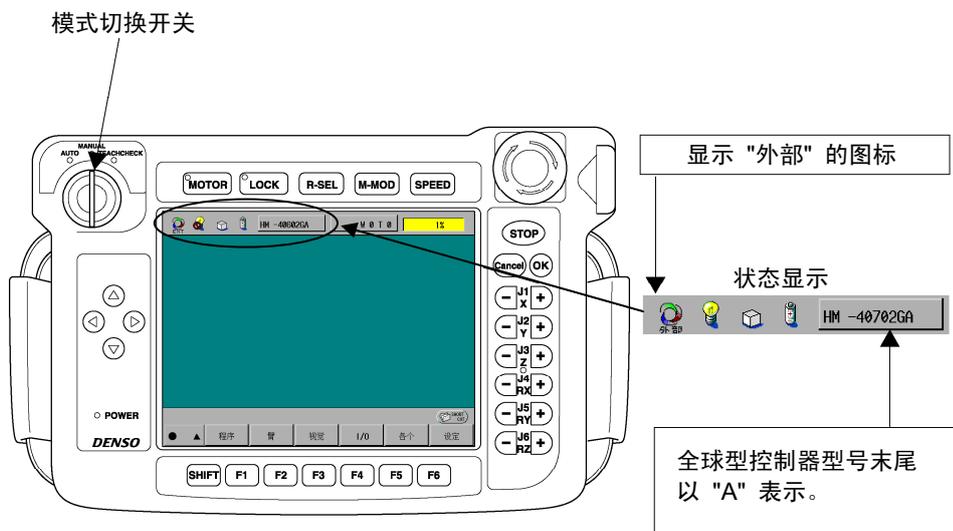
在此显示当前设定的 I/O 配置模式。



13.2 使用全球型控制器时的注意事项

为了从外部机器（PLC 等）上进行启动，需要将单一位置控制功能设定为 "外部自动限定模式"。只需要将模式切换开关切换为 AUTO 就可以对状态显示是否为 "外部" 进行确认。

需要变更设定时，请参照 "2.2.2 单一位置控制功能"。



13.3 通过 Mini I / O 专用模式的启动方法

在 Mini I / O 模式下，通过 "指令空间 (3 比特)" 和 "数据空间 (3 比特)" 的比特组合 (I / O 指令) 命令程序启动等。

另外，I / O 指令使用选通信号开始进行指令执行。

13.3.1 Mini I / O 指令专用输入信号的种类与功能

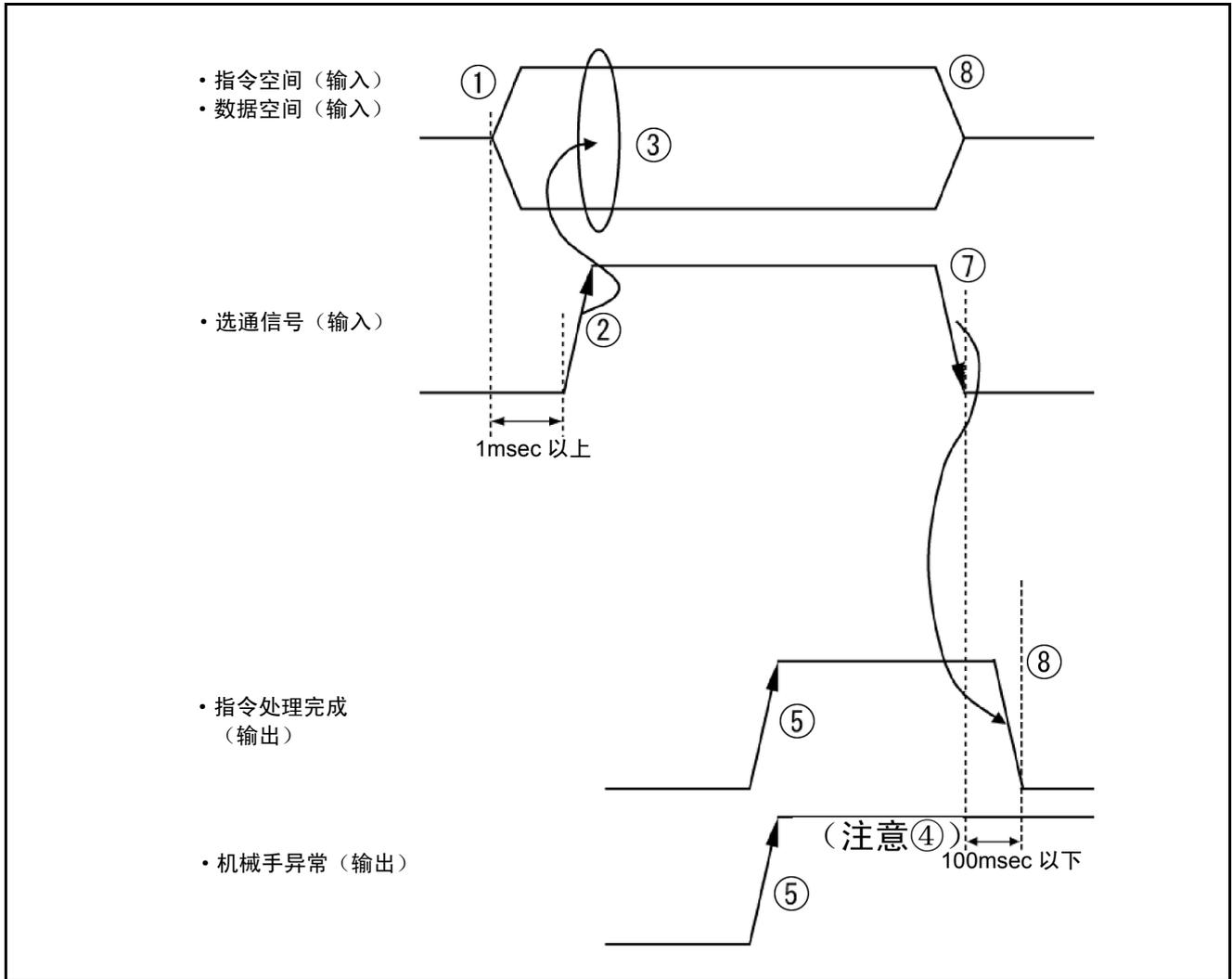
Mini I / O 模式下的专用输入信号如下表所示。

用途		输入信号名		功能	
		指令空间 (3 比特)	数据空间 (3 比特)		
开始	电机 ON CAL 执行	000	001	+ 选通信号	接通电机电源
	外部速度 (SP) 100	000	010		将外部速度设定为 100%
	外部模式切换	000	100		设置为外部模式
	电机 ON 执行 CAL 外部速度 (SP) 100 外部模式切换	000	111		全部进行上述 (起动) 内容
程序运行		010	程序编号 (0 ~ 7)		执行指定的程序
连续开始		011	---		执行连续开始
指定程序复位		100	程序编号 (0 ~ 7)		瞬时停止指定程序，进行程序复位 (初始化)
所有程序复位		101	---		瞬时停止所有程序，进行程序复位 (初始化)
清空机械手异常		001	---		解除 ERROR
停止		---	---	步骤停止 (所有任务)	通过信号释放 (OFF)，将运行中的所有 步骤停止

注意："+ 选通信号" 表示对指令空间 (3 比特) 和数据空间 (3 比特) 进行组合使用。

13.3.2 Mini I/O 指令处理方法

为了运行 I/O 指令，需要进行下图所示的处理。



I/O指令处理方法概要

① 对于机械手控制器的指令执行输出信号，从外部设定指令空间、数据空间（仅在需要时）。

② 设定结束之后，将选通信号 OFF → ON。

注意 ① 在①设定的数据需要在启动选通信号超出1msec之前进行确定。
 ② 在等待专用输出“机械手初始化完成”输出之后，再用选通信号进行指令输入。但是，清空机械手异常指令执行时，因为机械手初始化完成不被输出，所以机械手异常输出之后，请运行清空机械手异常指令。

- ③ 控制器用选通信号输入读入指令空间、数据空间。
- ④ 控制器根据读入的指令进行处理。
- ⑤ 完成指令处理之后，控制器将指令处理完成信号起动至 OFF → ON。
在处理过程中发生错误时，与指令处理结束信号同时输出机械手异常。
- ⑥ PLC 等待指令处理完成信号输入，确认机械手是否发生异常。
- ⑦ PLC 将指令、数据空间以及选通信号置于 OFF 状态。
- ⑧ 通过选通信号的 ON → OFF 的起动，控制器将指令处理完成输出置于 OFF 状态。伴随指令处理错误输出的机械手异常，在机械手异常清空指令执行之前保持 ON 状态。

注意 ③ 在⑧将选通信号ON → OFF之后，指令处理完成信号变为OFF所需的最长时间是100msec。

④ ⑤的指令处理完成信号OFF → ON之前，选通信号为OFF的情况，指令处理完成信号一旦输出之后，在100msec以内变为OFF状态。

13.3.3 Mini I / O 专用输出信号的种类与功能

Mini I / O 专用模式下的专用输出信号如下表所示。

Mini I / O 专用模式下的专用输出信号的种类与功能

用途	信号名称	功能
开始	机械手初始化完成（输出）	"运行准备指令" 将可以运行的状态向外部输出。
	自动模式（输出）	输出自动模式状态。
	运行准备完成（输出）	电机电源接通，并且输出外部自动模式状态。
程序运行	机械手运行中（输出）	输出运行过程中（程序运行过程中的任务为一个以上）状态。
错误、警告	CPU 正常（输出）	机械手控制器的CPU正常时输出。
	机械手异常（输出）	伺服异常、程序异常等，机械手发生异常时，输出。
	电池耗尽警告（输出）	编码器备份电池或存储备份电池的电压下降时，输出。
连续功能	连续开始许可（输出） 注：需要用 I / O 硬件设定进行设定	连续开始可以运行时，输出。
紧急停止电路 (标准型控制器)	紧急停止输出 (2 系统)	输出紧急停止状态。
	教导器紧急停止输出 (2 系统)	输出多功能教导器、小型教导器的紧急停止按钮的状态。
	双重安全开关输出 (允许开关输出) (2 系统)	输出多功能教导器、小型教导器的双重安全开关（允许开关）的状态。
安全回路 (全球型控制器)	教导器紧急停止输出 (2 系统)	输出多功能教导器、小型教导器的紧急停止按钮的状态。
	双重安全开关输出 (允许开关输出) (2 系统)	输出多功能教导器、小型教导器的双重安全开关（允许开关）的状态。
	接触器接点监视器输出	输出控制器内电机接触器的辅助接点的状态。这是电机ON时为ON、电机OFF时为OFF的接点输出。

13.4 用标准模式进行启动的方法

在标准模式下，通过 "指令空间 (4 比特)"、"数据空间 1 (8 比特)" 和 "数据空间 2 (16 比特)" 的比特组合 (I/O 指令) 下达程序启动等命令。
另外，I/O 指令使用选通信号开始进行指令执行。

13.4.1 标准 I/O 指令专用输入信号的种类与功能

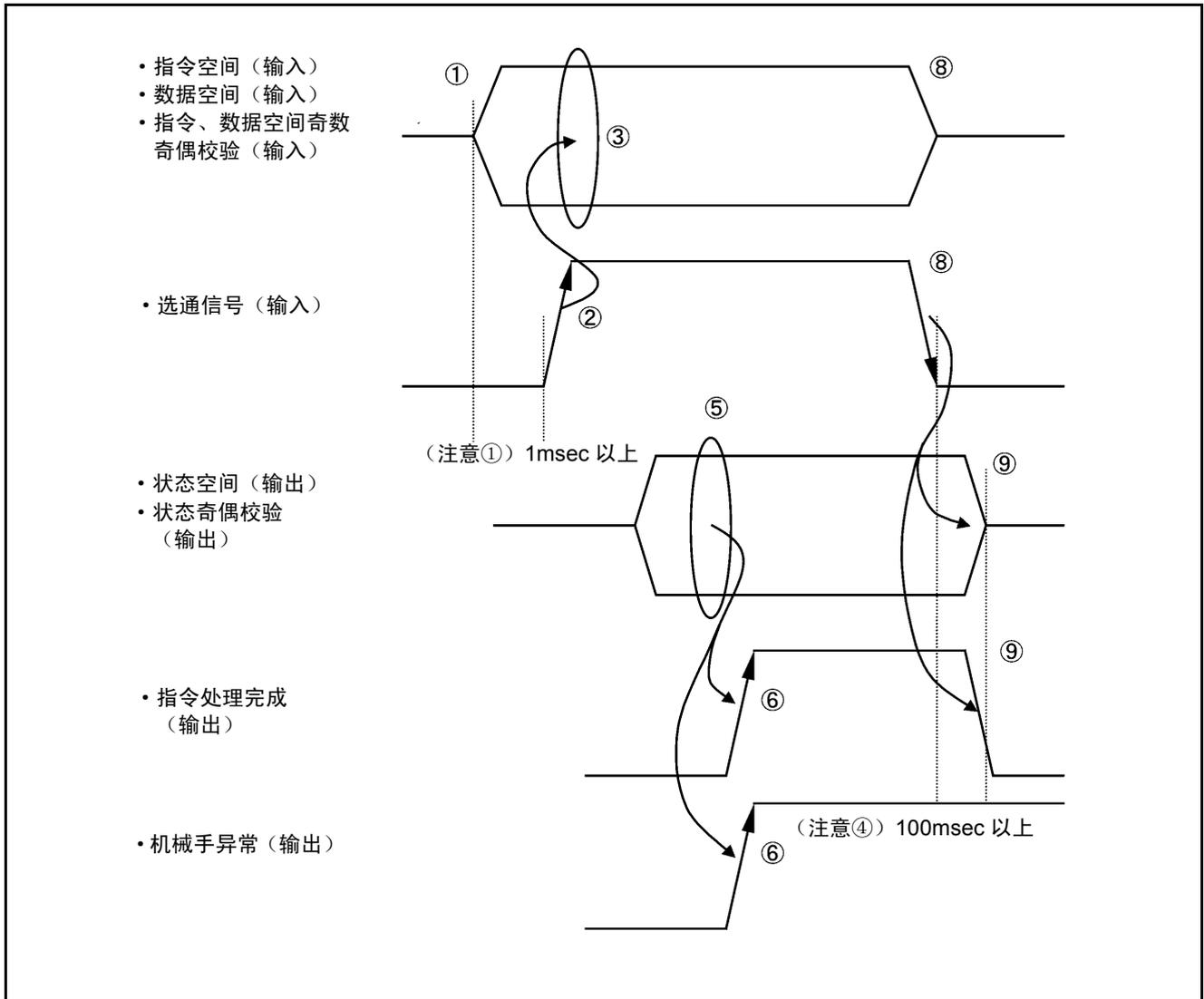
标准模式下的专用输入信号如下表所示。

用途		输入信号名			功能
		指令空间 (4 比特)	数据空间 1 (8 比特)	数据空间 2 (16 比特)	
程序 执行	程序复位 启动	0001	00000001	程序编号 (0~32767)	复位指定的程序后启动
	程序启动	0001	00000010	程序编号 (0~32767)	启动指定的程序。 步骤停止、瞬时停止时，从停止的下一个步骤 开始重新启动
	连续开始	0001	00000100	---	重新启动连续停止而停止的所有程序
	步骤停止	0001	00010000	程序编号 (0~32767)	步骤停止指定的程序
	瞬时停止	0001	00100000	程序编号 (0~32767)	瞬时停止指定的程序
	复位	0001	01000000	程序编号 (0~32767)	瞬时停止指定程序，进行程序复位 (初始化)
外部 速度、 加速度 设定	设定速度	0010	00000001	速度设定值 (1~100)	变更为指定的速度设定值 (1~100)
	设定加速度	0010	00000010	加速度设定值 (1~100)	变更为指定的加速度设定值 (1~100)
	设定减速度	0010	00000100	减速度设定值 (1~100)	变更为指定的减速度设定值 (1~100)
错误读出		0100	---	---	+ 选通信号 将当前发生的错误编号输出到状态空间
写入 I 型变量		0101	变量编号 (0~255)	变量设定值 (-32768~ 32768)	在指定的 I 型变量编号 (0~255) 上变更变量 设定值 (-32768~32768)
读出 I 型变量		0110	变量编号 (0~255)	---	将指定的 I 型变量编号 (0~255) 的当前值输出 到状态空间
开始	电机 ON CAL 执行	0111	00000001	---	接通电机电源
	外部速度 (SP) 100	0111	00000010	---	将外部速度设定为 100%
	外部模式切换	0111	10000000	---	设置为外部模式
	电机 ON 执行 CAL 外部速度 (SP) 100 外部模式切换	0111	10000011	---	全部进行上述 (起动) 内容
机械手 错误清空		1000	---	---	解除 ERROR
I/O 写入		1001	00000000 ~11111111	内部 IO 编号 (128~504)	将数据空间 1 的状态赋值到数据空间 2 上所指定 的编号开始的内部 IO 空间
I/O 读入		1010	---	内部 IO 编号 (128~504)	将数据空间 2 上所指定的编号开始的内部 IO 状态 输入到状态空间末尾的 8 比特上
停止	---	---	---	机械手停止	在信号释放 (OFF) 状态下，机械手停止
	---	---	---	瞬时停止 (所有任务)	通过信号释放 (OFF)，将运行中的所有程序瞬时 停止
	---	---	---	步骤停止 (所有任务)	通过信号释放 (OFF)，将运行中的所有步骤停止

注意: "+ 奇偶性 (奇数)" 是在指令空间、数据空间1、数据空间2的合计比特数为偶数时, 输入奇偶性 (奇数) 信号, 将合计比特数设为奇数。
 注意: "+ 选通信号" 表示指令空间、数据空间1、数据空间2、奇偶性 (奇数) 进行组合使用。

13.4.2 标准 I/O 指令处理方法

为了运行 I/O 指令, 需要进行下图所示的处理。



I/O指令处理方法概要 (标准模式)

- ① 对于机械手控制器的指令执行输出输入信号, 从外部设定指令空间、数据空间 (仅在需要时)、指令、数据空间奇偶性 (奇数)。
- ② 设定结束之后, 将选通信号 OFF → ON。

注意 ① 在①设定的数据需要在启动选通信号超出1msec之前进行确定。
 ② 在等待专用输出“机械手初始化完成”输出之后, 再用选通信号进行指令输入。

- ③ 控制器用选通信号输入读取指令空间、数据空间、指令、数据空间奇偶性（奇数）。
- ④ 控制器根据读入的指令进行处理。
- ⑤ 指令为状态输出时，控制器设定状态空间及状态奇偶校验。
- ⑥ 完成指令处理设定状态空间之后，控制器将指令处理完成信号起动至 OFF → ON。
在处理过程中发生错误时，与指令处理结束信号同时输出机械手异常。
- ⑦ PLC 等待指令处理结束信号输入，如果需要则可以获取状态空间的状态。此时，也要确认机械手是否发生异常。
- ⑧ PLC 状态读入结束之后，将指令、数据空间以及选通信号置于 OFF 状态。
- ⑨ 通过选通信号 ON → OFF 的启动，控制器将状态空间和指令处理完成输出置于 OFF 状态。
伴随指令处理错误输出的机械手异常，在机械手异常清空指令执行之前保持 ON 状态。

<p>注意 ③ 在⑧将选通信号ON → OFF之后，状态空间和指令处理完成信号变为OFF所需的最长时间是100msec。</p> <p>④ ⑥的指令处理完成信号OFF → ON之前，选通信号为OFF的情况，指令处理完成信号和状态空间一旦输出之后，在100msec以内变为OFF状态。</p>

13.4.3 标准 I/O 专用输出信号的种类与功能

标准模式下的专用输出信号如下表所示。

标准模式下的专用输出信号的种类与功能

用途	信号名称	功能
开始	机械手初始化完成	处于运行准备可以开始的状态时进行输出
	自动模式	机械手为自动模式时进行输出
	外部模式	机械手为外部模式时进行输出
	伺服 ON 状态	电机电源接通状态时进行输出
程序运行	机械手运行中	机械手运行过程中（程序执行过程中）进行输出
错误、警告	CPU 正常	机械手控制器的CPU的硬件正常时进行输出
	机械手异常	发生伺服异常、程序异常等重大异常时输出。
	机械手警告	发生轻微异常时进行输出
	电池耗尽警告	编码器备份电池或存储器备份电池的电压下降时输出。
连续功能	连续开始允许	可执行连续开始时输出。
SS 功能	SS 模式	在SS模式时进行输出 参照操作指南 "3.4.6 SS（安全启动）功能"
[紧急停止]	[紧急停止]	紧急停止电路的构成所需要的专用接点输出
I/O 指令处理	指令处理完成	输出I/O指令处理结束
	状态空间 奇偶性 (奇数)	状态空间（16比特）的输出比特合计为偶数时，输出奇偶性（奇数）
	状态空间 (16 比特)	输出错误读出、I型变量读出、I/O读入处理的结果

13.5 用标准模式进行启动的方法

在互换模式下，程序启动等的功能在每一比特上都赋予一定意义，并对该比特最先下达指示。

13.5.1 互换模式专用输出信号的种类与功能

互换模式下的专用输入信号如下表所示。

互换模式下的专用输出信号的种类与功能

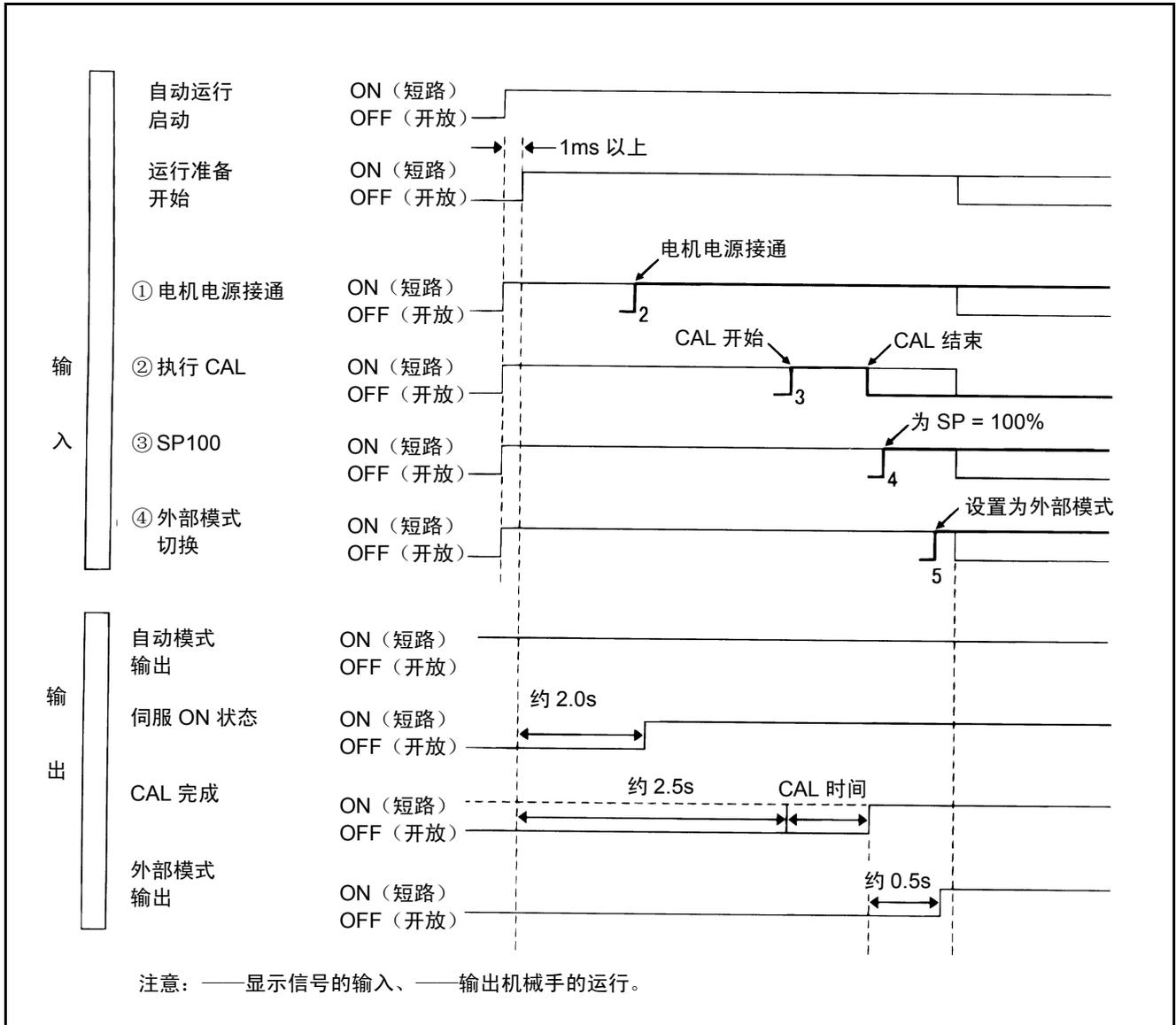
用途	信号名称	功能
开始	自动运行允许	可切换为自动模式。
	电机电源接通 + 运行准备开始	接通电机电源。
	执行 CAL + 运行准备开始	执行CAL。
	SPIOO + 运行准备开始	将速度设定为100%。
	外部模式切换 + 运行准备开始	设置为外部模式
	程序清零 + 运行准备开始	初始化处于停止状态的所有程序。 初始化后程序启动时，从程序的前头执行。
	程序 No. 选择 + 程序开始	执行指定的程序。
程序运行	程序清零 + 程序 No. 选择 + 程序开始	取消当前的程序，从最前面开始执行指定的程序。
	机械手停止	在信号释放状态下机械手停止。
停止	机械手停止	在信号释放状态下机械手停止。
	步骤停止	通过信号释放，步骤停止所有程序。
	瞬时停止	通过信号释放，瞬时停止所有程序。
错误解除	清空机械手异常 + 运行准备开始	解除ERROR。
程序中中断	中断跳跃	中止当前步骤的执行，开始下一个步骤的执行。
连续开始	连续开始信号 + 程序开始	执行连续开始。

注意：在信号名栏上记载多个信号名是为了表示组合使用。

13.5.2 互换模式的处理方法

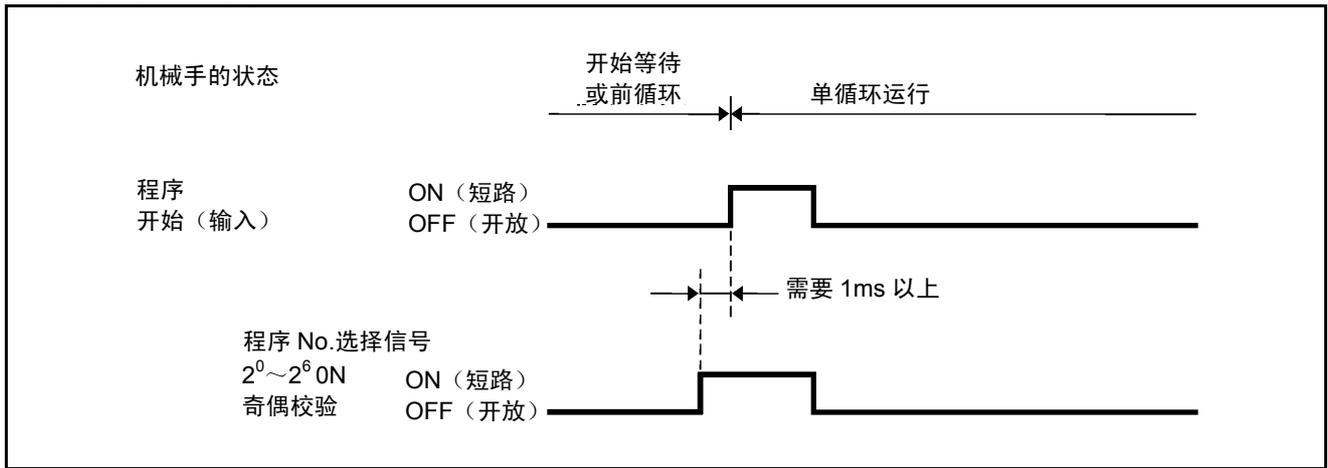
为了执行各功能，要进行下图所示的处理。

• 运行准备开始的示例



运行准备开始信号的时间图示例（互换模式）

• 程序开始的示例



程序No.选择信号 (互换模式)

13.5.3 互换 I/O 专用输出信号的种类与功能

互换模式下的专用输入信号如下表所示。

互换模式下的专用输出信号的种类与功能

用途	信号名称	功能
开始	机械手电源已投入	处于运行准备可以开始的状态时进行输出
	自动模式	机械手为自动模式时进行输出
	伺服 ON 状态	电机电源接通状态时进行输出
	CAL 完成	校准结束时进行输出
	外部模式	机械手为外部模式时进行输出
程序执行开始之前进行校验	教导演示	机械手处于手动模式或教导检查模式时进行输出
程序运行	程序开始 清零	受理程序开始信号，在开始执行程序时进行输出
	机械手运行中	机械手运行过程中（程序执行过程中）进行输出
程序结束	1 个循环结束	程序结束1个循环时进行输出
错误、警告	CPU 正常	机械手控制器的CPU（硬件）正常时进行输出
	机械手异常	伺服异常、程序异常等，机械手发生异常时，输出
	机械手警告	发生轻微异常时进行输出
	电池耗尽警告	编码器备份电池或存储器备份电池的电压下降时，输出
	错误编号	发生错误时，以BCD编码输出错误编号
连续功能	连续开始 允许	可以连续开始时进行输出
SS 功能	SS 模式	在SS模式时进行输出
紧急停止	紧急停止	紧急停止电路的构成所需要的专用接点输出

13.6 I/O 配置表

请对应所使用的配置模式并参照必要的 I/O 配置表。

关于参照的配置表，请参照 "3.3.1 不同配置模式的 I/O 配置一览" 的不同配置模式各个增设卡的配置表。关于 I/O 增设卡的配置，请参照选件机器说明书中的 [RC7M 用 I/O 增设卡]。

注意：在配置模式下使用 "I/O 转换 Box (互换)" 或 "I/O 转换 Box (标准)" 时，配置是不同的。请参照 RC7M 型控制器说明书中的 "第 8 章 使用 I/O 转换箱时的 I/O 配置表 (仅限标准型控制器)"。

13.6.1 HAND I/O (CN9)：全部模式通用的配置

在 RC7M 控制器上，按标准配置了夹治具 I/O (CN9) 信号。该夹治具 I/O 与模式选择种类无关，为全部模式通用配置。

■ HAND I/O (CN9)：NPN 型 I/O

端子No.	信号名称	端口编号	线色		端子No.	信号名称	端口编号	线色	
			标准	强韧				标准	强韧
1	夹治具输出	64	黑	蓝	11	夹治具输入	50	粉	白
2	夹治具输出	65	棕	黄	12	夹治具输入	51	粉	白
3	夹治具输出	66	黑	绿	13	夹治具输入	52	白	白
4	夹治具输出	67	棕	红	14	夹治具输入	53	白	白
5	夹治具输出	68	红	紫	15	夹治具输入	54	白	白
6	夹治具输出	69	橙	蓝	16	夹治具输入	55	白	棕
7	夹治具输出	70	黄	黄	17	夹治具用内部电源输出 E24V	—	白	棕
8	夹治具输出	71	绿	绿	18	夹治具用内部电源输出 E0V	—	白	棕
9	夹治具输入	48	蓝	红	19	未连接	—	白	棕
10	夹治具输入	49	紫	紫	20	未连接	—	白	棕

■ HAND I/O (CN9)：PNP 型 I/O

端子No.	信号名称	端口编号	线色		端子No.	信号名称	端口编号	线色	
			标准	强韧				标准	强韧
1	夹治具输出	64	黑	蓝	11	夹治具输入	50	粉	白
2	夹治具输出	65	棕	黄	12	夹治具输入	51	粉	白
3	夹治具输出	66	黑	绿	13	夹治具输入	52	白	白
4	夹治具输出	67	棕	红	14	夹治具输入	53	白	白
5	夹治具输出	68	红	紫	15	夹治具输入	54	白	白
6	夹治具输出	69	橙	蓝	16	夹治具输入	55	白	棕
7	夹治具输出	70	黄	黄	17	夹治具用内部电源输出 E0V	—	白	棕
8	夹治具输出	71	绿	绿	18	夹治具用内部电源输出 E24V	—	白	棕
9	夹治具输入	48	蓝	红	19	未连接	—	白	棕
10	夹治具输入	49	紫	紫	20	未连接	—	白	棕

13.6.2 Mini I/O（标准型控制器 CN5）：专用配置

端子 No.	信号名称	端口 编号	线色	端子 No.	名称	端口 编号	线色
1	自动允许输入（内部 +24V）	—	黑	35	自动允许	—	粉
2	外部紧急停止输入1b-1 （内部 +24V）	—	棕	36	外部紧急停止输入1b-2	—	粉
3	外部紧急停止输入2b-1 （内部 +24V）	—	红	37	外部紧急停止输入2b-2	—	粉
4	预约	—	橙	38	预约	—	粉
5	预约	—	黄	39	预约	—	粉
6	紧急停止输出1-1（小型继电器输出）	—	黑	40	紧急停止输出1-2（小型继电器输出）	—	白
7	紧急停止输出2-1（小型继电器输出）	—	棕	41	紧急停止输出2-2（小型继电器输出）	—	白
8	双重安全开关输出1-1 （小型继电器输出） （允许开关输出1-1）	—	红	42	双重安全开关输出1-2 （小型继电器输出） （允许开关输出1-2）	—	白
9	双重安全开关输出2-1 （小型继电器输出） （允许开关输出2-1）	—	橙	43	双重安全开关输出2-2 （小型继电器输出） （允许开关输出2-2）	—	白
10	—	—	黄	44	—	—	白
11	步骤停止（所有任务）（输入）	0	绿	45	CPU正常（输出）	16	白
12	选通信号（输入）	1	蓝	46	机械手运行中（输出）	17	白
13	数据空间 第0比特（输入）	2	紫	47	机械手异常（输出）	18	白
14	数据空间 第1比特（输入）	3	灰	48	机械手初始化完成（输出）	19	白
15	数据空间 第2比特（输入）	4	粉	49	自动模式（输出）	20	白
16	指令空间 第0比特（输入）	5	黑	50	运行准备完成（输出）	21	灰
17	指令空间 第1比特（输入）	6	黑	51	电池耗尽警告（输出）	22	紫
18	指令空间 第2比特（输入）	7	棕	52	指令处理完成（输出）	23	紫
19	通用输入	8	红	53	通用输出 / 连续开始许可（输出）	24	紫
20	通用输入	9	橙	54	通用输出	25	紫
21	通用输入	10	黄	55	通用输出	26	紫
22	通用输入	11	绿	56	通用输出	27	紫
23	通用输入	12	蓝	57	通用输出	28	紫
24	通用输入	13	灰	58	通用输出	29	紫
25	通用输入	14	粉	59	通用输出	30	紫
26	通用输入	15	棕	60	通用输出	31	灰
27	—	—	红	61	—	—	灰
28	教导器紧急停止输出 1b-1（试输出）	—	橙	62	教导器紧急停止输出 1b-2（试输出）	—	灰
29	教导器紧急停止输出 2b-1（试输出）	—	黄	63	教导器紧急停止输出 2b-2（试输出）	—	灰
30	Mini I/O卡的JP12短路时： 传送导引板用电源 （DC电源输出 +24V）	—	绿	64	Mini I/O卡的JP13短路时： 传送导引板用电源 （DC电源输出 +24V）	—	灰
31	—	—	蓝	65	—	—	灰
32	外部电源设定时：DC电源输入 +24V	—	粉	66	外部电源设定时：DC电源输入0V	—	灰
33	内部电源设定时：DC电源输出 +24V	—	黑	67	内部电源设定时：DC电源输出0V	—	蓝
34	—	—	棕	68	—	—	蓝

13.6.3 Mini I/O（全球型控制器 CN5）：专用配置

端子 No.	信号名称	端口 编号	线 色	端子 No.	名称	端口 编号	线色
1	预约	—	黑	35	预约	—	粉
2	预约	—	棕	36	预约	—	粉
3	预约	—	红	37	预约	—	粉
4	预约	—	橙	38	预约	—	粉
5	预约	—	黄	39	预约	—	粉
6	预约	—	黑	40	预约	—	白
7	预约	—	棕	41	预约	—	白
8	预约	—	红	42	预约	—	白
9	预约	—	橙	43	预约	—	白
10	—	—	黄	44	—	—	白
11	步骤停止（所有任务）（输入）	0	绿	45	CPU正常（输出）	16	白
12	选通信号（输入）	1	蓝	46	机械手运行中（输出）	17	白
13	数据空间 第0比特（输入）	2	紫	47	机械手异常（输出）	18	白
14	数据空间 第1比特（输入）	3	灰	48	机械手初始化完成（输出）	19	白
15	数据空间 第2比特（输入）	4	粉	49	自动模式（输出）	20	白
16	指令空间 第0比特（输入）	5	黑	50	运行准备完成（输出）	21	灰
17	指令空间 第1比特（输入）	6	黑	51	电池耗尽警告（输出）	22	紫
18	指令空间 第2比特（输入）	7	棕	52	指令处理完成（输出）	23	紫
19	通用输入	8	红	53	通用输出 / 连续开始许可（输出）	24	紫
20	通用输入	9	橙	54	通用输出	25	紫
21	通用输入	10	黄	55	通用输出	26	紫
22	通用输入	11	绿	56	通用输出	27	紫
23	通用输入	12	蓝	57	通用输出	28	紫
24	通用输入	13	灰	58	通用输出	29	紫
25	通用输入	14	粉	59	通用输出	30	紫
26	通用输入	15	棕	60	预约	31	灰
27	—	—	红	61	—	—	灰
28	预约	—	橙	62	预约	—	灰
29	预约	—	黄	63	预约	—	灰
30	Mini I/O卡的JP12短路时： 传送导引板用电源 （DC电源输出 +24V）	—	绿	64	Mini I/O卡的JP13短路时： 传送导引板用电源 （DC电源输出 +24V）	—	灰
31	—	—	蓝	65	—	—	灰
32	外部电源设定时：DC电源输入 +24V	—	粉	66	外部电源设定时：DC电源输入0V	—	灰
33	内部电源设定时：DC电源输出 +24V	—	黑	67	内部电源设定时：DC电源输出0V	—	蓝
34	—	—	棕	68	—	—	蓝

13.6.4 Mini I/O（标准型控制器 CN5）：互换 / 标准 / 全通用配置

端子 No.	信号名称	端口 编号	线色	端子 No.	名称	端口 编号	线色
1	自动启动输入（内部+24V）	—	黑	35	自动启动	—	粉
2	外部紧急停止输入1b-1 （内部 +24V）	—	棕	36	外部紧急停止输入1b-2	—	粉
3	外部紧急停止输入2b-1 （内部 +24V）	—	红	37	外部紧急停止输入2b-2	—	粉
4	预约	—	橙	38	预约	—	粉
5	预约	—	黄	39	预约	—	粉
6	紧急停止输出1-1（小型继电器输出）	—	黑	40	紧急停止输出1-2（小型继电器输出）	—	白
7	紧急停止输出2-1（小型继电器输出）	—	棕	41	紧急停止输出2-2（小型继电器输出）	—	白
8	双重安全开关输出1-1 （小型继电器输出） （启动开关输出1-1）	—	红	42	双重安全开关输出1-2 （小型继电器输出） （启动开关输出1-2）	—	白
9	双重安全开关输出2-1 （小型继电器输出） （启动开关输出2-1）	—	橙	43	双重安全开关输出2-2 （小型继电器输出） （启动开关输出2-2）	—	白
10	—	—	黄	44	—	—	白
11	通用输入	0	绿	45	CPU正常（输出）	16	白
12	通用输入	1	蓝	46	通用输出	17	白
13	通用输入	2	紫	47	通用输出	18	白
14	通用输入	3	灰	48	通用输出	19	白
15	通用输入	4	粉	49	通用输出	20	白
16	通用输入	5	黑	50	通用输出	21	灰
17	通用输入	6	黑	51	通用输出	22	紫
18	通用输入	7	棕	52	通用输出	23	紫
19	通用输入	8	红	53	通用输出	24	紫
20	通用输入	9	橙	54	通用输出	25	紫
21	通用输入	10	黄	55	通用输出	26	紫
22	通用输入	11	绿	56	通用输出	27	紫
23	通用输入	12	蓝	57	通用输出	28	紫
24	通用输入	13	灰	58	通用输出	29	紫
25	通用输入	14	粉	59	通用输出	30	紫
26	通用输入	15	棕	60	通用输出	31	灰
27	—	—	红	61	—	—	灰
28	教导器紧急停止输出 1b-1（试输出）	—	橙	62	教导器紧急停止输出 1b-2（试输出）	—	灰
29	教导器紧急停止输出 2b-1（试输出）	—	黄	63	教导器紧急停止输出 2b-2（试输出）	—	灰
30	Mini I/O卡的JP12短路时： 传送导引板用电源 （DC电源输出 +24V）	—	绿	64	Mini I/O卡的JP13短路时： 传送导引板用电源 （DC电源输出 +24V）	—	灰
31	—	—	蓝	65	—	—	灰
32	外部电源设定时：DC电源输入 +24V	—	粉	66	外部电源设定时：DC电源输入0V	—	灰
33	内部电源设定时：DC电源输出 +24V	—	黑	67	内部电源设定时：DC电源输出0V	—	蓝
34	—	—	棕	68	—	—	蓝

13.6.5 Mini I/O（全球型控制器 CN5）：互换 / 标准 / 全通用配置

端子 No.	信号名称	端口编号	线色	端子 No.	名称	端口编号	线色
1	预约	—	黑	35	预约	—	粉
2	预约	—	棕	36	预约	—	粉
3	预约	—	红	37	预约	—	粉
4	预约	—	橙	38	预约	—	粉
5	预约	—	黄	39	预约	—	粉
6	预约	—	黑	40	预约	—	白
7	预约	—	棕	41	预约	—	白
8	预约	—	红	42	预约	—	白
9	预约	—	橙	43	预约	—	白
10	—	—	黄	44	—	—	白
11	通用输入	0	绿	45	CPU正常（输出）	16	白
12	通用输入	1	蓝	46	通用输出	17	白
13	通用输入	2	紫	47	通用输出	18	白
14	通用输入	3	灰	48	通用输出	19	白
15	通用输入	4	粉	49	通用输出	20	白
16	通用输入	5	黑	50	通用输出	21	灰
17	通用输入	6	黑	51	通用输出	22	紫
18	通用输入	7	棕	52	通用输出	23	紫
19	通用输入	8	红	53	通用输出	24	紫
20	通用输入	9	橙	54	通用输出	25	紫
21	通用输入	10	黄	55	通用输出	26	紫
22	通用输入	11	绿	56	通用输出	27	紫
23	通用输入	12	蓝	57	通用输出	28	紫
24	通用输入	13	灰	58	通用输出	29	紫
25	通用输入	14	粉	59	通用输出	30	紫
26	通用输入	15	棕	60	预约	31	灰
27	—	—	红	61	—	—	灰
28	预约	—	橙	62	预约	—	灰
29	预约	—	黄	63	预约	—	灰
30	Mini I/O卡的JP12短路时： 传送导引板用电源 (DC电源输出 +24V)	—	绿	64	Mini I/O卡的JP13短路时： 传送导引板用电源 (DC电源输出 0V)	—	灰
31	—	—	蓝	65	—	—	灰
32	外部电源设定时：DC电源输入	—	粉	66	外部电源设定时：DC电源输入0V 内部电源设定时：DC电源输出0V	—	灰
33	+24V	—	黑	67		—	蓝
34	内部电源设定时：DC电源输出 +24V	—	棕	68		—	蓝

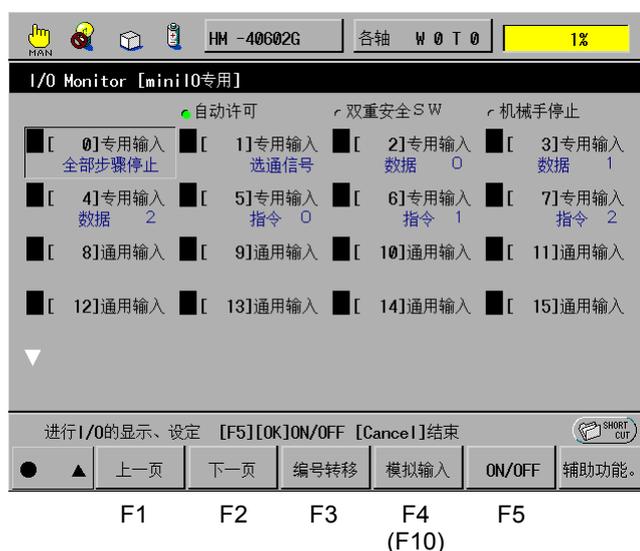
第 14 章 I / O 的监视与操作

可以对专用输入输出、通用输入输出、内部 I / O 的状态进行实时监视，并将通用输出、HAND 输出、内部 I / O 信号的强制输出和通用输入、HAND 输入信号进行模拟的 I / O 输入，对机械手动动作进行仿真。

14.1 由多功能教导器进行的操作

14.1.1 I / O 的监视

如果在基本画面上按压 [F4 I / O]，则显示下图所示的 [I / O Monitor] 视窗。
可以通过该画面对 I / O 的 ON、OFF 状态进行确认。



可以使用的功能键	
[F1 前一页]	显示 I / O 信号清单的前一页。
[F2 下一页]	显示 I / O 信号清单的后一页。
[F3 编号转移]	显示 [I / O 编号转移] 视窗。如果使用数字键输入 I / O 端口地址，按压 [OK]，则显示所需要的输入信号或者输出信号。
[F4 模拟输入]	在选择的系统输入端口上可以接受模拟输入。在对应的输入端口上显示 "!" 记号，在画面上部的状态栏上显示模拟 I / O 图标。该功能在测试程序时使用。
[F5 ON / OFF]	显示 "是否将 I / O [xxx] 置于 ON (或 OFF) ?" 系统讯息。如果按压 [OK] 按钮，则选择的输入端口为 ON (或 OFF)。可以在通用输出、HAND 输出、内部 I / O 上使用。 在指定不能进行 ON / OFF 编号时，会发生 ERROR 21FB (系统空间写入错误) 或 ERROR 73E4 (可以输入输出的范围之外)。
[F10 疑似清空]	解除疑似输入设定。

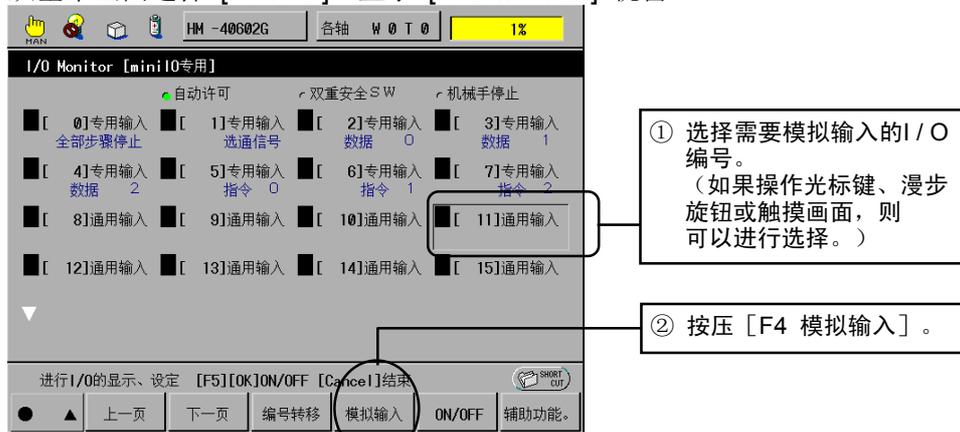
14.1.2 模拟 I / O 的操作方法

只要是通用输入、HAND 输入，就能够对模拟 I / O 输入进行设定。

如果设定模拟 I / O 输入，则可以在多功能教导器上对相应的端口进行 ON、OFF。

· 模拟 I / O 输入的设置

从基本画面选择 [F4 I / O]，显示 [I / O Monitor] 视窗。



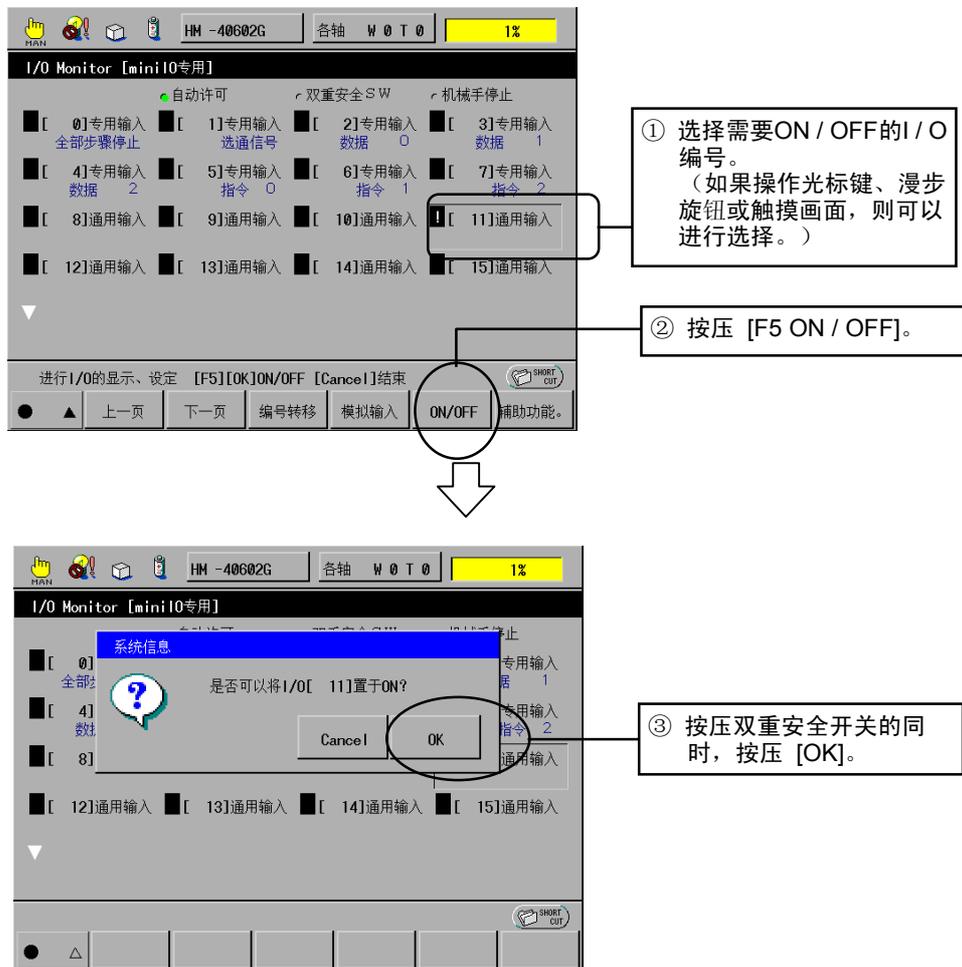
到此，模拟 I / O 输入的设置结束。

★参考★

为了解除模拟 I / O 输入，请再度进行①~③的操作，或进行 [F10 模拟清空] (模拟输入全部清空)。

· 模拟 I/O 的 ON / OFF 方法

在此，关于模拟 I/O 从 OFF 到 ON 的操作方法进行说明。



★注意★

如果对未显示惊叹号 "!" 的 IO 编号 ON / OFF, 则会发生 ERROR 73E4 (超出可输入输出范围)。



★参考★

为了将模拟 I/O 输入置于 OFF, 请重新进行 ①~③ 操作。

14.2 利用 WINCAPSIII的操作

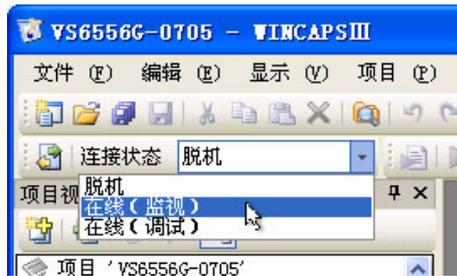
您可利用 WINCAPSIII，监视机械手控制器的 I/O 状态，或利用模拟 I/O 功能，进行程序的验证。

14.2.1 I/O 的监视

利用 WINCAPSIII 监视 I/O 的状态。

步骤 1

打开项目，将连接状态切换至 [在线 (监视)]。



步骤 2

选择 [显示 (V)] - [IO (I)], [I/O] 视窗即会打开。

滚动至想要监视的 I/O，确认 I/O 的状态。



The screenshot shows the 'I/O' monitoring window with a table of I/O points. The '智能显示' (Smart Display) button is active. The table columns are: No., 状态 (Status), 类型 (Type), 用途 (Purpose), 宏名 (Macro Name), 模拟 (Simulation), LOG, and 智能 (Smart).

No.	状态	类型	用途	宏名	模拟	LOG	智能
15	●	通用输入		UIN8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
16	●	专用输出	CPU normal (disable)	SOUT1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
17	●	专用输出	机械手运行过程中	SOUT2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
18	●	专用输出	机械手异常	SOUT3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
19	●	专用输出	机械手初始化结束	SOUT4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
20	●	专用输出	自动模式	SOUT5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
21	●	专用输出	运行准备结束	SOUT6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
22	●	专用输出	电池耗尽警告	SOUT7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
23	●	专用输出	指令处理结束	SOUT8	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
24	●	通用输出		UOUT1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

[状态] 栏显示绿色圆形则为 ON 状态。

步骤 3

利用只能显示功能，可仅显示希望查看的 I/O。

在希望其显示的 I/O 的 [智能] 栏上打勾。

按 [智能显示] 按钮，则只有 [智能] 栏打勾的 I/O 会显示出来。



The screenshot shows the 'I/O' monitoring window with the '智能显示' (Smart Display) button active. The table now only shows the I/O points where the '智能' (Smart) column has a checked box.

No.	状态	类型	用途	宏名	模拟	LOG	智能
18	●	专用输出	机械手异常	SOUT3	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
19	●	专用输出	机械手初始化结束	SOUT4	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
20	●	专用输出	自动模式	SOUT5	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

14.2.2 使用模拟 I/O

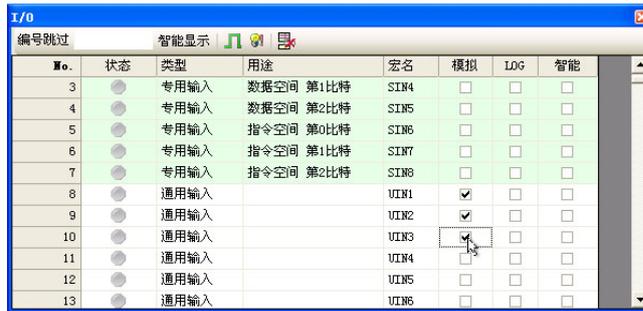
只有通用输入、HAND 输入，可使用能够模拟性地进行 I/O 输入的模拟 I/O 功能。使用模拟 I/O 功能，即可在 WINCAPSIII 的 [I/O] 视窗将 I/O 从 ON 转换至 OFF，或从 OFF 转换至 ON。

步骤 1

打开项目，将连接状态切换至 [在线 (监视)]。

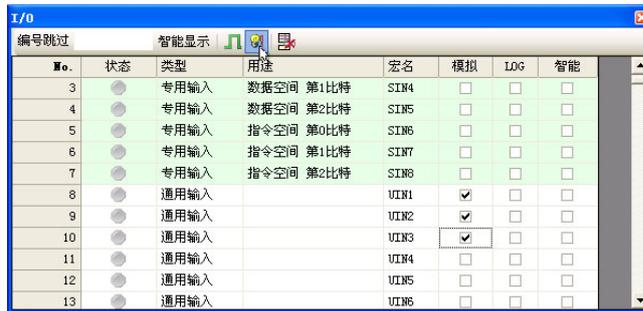
打开 [I/O] 视窗。

在希望使用模拟 I/O 功能之 I/O 的 [模拟] 栏打勾。

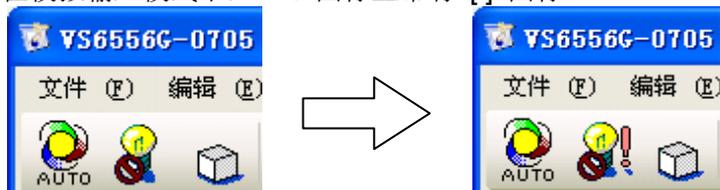


步骤 2

点击 [模拟输入] 按钮，即可通过 WINCAPSIII 操作已在 [模拟] 栏打勾的 I/O 了。



在模拟输入模式下，I/O 图标上带有 [!] 图标。



步骤 3

要想将 I/O 从 ON 转换至 OFF，或从 OFF 转换至 ON，可点击 [状态] 栏。



第 15 章 变量的监视和值的变更

15.1 由多功能教导器进行的操作

15.1.1 全局变量的监视和值的变更

操作路径：[F1 程序] — [F4 变量.]

监视、变更各种变量的值及正在使用的变量。

(1) 如果在 [程序一览] 视窗上按压 [F4 变量.]，则显示下图的 [选择变量类型] 视窗。



(2) 选择需要监视或需要变更的变量的类型。

如果按压 [F1 I. 整数]，则显示以下的 [整数型 (Integer)] 视窗。



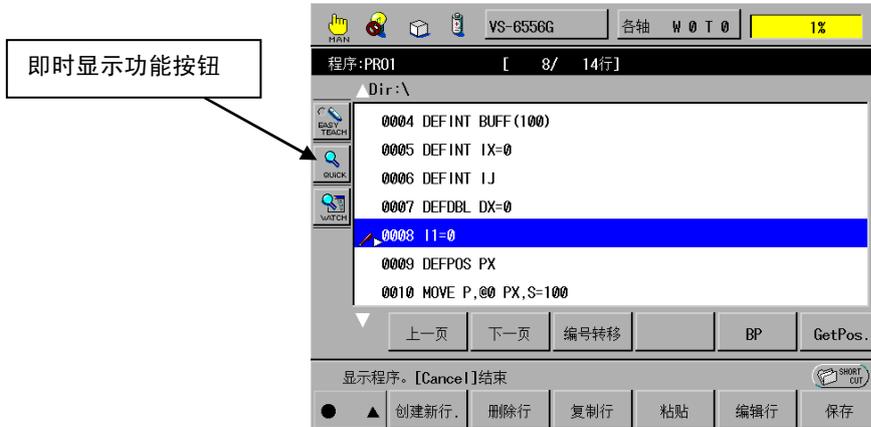
可以使用的功能键	
[F1 上一页]	显示变量清单的前一页。
[F2 下一页]	显示变量清单的后一页。
[F3 编号转移]	显示 [变量编号转移] 视窗。如果使用数字键输入变量名、按压 [OK]，则目的变量名被显示。
[F5 变更值]	显示数字键。如果从数字键上输入赋值变量的值、按压 [OK]，则其值被赋值变量。
[F7 复制变量]	复制所选的变量。
[F12 登录]	将所选变量登录至查看。

注意：在外部自动模式下，不能对值进行变更

15.1.2 局部 (Local) 变量的监视与值的变更

在显示程序文本的画面上，追加了即时显示功能按钮。

通过按压即时显示按钮，可参照当前光标行的局部 (Local) 变量、以及全局变量。

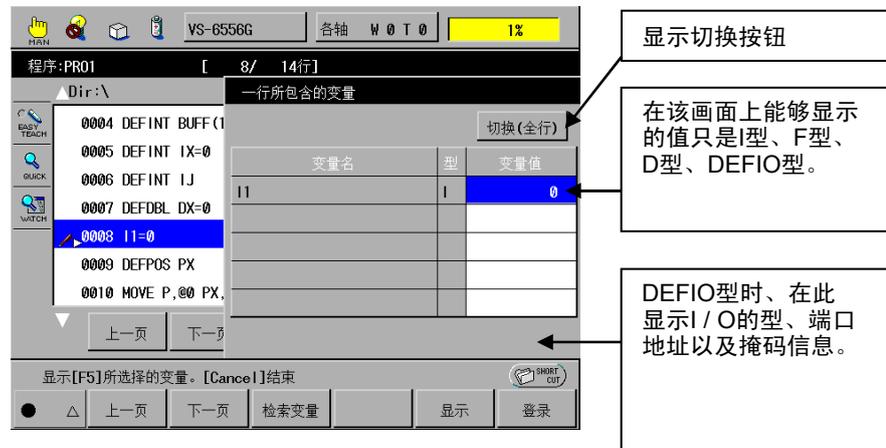


注意：光标行只有用手动模式不能变更。

通过按压即时显示功能按钮，包含在 1 行的变量会被显示。

用该画面，只能看到 I 型、F 型、D 型、DEFIO 型变量。

参照 DEFIO 变量的情况下，<I/O 变量的型>、<端口地址>、<掩码信息> 会被显示。下一示例显示的是步骤停止行的变量 "I1"。



注意1：变量的INDEX在范围之外、以及不是数值的情况下，变量名的INDEX部分显示 "?"。

例1. I型变量的使用数虽是100、但欲显示成I101了。

例2. 像I [slotnum] 那样，INDEX部分欲显示成是宏名的变量了。

INDEX部分在 "?" 的情况下、即使是I型、F型、D型、DEFIO型，其值也不被显示。请按压显示按钮，对准目的的INDEX。

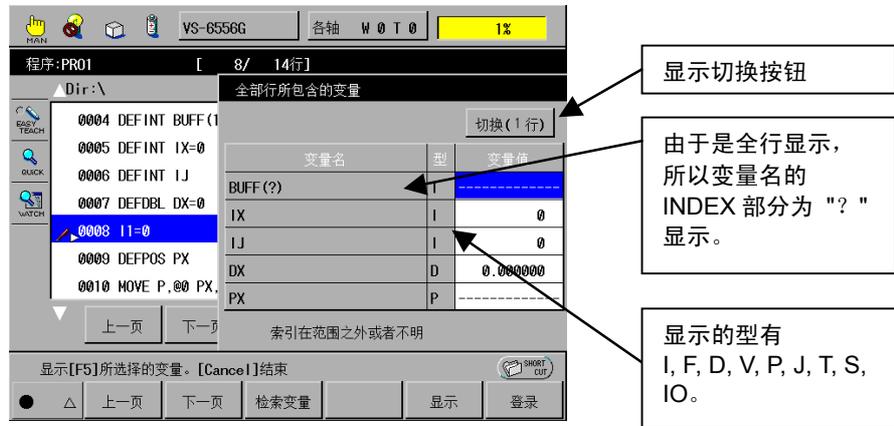
注意2：是DEFIO型变量端口地址在输入输出范围之外的情况下，为灰色显示。

而且，如果是SINGLE型的DEFIO型变量，IO的内容无法显示为单精度实数时，则显示 "NaN"。

注意3：自变量 (argument) 是排列的变量不被显示。

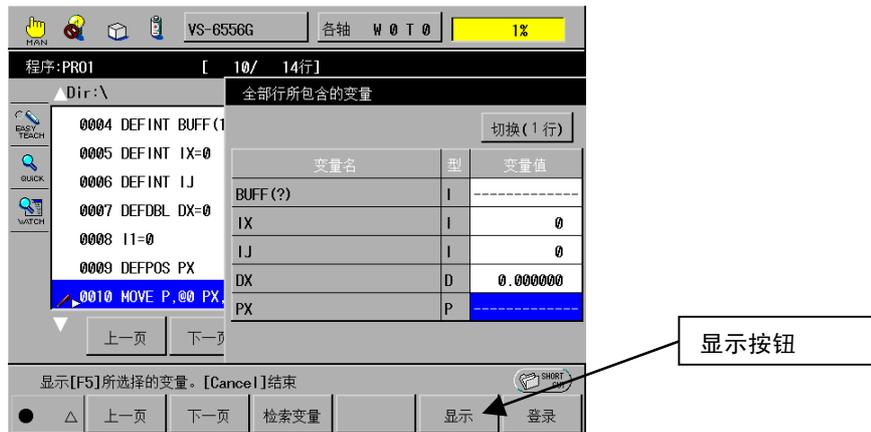
例. 是PROGRAM SUB1 (I1%, I2% (10)) 等的程序的情况下，自变量 (argument) 是排列变量的I2不被显示。

在即时显示功能上，通过 "显示切换按钮"，可以切换 1 行上所包含的变量和全行上所包含的变量的显示。如果切换为全行，则程序内的所有的局部 (Local) 变量会被显示。下一示例是切换为全行上所包含的变量画面的画面。



注意1: 1行显示的情况下，全局变量也会被显示，但全行显示的情况下，全局变量不会被显示。
 注意2: 全行显示的情况下，排列变量所有变量名为INDEX部分会以?显示。请按压显示按钮，向目的的INDEX对合。

通过按压显示按钮，所选择的变量会被显示。



注意: DEFIO变量的端口在范围以外的情况下，不能显示。

通过按压显示按钮，所选择的变量会被显示。

下一个画面是显示P型局部（Local）变量PX（3）的示例。

在该画面上能够执行值变更及变量移动等与全局变量同等的操作。该画面上的变量的登录为 "F12 登录"。



如果按压教导器的SHIFT按钮，则此处为F12的登录。

- 注意1: 变量的INDEX部分为 "?" 的情况下，如果按压显示按钮，则INDEX为0的变量会被显示。请将光标对合目的的INDEX。
- 注意2: 在对DEFIO变量的值进行变更时，需要和通常I / O一样按压双重安全开关。
- 注意3: 没有向局部 (Local) 变量的位置输入。

在值变更时，如果在该画面上按压 [F5 变更值]，则就会显示数字键。如果从数字键上输入赋值变量的值、按压 [OK]，则其值被赋值变量。

注意: 在外部自动模式下，不能对值进行变更

15.1.3 变更全局变量的使用个数

操作路径 [F1 程序] — [F4 变量.] — [F12 使用变量数]

对各个类型的全局变量的使用个数进行变更。

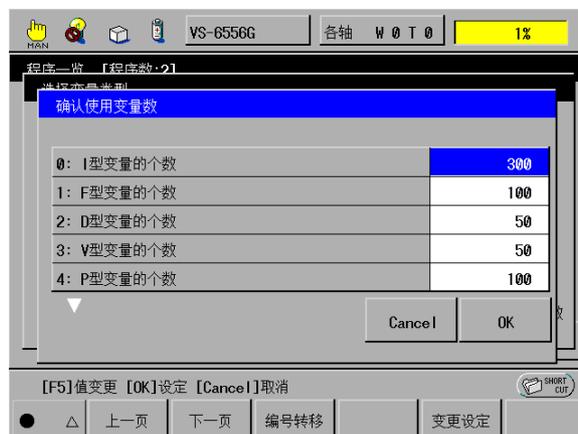
(1) 如果按压 [F12 使用变量数]，则显示以下的 [确认使用变量数] 视窗。



(2) 如果选择想要变更使用个数的项目、按 "F5 变更设定"，则显示以下的 [变更参数] 视窗。



(3) 如果输入变更值后按压 [OK]，则被变更的值就会显示在 [确认使用变量数] 视窗上。



- (4) 确认设定内容之后、如果按压 [OK]，则如下所示显示系统讯息。如果在此按压 [OK]，则进行程序编译。如果程序编译加载正常完成，则变量的使用个数在该时刻被变更。



- (5) 如果在 (4) 中按压 [Cancel]，则变量的使用个数在该时刻不被变更，直到进行下一次程序编译时使用个数的变更才有效。

注：

[关于全局变量的个数]

在该控制器上，只有在执行程序进行加载时，才能对变量的使用个数进行变更。为此，如果变更变量的使用个数的设定，则首先要通过程序编译对反映了变量使用个数的变更的执行文件进行创建，之后再加载。由此从加载完成的时刻开始，可以使用变更了的变量。

15.2 通过 WINCAPSIII的操作

可利用 WINCAPS 监视或编辑机械手控制器的通用变量与局部变量。

15.2.1 全局变量的监视和值的变更

可利用 WINCAPSIII对机械手控制器的全局变量进行监视与数值的变更。

步骤 1

打开项目，将连接状态切换至 [在线 (监视)]。

选择 [显示 (V)] - [变量 (V)]，并选择希望监视的变量型。
所选择的变量型视窗即会显示出来。

No.	值	用途	宏名	智能
0	5			<input type="checkbox"/>
1	300			<input type="checkbox"/>
2	50			<input type="checkbox"/>
3	11			<input type="checkbox"/>
4	1515458			<input type="checkbox"/>
5	0			<input type="checkbox"/>
6	0			<input type="checkbox"/>
7	0			<input type="checkbox"/>
8	0			<input type="checkbox"/>
9	0			<input type="checkbox"/>
10	0			<input type="checkbox"/>
11	0			<input type="checkbox"/>

步骤 2

通过在 [值] 栏输入数值，可对机械手控制器的变量进行编辑。

通过编辑变量可变更机械手控制器内的数据，但无法变更 WINCAPSIII项目的数值。
如希望保存，请利用 [发送接收数据] 从控制器接收变量数据。

15.2.2 局部 (Local) 变量的监视与值的变更

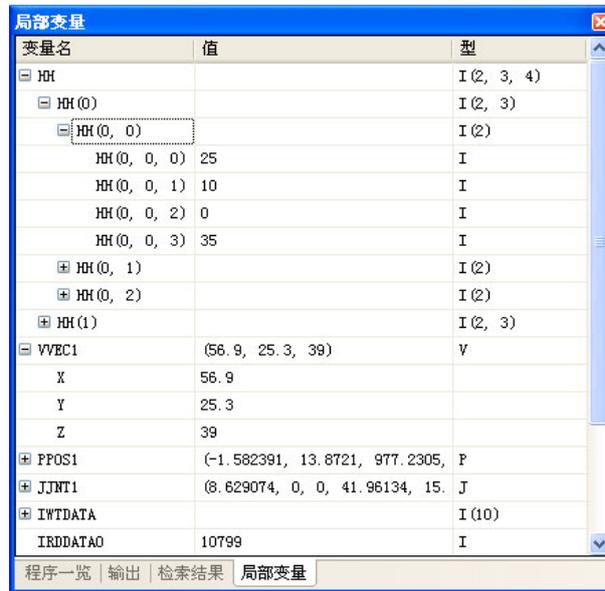
用 WINCAPSIII，可监视机械手控制器的局部变量和变更数值。

步骤 1

打开项目，将连接状态设定为 [在线 (监视)]。

选择 [显示 (V)] - [局部变量 (L)]。

您在 [项目视窗] 或 [程序一览] 视窗中选择的程序的局部变量即会显示出来。



变量名	值	型
HH		I (2, 3, 4)
HH (0)		I (2, 3)
HH (0, 0)		I (2)
HH (0, 0, 0)	25	I
HH (0, 0, 1)	10	I
HH (0, 0, 2)	0	I
HH (0, 0, 3)	35	I
HH (0, 1)		I (2)
HH (0, 2)		I (2)
HH (1)		I (2, 3)
VVEC1	(56.9, 25.3, 39)	V
X	56.9	
Y	25.3	
Z	39	
PPOS1	(-1.582391, 13.8721, 977.2305, P	
JJNT1	(8.629074, 0, 0, 41.96134, 15.	J
IWTDATA		I (10)
IRDDATA0	10799	I

步骤 2

通过在 [值] 栏输入数值，即可对机械手控制器的变量进行编辑。

注意：当利用DEFIO定义通用输入端口与夹治具输入端口时，须事先将I/O设置为模拟输入。

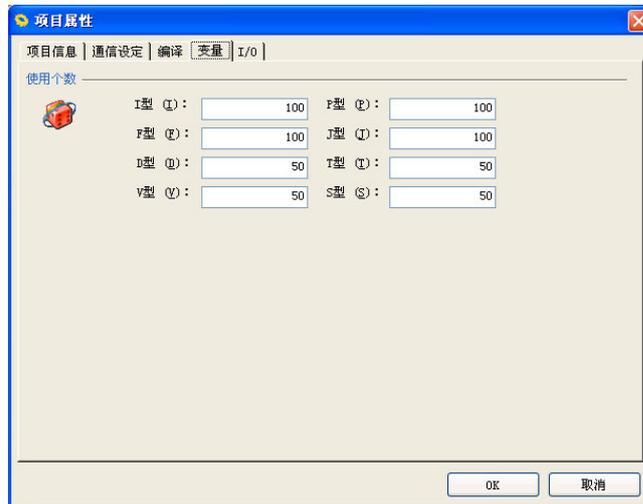
15.2.3 变量使用个数的变更

在 WINCAPSIII 中，可设定变量的使用个数。

步骤 1

打开项目，选择 [项目 (P)] - [属性 (P)]。

[项目属性] 视窗即会打开。
点击 [变量] 图标。



变更变量的个数，按 [OK]。

步骤 2

将数据发送至机械手控制器。

选择 [通信 (N)] - [发送接收数据 (T)], [发送接收数据] 视窗即会显示出来。
在 WINCAPSIII 侧 [参数] - [程序 参数] 处打勾，点击 [发送 (S) →]。



步骤 3

在机械手控制器中选择 [F1 程序] - [F6 辅助功能] - [F12 编译]。

程序会被编译，项目会自动上载。即会显示变量使用数的变更。

第5部

高级使用方法

第 16 章 使用条件的最佳设定

16.1 机械手安装条件（地面设置、悬挂）的设定方法 [6 轴机械手]

在 6 轴机械手中，需要根据地面安装、悬挂等设置方法进行参数设定。
地面设置时设定为 "0"，悬挂时设定参数为 "1"。
出厂时，设定为地面设置 "0"，所以悬挂使用时，请对设定进行变更。

16.1.1 机械手安装条件（地面设置、悬挂）设定的目的

使用电流限制功能和力限制功能时，需要将重力补偿设定设为有效。根据安装条件（地面设置、悬挂）判断该重力补偿的方向。

16.1.2 由多功能教导器进行的设定

操作路径：基本画面— [F2 臂] — [F6 辅助功能.] — [F7 使用条件.]

如果操作多功能教导器、经由上述的操作路径，则出现 [使用条件（参数数:）] 画面。



在该 [使用条件（参数数:）] 画面中，选择 [地面设置、悬挂设定] 的项目，按下 [F5 变更设定]，则数字键盘形式的 [变更参数] 画面被显示，可以在此更改各个参数。

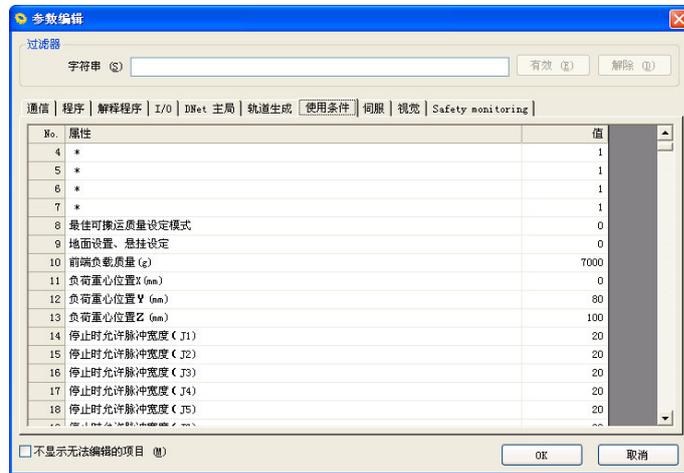
请将安装条件设定为 "0"、"1" 中的某一个。如果输入上述以外的值，则被输入的值在范围之外。



注意：关于在多功能教导器上变更的使用条件，请通过WINCAPSIII的 [参数] - [臂参数] 来接收。

16.1.3 利用 WINCAPSIII 进行的设定

下面来说明一下利用 WINCAPSIII 对地面设置 "0"、悬挂 "1" 进行设定的方法。
选择 [项目 (P)] - [参数 (T)] - [使用条件]。



双击 [地面设置、悬挂设定] 的数值栏，即可输入或变更参数值。

注意：关于 WINCAPSIII 上所设定的使用条件，请利用发送接收数据功能接收 [臂参数]。

设定参数值之后，将数据传送给机械手控制器。
首先，用多功能教导器的 [MOTOR]，将电机电源置于 OFF。
选择 [通信 (N)] - [发送接收数据 (T)]。

在 [参数] - [臂参数] 处打勾，点击 [发送 (S) →]。



16.2 最佳可搬运质量设定功能

是与机械手的前端负荷或姿势相应，设定最佳速度、加速度的功能。可以选择表 4-2 所示的 4 个模式。

表 16-1 最佳可搬运质量的模式

模式	设定条件	设定内容	
		PTP 动作	CP 动作
0	负荷条件值	最高加速度	最高加速度
1	负荷条件值 机械手的姿势	最高速度、加速度	与模式0相同
2		与模式0相同	最高速度、加速度
3		与模式1相同	与模式2相同

16.2.1 模式 0

是出厂时的默认缺省模式。通过机械手的负荷条件值，设定 PTP 动作、CP 动作时的最高加速度。

关于固定时间，请参照机械手概述说明书 "第 3 章 机械手的固定时间"。

16.2.2 模式 1

通过机械手的负荷条件值与动作中的机械手的姿势，设定 PTP 动作时的 1、2、3 轴的最高速度与加速度。PTP 动作时的 4、5、6 轴和 CP 动作与模式 0 相同。

使用模式 1 的情况下

需要缩短 PTP 动作的动作时间的时候，请设定为模式 1。

模式 1 使用时的注意事项

在动作过程中，有时会发生过载错误和偏差过大错误。关于负荷率，请在调整时利用多功能教导器的过载预定值（参照操作指南 "5.3 机械手的当前位置显示、[F2] - [F6] - [F10]"）、或者利用 WINCAPSIII 的 LOG 功能进行确认。

发生过载错误时，请设定计时器和内部速度、加速度，调整电机负荷。

发生偏差过大错误时，请调整速度、加速度。

由于动作速度过快，通过 (Pass) 轨道会发生 20mm 左右的变化，在这种状况下进行通过 (Pass) 传输动作可能被附近障碍物干扰，因此要在模式 0 下执行。

16.2.3 模式 2

通过机械手的负荷条件值与动作中的机械手的姿势，设定 CP 动作时的最高速度与加速度。PTP 动作时与模式 0 相同。

使用模式 2 的情况下

以下的 2 种情况，请设定为模式 2。

- ① 需要缩短 CP 动作时的动作时间时
- ② 在需要避免指令速度限制超程错误时

在 CP 动作中发生指令速度限制溢出错误 (6081~6086) 时，当 CP 动作时的轨道通过不规则点附近（参照操作指南 "4.1.3 关于机械手手腕、肘、臂的形态 [2] 形态的边界"）或动作范围限制附近时，有时会发生指令速度限制溢出错误而停止。此时，则在指令速度限制内自动降低速度，并且可以不发生错误动作。

模式 2 使用时的注意事项

- 有时会在动作中发生过负荷错误。在调整时，请使用多功能教导器的过负荷预想值（参照操作指南 "5.3 机械手的当前位置的表示、[F2] - [F6] - [F10]"）或 WINCAPSIII 的 LOG 功能，确认负荷率。发生过载错误时，请设定定时器和内部速度、加速度，调整电机负荷。
- 由于动作速度过快，通过 (Pass) 轨道会发生 20mm 左右的变化，在这种状况下进行通过 (Pass) 动作可能被附近障碍物干扰，因此要在模式 0 下执行。
- 当 CP 动作处于等速移动区间时，对于有必要进行等速移动的作业，为防止速度发生变化，请在模式 0 或模式 1 状态下执行。
- 在 CP 动作时，有时会发生指令加速度限制溢出错误 (6761~6766)、偏差过大错误 (6111~6116)。当发生错误时，请通过内部速度、内部加速度对速度、加速度进行调整。此外，有时在高速动作时会产生最大 5mm 左右的轨迹错位。当动作附近有障碍物时，请降低速度使用。
- 当在不规则点附近（参照操作指南 "4.1.3 关于机械手手腕、肘、臂的形态 [2] 形态的边界"）在动作速度正在降低时使其瞬时停止时，有时瞬时停止时间会延长。但是，瞬时停止距离没有变化。

16.2.4 模式 3

PTP 动作时与模式 1 同样运行，CP 动作时与模式 2 同样动作。

16.3 最佳可搬运质量功能的设定方法

16.3.1 由多功能教导器进行的设定

操作路径：基本画面 - [F2 臂] - [F6 辅助功能.] - [F7 使用条件.]

如果操作多功能教导器、经由上述的操作路径，则出现 [使用条件 (参数数:)] 画面。当前的内部负荷条件值和内部模式等被显示。



在该 [使用条件 (参数数:)] 画面中，选择以下项目，按下 [F5 变更设定]，则数字键盘形式的 [变更参数] 画面被显示，可以在此更改各个参数。

设定项目：

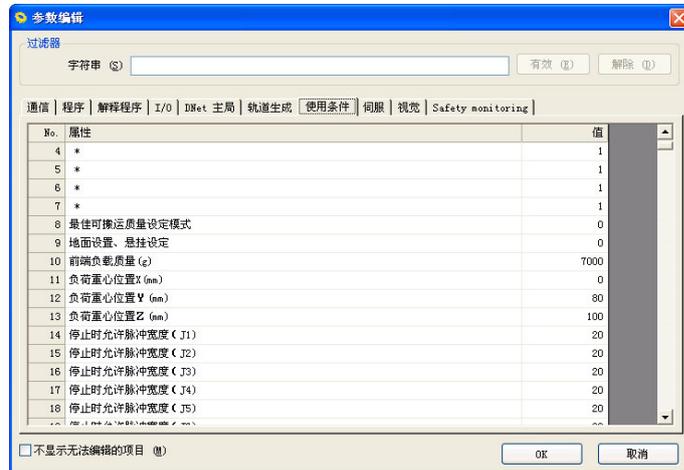
[7: 最佳可搬运质量设定模式]、[9: 前端负载质量 (g)]、[10: 负荷重心位置 X (mm)]
[11: 负荷重心位置 Y (mm)]、[12: 负荷重心位置 Z (mm)]
其中，4 轴机械手的 Ver. 1.9 以后版本是 [12: 负荷惯性力矩 (kgcm²)]

- 最佳可搬运质量初始化设定是 0~3，如果输入以外的设置，则会发生 "输入的数值超过范围"。
- 前端负载质量，如果输入的数值超过每个机械手所设定的允许值，则会发生 "输入的数值超过范围"。
- 负荷重心位置的输入要满足允许范围。在不满足每个机械手所设定的允许范围时，则会发生 "输入的数值超过范围"。

16.3.2 利用 WINCAPSIII 的设置

下面来说明一下利用 WINCAPSIII 对外部负荷条件值（前端负荷质量、重心位置）和外部模式进行设定的方法。

选择 [项目 (P)] - [参数 (T)] - [使用条件] 图标。



双击以下项目的设定值栏，即可进行各参数值的输入及更改。

设定项目：

[最佳可搬运质量设定模式]、[前端负载质量 (g)]、[负荷重心位置 X (mm)]、
[负荷重心位置 Y (mm)]、[负荷重心位置 Z (mm)]
其中，4 轴机械手的 Ver. 1.9 以上版本是 [负荷惯性力矩 (kgcm²)]

设定各参数值之后，将数据传送给机械手控制器。

首先，用多功能教导器的 [MOTOR]，将电机电源置于 OFF。

选择 [通信 (N)] - [发送接收数据 (T)]。

在 [参数] - [臂 参数] 处打勾，点击 [发送 (S) →]。



16.4 最佳可搬运质量初始化设定的设定方法 [V1.4 以后版本]

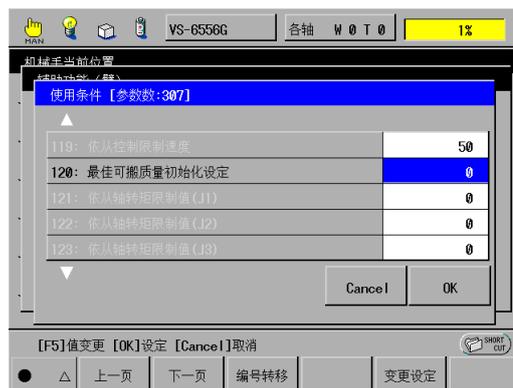
控制器电源 ON 时，就需要将 [7: 最佳可搬运质量设定模式] 参数初始化为模式 0 时或需要保持当前设定值的情况进行说明。

设定值	设定值的含义
0	控制器电源接通时，将最佳可搬运质量设定模式初始化为模式0。 (出厂时)
1	即使接通控制器电源，也不将最佳可搬运质量初始化。 (保持当前的值)

16.4.1 由多功能教导器进行的设定

操作路径：基本画面— [F2 臂] — [F6 辅助功能] — [F7 使用条件]

如果操作多功能教导器、经由上述的操作路径，则出现 [使用条件 (参数数:)] 画面。



在该 [使用条件 (参数数:)] 画面中，选择 [120: 最佳可搬质量初始化设定] 项目，按下 [F5 变更设定]，则数字键盘形式的 "变更参数" 画面被显示，可以在此更改各个参数。

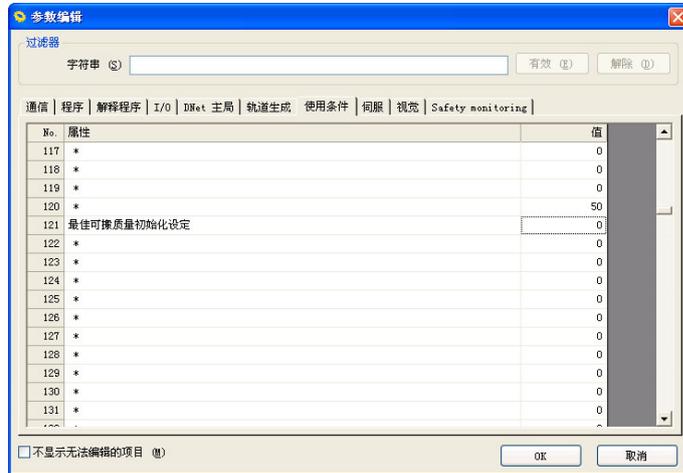
0: 无效 → 控制器电源接通时初始化 (出厂时)

1: 有效 → 即使接通控制器电源也不初始化 (保持当前的值)

16.4.2 利用 WINCAPSIII 的设置

下面说明一下利用 WINCAPSIII，对最佳可搬运质量初始化设定进行设置的方法。

选择 [项目 (P)] - [参数 (T)] - [使用条件] 图标。



双击最佳可搬运质量初始化设定项目，则可以进行参数值的输入及更改。

设定各参数值之后，将数据传送给机械手控制器。

首先，用多功能教导器的 [MOTOR]，将电机电源置于 OFF。

选择 [通信 (N)] - [发送接收数据 (T)]。

在 [参数] - [臂 参数] 处打勾，点击 [发送 (S) →]。



第 17 章 程序语言（机械手控制语句）

17.1 机械手动作的种类

17.1.1 绝对动作与相对动作

绝对动作

向被教导的动作位置进行移动的动作称为绝对动作。

绝对动作不会影响其前一动作，向必须被教导的位置移动。

执行绝对动作的指令如下所示。

APPROACH、MOVE、GOHOME、DRIVEA

相对动作

从当前位置开始，仅移动被教导的移动量的动作称为相对动作。

当前位置是执行即将动作之前的指令的结果。相对动作是以当前位置为基准的，所以相对运行受即将动作之前的运行指令的影响。

执行相对动作的指令如下所示。

DEPART、DRAW、DRIVE、ROTATE、ROTATEH

17.1.2 插补控制

当机械手臂的前端移动时，其路径不止一个。各轴的动作相互结合，可以实现多种路径。还可以控制使其为直线或圆弧。下面对与动作路径的种类相应的控制方法进行说明。

用以下的指令对插补方法（PTP 控制、CP 控制、圆弧插补控制）进行指定。

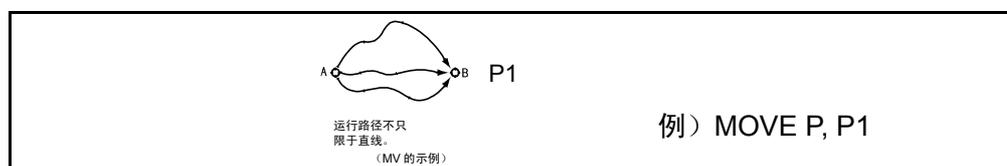
指定插补方法的指令：

APPROACH、DEPART、DRAW、MOVE

PTP 控制

所谓 PTP (Point to Point) 是指从点向点的移动。移动的路径依存于机械手的姿势，不只限于直线运动。

在用动作控制指令指定插补方法时，进行 "P" 的指定。



在 PTP 动作目标位置，指定 P 型、T 型变量时，如果指定机械手形态，则会向成为指定的机械手形态进行移动。不指定机械手形态的情况下，与现在的机械手形态成为同一形态。

CP 控制

CP 控制是进行插补控制使到达动作目标位置的路径成为直线。

在用动作控制指令指定插补方法时，进行 "L" 的指定。



- 作为原则，不能向与当前的机械手形态不同的形态移动。如果指定为不同形态时，则会发生 "607F 机械手形态不一致" 的错误。但是，可以移动时，则不发生错误。
- 机械手形态为与当前的姿势相近的形态被选择。因此，即使通过 P 型、T 型变量指定机械手形态，也不限于变为所指定的形态。如果指定为不同形态时，则会发生 "601C 请更改形态" 的警告。
- 如果通过 CP 控制执行程序中的最初的动作指令时，由于机械手的位置有时可能无法动作。推荐对程序中的最初动作指令使用 PTP 控制。

圆弧插补控制

圆弧插补控制，将到达动作目标位置的路径进行插补控制来描绘圆弧。

在用动作控制指令指定插补方法时，进行 "C" 的指定。



- 作为原则，与 CP 控制相同，不能向与当前的机械手形态不同的形态移动。如果指定为不同形态时，则会发生 "607F 机械手形态不一致" 的错误。但是，可以移动时，则不发生错误。
- 机械手形态为与当前的姿势相近的形态被选择。因此，即使通过 P 型、T 型变量指定机械手形态，也不限于变为所指定的形态。如果指定为不同形态时，则会发生 "601C 请更改形态" 的警告。
- 如果通过圆弧插补控制执行程序中的最初的动作指令时，由于机械手的位置有时可能无法动作。推荐对程序中的最初动作指令使用 PTP 控制。

17.2 机械手控制指令

17.2.1 DRIVE A

功能
格式

进行各轴的绝对动作。

DRIVEA □ [**<@通过 (Pass) 开始位移>**[□]] (<轴编号>, <轴坐标> [, (<轴编号>, <轴坐标>)...]) [, <动作选择>] [, NEXT]

说明

将 <轴编号> 上指定的轴向 <轴坐标> 所指定的角度 (DEG) 移动。

将相同的轴进行多个指定时, 后面的指定为有效。

在 <@通过 (Pass) 起始位移> 上存在 @0、@P (@1~@255)、@E。

通过 (Pass)起始位移	意思
@0	以末端动作方式运行。(如果省略, 则作为 @0处理)
@P (或者是 @1~@255)	以通过 (Pass) 动作方式运行。注意: 在进行数值指定时, 如果在以目标位置为中心的球半径 (mm指定) 上输入动作指令, 则移向下一个控制。是通过 (Pass) 起始的大致目标值, 不是保证值。
@E	通过编码器值对到达目标位置进行确认。

在 <动作选择> 中有 SPEED、ACCEL、DECEL。

动作选择	意思
SPEED (或标记为S)	指定移动速度。
ACCEL	指定加速度。
DECEL	指定减速度。

如果使用 <NEXT 选择>, 则不等待机械手的动作结束而继续下一个非动作命令。

但是, 以下的命令要一直等待机械手的动作结束 (通过 (Pass) 开始)。机械手动作命令 (CHANGETOOL、CHANGEWORK、SPEED、JSPEED、ACCEL、JACCEL、DECEL、JDECEL)、最佳可搬运质量设定共用程序库 (Library) (aspACLD、aspChange)、臂动作共用程序库 (Library) (mvSetPulseWidth 等)

此外, 与动作选择并用的情况下, NEXT 选择变为无效。若使用 <NEXT 选择> 执行步骤停止, 在等待下一个动作命令执行时, 在结束其动作之后停止。请注意达到停止时的移动延长。

注意: 在教导检查模式下, NEXT选择变为无效。

应用示例

例 1: DEFINT li1, li2, li3

DEFSNG lf1, lf2, lf3

DRIVEA (li1, 30)

DRIVEA (li1, lf1)

DRIVEA @P (li1, 0.78RAD), (li2, lf2), (li3, lf3)

'将 li1 值的轴移动到 30 度 (deg) 的位置。

'将 li1 值的轴移动到 lf1 值的位置。

'将 li1 的值的轴向 0.78 (rad)、将 li2 向 lf2 的值、

'将 li3 向 lf3 的值的值的位置移动。

17.2.2 DRIVE

功能格式

进行各轴的相对动作。

DRIVEA □ [<@通过 (Pass) 开始位移>] (<轴编号>, <相对移动量>)

[, (<轴编号>, <相对移动量>) ...] [, <动作选择>] [, NEXT]

说明

将 <轴编号> 上指定的轴向 <相对移动量> 所指定的角度 (DEG) 移动。

<相对移动量> 为正时, 让指定轴向 + 方向, 为负时, 向 - 方向动作。

将相同的轴进行多个指定时, 后面的指定为有效。

在 <@通过 (Pass) 起始位移> 上存在 @0、@P (@1~@255)、@E。

通过 (Pass) 开始位移	意思
@0	以末端动作方式运行。(如果省略, 则作为@0处理)
@P (或 @1~@255)	以通过 (Pass) 动作方式运行。注意: 在进行数值指定时, 如果在以目标位置为中心的球半径 (mm指定) 上输入动作指令, 则移向下一个控制。是通过 (Pass) 起始的大致目标值, 不是保证值。
@E	通过编码器值对到达目标位置进行确认。

在 <动作选择> 上存在 @0、@P (@1~@255)、@E。

动作选择	意思
SPEED (或标记为S)	指定移动速度。
ACCEL	指定加速度。
DECEL	指定减速度。

如果使用 <NEXT 选择>, 则不等待机械手的动作结束而继续下一个非动作命令。但是, 以下的命令要一直等待机械手的动作结束 (通过 (Pass) 开始)。机械手动作命令 (CHANGETOOL、CHANGEWORK、SPEED、JSPEED、ACCEL、JACCEL、DECEL、JDECEL)、最佳可搬运质量设定共用程序库 (Library) (aspACLD、aspChange)、臂动作共用程序库 (Library) (mvSetPulseWidth 等)

此外, 与动作选择并用的情况下, NEXT选择变为无效。若使用 <NEXT选择> 执行

步骤停止, 在等待下一个动作命令执行时, 在结束其动作之后停止。请注意达到停止时的移动延长。

注意: 在教导检查模式下, NEXT选择变为无效。

应用示例

例 1: DEFINT li1, li2, li3
DEFSNG lf1, lf2, lf3
DRIVE (li1, 30)
DRIVE (li1, lf1)
DRIVE (li1, 0.78RAD), (li2, lf2), (li3, lf3)

'从当前位置将 li1 移动 30 度 (deg)。

'从当前位置将值为 li1 的轴移动 lf1 值的量。

'从当前位置将 li1 移动 0.78 (rad) 值的量、

'将 li2 移动 lf2 所示值的量、将 li3 移动 lf3 值

'的量。

17.2.3 DRAW

功能格式

通过工件作业坐标系的指定执行相对动作。

DRAW_L <插补方法>, [**<@通过 (Pass) 开始位移量>**] <并进移动量> [, <动作选择>] [,NEXT]

说明

从当前位置移动 <并进移动量> 的距离。

在 <插补方法> 中可以在P（标记为PTP也可）、L中任意选择。

插补方法	意思
P（或标记为PTP）	用PTP控制移动。
L	用CP控制移动。

在 <@通过 (Pass) 起始位移量> 上存在 @0、@P (@1~@255)、@E。

通过 (Pass) 起始位移	意思
@0	以末端动作方式运行。（如果省略，则作为@0处理）
@P （或 @1~@255）	以通过 (Pass) 动作方式运行。注意：在进行数值指定时，如果在以目标位置为中心的球半径（mm指定）上输入动作指令，则移向下一个控制。是通过 (Pass) 起始的大致目标值，不是保证值。
@E	通过编码器值对到达目标位置进行确认。

在 <动作选择> 中有 SPEED、ACCEL、DECEL。

动作选择	意思
SPEED（或标记为S）	指定移动速度。
ACCEL	指定加速度。
DECEL	指定减速度。

如果使用 <NEXT 选择>，则不等待机械手的动作结束而继续下一个非动作命令。

但是，以下的命令要一直等待机械手的动作结束（通过 (Pass) 开始）。机械手动作命令 (CHANGETOOL、CHANGEWORK、SPEED、JSPEED、ACCEL、JACCEL、DECEL、JDECEL)、最佳可搬运质量设定共用程序库 (Library) (aspACLD、aspChange)、臂动作共用程序库 (Library) (mvSetPulseWidth 等)

此外，与动作选择并用的情况下，NEXT选择变为无效。若使用 <NEXT选择> 执行步骤停止，在等待下一个动作命令执行时，在结束其动作之后停止。请注意达到停止时的移动延长。

注意：在教导检查模式下，NEXT选择变为无效。

备注 DRAW 可以用 MOVE 改写。

例：DRAW L, (50, 10, 50) '与 MOVE L, P0 + (50, 10, 50) 相同

应用示例

DEFVEC lv1

DRAW L, (50, 10, 50) '向离开当前位置 (X = 50, Y = 10, Z = 50) 的位置(CP 控制)移动。

DRAW L, lv1, SPEED = 90 '从当前位置向离开 lv1 的位置 (CP 控制, S = 90) 移动。

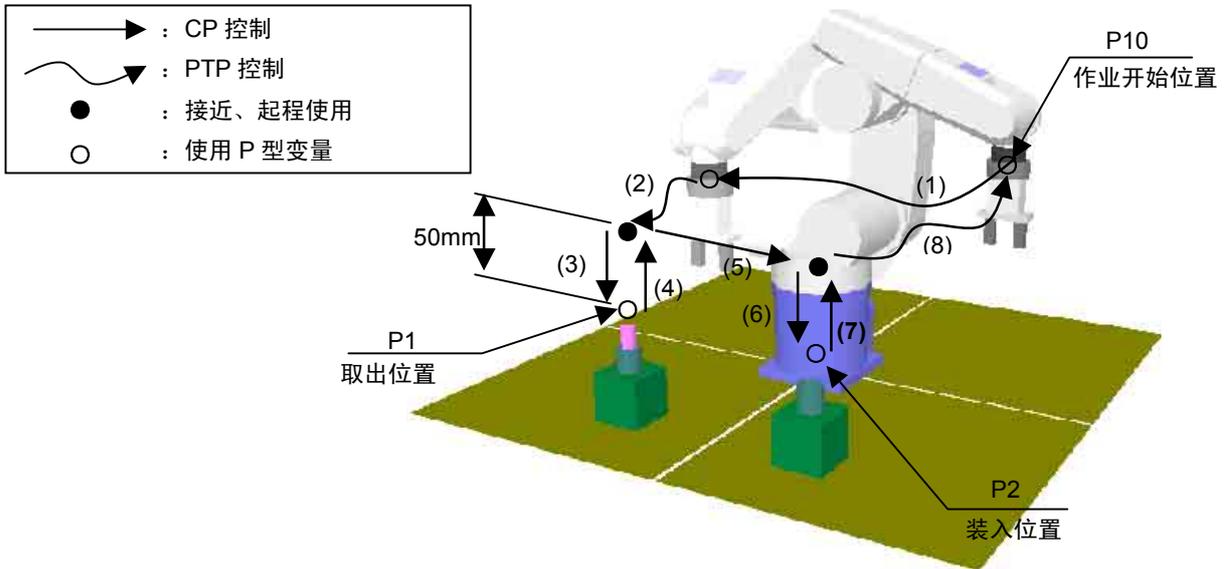
注意事项

DRAW 动作位置的形态，变为动作起始时的形态。

17.3 练习题 机械手控制指令

练习题 1

请使用机械手控制语句，编写向工件取出、装入位置动作的程序。



■ 动作规格

- ① (1) 的动作，请使用只将J1轴向0度位置动作的指令。
- ② 在接近、起程点上请使用通过 (Pass) 动作。
- ③ 向P1以及P2点进行下降动作时的速度设为20%。
- ④ 请将向P1以及P2点的动作设为编码器值确认动作。

程序	运行说明
'TITLE "练习程序1"	'程序标题
PROGRAM PRO1	'程序名的定义
TAKEARM	'获取臂控制权
SPEED 100	'内部速度100%
_____	'(1): 移动J1轴至0度位置
_____	'(2): P1的夹治具方向 向上空移动50mm
_____	'(3): 移动至P1
_____	'(4): P1的夹治具方向 向上空移动50mm
_____	'(5): P2的夹治具方向 向上空移动50mm
_____	'(6): 移动至P2
_____	'(7): P2的夹治具方向 向上空移动50mm
MOVE P, @0 P10	'(8): 移动至P10
END	'程序停止

第 18 章 编程语言（流程控制语句）

18.1 流程控制语句的种类

为了控制程序的各个语句的执行顺序，使用流程控制语句。
通过使用流程控制语句，可以编写高效的程序。

流程控制指令大致分为以下的 4 种。

- ① 调出
 - CALL
 - GOSUB

- ② 无条件分支
 - GOTO

- ③ 条件分支
 - IF~ENDIF
 - SELECT CASE~END SELECT

- ④ 反复
 - FOR~NEXT
 - DO~LOOP
(DO WHILE~LOOP, DO~LOOP WHILE, DO UNTIL~LOOP,
DO~LOOP UNTIL)

18.2 调出的指令

18.2.1 CALL

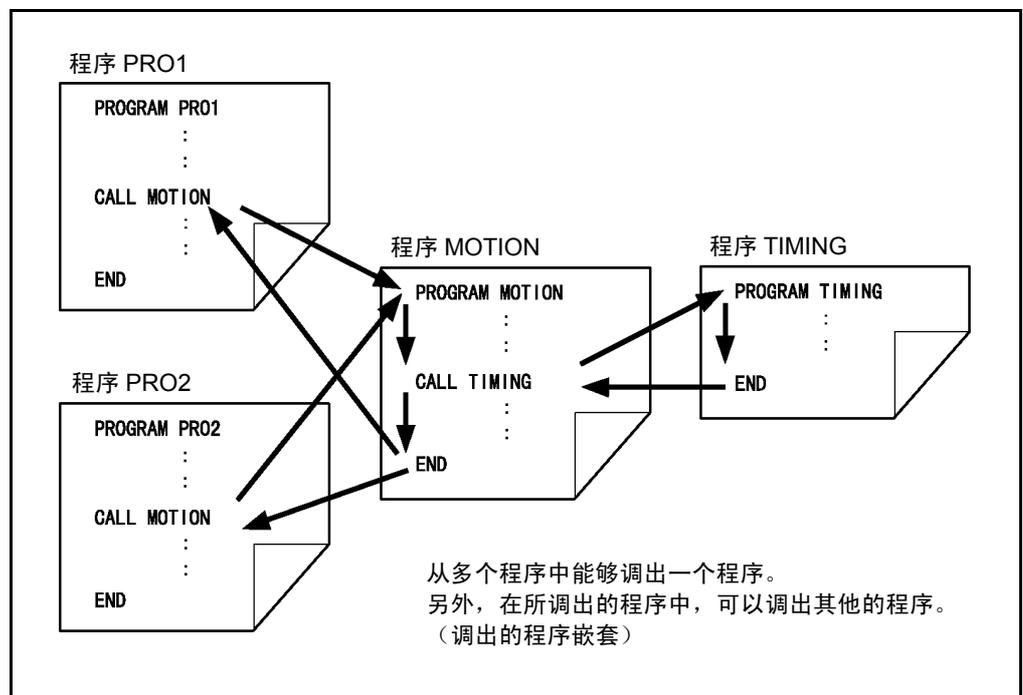
功能 调出程序并执行。

格式 CALL □ <程序名>

说明

可以与主要执行的程序相区别创建程序，可如子程序那样调出该程序来使用。如果使用 CALL 语句调出程序，则控制将转移到所调出的程序。如果执行了位于所调用程序的最末行的 END 语句，则控制将返回到调用源的程序。

被调出的程序还可以调出其他的程序（将其称为调出的嵌套）。但是，被调出的程序不可以调出调出源程序。



注意：调出的程序嵌套包括CALL，GOSUB在内最多为31次。

应用示例

```
CALL PRO1           '调出程序名 <PRO1> 执行
CALL SampleProgram '调出程序名 <SampleProgram> 执行
```

18.2.2 GOSUB

功能 调出辅助程序。

格式 GOSUB_ <*标签名>

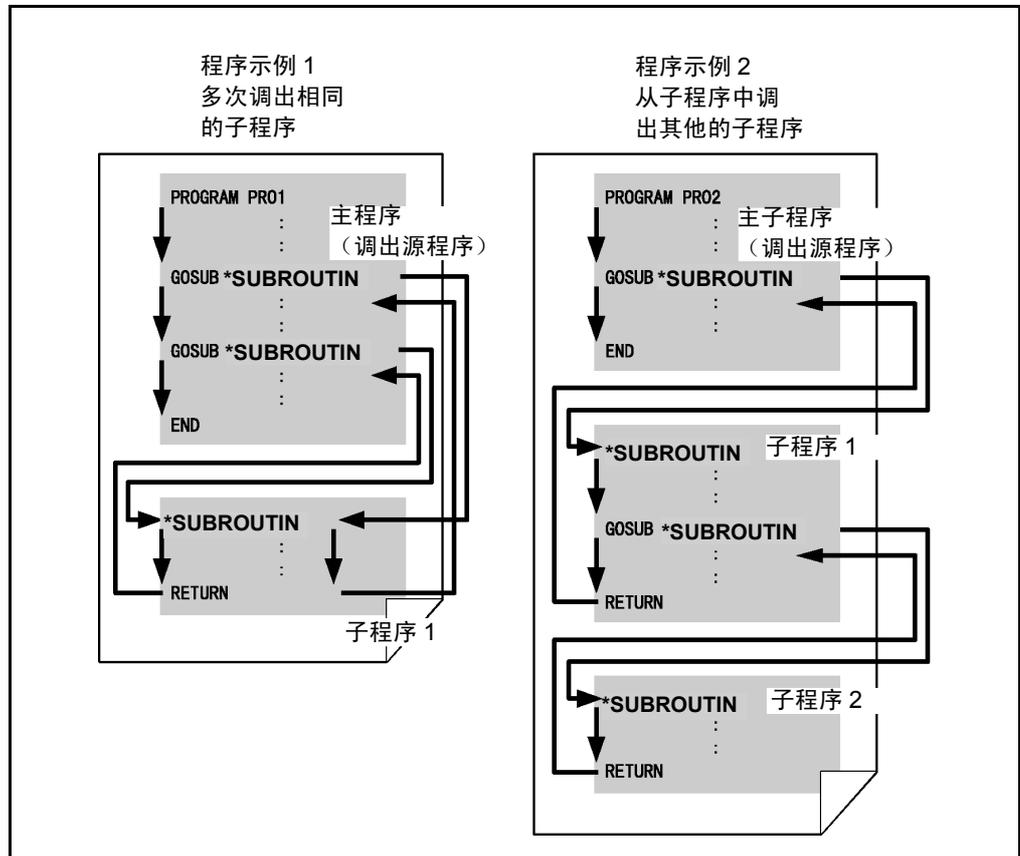
说明

在一个程序中、需要在多个不同的位置使用相同处理时，将该处理作为辅助程序记述，将其从多个位置调出使用。
子程序必须与调出的程序在相同的文件中。

若用 GOSUB 语句调出子程序，则控制将转移到子程序。如果执行了位于所调出子程序的最末行的 RETURN 语句，则控制将返回到调出子程序的程序原来的下一行。

也可以从辅助程序中调出其他的子程序（将其称为调出的嵌套）。

注意：调出的程序嵌套包括CALL，GOSUB在内最多为31次。此外，关于标签名的规定事项，请参照 "18.3.1 GOTO"。



应用示例

```
IF IO [128] = ON THEN GOSUB * Line1  
IF I2 = 5 THEN  
    GOSUB * Label2  
ENDIF
```

'IO [128] 为ON时，向 *Line1: 标签转移
'I2为5的情况。
'向标签名 <*Label2:> 转移

18.3 无条件分支指令

18.3.1 GOTO

功能 无条件执行程序的分支。

格式 GOTO \square <*标签名> (或 GO \square TO \square <*标签名>)

说明 向指定的标签位置转移，从转移位置开始继续执行。

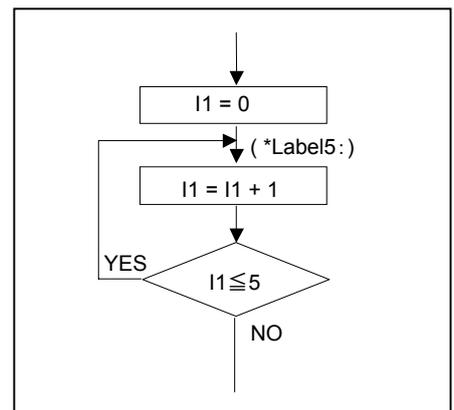
使用标签名时的限制事项

- 标签的名称以星号 (*) 开始。
- 标签名称的第 2 个字符，必须是任意的一个字母。
- 标签名的第 3 个字符之后可以使用英文字母和数字的任意组合。
- 不能在标签名称上使用保留字。
- 可参照标签的范围只是其标签所在的程序中。
- 被参照的标签名的最后字符必须是冒号 (:)。

应用示例

例) I1 的值在 5 以下时，转移至指定标签位置。

```
I1 = 0
*Label5:
I1 = I1 + 1
IF I1 <= 5 THEN      'I1 在 5 以下时
    GOTO *Label5     '向标签名 <*Label5:> 转移
ENDIF
```



18.4 条件分支指令

18.4.1 IF ~ ENDIF

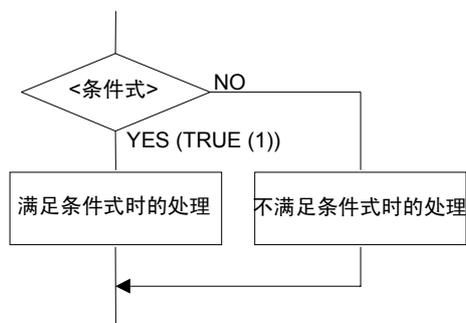
功能 执行IF~END IF之间的条件判断表达式中的条件判断。

格式

```
IF □ <条件式> □ THEN
    : '满足 <条件式> 时的处理
[ELSE]
    : '不满足 <条件式> 时的处理
ENDIF (或者是END IF)
```

说明 IF 语句的 <条件式> 如果为真 (TRUE (1)), 则执行 IF~ELSE 之间的语句; 如果不是真, 则执行 ELSE~ENDIF 之间的语句。

※在 <条件式> 中可以使用以下的运算符



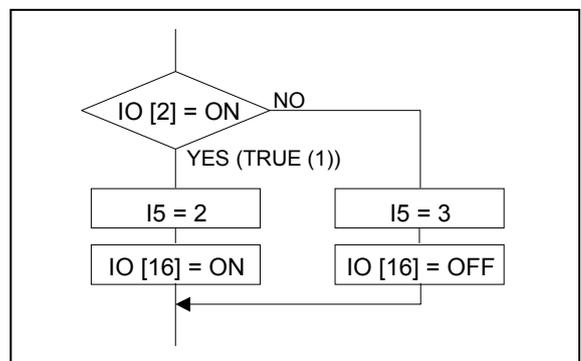
关系运算符	运算内容
=	等于
=.	几乎相等 (近似比较)
<>	不等于
<	小于
>	大于
<=	小于或等于
>=	大于或等于

备注
近似比较运算符 (=.) 的比较精度可以在 [RPJ 设定] 的 "近似比较精度" 上进行设定。

应用示例

例) 如果IO [2] 为 ON, 则将2赋值 I5、将IO [16] 置于ON。如果是ON之外, 则将3赋值I5之后, 将IO [16] 置于OFF。

```
IF IO [2] = ON THEN          'IO [2] 为ON时
    I5 = 2
    SET IO [16]
ELSE                          '除此之外时
    I5 = 3
    RESET IO [16]
ENDIF
```



18.4.2 SELECT CASE

功能 执行多个条件判断。

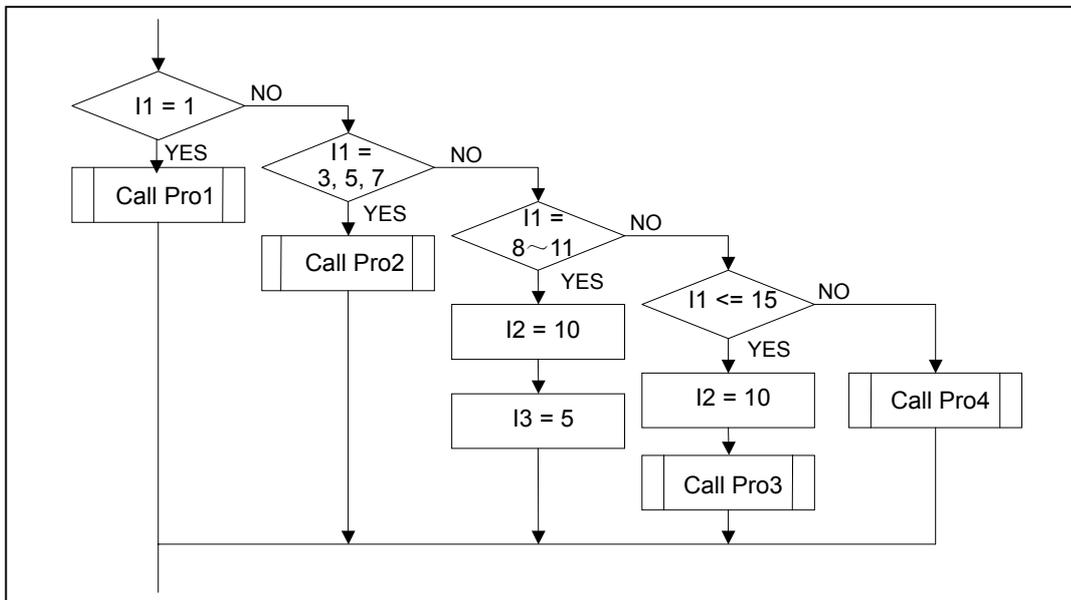
格式

```
SELECT CASE <式>
CASE <项目>
: <式> 的值与 CASE 语句的 <项目> 一致时的处理
[CASE ELSE]
: 与所有的 CASE 语句的 <项目> 不一致时的处理
END SELECT
```

说明 在 SELECT CASE 语句中, 在 SELECT CASE 之后放置 <式>, 该 <式> 的值与 CASE 语句 <项目> 一致时, 在其 CASE 以后执行到下一个 CASE 之间的语句。所有的 CASE 语句不一致时, 执行 CASE ELSE~END SELECT 之间的语句。

应用示例

例) I1 的值与各个项目相一致时, 分别进行各自的处理。



```
SELECT CASE I1
CASE 1                                'I1 为 1 时
CALL PRO1
CASE 3, 5, 7                          'I1 为 3、5、7 时
CALL PRO2
CASE 8 to 11                          'I1 为 8~11 时 (范围指定使用 "** to **")
I2 = 10
I3 = 5
CASE IS <= 15                          'I1 为 15 以下时 (比较条件 IS <比较运算符> <比较值 (数值)>
I2 = 10
CALL PRO3
CASE ELSE                              '项目不一致时
CALL PRO4
END SELECT
```

18.5 反复指令

18.5.1 FOR ~ NEXT

功能 循环执行位于FOR~NEXT之间的一系列的指令。

格式 FOR <变量名> = <初始值> TO <最终值> [STEP <增量>]
: '在 FOR 行满足指定条件时的处理
NEXT

说明 根据 FOR 行指定的条件，反复执行位于 FOR~NEXT 区间内的一系列命令。

<初始值> 设定 <变量名> 中指定的变量的初始值。

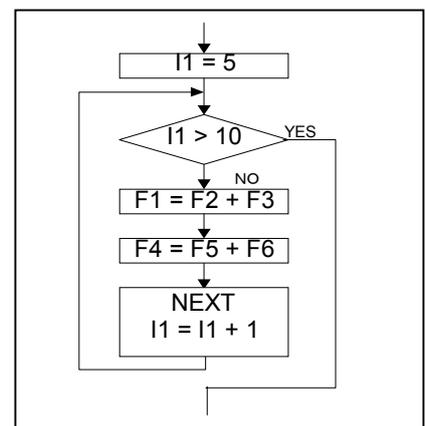
<最终值> 设定 <变量名> 中指定的变量的最终值。

<增量> 设定初始值和最终值之间的增量。如果省略 STEP，则增量被视为 1。

应用示例

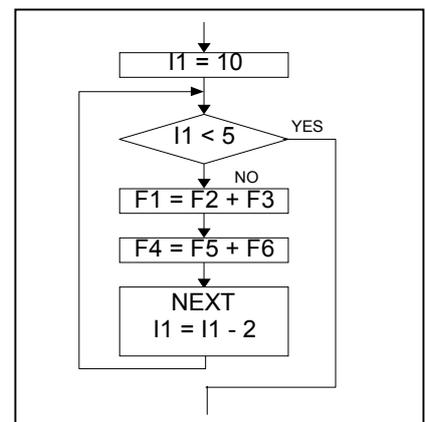
例1) 从5开始以1为单位对I1进行增量，进行循环处理直至I1超过10为止。I1变为5、6、...、10，在达到11层级时结束循环处理。

```
FOR I1 = 5 TO 10      '将初始值5置位在I1上（增量值1）  
                      'I1为10以下时继续处理  
    F1 = F2 + F3  
    F4 = F5 + F6  
NEXT                  '将I1计数 +1
```



例2) 从10开始以2为单位对I1作减法，进行循环处理直至I1小于5为止。I1变为10、8、6，在达到4层级时结束循环处理。

```
FOR I1 = 10 TO 5 STEP -2  '将初始值10置位在I1上（减去值2）  
                          'I1为5以上时继续处理  
    F1 = F2 + F3  
    F4 = F5 + F6  
NEXT                      '将I1计数 -2
```



18.5.2 DO ~ LOOP

功能 进行迭代（重复）。

格式

```

DO [WHILE (或UNTIL) <条件式>]
: '满足条件时 (WHILE 时) 或
: '条件满足之前 (UNTIL 时), 执行 DO~LOOP 之间的处理。
LOOP
或
DO
: '满足条件时 (WHILE 时) 或
: '条件满足之前 (UNTIL 时), 执行 DO~LOOP 之间的处理。
LOOP [WHILE (或UNTIL) <条件式>]
    
```

说明 DO WHILE 和 DO UNTIL 是前迭代。LOOP WHILE 和 LOOP UNTIL 是后迭代。WHILE 语句在满足 <条件式> 时 (TRUE (1)) 进行重复，UNTIL 语句在满足 <条件式> 之前进行重复。

- DO ~LOOP : WHILE、UNTIL 的省略是可以的，但会形成无限的循环。
- DO WHILE~LOOP : 在满足条件时进行反复处理。（前判定）
- DO ~LOOP WHILE: 在满足条件时进行反复处理。（后判定）
- DO UNTIL ~LOOP : 在满足条件之前进行反复处理。（前判定）
- DO ~LOOP UNTIL : 在满足条件之前进行反复处理。（后判定）

注意：要从DO~LOOP语句中强制退出，转移到DO~LOOP语句的下一个指令，则使用EXT DO指令。

应用示例

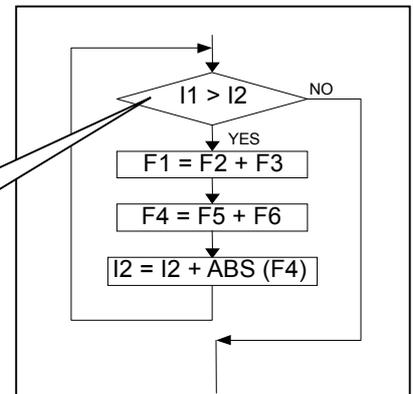
- DO WHILE ~ LOOP（前判定）

例1) I1 > I2时，进行反复处理。

```

DO WHILE I1 > I2      'I1 > I2时，在DO~LOOP之间反复运行
  F1 = F2 + F3
  F4 = F5 + F6
  I2 = I2 + ABS ( F4 )
LOOP
    
```

在此对条件进行判断，在满足条件时进行反复运行。



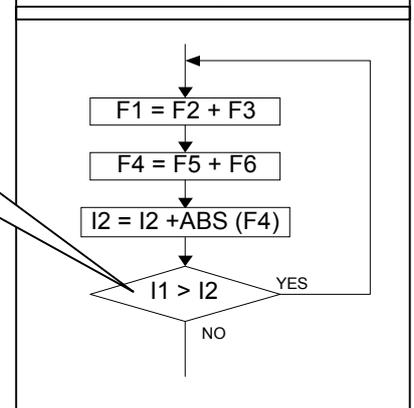
- DO ~ LOOP WHILE（后判定）

例2) I1 > I2时，进行反复处理。

```

DO
  F1 = F2 + F3
  F4 = F5 + F6
  I2 = I2 + ABS ( F4 )
LOOP WHILE I1 > I2    'I1 > I2时，在DO~LOOP之间反复运行
    
```

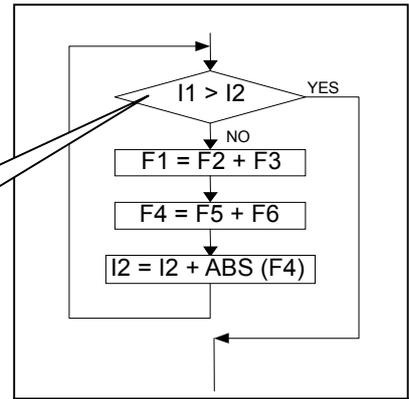
在此对条件进行判断，在满足条件时进行反复运行。



- DO UNTIL ~ LOOP (前判定)
例3) 在达到 $I1 > I2$ 之前进行反复处理。

DO UNTIL $I1 > I2$ '达到 $I1 > I2$ 之前，在DO~LOOP之间反复运行
 $F1 = F2 + F3$
 $F4 = F5 + F6$
 $I2 = I2 + ABS(F4)$
 LOOP

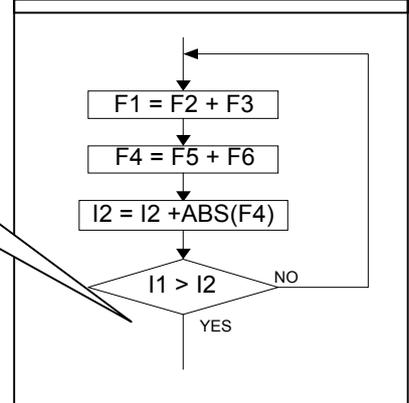
在此对条件进行判断，在满足条件之前进行反复运行。



- DO ~ LOOP UNTIL (后判定)
例4) 在达到 $I1 > I2$ 之前进行反复处理。

DO
 $F1 = F2 + F3$
 $F4 = F5 + F6$
 $I2 = I2 + ABS(F4)$
 LOOP UNTIL $I1 > I2$ '达到 $I1 > I2$ 之前，在DO~LOOP之间反复运行

在此对条件进行判断，在满足条件之前进行反复运行。

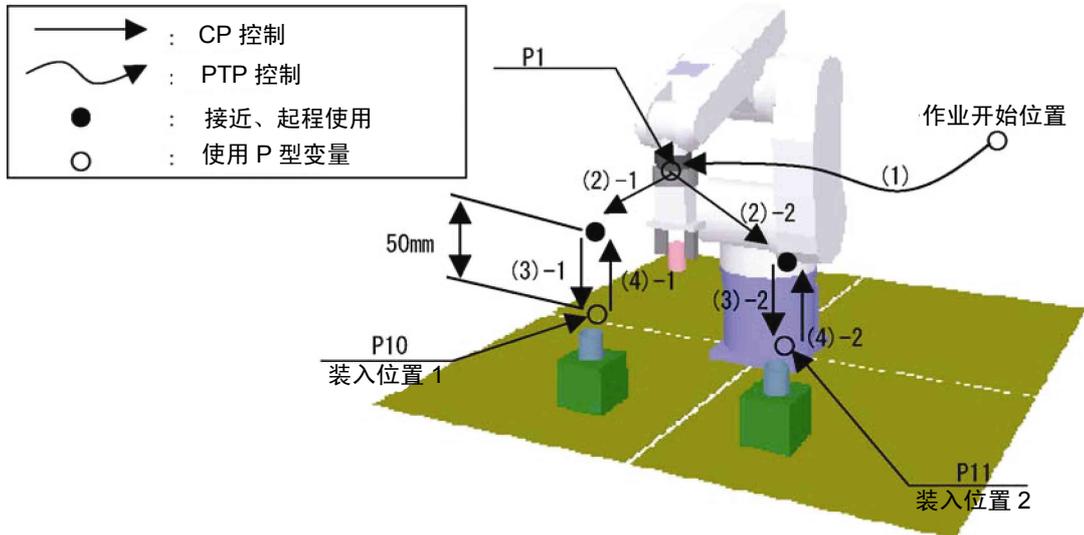


注意：在后判定时，必须执行DO~LOOP之间的处理。

18.6 练习题 流程控制指令

练习题 2

请使用流程控制语句 (IF~ENDIF)，编写判断工件装入位置的程序。



■ 动作规格

- ① 向P1的位置移动后，使用IF语句确认I [5] 的值。若I [5] = 0，则至 (2)-1，若I [5] = 1，则向 (2)-2 移动，完成工件装入动作。
- ② 在向 (3) -1及 (3) -2的动作到达位置调出夹治具打开的程序 (HAND_OPEN)。

程序	运行说明
'TITLE "练习程序2"	'程序标题
PROGRAM PRO2	'程序名的定义
TAKEARM	'获取臂控制权
SPEED 100	'内部速度100%
MOVE P, P1	'(1): 移动至P1
_____	'(2): IF语句 若I5 = 0则移至下个命令
APPROACH L, P10 50	'(2)-1: P10的夹治具方向 向上空移动50mm
MOVE L, P10	'(3)-1: 移动至P10
_____	'(2): IF语句 若I5 = 0不成立，移至下个命令
APPROACH L, P11 50	'(2)-2: P11的夹治具方向 向上空移动50mm
MOVE L, P11	'(3)-2: 移动至P11
_____	'(2): IF语句的结束
_____	'调出HAND_OPEN程序
DEPART L, 50	'(4)-1, (4)-2: P10或P11的夹治具方向 向上空移动50mm
END	'程序停止

第 19 章 程序语言（输入输出语句）

19.1 时间控制指令

命令机械手在指定的时间内待命。

19.1.1 DELAY

功能 在所指定的时间内，停止机械手的程序的执行。

格式 DELAY [□] <延迟时间>

说明 等待 <延迟时间> 所指定的时间经过。延迟时间（单位：ms）1 秒作为 1000 输入。

应用示例

DELAY 300 '等待 300ms (0.3s) 的时间经过。

DELAY I15 '等待 I15 的数值的时间经过。

19.1.2 WAIT

功能 执行条件程序的停止。

格式 WAIT [□] <条件式> [,<超时时间>] [,<存放变量>]

说明 到指定的 <条件式> 成立为止停止程序的执行。
一旦设定 <超时时间>，在经过指定时间之后结束 WAIT 语句的执行，控制移到下一个指令。通过这样，可以防止无限期停止。
<超时时间> 的单位是 ms。
若设定 <存放变量> 条件式成立，TRUE (1) 进入遗漏 WAIT 指令时所指定的存放变量，FALSE (0) 进入在超时遗漏时所指定的变量。

应用示例

DEFINT I1

WAIT I1 = 1 '等待 I1 = 1 成立。

WAIT IO [10] = ON '在 IO [10] 为 ON 之前一直等待。

WAIT IO [5] = 0, 2000 '在 IO [5] 为 OFF 之前等待，经过 2 秒钟之后仍不 OFF 时，则转向下一个语句。

WAIT I3 = 5, 1000, I4 '在 I3 = 5 成立之前等待，如果成立则设为 I4 = 1；经过 1 秒钟之后仍不成立时，则设为 I4 = 0，转移到下一个语句。

19.2 I/O 端口控制

在此以机械手的夹紧动作为例对输出命令进行介绍。输出命令在将输出接口置于 ON 命令时使用 SET；在将输出接口置于 OFF 时使用 RESET。

19.2.1 SET

功能 将 I/O 端口置为 ON。

格式 SET [□] <I/O 变量> [, <输出时间>]

说明 将在 <I/O 变量> 中指定的端口置于 ON。使用端口编号或 I/O 变量对端口进行指定。如果指定 <输出时间>，则脉冲输出。（输出时间的单位是 ms。）如果指定 <输出时间>，则在其时间没有经过之前，不向下一个命令进行。此外，输出时间的设定值是最低输出时间，实际的输出时间会受任务的优先顺序影响而变化。

应用示例

SET IO [64] ' (或 SET IO64) 将端口编号 64 设置为 ON。
SET IO [128],50 ' (或 SET IO128, 50) 将端口编号 128 设置为 ON，则 50ms 之后 OFF，转移到下一个命令。

19.2.2 RESET

功能 将 I/O 端口置为 OFF。

格式 RESET [□] <I/O 变量>

说明 将在 <I/O 变量> 中指定的端口置于 OFF。使用端口编号或 I/O 变量对端口进行指定。

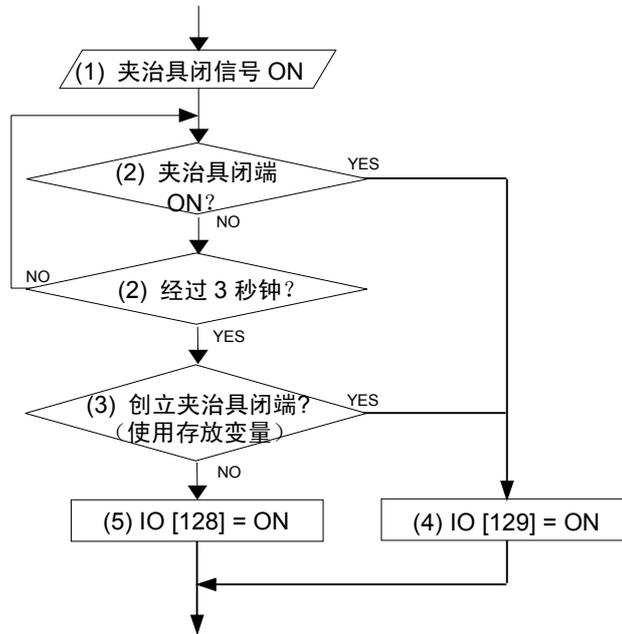
应用示例

RESET IO [64] ' (或 SET IO64) 将端口编号 64 设置为 OFF。

19.3 练习题 I/O 端口控制指令

练习题 3

请使用 I/O 端口控制语句对进行以下的流程控制的动作程序进行创建。



■ 动作规格

- ① 夹治具关闭信号使用IO [64]
- ② 夹治具关闭端信号使用IO [48]，使用WAIT语句等待信号输入
- ③ 判定手动夹治具成立要使用WAIT语句的存放变量 (I [20])

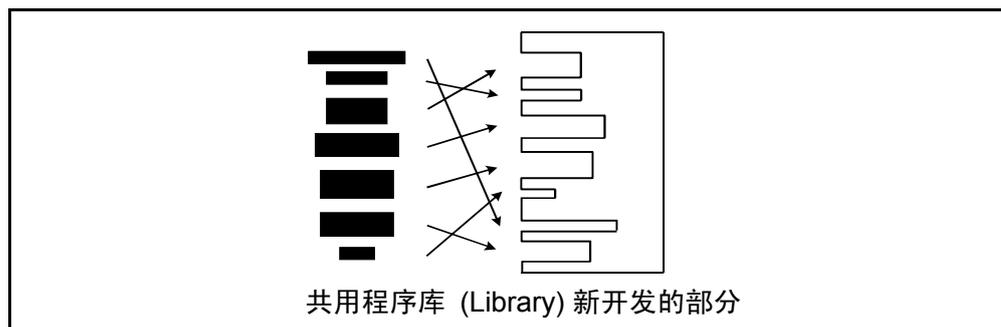
程序	运行说明
'TITLE "练习程序3"	'程序标题
PROGRAM PRO3	'程序名的定义
_____	'(1): 打开夹治具关闭信号IO [64]
_____	'(2): IO [64] 输入等待3秒，使用存放变量数I20
IF I [20] = 1 THEN	'(3): 若保存放变量I20 = 1 (成立)，则转移到下一个命令
_____	'(4): 打开IO [129]
ELSE	'(3): 存放变量I20 = 1 (成立) 以外移到下一个命令
_____	'(5): 打开IO [128]
ENDIF	'(3): IF语句的结束
END	'程序停止

第 20 章 程序语言（共用程序库 (Library) 篇）

20.1 共用程序库 (Library) 程序的运用

20.1.1 共用程序库 (Library) 的概念

将具有通用性的程序象零部件一样进行集中，以便按照用途进行利用的软件，被称为程序的共用程序库 (Library)。在 PAC 语言中，因为可以从程序中调出其他的程序，所以可以利用共用程序库 (Library) 中的程序，或者将编写的程序登录在共用程序库 (Library) 中，能够进行高效率的程序开发。



使用共用程序库 (Library) 进行程序开发的构思

20.1.2 程序库 (Program Bank) 功能

在 WINCAPSIII, 准备有用于使用库的程序库 (Program Bank)。程序库 (Program Bank) 是将程序作为库进行登录或用于将所登录的程序项目加入到工程中的工具。

为了使用程序库 (Program Bank) 中登录的共用程序库 (Library), 需要进行库存程序的追加 (引入)。关于库存共用程序库 (Library) 的追加方法, 请参照 "20.1.4 共用程序库 (Library) 的追加 (引入) 方法"。

20.1.3 共用程序库 (Library) 的分类

标准的共用程序库 (Library), 分为以下 6 种等级进行提供。

标准共用程序库 (Library) 的等级

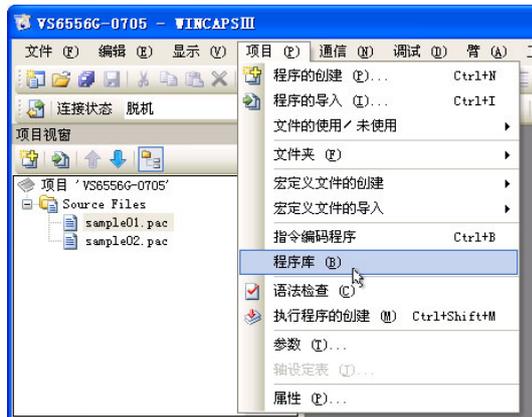
	等级名称	内容
1	原有语言	提供与原有语言的指令同等的功能。
2	码垛堆积	提供码垛堆积功能。
3	工具操作	提供工具操作相关的功能。
4	输出入	提供DIO、RS232C输出入相关的功能。
5	臂动作	提供上述之外的臂动作相关的功能。
6	视觉	提供视觉动作相关的功能。
7	Ver. 1.2互换	提供控制器的软件版本在Ver. 1.2 *以前可以使用的共用程序库 (Library)。对象有以下3种。如果用Ver. 1.2 *从上述等级使用以下3种的程序, 则会发生编译程序异常。除该3种以外, 请使用上述1~6的共用程序库 (Library)。 ndVcom、pltMove、pltMoveO

20.1.4 共用程序库 (Library) 的追加 (引入) 方法

请尝试把程序 "dioSetAndWait" 从程序库 (Program Bank) 项目中追加到计划中。

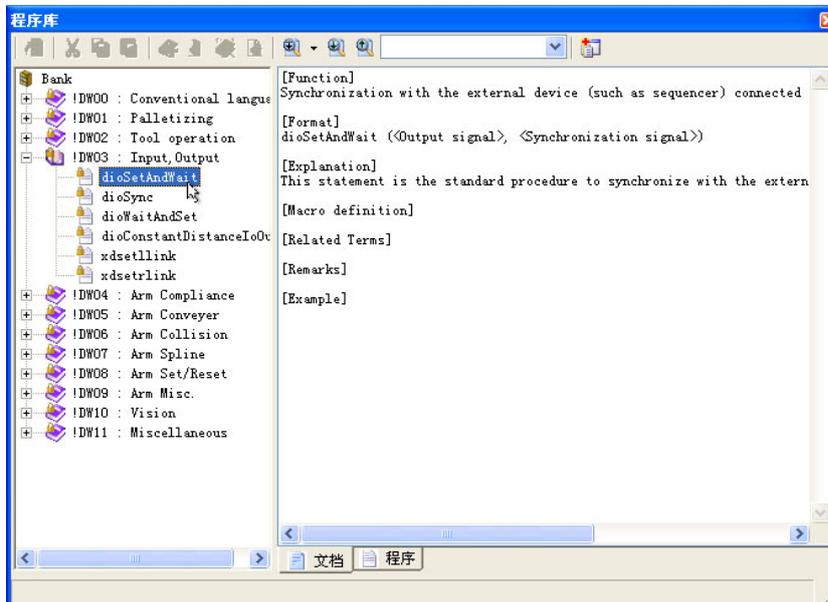
步骤 1 在 WINCAPSIII 中打开项目。

步骤 2 选择 [项目 (P)] - [程序库 (B)]。



步骤 3 选择程序。

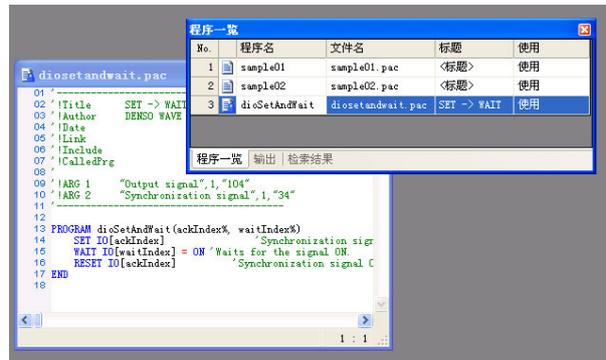
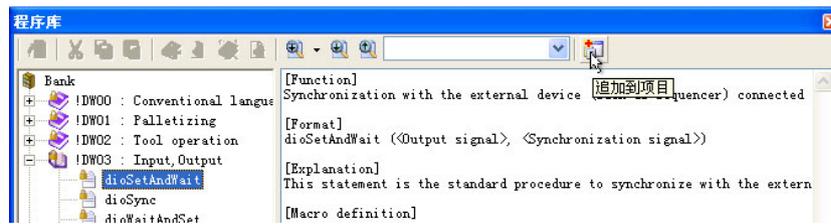
"dioSetAndWait" 在 "!DW03:Input, Output" 中, 因此请打开 [!DW03:Input, Output], 选择 [dioSetAndWait]。



通过 [文档] 图标与 [程序] 图标确认内容。

步骤 4

点击 [追加到程序] 所显示出来的程序将会登录至项目内。



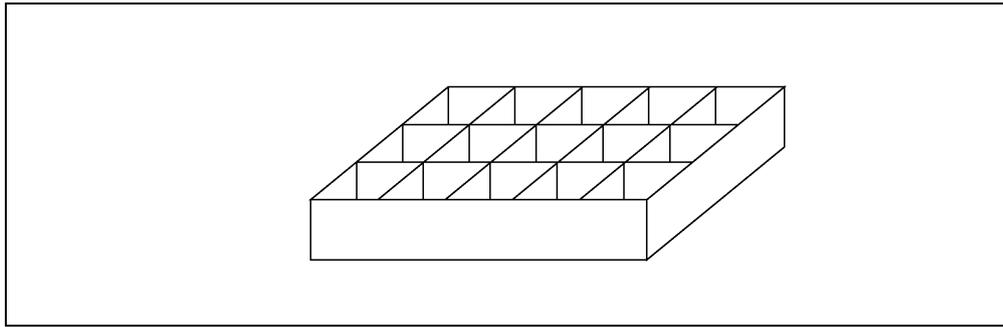
20.2 码垛共用程序库 (Library) 的运用

20.2.1 码垛堆积的说明

如下图所示的有间隔的托板上，顺次进行零部件的装入和取出的操作，称为码垛堆积。

码垛堆积用的共用程序库 (Library) 程序为：在只进行教导的托板的间隔数和四角的位置的情况下，即可以进行码垛堆积动作。

码垛堆积的程序，在每次运行调出时，会顺序改变托板的取出位置。



有隔板的托板

20.2.2 简易码垛堆积共用程序库 (Library)

为了进行码垛堆积运行，需要通过程序库 (Program Bank)，将 xdGetPalt 共用程序库 (Library) 预先追加 (引入) 到项目内。



简易码垛堆积共用程序库 (Library) (xdGetPa1t) 被引入状态。

20.2.3 码垛堆积的参数

进行码垛堆积所需要的参数列于下图 -1)、-2)、-3) 和表 -4) 中。
在 PAC 语言中，将这些参数作为变量的值保存起来。

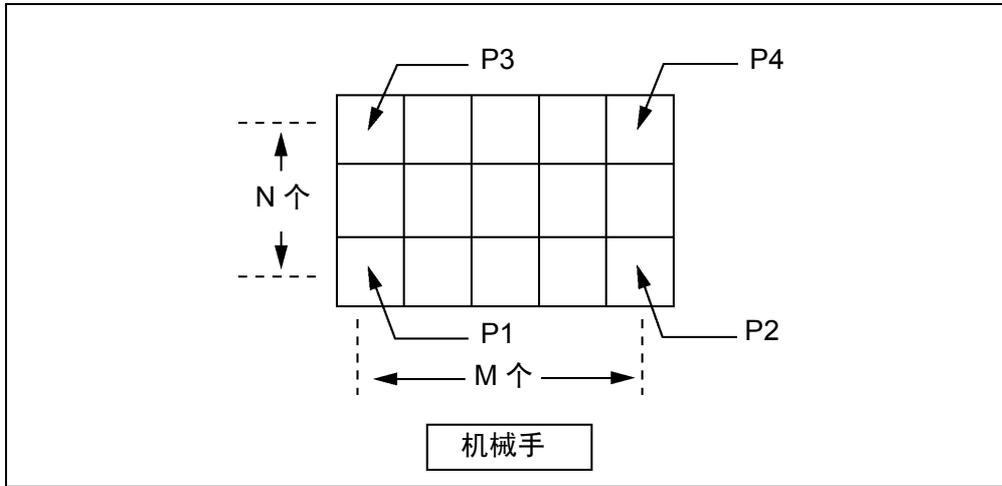


图 -1) 托板的俯视图

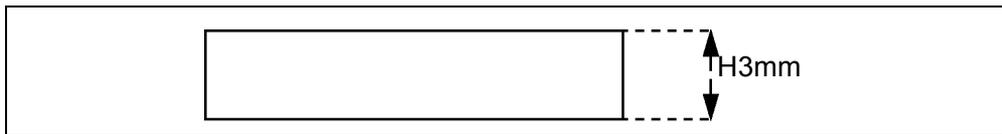


图 -2) 托板的侧视图

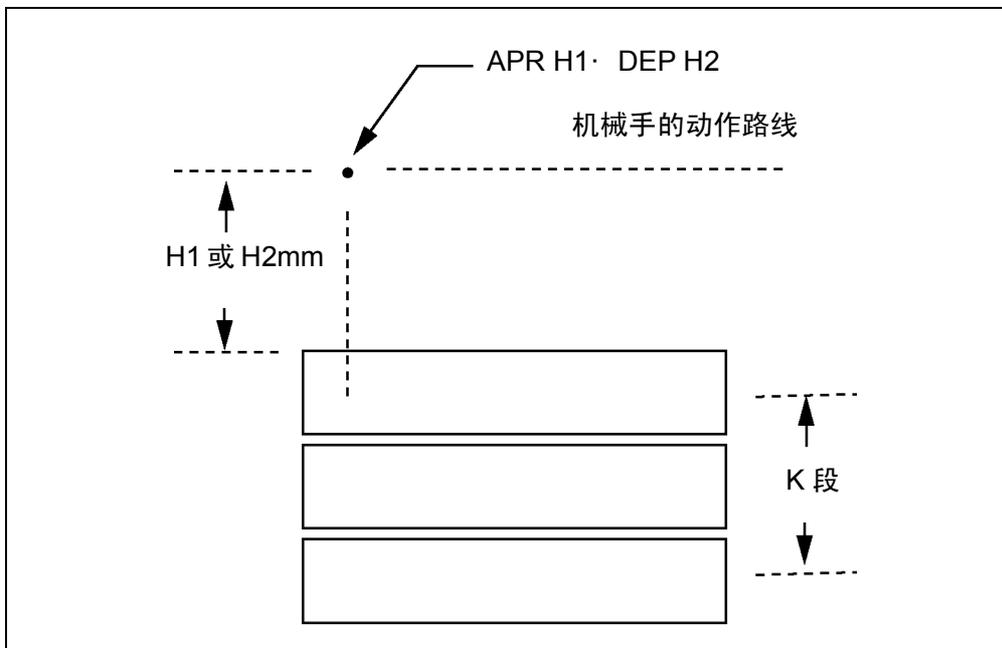


图 -3) 托板的段积图

表 -4) 码垛堆积所需的参数

记号	名称	意思	单位
	码垛堆积编号	码垛堆积的索引编号	无 (整数)
N	横拆分数	从P1到P3方向的拆分数	个 (整数)
M	纵拆分数	从P1到P2方向的拆分数	个 (整数)
K	段积数	托板的段积数	个 (整数)
H1	接近长度	机械手靠近托板时的接近长度	mm (单精度实数)
H2	离开长度	机械手离开托板时的起程长度	mm (单精度实数)
H3	托板高度	托板的1段的高度	mm (单精度实数)
	但是, H1和H2必须满足以下的条件。 $H1 > \{H3 \times (K-1)\} + 5$ $H2 > \{H3 \times (K-1)\} + 5$		
P1 P2 P3 P4	图16-3所示的托板四角的点。各点的相对位置关系, 不能进行互换。 还有机械手的姿势是教导P1的位置时的姿势, 在所有的点上均维持该姿势。		

N 横向拆分数

表示托板的横向的拆分数。
 在图 16-3 中成为 3 列。

M 纵向拆分数

表示托板的纵向的拆分数。
 在图 16-3 中成为 5 列。

K 段积数

表示托板的段积数。
 在图 16-5 中成为 3 段。

H1 接近长度

表示机械手靠近托板时的接近长度。
 每次调出码垛堆积均使用相同的接近长度。

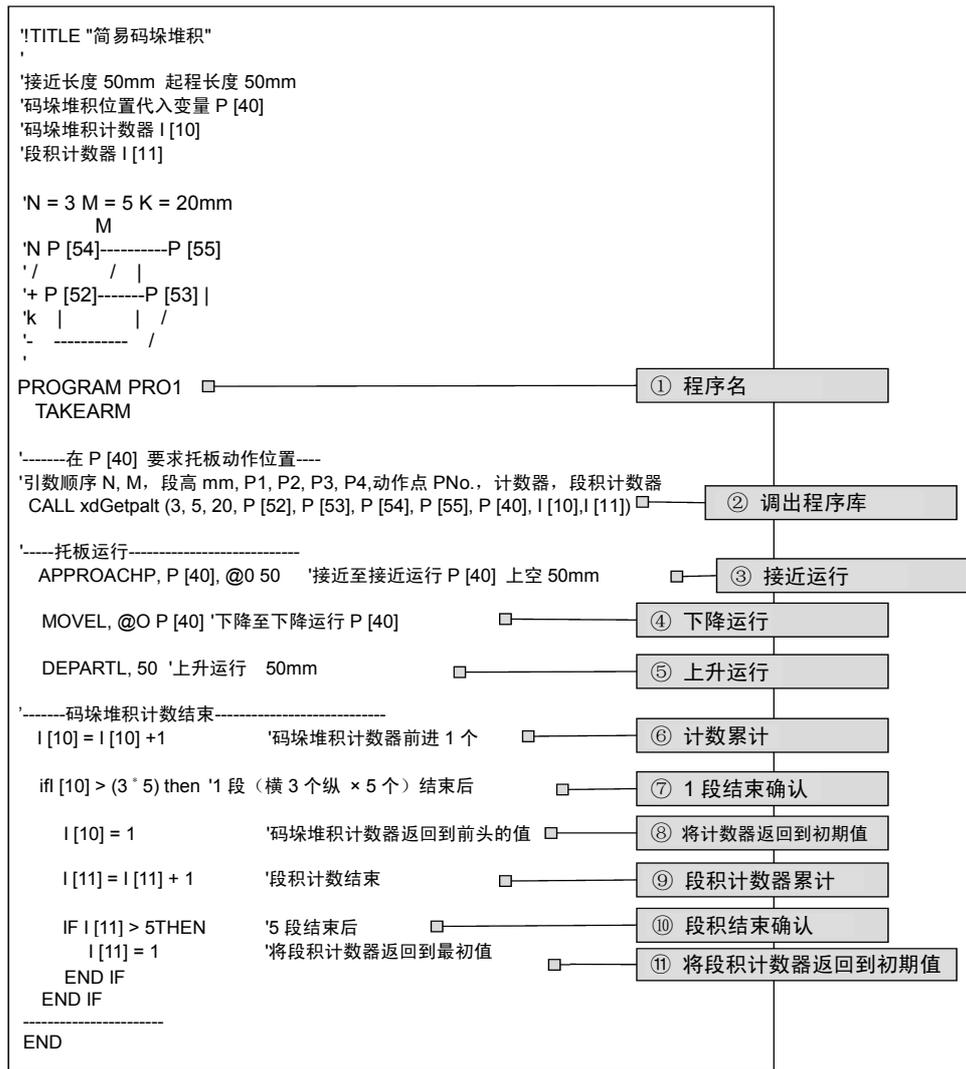
H2 起程长度

显示机械手离开托板时的起程长度。
 每次调出码垛堆积均使用相同的起程长度。

H3 托板高度

显示托板的一段的高度。
 托板逐渐向上累积时, 输入正的数值。
 托板逐渐减少时, 输入负的数值。
 托板的段数不变化时, 输入 0。

20.2.4 简易码垛堆积程序 "PRO1"



简易码垛堆积程序 "PRO1" 的说明

码垛堆积计数器和分段累积计数器均需要用户进行加法处理。

在 PRO1 上所使用的变量

码垛堆积目标位置赋值变量 (P 型: 在示例中使用 P40)

- 码垛堆积计数器变量 (I 型: 在示例中使用 I10)
- 段累积计数器变量 (I 型: 在示例中使用 I11)
- 托板 4 角位置变量 P1~P4 (P 型: 在示例中使用 P52~P55)

需置于 PRO1 动作之前

- 需要在码垛堆积计数器 I10、段累积计数器 I11 上赋值初始值 1。
- 需要将托板四角的位置教导向托板 4 角位置变量 P1~P4。

以下分为各个部分, 对程序 PRO1 进行解说。

① 程序名

```
,  
PROGRAM PRO1  
TAKEARM
```

请变更程序的名称

② 调出程序库

```
----- 在 P [40] 要求托板动作位置 -----  
'引数顺序 N, M, 段高 mm, P1, P2, P3, P4, 动作点 PNo., 计数器, 段积计数器  
CALL xdGetPalt (3, 5, 20, P [52], P [53], P [54], P [55], P [40], I [10], I [11])
```

在该共用程序库 (Library) 中设定以下所示的自变量 (argument), 目标位置被赋值第8号自变量 (argument) 所指定的变量。

自变量 (argument)

第1号… 横向拆分数N, 需要在1以上。

在示例中为3列。

第2号… 纵向拆分数M, 需要在1以上。

在示例中为5列。

第3号… 段累积高度 (mm)

在累积上升的情况下, 指定正值; 在下降的情况下, 指定负值。

在示例中指定20mm。

第4号~第7号… 分别指定表示托板的四角位置的P型变量。在示例中指定P52~P55。

第8号… 目标位置赋值变量

指定赋值动作位置的P型变量。

这个位置是从现在的计数器求出的码垛堆积的位置。在示例中指定P40。

第9号… 是码垛堆积计数器。需要在1以上、 $N * M$ 以下。用这个值指定码垛堆积目标位置。

第10号… 是段累积计数器。需要1以上的数值。用这个值指定段位置。

- ③ 接近运行
- ④ 下降运行
- ⑤ 上升运行

```

----- 托板运行 -----
APPROACH P,P[40],@0 50      '接近至接近运行 P [40] 上空 50mm

MOVE L,@0 P[40]             '下降至下降运行 P [40]

DEPART L,50                  '上升运行 50mm

```

执行 "② 共用程序库 (Library) 调出" 的结果，是目标位置被赋值P40，所以要在P40上进行动作。

通常，在这些运行的过程中，要插入夹紧、松开的处理。

- ⑥ 计数器累计
- ⑦ 一段结束确认
- ⑧ 将计数器返回到初期值
- ⑨ 段积计数器累计
- ⑩ 段积结束确认
- ⑪ 将段积计数器返回到初期值

```

----- 码垛堆积计数结束 -----
I[10] = I[10] + 1           '码垛堆积计数器前进 1 个

IF I[10] > (3 * 5) then    '1 段（横 3 个纵 × 5 个）结束后
  I[10] = 1                '码垛堆积计数器返回到最初的值
  I[11] = I[11] + 1       '段积计数结
  IF I[11] >= 5 THEN      '束 5 段结束后
    I[10] = 1             '将段积计数器返回到前头的值
  END IF
END IF

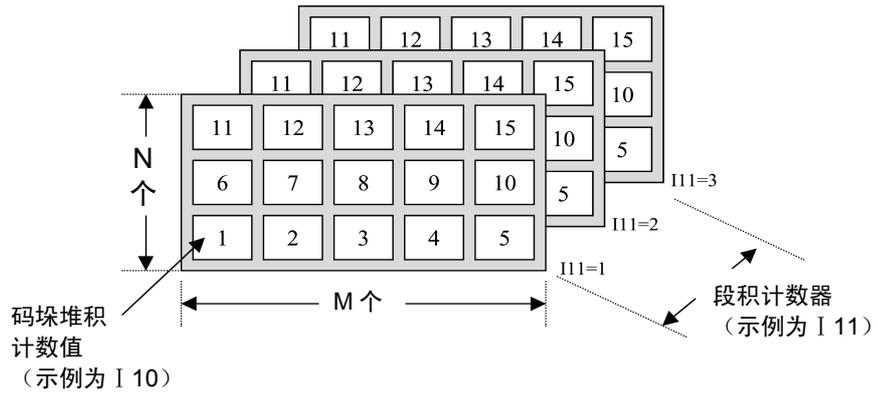
```

在此，要进行码垛堆积计数器和段累积计数器的计数增加以及步骤结束确认处理。

在简易码垛堆积中，与通常的码垛堆积不同，作为码垛堆积计数器，采用通常的I型变量。在示例中，使用I10。

还有阶段累积计数器也同样使用通常的I型变量。在示例中，使用I11。

在 "② 共用程序库 (Library) 调出" 中，从I10和I11的值计算出位置，并赋值P40。



还有只在1阶段的码垛堆积时，可以按以下方法进行单纯化。

```

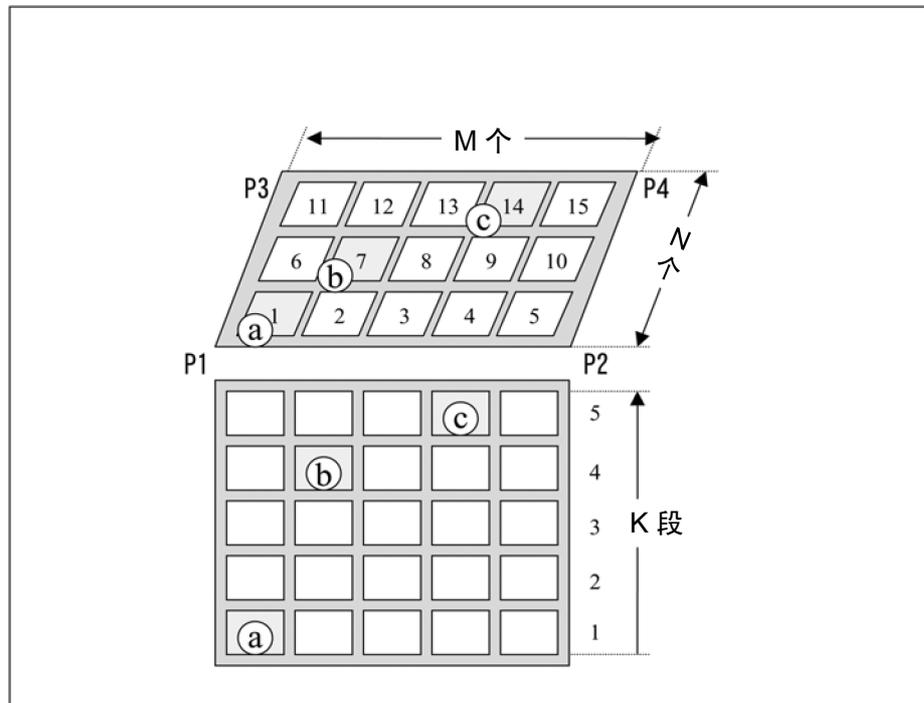
----- 码垛堆积计数结束 -----
I[10] = I[10] + 1           '码垛堆积计数器前进 1 个

if I[10] > (3 * 5) then    '1 段 (横 3 个纵 × 5 个) 结束后
    I[10] = 1              '码垛堆积计数器返回到最初的值
    I[11] = I[11] + 1     '段积计数结束
    IF I[11] >= 5 THEN    '5 段结束后
        I[10] = 1        '将段积计数器返回到前头的值
    END IF
END IF

```

一阶段的码垛堆积时，可以删除。

在简易码垛堆积中，位置和计数器的关系



托板 $N = 3$ 、 $M = 5$ 、码垛堆积计数器 I_{10} ，阶段累积计数器 I_{11} 的情况下

- Ⓐ $I_{10} = 1$, $I_{11} = 1$
- Ⓑ $I_{10} = 7$, $I_{11} = 4$
- Ⓒ $I_{10} = 14$, $I_{11} = 5$

附录

练习题答案示例

■ 练习题 1（17.3 练习题 机械手控制指令）

程序	运行说明
'TITLE "练习程序1"	'程序标题
PROGRAM PRO1	'程序名的定义
TAKEARM	'获取臂控制权
SPEED 100	'内部速度100%
DRIVEA @ 0 (1, 0)	'(1): 移动J1轴至0度位置
APPROACH P, P1, @ P 50	'(2): P1的夹治具方向 向上空移动50mm
MOVE L, @ E P1, S = 20	'(3): 移动至P1
DEPART L, @ P 50	'(4): P1的夹治具方向 向上空移动50mm
APPROACH P, P2, @ P 50	'(5): P2的夹治具方向 向上空移动50mm
MOVE L, @ E P1, S = 20	'(6): 移动至P2
DEPART L, @ P 50	'(7): P2的夹治具方向 向上空移动50mm
MOVE P, @ 0 P10	'(8): 移动至P10
END	'程序停止

■ 练习题 2（18.6 练习题 流程控制指令）

程序	运行说明
'TITLE "练习程序2"	'程序标题
PROGRAM PRO2	'程序名的定义
TAKEARM	'获取臂控制权
SPEED 100	'内部速度100%
MOVE P, P1	'(1): 移动至P1
IF I [5] = 0 THEN	'(2): IF语句 若I5 = 0则移至下个命令
APPROACH L, P10 50	'(2) -1: P10的夹治具方向 向上空移动50mm
MOVE L, P10	'(3) -1: 移动至P10
ELSE	'(2): IF语句 若I5 = 0不成立, 移至下个命令
APPROACH L, P11 50	'(2) -2: P11的夹治具方向 向上空移动50mm
MOVE L, P11	'(3) -2: 移动至P11
ENDIF	'(2): IF语句的结束
CALL HAND-OPEN	'调出HAND_OPEN程序
DEPART L, 50	'(4) -1, (4) -2: P10或P11的夹治具方向 向上空移动50mm
END	'程序停止

■ 练习题 3 (19.3 练习题 I/O 卡控制指令)

程序	运行说明
'TITLE "练习程序3"	'程序标题
PROGRAM PRO3	'程序名的定义
SET IO [64]	'(1): 打开夹治具关闭信号IO [64]
WAIT IO [64], 3000, I [20]	'(2): IO [64] 输入等待3秒, 使用保存变量数I20
IF I [20] = 1 THEN	'(3): 若存放变量I20 = 1 (成立), 移至下个命令
SET IO [129]	'(4): 打开IO [129]
ELSE	'(3): 存放变量I20 = 1 (成立) 以外, 移至下个命令
SET IO [128]	'(5): 打开IO [128]
ENDIF	'(3): IF语句的结束
END	'程序停止

按功能单位排列的指令一览

各功能指令一览所示的说明页表示编程手册1
的页。
请参照编程手册 1

按功能单位排列的指令一览

4轴 6轴 视觉
装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

功能区分	指令	功能	4轴	6轴	视觉装置	说明页
定义语句						
程序名	PROGRAM	定义程序名。	◎	◎	◎	9-1
干扰区域坐标	AREA	定义进行干扰检查的区域。	○	○		9-2
用户函数	DEF FN	定义用户定义函数。	◎	◎	◎	9-4
初始坐标	HOME	将任意的坐标作为初始位置定义。	○	○		9-5
工具坐标	TOOL	定义工具坐标系。	○	○		9-6
工件坐标	WORK	定义用户坐标系。	○	○		9-7
局部 (Local) 变量	DEFINT	定义整数型变量。整数的范围为-2147483648~2147483647。	◎	◎	◎	9-8
I型	DEFSNG	定义单精度实数型变量。单精度实数的范围为-3.4E-38~3.4E+38。	◎	◎	◎	9-8
F型	DEFSNG	定义单精度实数型变量。单精度实数的范围为-3.4E-38~3.4E+38。	◎	◎	◎	9-8
D型	DEFDBL	定义双精度实数型变量。双精度实数的范围是-1.7D+308~1.7D+308。	◎	◎	◎	9-9
S型	DEFSTR	定义字符串型变量。字符串的长度为最大243个字符。	◎	◎	◎	9-9
V型	DEFVEC	定义矢量型变量。	◎	◎		9-10
P型	DEFPOS	定义位置型变量。	○	○		9-10
J型	DEFJNT	定义关节型变量。	○	○		9-11
T型	DEFTRN	定义齐次变换型变量。	◎	◎		9-11
I/O型	DEFIO	定义与输入输出端口相对应的I/O变量。	◎	◎	◎	9-12
排列	DIM	排列定义数组。	◎	◎	◎	9-13
文件夹功能	FOLDER	定义从外部程序能够访问的局部(Local)变量。	V2.2	V2.2		9-14
	EXTERN	定义参照用其他的程序所定义的FOLDER变量的行为。	V2.2	V2.2		19-17
赋值语句						
变量	LET	对变量赋值。	○	○	○	10-1
矢量	LETA	对齐次变换型的近似矢量赋值。		◎		10-2
	LETO	对齐次变换型的定位矢量赋值。	◎	◎		10-3
	LETP	对位置型或齐次变换型的位置向量赋值。	◎	◎		10-4
形态	LETF	对位置型或齐次变换型的形态分量赋值。	◎	◎		10-5
连接角	LETJ	对关节型的指定连接角赋值。	◎	◎		10-6
姿势	LETR	对位置型的3个旋转分量赋值。		◎		10-7
旋转成分	LETRX	对位置型的X轴旋转分量赋值。		◎		10-8
	LETRY	对位置型的Y轴旋转分量赋值。		◎		10-9
	LETRZ	对位置型的Z轴旋转分量赋值。		◎		10-10
	LETT	对位置型的T分量赋值。	◎			10-11

4轴 6轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

功能区分	指令	功能	4轴	6轴	视觉装置	说明页
轴成分	LETX	对矢量型 / 位置型/齐次变换型的 X 轴分量赋值。	◎	◎		10-12
	LETY	对矢量型 / 位置型/齐次变换型的 Y 轴分量赋值。	◎	◎		10-13
	LETZ	对矢量型 / 位置型/齐次变换型的 Z 轴分量赋值。	◎	◎		10-14
流程控制语句						
程序停止	END	定义基于程序的动作结束。	◎	◎	◎	11-1
	STOP	结束程序的执行。	◎	◎	◎	11-2
	STOPEND	是循环停止循环起动或带有循环选项的启动程序的语句。循环起动包括该语句的程序时，即使执行该语句也不会影响运行。	◎	◎		11-3
调出	CALL	调出程序并执行。	◎	◎	◎	11-4
	GOSUB	调出子程序。	◎	◎	◎	11-6
	ON~GOSUB	根据表达式的值，执行对应子程序的调出。	◎	◎	◎	11-7
反复	RETURN	从子程序中返回。	◎	◎	◎	11-8
	DO~LOOP	进行迭代（循环）。	◎	◎	◎	11-9
	EXIT DO	执行从 DO~LOOP 中的强制退出。	◎	◎	◎	11-11
	FOR~NEXT	循环执行位于 FOR~NEXT 之间的一系列指令。	◎	◎	◎	11-12
	EXIT FOR	执行从 FOR~NEXT 中的强制退出。	◎	◎	◎	11-14
条件分支	REPEAT~UNTIL	执行后迭代。	◎	◎	◎	11-15
	WHILE~WEND	执行前迭代。	◎	◎	◎	11-16
	IF~END IF	执行 IF~END IF 之间的条件判断表达式中的条件判断。	◎	◎	◎	11-17
无条件分支	IF~THEN~ELSE	执行逻辑表达式的条件判断。	◎	◎	◎	11-18
	SELECT CASE	执行多个条件判断。	◎	◎	◎	11-19
	GOTO	无条件执行程序的分支。	◎	◎	◎	11-21
备注	ON~GOTO	执行根据表达式的值的无条件分支。	◎	◎	◎	11-22
	REM	进行备注的记述。	◎	◎	◎	11-23
机械手控制语句						
动作控制	APPROACH	执行工具坐标系指定的绝对动作。	○	○		12-1
	DEPART	通过工具坐标系的指定执行相对动作。	○	○		12-4
	DRAW	通过工件作业坐标系的指定执行相对动作。	◎	◎		12-7
	DRIVE	进行各轴的相对动作。	◎	◎		12-9
	DRIVEA	进行各轴的绝对动作。	◎	◎		12-11
	GOHOME	向用 HOME 语句定义的位置点(原点)移动。	◎	◎		12-13
	MOVE	将机械手向指定坐标移动。	○	○		12-14
	ROTATE	执行指定的旋转轴的旋转动作。	○	○		12-19
	ROTATEH	执行以接近矢量为轴的旋转动作。	◎	◎		12-22
	CURJNT	以 J 型取得机械手的当前角度。	○	○		12-24
	CURPOS	以 P 型数据得出工具坐标系中的当前位置。	○	○		12-25
	CURTRN	以 T 型数据得出工具坐标系中的当前位置。	◎	◎		12-26
	CUREXJ	以 F 型获取附加轴的当前角度。	V1.5	V1.5		12-27
	DESTJNT	以 J 型获取当前的动作命令目标位置。	○	○		12-28

4轴 6轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

功能区分	指令	功能	4轴	6轴	视觉装置	说明页
	DESTPOS	以 P 型获取当前的动作命令目标位置。 机械手停止时，获取当前的位置（指令值）。	○	○		12-29
	DESTTRN	以 T 型获取当前的动作命令目标位置。 机械手停止时，获取当前的位置（指令值）。	◎	◎		12-30
	DESTEXJ	以 F 型获取附加轴的以当前的动作指令所指定的目标位置。停止时，获取当前的位置（指令值）。	V1.5	V1.5		12-31
	ARRIVE	通过设定针对动作命令的全移动距离的动作比例，使程序待机直到到达机械手所设定的动作比例为止。	◎	V1.2		12-32
	POSCLR	将轴的当前位置强制性置为 0mm 或 0 度。	V1.5	V1.5		12-34
	SETSPLINEPOINT	登录自由曲线动作的通过点。	V2.3	V2.3		12-35
	CLRSPLINEPOINT	将自由曲线动作的通过点全部删除。	V2.3	V2.3		12-36
	GETSPLINEPOINT	取出已经登录的自由曲线动作的通过点。	V2.3	V2.3		12-37
形态控制	CURFIG	得出机械手形态的当前值。	◎	◎		12-38
	FIGAPRL	计算能够执行 CP 动作的接近位置和基准位置的形态。	○	○		12-40
	FIGAPRP	计算能够执行 PTP 动作的接近位置与基准位置的形态。	○	○		12-42
停止控制	HOLD	暂时停止程序的执行。	◎	◎		12-43
	HALT	停止程序的执行。	◎	◎		12-44
	INTERRUPT ON/OFF	中断机械手的动作。	◎	◎		12-45
速度控制	SPEED	指定当前所获取的臂组轴的内部移动速度。	◎	◎		12-47
	JSPEED	指定当前所获取的臂组轴的内部轴速度。	◎	◎		12-49
	ACCEL	指定当前所获取的臂组轴的内部加速度、内部减速度。	◎	◎		12-50
	JACCEL	指定当前所获取的臂组轴的内部轴加速度、内部轴减速度。	◎	◎		12-51
	DECEL	指定当前所获取的臂组轴的内部减速度。	◎	◎		12-52
	JDECEL	指定当前所获取的臂组轴的内部轴减速度。	◎	◎		12-53
	CURACC	获取当前所获取的臂组轴的内部加速度。	◎	◎		12-54
	CURJACC	获取当前所获取的臂组轴的内部轴加速度。	◎	◎		12-55
	CURDEC	获取当前所获取的臂组轴的内部减速度。	◎	◎		12-56
	CURJDEC	获取当前所获取的臂组轴的内部轴减速度。	◎	◎		12-57
	CURJSPD	获取当前所获取的臂组轴的内部轴速度。	◎	◎		12-58
	CURSPD	获取当前所获取的臂组轴的内部移动速度。	◎	◎		12-59
	CUREXTACC	得出外部加速度的当前值。	V1.4	V1.4		12-60
	CUREXTDEC	得出外部减速度的当前值。	V1.4	V1.4		12-61

4轴 6轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

功能区分	指令	功能	4轴	6轴	视觉装置	说明页
时间控制	CUREXTSPD	得出外部速度的当前值。	V1.4	V1.4		12-62
	EXTSPEED	设定外部速度。	V1.98	V1.98		12-62
	DELAY	在所指定的时间内，停止程序的执行。	◎	◎	◎	12-63
	WAIT	执行条件程序的停止。	◎	◎		12-64
坐标转换	CHANGETOOL	更改工具坐标系。	◎	◎		12-65
	CHANGEWORK	更改用户坐标系。	◎	◎		12-66
	CURTOOL	得出当前所设定的 TOOL 编号。	V1.4	V1.4		12-67
干扰检查	CURWORK	得出当前所设定的 WORK 编号。	V1.4	V1.4		12-68
	SETAREA	选择进行干扰检查的区域。	◎	◎		12-69
	RESETAREA	初始化干扰检查。	◎	◎		12-70
伺服内部数据	GetSrvData	获取机械手轴的伺服内部数据。	V1.5	V1.5		12-71
	GetJntData	获取指定轴的伺服内部数据。				12-72
电动机电源	MOTOR {ON OFF}	执行电机电源的 ON / OFF。	V1.5	V1.5		12-73
	CAL	执行 CAL。	V1.5	V1.5		12-74
特殊控制	ST_aspACLD	变更内部负荷条件值。以前端负荷质量 (g)、负荷重心位置 (mm) 指定所有负荷条件值。	V1.9	V1.9		12-75
	ST_aspChange	选择最佳搬运质量设定模式的内部模式。	V1.9	V1.9		12-76
	ST_SetGravity	修正各个关节的静负荷（重力转矩），设定重力平衡。	V1.9	V1.9		12-77
	ST_ResetGravity	将重力平衡设为无效。	V1.9	V1.9		12-78
	ST_SetGrvOffset	通过各个关节的重力转矩修正重力补偿值。	V1.9	V1.9		12-78
	ST_ResetGrvOffset	将重力补偿值的修正设为无效。	V1.9	V1.9		12-79
	ST_SetCurLmt	限制指定轴的电机电流值。	V1.9	V1.9		12-79
	ST_ResetCurLmt	解除指定轴的电机电流限制。	V1.9	V1.9		12-81
	ST_SetEralw	变更指定轴的偏差允许值。	V1.9	V1.9		12-82
	ST_ResetEralw	将指定轴的偏差允许值返回到默认值。	V1.9	V1.9		12-82
	ST_OnSrvLock	将指定轴置为伺服锁定状态。	V1.9			12-83
	ST_OffSrvLock	解除指定轴的伺服锁定。	V1.9			12-84
	ST_SetCompControl	将力限制功能设为有效。		V1.9		12-85
	ST_SetCompFControl	将力限制功能设为有效。		V1.9		12-86
	ST_ResetCompControl	将力限制功能设为无效。		V1.9		12-87
	ST_SetFrcCoord	选择力限制设定坐标系。		V1.9		12-88
	ST_SetFrcLimit	设定力限制比例。		V1.9		12-89
	ST_ResetFrcLimit	将力限制比例初始化。		V1.9		12-90
	ST_SetCompRate	设定力限制时的柔度的比例。		V1.9		12-91
	ST_ResetCompRate	将力限制时的柔度的比例初始化。		V1.9		12-92
	ST_SetFrcAssist	设定力限制时的偏移。		V1.9		12-92
	ST_ResetFrcAssist	将力限制时的偏移初始化。		V1.9		12-93
	ST_SetCompJLimit	设定力限制时的电流限制值。		V1.9		12-93
	ST_ResetCompJLimit	将力限制时的电流限制值初始化。		V1.9		12-94
	ST_SetCompVMode	设定力限制时的速度控制模式。		V1.9		12-95
	ST_ResetCompVMode	将力限制时的速度控制模式设为无效。		V1.9		12-95
	ST_SetCompEralw	设定力限制时的工具端的位置、姿势偏差允许值。		V1.9		12-96
ST_ResetCompEralw	将力限制时的工具端的位置、姿势偏差允许值初始化。		V1.9		12-97	
ST_SetDampRate	设定力限制时的粘性比例。		V1.9		12-97	

4轴 6轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

功能区分	指令	功能	4轴	6轴	视觉装置	说明页
	ST_ResetDampRate	将力限制时的粘性比例初始化。		V1.9		12-98
	ST_SetZBalance	设定Z轴、T轴的重力补偿值。	V1.9			12-99
	ST_ResetZBalance	将重力补偿值的修正设为无效。	V1.9			12-99
输出控制语句						
I/O 端口	IN	从通过I/O变量表示的I/O端口中读取数据。	◎	◎	◎	13-1
	OUT	向通过I/O变量表示的I/O端口输出数据。	◎	◎	◎	13-2
	IOBLOCK ON/OFF	在执行动作指令时，同步执行I/O指令、运算指令等非动作指令。	◎	◎		13-3
	SET	将I/O端口置为ON。	◎	◎	◎	13-5
	RESET	将I/O端口置为OFF。	◎	◎	◎	13-7
	INPUT	得出来自RS232C及Ethernet端口的数据。	◎	◎	◎	13-8
	LINEINPUT	从RS232C及Ethernet端口中读取数据直到间隔符，并对字符串型变量赋值。	◎	◎	◎	13-10
	PRINT	从RS232C及Ethernet端口输出数据。	◎	◎	◎	13-11
	WRITE	从RS232C及Ethernet端口输出数据。	◎	◎	◎	13-12
	FLUSH	清空输入缓冲区。		◎	◎	◎
串行二进制通信	PRINTB	向RS-232C及Ethernet端口输出1字节(Byte)数据。	V1.5	V1.5	V1.9	13-14
	INPUTB	从RS-232C及Ethernet端口输入1字节(Byte)数据。	V1.5	V1.5	V1.9	13-15
	LPRINTB	从RS-232C及Ethernet端口输出多字节(Byte)数据。	V1.5	V1.5	V1.9	13-16
	LINPUTB	从RS-232C及Ethernet端口输入多字节(Byte)数据。	V1.5	V1.5	V1.9	13-17
	com_encom	为了进行二进制通信占有RS-232C端口。(占有COM端口)	V1.5	V1.5	V1.9	13-18
	com_discom	为了结束二进制通信释放RS-232C端口。(释放COM端口)	V1.5	V1.5	V1.9	13-19
多功能教导器	com_state	获取RS-232C端口的状态。	V1.5	V1.5	V1.9	13-20
	PRINTMSG	将讯息附上字幕和图标，在多功能教导器的彩色液晶中显示。	◎	◎		13-21
	PRINTDBG	向调试视窗输出数据。	◎	◎		13-22
	BUZZER	鸣响蜂鸣器。	◎	◎		13-22
	PRINTWARNING	在教导器警告讯息显示部分显示讯息。	V2.2	V2.2		13-23
TP 简易操作盘	PRINTLBL	向用户定义按钮设定标签(字幕)。	◎	◎		13-23
	set_button	进行按钮参数的设定。	V1.5	V1.5		13-27
	set_page	进行页面参数的设定。	V1.5	V1.5		13-31
	change_bCap	进行按钮的字幕设定。	V1.5	V1.5		13-33
	change_pCap	进行页面的字幕设定。	V1.5	V1.5		13-34
	disp_page	进行页面的显示。	V1.5	V1.5		13-35
多项任务控制语句						
任务控制	RUN	与其它程序同时启动。	◎	◎		14-1
	KILL	强制结束任务。	◎	◎		14-2
	SUSPEND	暂停任务。	◎	◎		14-3

4 轴 6 轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于 4 轴机械手、6 轴机械手或视觉装置的规格而异。
◎	V1.2		可在 4 轴机械手和版本 1.2 以上的 6 轴机械手上使用

功能区分	指令	功能	4 轴	6 轴	视觉装置	说明页
	DEFEND	保护任务。	◎	◎		14-4
	STATUS	得出程序的状态。	◎	◎		14-5
	SUSPENDALL	停止特权任务以外的所有的程序。	V1.98	V1.98		14-6
	KILLALL	强制结束特权任务以外的所有的任务。	V1.98	V1.98		14-7
	CONTINUERUN	进行连续启动。	V1.98	V1.98		14-7
	ROBOTSTOP	停止机械手。	V1.98	V1.98		14-8
	TAKEARMSTATE	返回臂组控制权的获取状态。	V2.2	V2.2		14-9
	LOCKSTATE	获取机器锁定的状态。	V2.2	V2.2		14-9
	DEADMANSTATE	获取双重安全开关的状态。	V2.2	V2.2		14-10
	SEMIDSTATE	返回所指定的信号(semaphore) ID 是有效还是无效。	V2.2	V2.2		14-11
信号 (semaphore)	CREATESEM	生成信号 (semaphore)。	◎	◎		14-12
	DELETESEM	删除信号 (semaphore)。	◎	◎		14-15
	FLUSHSEM	释放信号 (semaphore) 等待任务。	◎	◎		14-16
	GIVESEM	释放信号 (semaphore) 等待任务。	◎	◎		14-17
	TAKESEM	获取拥有所指定的信号(semaphore) ID 的信号 (semaphore)。	◎	◎		14-18
臂 信 号 (semaphore)	TAKEARM	获取臂组。当省略了臂组 NO 时，取得臂组 0 (仅限机械手) 的控制权。获取时，将内部速度、加速度、减速度设定为 100。 当获取的臂组包含机械手轴时，将工具坐标、工件坐标返回到原点。	◎	◎		14-19
	GIVEARM	释放机械手控制权。	◎	◎		14-24
	TAKEVIS	获取视觉处理权。	◎	◎		14-25
	GIVEVIS	释放视觉处理权。	◎	◎		14-26
特权任务	INIT	根据特权任务参数的状态，进行电机 ON、CAL、速度的设定。	V1.7	V1.7		14-27
	SETOCCUPATIONTIME	改变特权任务的占有时间。	V2.0	V2.0		14-28
	INITWAITERR	初始化错误检测。(特权任务专用命令)	V2.2	V2.2		14-29
	WAITERROR	检测错误发生。(特权任务专用命令)	V2.2	V2.2		14-30
	CURERRSTATUS	返回错误发生状态。(特权任务专用命令)	V2.2	V2.2		14-31
函数						
算术函数	ABS	得出算式的值的绝对值。	◎	◎	◎	15-1
	EXP	得出将自然对数作为基数的指数函数。	◎	◎	◎	15-2
	INT	得出不超过被指定值的最大的整数。	◎	◎	◎	15-3
	LOG	得出自然对数。	◎	◎	◎	15-4
	LOG10	得出常用对数。	◎	◎	◎	15-5
	POW	求取幂乘。	◎	◎	◎	15-6
	MAX	抽出最大值。	◎	◎	◎	15-7
	MIN	抽出最小值。	◎	◎	◎	15-8
	RND	发生大于 0 小于 1 的随机数。	◎	◎	◎	15-9
	SGN	调查符号。	◎	◎	◎	15-10
	SQR	得出平方根。	◎	◎	◎	15-11
三角函数	ACOS	得出反余弦 (arc cosine)。	◎	◎	◎	15-12
	ASIN	得出反正弦 (arc sine)。	◎	◎	◎	15-13
	ATN	得出反正切 (arc tangent)。	◎	◎	◎	15-14

4轴 6轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

功能区分	指令	功能	4轴	6轴	视觉装置	说明页	
角度转换	ATN2	得出以算式 1 除以算式 2 之后的反正切 (arc tangent)。	◎	◎	◎	15-15	
	COS	得出余弦 (cosine)。	◎	◎	◎	15-16	
	SIN	得出正弦 (sine)。	◎	◎	◎	15-17	
	TAN	得出正切 (tangent)。	◎	◎	◎	15-18	
	DEGRAD	将单位转换为弧度。	◎	◎	◎	15-19	
	RAD	将通过弧度赋予的数值转换为角度。	◎	◎	◎	15-20	
	RADDEG	将单位转换为度。	◎	◎	◎	15-21	
速度转换	MPS	转换速度的表示。	◎	◎		15-22	
时间函数	SEC	将以秒为单位赋予的数值转换为毫秒。	◎	◎	◎	15-23	
矢量	AVEC	抽出近似矢量。	◎	◎		15-24	
	OVEC	抽出定向矢量。	◎	◎		15-25	
	PVEC	抽出位置矢量。	○	○		15-26	
	MAGNITUDE	得出矢量的大小。	◎	◎		15-27	
姿势 (Pose) 数据类型转换	J2P	从关节型转换为位置型。	○	○		15-28	
	J2T	从关节型转换为齐次变换型。	○	○		15-29	
	P2J	从位置型转换为关节型。	○	○		15-30	
	P2T	从位置型转换为齐次变换型。	○	○		15-31	
	T2J	从齐次变换型转换为关节型。	◎	◎		15-32	
	T2P	从齐次变换型转换为位置型。	◎	◎		15-33	
	TINV	计算齐次变换型的逆矩阵。	◎	◎		15-34	
	NORMTRN	进行齐次变换型的正规化计算。	V1.8	V1.8		15-34	
	距离抽出	DIST	返回 2 点间的距离。	○	○		15-35
	形态成分	FIG	抽出形态。	○	○		15-36
角度成分	JOINT	从关节型坐标抽出角度。	○	○		15-37	
	轴成分	POSX	抽出 X 分量。	○	○		15-38
	POSY	抽出 Y 分量。	○	○		15-39	
	POSZ	抽出 Z 分量。	○	○		15-40	
旋转成分	POSRX	抽出 X 轴旋转分量。		◎		15-41	
	POSRY	抽出 Y 轴旋转分量。		◎		15-42	
	POSRZ	抽出 Z 轴旋转分量。		◎		15-43	
	POST	抽出 T 轴旋转分量。	◎			15-44	
姿势成分	RVEC	抽出姿势。		◎		15-45	
	位置函数	AREAPOS	以位置型返回进行干扰检查的区域的中心位置和长方体的方向。	◎	◎		15-46
	AREASIZE	将定义进行干扰检查的区域的长方体的大小 (各个边的长度) 以矢量型进行返回。	◎	◎		15-47	
字符串函数	TOOLPOS	将工具坐标系以位置型进行返回。	◎	◎		15-48	
	WORKPOS	将用户坐标系以位置型进行返回。	◎	◎		15-49	
	ASC	转换为字符编码。	◎	◎	◎	15-50	
	BIN\$	将算式的值转换为 2 进制数的字符串。	◎	◎	◎	15-51	
	CHR\$	将 ASCII 编码转换为字符。	◎	◎	◎	15-52	
	SPRINTF\$	转换为指定了表达式的格式化，并作为字符串返回。	◎	◎	◎	15-53	
	HEX\$	以字符串形式得出从 10 进制数转换为 16 进制数后的值。	◎	◎	◎	15-56	
	LEFT\$	抽出左部分字符串。	◎	◎	◎	15-57	
	LEN	以字节 (Byte) 数得出字符串的长度。	◎	◎	◎	15-58	

4轴 6轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

功能区分	指令	功能	4轴	6轴	视觉装置	说明页
	MID\$	从字符串中取出指定的字符数部分的字符串。	◎	◎	◎	15-59
	ORD	转换为字符编码。	◎	◎	◎	15-60
	RIGHT\$	抽出字符串的右边部分。	◎	◎	◎	15-61
	STRPOS	得出字符串的位置。	◎	◎	◎	15-62
	STR\$	将数值转换为字符串。	◎	◎	◎	15-63
	VAL	从字符串转换为数值。	◎	◎	◎	15-64
常量						
内置常量	OFF	赋予 OFF (0) 值。	◎	◎	◎	16-1
	ON	赋予 ON (1) 值。	◎	◎	◎	16-2
	PI	赋予 π 值。	◎	◎	◎	16-3
	FALSE	赋予布尔值的假 (0) 的值。	◎	◎	◎	16-4
	TRUE	赋予布尔值的真 (1) 的值。	◎	◎	◎	16-5
时间 / 日期控制						
时间 / 日期	DATE\$	得出当前的日期。	◎	◎		17-1
	TIME\$	得出当前的时间。	◎	◎		17-2
	TIMER	得出经过时间。	◎	◎		17-3
错误控制						
错误信息	ERRMSG\$	赋予错误讯息。	◎	◎	◎	18-1
	SETERR	将用户定义错误设定到 I 型变量区域。	V1.98	V1.98		18-1
	GETERR	从通过错误存放功能定义的缓冲区中获取错误编码。	V1.98	V1.98		18-2
	CLRERR	清空错误。	V1.98	V1.98		18-3
	GETERRLVL	赋予错误编码的等级。	V1.98	V1.98		18-3
系统信息						
系统	GETENV	获取系统的环境设定值。	◎	◎	◎	19-1
	LETENV	设定系统的环境设定值。	◎	◎	◎	19-2
	VER\$	获取各个模块的版本。	◎	◎	◎	19-3
	GETLANGUAGE	获取当前的语言设定。	V2.2	V2.2		19-3
记录	STARTLOG	开始伺服控制 LOG 的记录。	◎	◎		19-4
	CLEARLOG	对伺服控制 LOG 的记录进行初始化。	◎	◎		19-5
动作模式	STOPLOG	停止伺服控制 LOG 的记录。	◎	◎		19-6
	CHGEXTMODE	进行外部自动模式切换。	V1.98	V1.98		19-7
	CHGINTMODE	进行内部自动模式切换。	V1.98	V1.98		19-8
	CUROPTMODE	获取动作模式。	V1.98	V1.98		19-8
	SYSSTATE	获取控制器的状态。	V1.98	V1.98		19-9
预处理器						
符号常量、宏定义	#define	使用指定字符串替换程序中所指定的常量或宏名。	◎	◎	◎	20-1
	#undef	将通过 #define 定义的符号常量或宏的定义置为无效。	◎	◎	◎	20-2
	#error	如果执行 #error 指令，则强制性置为编译错误。	◎	◎	◎	20-3
文件取得	#include	读入预处理程序。	◎	◎	◎	20-4
最佳化	#pragma optimize	指定按程序单位执行的最优化处理。	◎	◎	◎	20-6
视觉控制						
图像输出	CAMIN	将来自照相机的图像存放在图像存储器（处理画面）中。	◎	◎	◎	21-3
	CAMMODE	设定存放照相机图像时的功能。	◎	◎	◎	21-4

4轴 6轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

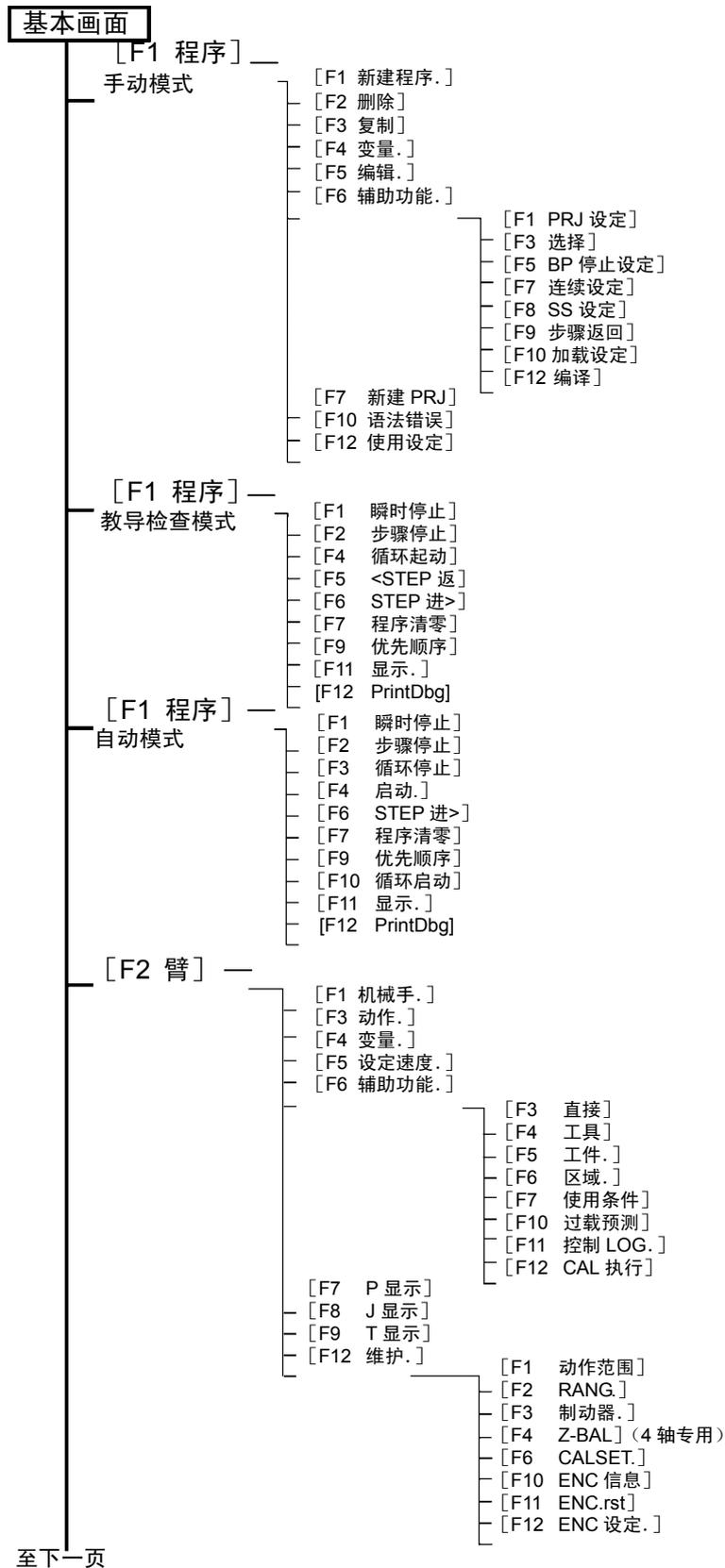
功能区分	指令	功能	4轴	6轴	视觉装置	说明页
视窗设定	CAMLEVEL	设定照相机图像的输入等级。	◎	◎	◎	21-6
	VISCAMOUT	将来自照相机的图像显示在监视器上。	◎	◎	◎	21-7
	VISPLNOUT	将存放存储器的图像显示在监视器上。	◎	◎	◎	21-8
	VISOVERLAY	将绘制画面的信息显示在监视器上。	◎	◎	◎	21-9
	VISDEFTABLE	取得照相机图像，并设定图像输出时的一览表数据。	◎	◎	◎	21-11
	VISREFTABLE	参照一览表的数据。	◎	◎	◎	21-13
	WINDMAKE	指定图像处理的范围。	◎	◎	◎	21-14
	WINDCLR	消除所设定的视窗信息。	◎	◎	◎	21-19
	WINDCOPY	复制视窗的数据。	◎	◎	◎	21-20
	WINDREF	取得视窗的信息。	◎	◎	◎	21-22
描绘	WINDDISP	描绘所设定的视窗。	◎	◎	◎	21-23
	VISSCREEN	指定描绘的画面。	◎	◎	◎	21-24
	VISBRIGHT	指定描绘时的亮度值。	◎	◎	◎	21-26
	VISCLS	用指定辉度全面涂抹（清空）模式所指定的画面。	◎	◎	◎	21-27
	VISPUTP	在画面上描绘点。	◎	◎	◎	21-29
	VISLINE	在画面上描绘直线。	◎	◎	◎	21-30
	VISPTP	在画面上描绘连接2点间的直线。	◎	◎	◎	21-31
	VISRECT	在画面上描绘矩形。	◎	◎	◎	21-32
	VISCIRCLE	在画面上描绘圆。	◎	◎	◎	21-33
	VISELLIPSE	在画面上描绘椭圆。	◎	◎	◎	21-34
图像处理	VISSECT	在画面上描绘扇形。	◎	◎	◎	21-35
	VISCROSS	在画面上描绘十字标记。	◎	◎	◎	21-36
	VISLOC	指定文字的显示位置。	◎	◎	◎	21-37
	VISDEFCHAR	指定文字的大小与显示方法。	◎	◎	◎	21-39
	VISPRINT	在画面上显示文字、数字。	◎	◎	◎	21-40
	VISWORKPLN	指定处理对象的存放存储器（处理画面）。	◎	◎	◎	21-41
	VISGETP	从存放存储器（处理画面）中获取指定坐标的亮度。	◎	◎	◎	21-42
	VISHIST	得到画面的直方柱状体（辉度分布）。	◎	◎	◎	21-43
	VISREFHIST	读出直方柱状体结果。	◎	◎	◎	21-44
	VISLEVEL	根据直方柱状体结果求取2值化基准值等级。	◎	◎	◎	21-45
编码识别 标注	VISBINA	2值化基准值处理画面。	◎	◎	◎	21-47
	VISBINAR	2值化基准值显示画面。	◎	◎	◎	21-49
	VISFILTER	对图像进行过滤器处理。	◎	◎	◎	21-50
	VISMASK	执行图像间演算。	◎	◎	◎	21-52
	VISCOPY	复制画面。	◎	◎	◎	21-54
	VISMEASURE	测量视窗内的特征（面积、重心、主轴角）。	◎	◎	◎	21-55
	VISPROJ	测量视窗内的投影数据。	◎	◎	◎	21-58
	VISEGE	测量视窗内的边缘。	◎	◎	◎	21-60
	VISREADQR	读取QR编码。	◎	◎	◎	21-64
	BLOB	执行贴标。	◎	◎	◎	21-67
BLOBMEASURE	进行对象标签编号的特征测量。	◎	◎	◎	21-70	
BLOBLABEL	获取指定坐标的标签编号。	◎	◎	◎	21-72	

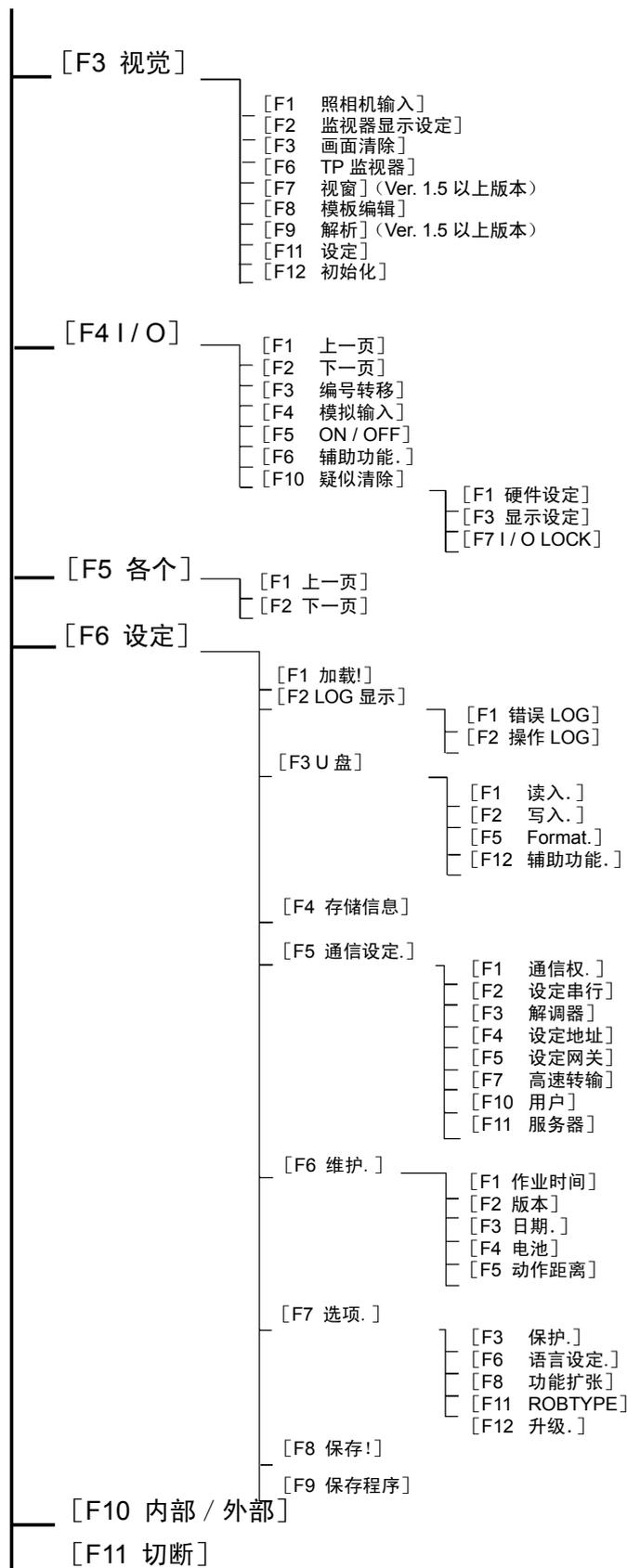
4轴 6轴 视觉装置

◎	◎	◎	可在所有机械手及视觉装置上使用
○	○	○	可在所有机械手上使用，但是用于4轴机械手、6轴机械手或视觉装置的规格而异。
◎	V1.2		可在4轴机械手和版本1.2以上的6轴机械手上使用

功能区分	指令	功能	4轴	6轴	视觉装置	说明页
检索功能	BLOBCOPY	复制对象标签编号。	◎	◎	◎	21-74
	SHDEFMODEL	执行检索模型的登录。	◎	◎	◎	21-76
	SHREFMODEL	参照登录模型数据。	◎	◎	◎	21-78
	SHCOPYMODEL	复制登录模型。	◎	◎	◎	21-79
	SHCLRMODEL	消除登录模型。	◎	◎	◎	21-80
	SHDISPMODEL	将登录模型显示在画面上。	◎	◎	◎	21-81
	SHMODEL	检索模型。	◎	◎	◎	21-82
	SHDEFCORNER	设定棱角检索的条件。	◎	◎	◎	21-86
	SHCORNER	检索棱角。	◎	◎	◎	21-87
	SHDEFCIRCLE	设定圆检索的条件。	◎	◎	◎	21-89
结果获取	SHCIRCLE	检索圆。	◎	◎	◎	21-90
	VISGETNUM	从存放存储器中获取图像处理结果。	◎	◎	◎	21-93
	VISGETSTR	获取编码识别结果。	◎	◎	◎	21-94
	VISPOSEX	从存放存储器中获取图像处理结果(X坐标)。	◎	◎	◎	21-95
	VISPOSY	从存放存储器中获取图像处理结果(Y坐标)。	◎	◎	◎	21-96
	VISSTATUS	监视各个命令的处理结果。	◎	◎	◎	21-97
	VISREFCAL	获取CAL(视觉-机械手坐标转换)数据。	◎	◎	◎	21-98

多功能教导器的功能菜单



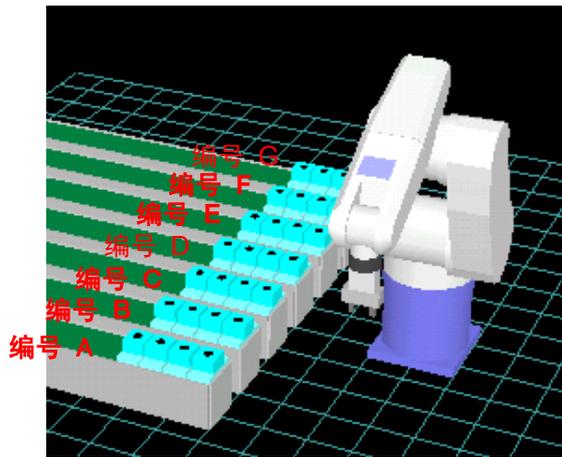


示例程序集

■ 工件取出编号选择

功能 从外部机器获取编号信息。

说明 机械手获取自外部机器作为编号信息输出的I/O信号转换为十进制数。在此，就获取每条传送带所区分的工件编号信息，按照不同编号取出~组装动作所选择的程序进行说明。



程序

【初期输入参数（使用变量）】

未使用	
-----	--

【程序示例】

```
'TITLE"工件取出型号选择"
PROGRAM Sample

TAKEARM
DEFIO type = BYTE,14,&B00011111

WAIT IO[34] = ON
IN I[1] = type

SELECT CASE I[1]
CASE 1
CALL pick_A
CASE 4
CALL pick_B
CASE 5
CALL pick_C
CASE 6
CALL pick_D
CASE 7 TO 15
CALL pick_E
CASE 16
CALL pick_F
CASE IS >= 20
CALL pick_G
CASE ELSE
PRINTMSG "编号数据读入异常", 2, "错误"
STOP
END SELECT

MOVE P,@0 P10

END
```

'定义 I/O 变量，读入数据。
'在独立变量 type1 内仅保存从专用输入端口 [14] 开始的
'5 比特的范围的输出状态
'编号数据输出设置完成信号
'将 type 的数据读入 I[1]（将 2 进制变更为 10 进制）

I[1] 为 1 时
'部件 A 取出~组装运行
I[1] 为 4 时
'部件 B 取出~组装运行
I[1] 为 5 时
'部件 C 取出~安装运行
I[1] 为 6 时
'部件 D 取出~组装运行
I[1] 为 7-15 时
'部件 E 取出~组装运行
I[1] 为 16 时
'部件 F 取出~组装运行
I[1] 为 20 以上时
'部件 G 取出~安装运行
条件不一致时

'机械手定位

示例程序集

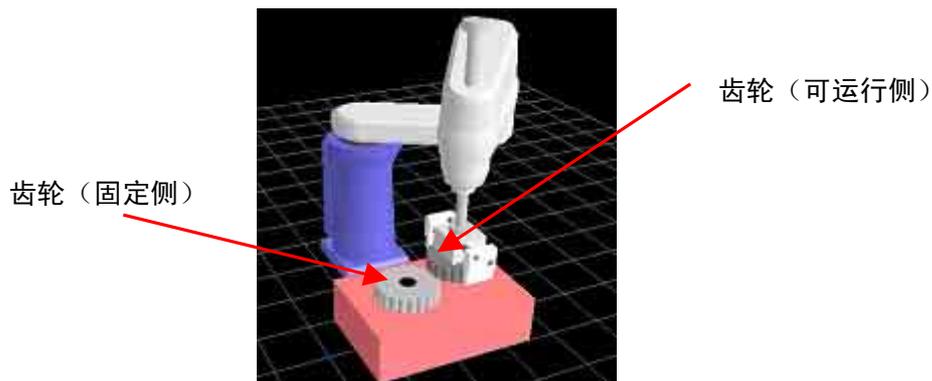
■ 齿轮啮合插入 (HS-E 机械手)

功能 使用电流限制功能，用柔和的力按压插入使齿轮啮合。

说明 使用上下轴的电流限制功能，用柔和的力按压齿轮（作业侧），同时让其进行旋转动作并进行啮合。在动作过程中监视上下轴的选择位置，在到达插入OK判定位置的时刻停止旋转动作。

将判断为齿轮啮合，插入 OK 的任意位置设定为插入 OK 判定位置。

在齿轮以外的产品中，也可以应用在啮合插入上。



程序

【初期输入参数 (使用变量)】

F1	插入 OK 判断位置 (Z 轴坐标值)【需要输入】
----	---------------------------

【程序示例】

```

'TITLE "齿轮啮合插入 (HS-E 机械手)"
PROGRAM sample

TAKEARM
APPROACH P,P1,@0 50,S=10      '上空位置
ST_SetZBalance                 'Z·T 轴重力补偿
ST_SetEralw 3,30              'Z 轴偏差容许值设定
ST_SetCurLmt 3,20            'Z 轴电流限制开始
MOVE L,@0 P1,S=10            '齿轮啮合插入到达位置
judge% = 0                    '齿轮啮合插入标志初始化
IF POSZ(CURPOS) >= F1 THEN    '插入高度判断
  judge% = 1                  '啮合 NG (judge = 1)
  SPEED 5                    '旋转运行速度 5%
  ROTATEH 30,NEXT            '30 度旋转 (NEXT 选择)
  -----ROTATEH 与运行并行处理-----
  flg% = 0                    '标志初始化
  DO
    IF POSZ(CURPOS) <= F1 THEN '插入高度判断
      CALL MotionSkip          '途中运行跳过【程序库使用】
      judge% = 0              '啮合 OK (judge = 0)
      EXIT DO                 'DO-LOOP 语句强制退出
    ENDIF
    CALL MotionComp(fl%)      '运行 (ROTATE 命令) 完成确认【程序库】
  LOOP UNTIL fl%=1           '反复 DO-LOOP 直至运行完成
ENDIF
ST_ResetZBalance              'Z·T 轴重力补偿解除
ST_ResetEralw 3              'Z 轴偏差容许值设定解除
ST_ResetCurLmt 3            'Z 轴电流限制解除
CALL HAND_OPEN               '夹治具松开【用户编辑程序】
DEPART L,50,S=100           'Z 轴上升

IF judge% = 1 THEN           '啮合 NG 时
  PRINTMSG "啮合 NG", 2, "错误" '显示在教导器
ENDIF
MOVE L,@0 P5,S=100          '向规定位置移动

END
  
```

使用共用程序库 (Library) MotionSkip, MotionComp

示例程序集

■ 定位返回 1 (Z 轴上升)

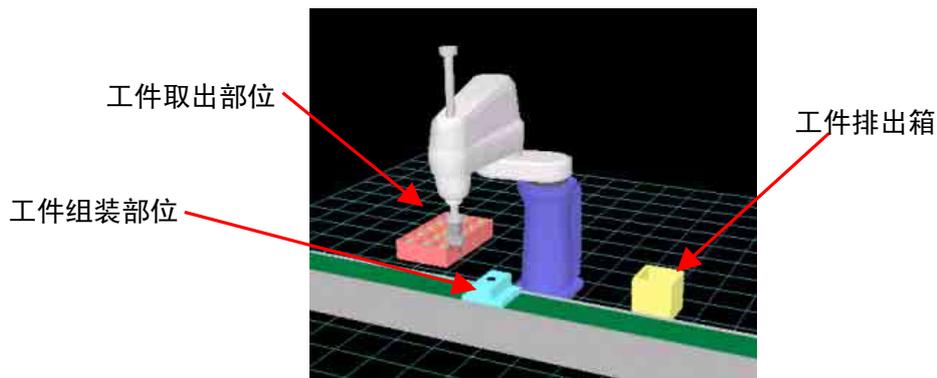
功能 Z轴上升之后臂向规定位置移动。

说明

这是在动作途中机械手停止之后、向规定位置复位的程序。

如下图所示，在臂的可运行范围之内没有干扰物时，通过从最初停止位置开始让 Z 轴（上下轴）上升，可以避开干扰物，使机械手向规定位置移动。

在采样程序中，夹治具柄卡盘信号置于 ON 时，视为夹治具内有剩余工件，要在工件排出动作之后再向规定位置进行复位。



程序

【初期输入参数（使用变量）】

P50	输入当前位置用【自动输入】
F1	Z 轴上升坐标值【需要输入】

【程序示例】

```
'TITLE "定位返回 1 (Z 轴上升)"
PROGRAM Sample

TAKEARM
HOME (200, 300, 350, 45, 1)      '定位坐标设定

P50 = CURPOS                      '当前位置读入
LETZ P50 = F1                    '上下轴移动坐标值的代入
MOVE P, @0 P50,S=30              '上下轴运行
                                  '向从当前位置仅指定上下轴的坐标值移动

IF IO[64] = ON THEN              '确认夹治具内是否残留工件
  CALL DiscardProduct            'IO [64] = ON 时，执行排出运行程序
ENDIF

GOHOME                            '向规定位置移动

END
```

示例程序集

■ 定位返回 2（区域判别）

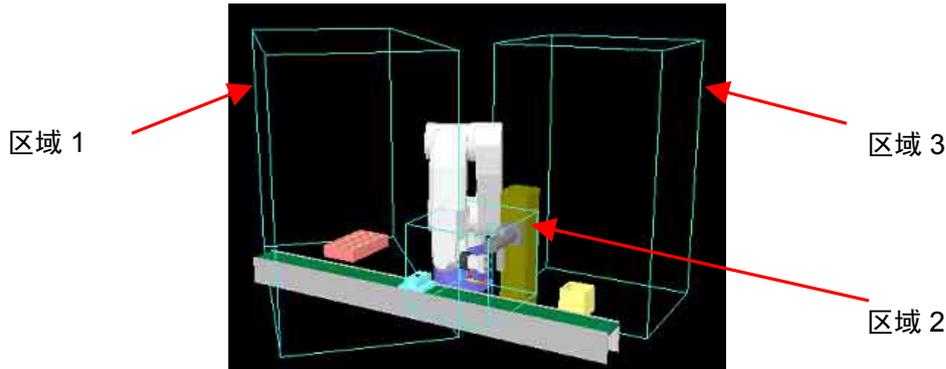
功能

根据臂前端位置情况，避免周边机器干扰，将臂向定位进行移动。

说明

是机械手在任意的位置停止之后向规定位置进行移动的程序。

如下图所示，在臂的可运行范围内有干扰物时，要事先设定区域，通过判断臂前端停止在哪一个设定区域内，在避免周围机器干扰的同时向规定位置移动。



程序

【初期输入参数（使用变量）】

P10	输入当前位置用【自动输入】
IO [221]	区域 1 范围内输出信号【自动输出信号】
IO [222]	区域 2 范围内输出信号【自动输出信号】
IO [223]	区域 3 范围内输出信号【自动输出信号】

【程序示例】

```
!TITLE "定位返回 2（区域判别）"
PROGRAM Sample
```

```
TAKEARM
HOME P1
P10 = CURPOS
```

'将规定位置设定在 P1
'将当前位置代入 P10

```
IF (IO[221]=OFF) AND (IO[222]=OFF) AND (IO[223]=OFF) THEN
  PRINTMSG "当前位置在区域范围外", 2, "错误"
  STOP
ELSEIF (IO[221]=ON) OR (IO[223]=ON) THEN
  LETZ P10 = 450
ELSEIF IO[222] = ON THEN
  LETY P10 = 0
END IF
MOVE P,@0 P10,S=50
```

'臂前端未在指定区域内时
'显示错误讯息
'程序停止
'臂前端位于区域编号 [1,3] 内时
'在 Z 轴坐标设定 450mm
'臂前端位于区域编号 [2] 内时
'在 Y 轴坐标设定 0mm
'从当前位置的回避运行

```
-----夹夹具关闭（工件夹持）时，排出运行-----
```

```
IF IO[64] = ON THEN
  MOVE P,@0 P22,S=50
  APPROACH P,P21,@0 100,S=50
  MOVE P,@0 P21,S=50
  CALL PRPDUCT_RELEASE
  DEPART P,@0 100,S=50
  MOVE P,@0 P22,S=50
  GOHOME
```

'夹夹具关闭时
'向干涉部回避位置移动
'向排出箱上空 100mm 移动
'向排出箱移动
'松开运行【用户编辑程序】
'向排出箱上空 100mm 移动
'向干涉部回避位置移动
'向规定位置移动

```
-----夹夹具关闭（工件无）时，返回规定位置-----
```

```
ELSEIF IO[64] = OFF THEN
  IF IO[222] = ON THEN
    MOVE P,@0 P22,S=50
  ENDIF
  GOHOME
ENDIF
```

'夹夹具打开时
'臂前端位于区域编号 "2" 内时
'向干涉部回避位置移动
'向规定位置移动

```
END
```

示例程序集

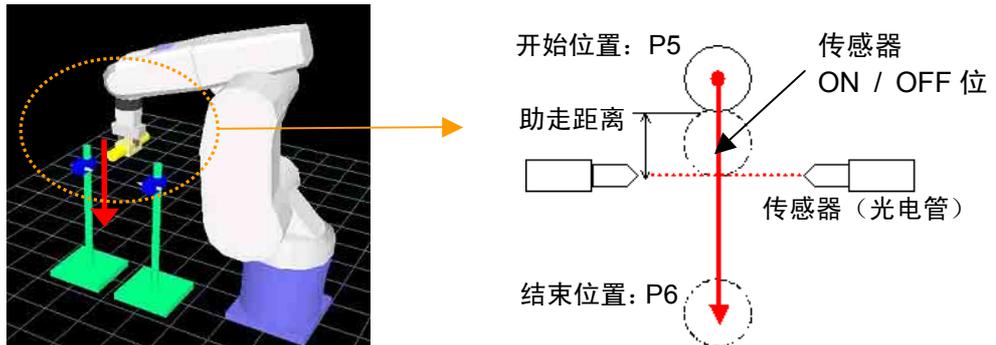
■ 由传感器进行的工件尺寸测量

功能

使用传感器测量工件的尺寸。

说明

安装用于工件尺寸测量的传感器。工件通过传感器，读取ON / OFF切换时的机械手坐标位置。从传感器的ON → OFF、OFF → ON时的坐标差测量工件的尺寸。另外，开始位置是为了稳定测量值而预测的在传感器位置机械手到达等速时的助走距离的位置。
【注意】：因测量精度依存于传感器的精度和机械手的速度，所以不能进行需要高精度的尺寸测量。



程序

【初期输入参数（使用变量）】

P5	机械手动作开始位置【需要输入】
P6	机械手动作结束位置【需要输入】
P10	读取传感器 ON → OFF 时的机械手当前值用【自动输入】
P11	读取传感器 OFF → ON 时的机械手当前值用【自动输入】
F1	2 点间 (P10 - P11) 距离计算结果【自动输入】

【程序示例】

```

!TITLE "由传感器进行的工件尺寸测量"
PROGRAM Sample011

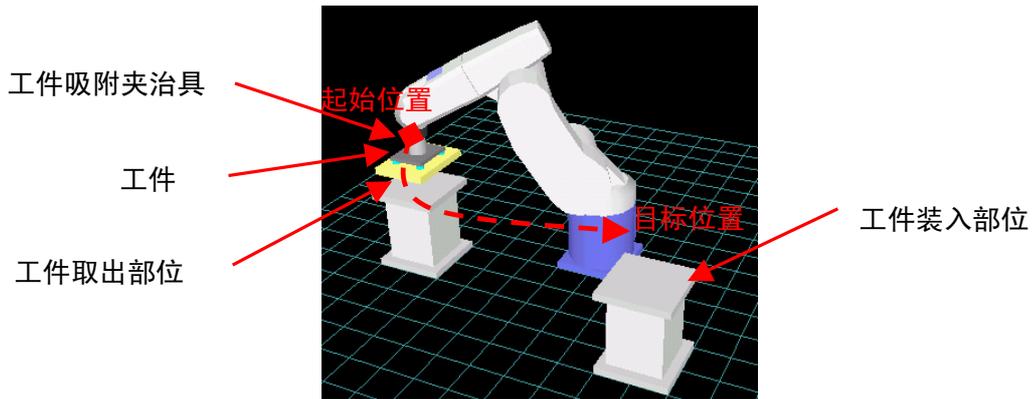
TAKEARM
MOVE P,@E P5,S=100          '向测量开始位置移动
SPEED MPS(20)              '将运行速度设定为 20mm / s
ACCEL 100,100              '将加减速速度设定为 100%
Inspect%=0                  '检查标志初始化
IF IO[34]=ON THEN          '光电传感器 ON 时
  PRINTMSG "传感器在测量开始点 ON", 2, "错误"
  STOP                      '程序结束
ENDIF
MOVE L,@0 P6,NEXT          '测量结束位置 (NEXT 选择)
'-----向 P6 的移动运行与并列处理-----
DO
  IF Inspect%= 0 THEN
    IF IO[34]= ON THEN
      P10=CURPOS
      Inspect%= 1
      '光电传感器 ON 时
      '在 P [10] 内读入当前值
      '尺寸测量开始 (标志 = 1)
    ENDIF
    ELSEIF IO[34]= OFF THEN
      P11=CURPOS
      Inspect%= 2
      '光电传感器 OFF 时
      '在 P [11] 内读入当前值
      '尺寸测量结束 (标志 = 2)
      EXIT DO
      'DO~LOOP 语句强制退出
    ENDIF
  LOOP
'-----
IF (Inspect%= 0) OR (Inspect%= 1) THEN
  PRINTMSG "请确认测量 NG 传感器的 ON / OFF", 2"错误"
  STOP                      '程序结束
ENDIF
'-----代入 2 点间的测量结果-----
F1 = DIST(P10, P11)        '代入 2 点间距离
'-----
DEPART L,@0 100,S=50
END
  
```

示例程序集

■ 臂动作时的工件落下监视

功能 利用吸附传感器的ON / OFF状态，对臂动作过程中的工件掉落进行监视。

说明 在臂动作过程中，传感器OFF（发生工件掉落、滑脱等）时，会中途停止臂的动作，向外部机器输出异常信号。防止吸附不良的工件流向后道工序。



程序

【初期输入参数（使用变量）】

P0	机械手动作开始位置【需要输入】
P1	机械手动作目标位置【需要输入】

【程序示例】

```

!TITLE "臂运行时的工件落下监视"
PROGRAM Sample

TAKEARM
flg% = 0 '吸附传感器状态标志初始化
comp% = 0 '运行命令完成标志初始化
MOVE P,P0
SPEED 50
MOVE P,P1,NEXT '向运行开始位置 (P0) 移动
'-----向 P1 的移动运行与并列处理 -----
DO
  IF IO[35] = OFF THEN '吸附传感器 OFF 时
    flg% = 1 '工件落下记忆 (flg = 1)
    CALL MotionSkip '中断臂运行命令【程序库使用】
    EXIT DO 'DO~LOOP 语句强制退出
  ENDF
  CALL MotionComp(comp%) '运行完成确认【程序库使用】
LOOP UNTIL comp% = 1 '反复 DO~LOOP 直至运行完成
'-----在运行途中传感器 OFF（工件落下）时的处理 -----
IF flg% = 1 THEN
  PRINT MSG "吸附工件落下", 2, "Error" '确认运行中的吸附传感器状态
  SET IO[104] '教导器显示
  STOP '输出工件落下异常信号 (IO [104])
  ENDF '程序结束
END
  
```

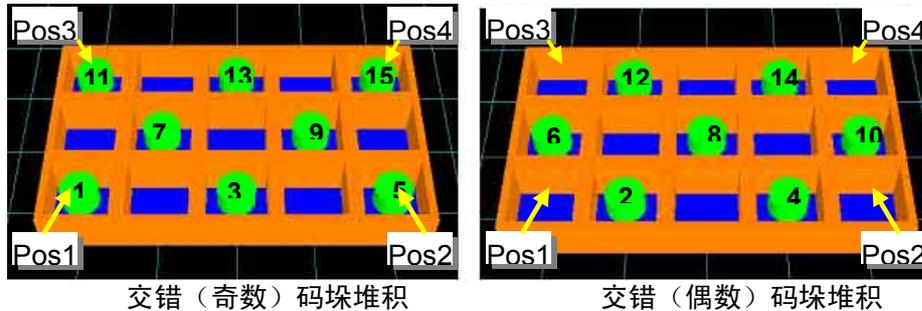
使用共用程序库 (Library) MotionSkip, MotionComp

示例程序集

■ 交错（奇数、偶数）码垛堆积

功能 跳过1个进行码垛堆积取出动作。

说明 让码垛堆积进行跳过1个（将其交错交错动作）的动作。
就奇数、偶数位置取出所对应的程序进行说明。



程序

【初期输入参数（使用变量）】

P11	托板四角 Pos1 位置【需要输入】
P12	托板四角 Pos2 位置【需要输入】
P13	托板四角 Pos3 位置【需要输入】
P14	托板四角 Pos4 位置【需要输入】
P15	码垛堆积动作位置【自动输入】
I1	在码垛堆积取出计数器 第一次执行时，奇数交错输入 1、偶数交错输入 2【需要输入】

【程序示例】

```

!TITLE "交错码垛堆积（奇数）"
PROGRAM Sample003

TAKEARM
CALL xdGetPalt(3,5,0,P11,P12,P13,P14,P15,I1,1) '简易码垛堆积程序库
APPROACH P,P15,100,S=100 '向码垛堆积点上空移动
MOVE L,@E P15,S=70 '向码垛堆积点移动
CALL CloseGripper '夹治具关闭【用户编辑程序】
I1 = I1 + 2 '码垛堆积计数器 UP（交错时 + 加算 + 2）
IF I1 > (3 * 5) THEN '工件取出数完成时
    I1 = 1 '计数器清除（奇数交错时初始值 = 1）
ENDIF
DEPART L,@P 100,S=80 '上升

END

!TITLE "交错码垛堆积（偶数）"
PROGRAM Sample003

TAKEARM
CALL xdGetPalt(3,5,0,P11,P12,P13,P14,P15,I1,1) '简易码垛堆积程序库
APPROACH P,P15,100,S=100 '向码垛堆积点上空移动
MOVE L,@E P15,S=70 '向码垛堆积点移动
CALL CloseGripper '夹治具关闭【用户编辑程序】
I1 = I1 + 2 '码垛堆积计数器 UP（交错时 + 加算 2）
IF I1 > (3 * 5) THEN '工件取出数完成时
    I1 = 2 '计数器清除（偶数交错时初始值 = 2）
ENDIF
DEPART L,@P 100,S=80 '上升

END
    
```

使用共用程序库 (Library) MotionSkip, MotionComp

用语

A—E

安装台架 (installation frame)

安装机械手的台子

保存 (save)

指用机械手控制器将程序、臂数据等保存在软盘上。

臂配置宏定义文件 (arm configuration macro definition file)

臂设定数据的宏定义信息的集合文件。

臂信号 (semaphore) (arm semaphore)

机械手控制权。只有拥有该权利的任务才能动作机械手

臂文件 (arm file)

记录机械手固有信息的文件。

臂形态 (arm figure)

6轴机械手的第1轴~第3轴的值所决定的形态。基准有正手、反手2种

变量表 (variable table)

指机械手控制器保持的变量编号和值成对的数据群

编码器值确认动作 (encoder value check motion)

对于被教导的动作目标位置,编码器值进入指定脉冲以内时,判断到达目标位置的动作。

边缘 (edge)

辉度的变化点 (视觉用语)

标注 (labeling)

指对二值化的白、黑区域加注编号 (视觉用语)

标准模式 (normal mode)

I/O的标准配置模式

单步运行 (step check)

要在教导检查模式下单步运行执行程序。

步骤启动 (single-step start)

单步运行执行程序的启动方法。执行单步运行之后停止

不翻转 (NONFLIP)

6轴机械手的手掌形态之一 (⇔ 翻转)

CAL (CAL)

接通机械手控制器的电源后,为了检查机械手的当前位置,使全部的轴微动

CALSET (CALSET)

校正控制器识别的位置讯息与机械手本体实际位置的关系

CP控制 (CP control)

插补控制到达动作目标位置时的路径,使其呈直线或圆弧 (⇔ PTP控制)

菜单分类目录 (menu tree)

指对功能键的功能菜单一览进行分类目录状显示。在操作向导上进行介绍

操作LOG (operation log)

使用多功能教导器等的操作记录。

Operating panel (operating panel)

与控制器相连接的固定操作盘。没有教导功能

常规矢量 (normal vector)

机械界面坐标系中的X轴正方向的矢量

程序传送 (transfer)

指在机械手控制器与WINCAPSIII (电脑)之间发送接收机械手程序。

程序复位 (program reset)

从前面强制执行程序的输入信号

共用程序库 (library)

以再利用为目的的程序的集合。可以使用WINCAPSIII的程序库进行登陆、利用。

程序启动 (program start)

启动程序的输入信号。步骤停止的情况下,从下一个步骤重新开始执行、瞬时停止的情况下,从同一个步骤的后面重新开始执行

错误编码 (error code)

表示机械手上发生之异常的原因、状况的4位16进制数。各错误编码的含义需参照错误编码表。

错误LOG (error log)

错误内容和发生错误的时刻记录

D型变量 (D type variable)

值为双精度 (有效位精度为15位) 实数的变量

地址设定 (IP address)

要设定控制器的IP地址。用以太网通信时需要

单 (SINGLE)

6轴机械手的第6轴形态之一 (⇔ 双)

单4 (SINGLE4)

6轴机械手的第4轴形态之一 (⇔ 双4)

单轴CALSET (CALSET of a single axis)

指只对指定的轴进行CALSET

当前位置 (current position)

工具坐标系原点的当前位置

第一臂 (first arm)

机械手臂中,靠近底座的臂

第二臂 (second arm)

机械手臂中,远离底座的臂

底座 (base)

安装机械手第1轴的部分

底座面 (base mounting surface)

底座和台架的接合面

底座坐标系 (base coordinates)
以机械手的底座为原点的三维正交坐标

电机电源断开 (powering OFF the motor)
指切断机械手的电机电源

电机电源接通 (powering ON the motor)
指接通机械手电机电源

电源断开 (powering OFF the robot controller)
将机械手控制器的电源切断。

电源接通 (powering ON the robot controller)
接通机械手控制器的电源。

定向矢量 (orient vector)
机械界面坐标系中的Y轴正方向的矢量

定义类型 (type declaration)
指在程序内定义变量的类型

局部 (local) 变量 (local variable)
只在一个任务内可以利用的变量

多项任务 (multitasking)
表面上看是同时执行多个程序的状态。机械手控制器基准的CPU通过在短时间内依次替换程序的执行而实现

二值化 (binarization)
指以界值(二值化值)为界限,将各个像素的辉度变更为白(0)、黑(1)

二值化值 (binarization level)
二值化的界值(视觉用语)

F—J

FIG (FIG)
表示形态的数字

F型变量 (F type variable)
值为单精度(有效位精度为7位)实数的变量

反手 (LEFTY)
6轴机械手的手腕形态之一(⇔ 正手)

翻转 (FLIP)
6轴机械手的手掌形态之一(⇔ 不翻转)

干涉区域 (interference area)
为了监视工具与设备是否相干涉由用户设置的区域。基准如果工具坐标系的原点进入区域内,则从指定的I/O基准端口有输出

工件坐标系 (work coordinates)
以机械手作业目标的工件为原点的三维正交坐标

工具 (tool)
机械手直接作用于工件的部分。与末端执行器(JIS)基准为同义词

工具定义 (Defining tool coordinates)
指定义工具坐标系。以机械界面坐标系为基准,定义基准原点偏移量和围绕各个轴的旋转角度。可以从TOOL1基准到TOOL63定义

工具坐标系 (tool coordinates)
是在工具上设定原点的坐标系对机械界面坐标系的基准原点在任意点进行偏移,围绕各个轴旋转

功能键 (function keys)
在教导器画面下方排列的按钮。功能名称显示在画面的下部,如果按压则发挥功能

各轴模式 (joint mode)
每个轴进行手动动作的模式

宏 (macro)
指用变量编号、端口编号等12字符串定义名称。基准执行程序时名称被置换为编号

宏定义文件 (macro definition file)
定义宏的文件

互换模式 (compatible mode)
为了保持I/O的配置与原来序列的机械手的互换性基准而设定的模式。用软件进行切换

辉度 (brightness)
表示各个像素所具有的亮度数值(0~255)(视觉基准用语)

辉度积分值 (brightness integral value)
视窗内的所有像素辉度值的合计值(视觉用语)

I/O (I/O)
输入输出信号

I/O指令 (I/O command)
通过外部机器使用I/O端口发出的处理指令。基准机械手控制器按照该指令进行处理

I型变量 (I type variable)
数值为整数的变量

J型变量 (J type variable)
各轴的值所表示的变量

机器锁定 (machine lock)
指不实际动作机械手,只是在机械手控制器上进行仿真动作的状态

机械界面坐标系 (mechanical interface coordinates)
以凸缘中心为原点的三维正交坐标

机械末端 (mechanical end)
由机械限位器设定的机会动作界限(⇔ 软件限位)

机械手警告 (robot warning)
I/O指令和伺服处理上发生轻微异常的通知输出信号

机械手停止 (robot stop)
立即停止程序,切断电机电源的停止方法

机械手异常 (robot error)
通知伺服异常、程序异常等的机械手上发生异常的基准输出信号

机械限位器 (mechanical stopper)
是对机械手的轴的动作进行物理限制的机构

夹治具 (end-effector)
抓取工件的部分。与工具相同

加载 (load)
指从软盘/U盘上将程序、臂数据读入到机械手控制器。

间距角 (pitch angle)
围绕Y轴的旋转角

监视器 (monitor)
显示机械手的当前状态

检索 (search)
指寻找与标准的图像数据(检索模型)一致的位置基准(视觉用语)

教导 (teaching)
指使用多功能教导器将操作所需要的信息输入到基准机械手。

教导检查 (teach check)

指检查程序所进行的动作。

教导器缺省运转 (pendantless state)

指多功能教导器在未连接状态下,从外部机器进行运转。

近似矢量 (approach vector)

在机械界面坐标系上的Z轴正方向矢量

结束动作 (end motion)

伺服的指令位置与被教导的动作目标位置一致时,判断到达目标位置的動作

步骤停止 (step stop)

执行程序单步运行后停止的停止方法

蜷起 (ABOVE)

6轴机械手的肘形态之一(⇔ 伸直)

旧语言 (conventional language)

DENSO机械手原来使用的机械手语言

绝对动作 (absolute motion)

向被教导的动作目标位置移动的动作(⇔ 相对动作)

K-O**可搬运质量 (load capacity)**

机械手可以提起的工具和工件的合计质量

可运行范围 (motion space)

机械手可动作的范围

控制LOG (control log)

指令值、编码器值、电流值、负载率的记录。基准在各个动作轴上记录

快拍 (snapshot)

记录机械手当前状态的功能

LOG (log)

机械手的操作和动作等的记录。
有错误LOG、操作LOG、控制LOG、通信LOG4种

码垛堆积 (palletizing)

指在分隔的板台上顺次进行零部件的装入和取出。

每个操作 (panel operation)

指从多功能教导器的画面上操作内部I / O的基准ON / OFF。

面积 (area)

将图像数据二值化时,视窗内的白、黑的像素数(视觉基准用语)

模式法 (mode method)

直方柱状体双峰型分布时,在谷处设定二值化值的方法

模式切换开关 (mode switch)

教导器上的开关。可切换运行模式。

NLIM (NLIM)

软件限位负方向端点的值基准(⇔ PLIM)

内部加速度 (internal acceleration)

指在程序中设定的加速度

内部减速度 (internal deceleration)

指在程序中设定的减速度

内部模式 (internal mode)

使用多功能教导器,可以进行机械手动作和教导等的模式

内部速度 (internal speed)

指在程序中设定的速度

内部自动运行 (internal automatic run)

指从Operating panel和多功能教导器上执行程序

Operator (operator)

WINCAPS III的用户等级之一重要的参数不能变更。基准不需要输入密码

P-T**PAC (PAC)**

DENSO机械手所使用的新机械手语言。
是SLIM (JIS的产业用机械手语言)的高级互换语言

PAC程序管理器 (PAC program manager)

支持PAC程序开发的软件。具有编辑程序、指令编码程序、程序库 (Program Bank) 等功能

PLIM (PLIM)

软件限位正方向端点的值基准(⇔ NLIM)

Programmer (programmer)

WINCAPS III的用户等级之一。一般的操作都可以进行。要置于该模式,需要输入密码。

PTP控制 (PTP control)

不进行插补,只移动至动作目标位置的控制。到达的路径不只限于直线。(⇔ CP控制)

P瓷砖法 (P tile method)

为使目标物的面积和黑(或者白)部分的面积相同而设定二值化值的方法(视觉用语)

P型变量 (P type variable)

位置、姿势以及形态所表示的变量

判别分析法 (discrimination analysis method)

从直方柱状体使用统计方法设定二值化值的方法(视觉基准用语)

区域设定 (defining interface area)

要设定干涉区域。有两种方法:使用多功能教导器和WINCAPS III进行设定的方法和用程序指令进行设定的方法。

全局变量 (global variable)

任何任务均可以利用的变量

RANG (RANG)

决定机械手的基准位置和机械末端关系的角度

RX成分 (component RX)

围绕X坐标轴的旋转角度

RY成分 (component RY)

围绕Y坐标轴的旋转角度

RZ成分 (component RZ)

围绕Z坐标轴的旋转角度

任务 (task)

指同时对多个程序进行管理时,各程序所形成的动作基准过程。

日常点检 (daily inspection)

每天作业之前的点检

软件限位 (software motion limit)

软件上所决定的机械手的动作范围界限(⇔ 机械末端)

STOP键 (STOP key)

教导器的按钮之一。如果按压,则可以瞬时停止所有基准程序

伸直 (BELOW)

6轴机械手的肘形态之一 (⇔ 蜷起)

视窗 (window)

处理图像的范围 (视觉用语)

视觉功能 (visual function)

处理从照相机输入的图像, 将需要的数据赋予机械手基准控制功能的功能

视觉装置 (visual device)

处理从照相机输入的图像, 将需要的数据赋予机械手的基准装置

手动操作 (manual robot operation)

指用户使用多功能教导器操作键操作机械手。

手腕形态 (wrist figure)

6轴机械手的第4轴~第5轴的值所决定的形态。有翻转和不翻转2种

数据区域 (data area)

指指定I/O指令所需要数据的I/O端口群。

双 (DOUBLE)

6轴机械手的第6轴形态之一 (⇔ 单)

双4 (DOUBLE4)

6轴机械手的第4轴形态之一 (⇔ 单4)

双重安全开关 (deadman switch)

为了安全, 仅在与移动方向键同时按压时, 才使机械手动作的开关。如果释放移动方向键或者双重安全开关的任何一个, 机械手就会立即停止。也称为允许开关。

伺服ON (servo ON)

将接通电机电源情况通知外部的信号

TOOL0 (TOOL0)

是工具定义的特殊型, 原点偏移为0, 即意味着机械基准界面坐标系

TOOL模式 (TOOL mode)

工具坐标系上手动操作模式

T型变量 (T type variable)

位置矢量、定位矢量、输入矢量以及形态所表示的变量

特异点 (singular point)

两个形态界限上的位置

通过 (Pass) 动作 (pass motion)

通过被教导的动作目标位置附近的动作

通信LOG (communication log)

电脑和机械手的通信状态的记录

通信权设定 (set communication permission)

指设定机械手控制器的各个通信端口的使用许可

通信设定 (set communication)

指设定机械手控制器的各个通信端口的使用条件 (通信速度等)

通用输出信号 (user I/O signals)

可以用用户程序控制的输入输出信号

凸缘部 (plate mechanical interface)

指机械手的机械臂前端安装工具的部分。

凸缘面 (mechanical interface)

凸缘和工具的接合面。指机械界面 (JIS)

U-Z

外部加速度 (external acceleration)

指在多功能教导器上设定的加速度。输入最高加速度基准的比例 (%)

外部减速度 (external deceleration)

指在多功能教导器上设定的减速度。输入最高加速度基准的比例 (%)

外部模式 (external mode)

指从外部机器能够运转机械手的模式

外部速度 (external speed)

指在多功能教导器上设定的速度。输入最高速度的基准比例 (%)

外部自动运行 (external automatic run)

指从外部机器执行程序

漫步旋钮 (jog dial)

教导器上的旋钮。在输入画面上用于光标的移动和基准切换选择臂。

位置数据 (position data)

机械手的凸缘中心 (工具定义有效时为工具前端) 的基准位置和表示机械手姿势的底座坐标系的数据

X-Y模式 (X-Y mode)

在底座坐标系上进行手动动作的模式

系统变量 (system variable)

在程序内检查系统状况的变量

系统项目 (project)

指程序和与此相关的数据群。

相对动作 (relative motion)

从当前位置开始仅以被教导的移动量移动的动作

像素 (pixel)

指构成画面的点 (视觉用语)

信号 (semaphore)

指任务的执行权。在任务之间进行同步或者非同步运行的任务之间, 进行排他控制时使用

形态 (figure)

机械手的各轴 (关节) 所能选择的状态。对于相同位置、姿势可以选择多个形态

形态成分 (figure component)

决定形态的要素。在6轴机械手中, 有臂、肘、手腕、第6轴、第4轴5个要素

悬挂规格 (overhead version)

将底座设为上、将机械臂设为下, 安装为从屋顶悬挂规格的机械手。在操作台上不需要机械手的安装空间, 能够扩大作业区域

选通信号 (strobe signal)

指示开始进行I/O指令处理的输入信号

循环启动 (single-cycle start)

执行程序单循环的启动方法。执行单循环 (至程序的基准最终步骤) 后结束

循环停止 (cycle stop)

执行程序单循环后停止的停止方法

以太网网板 (ethernet board)

控制器的增设卡之一。按照TCP/IP通信协议, 与WINCAPSIII进行通信时使用。

用户等级 (user level)

为了保证数据管理的安全所设置的用户等级。在每个等级上对可以访问的信息和操作进行限制

用户坐标系 (user coordinates)

指用户可以定义的坐标系

优先顺序 (priority)
优先执行任务的顺序。按照优先顺序高的序次执行程序

余角 (yaw angle)
围绕X轴的旋转角度

远程操作 (remote operation)
指在WINCAPSIII上使显示的机械手臂动作。

动作模式 (operating modes)
用手动进行动作的模式。有各个轴模式、X-Y模式、TOOL模式3种

瞬时停止 (suspend, halt)
立即停止程序的停止方法电机电源没有切断

制动器OFF (releasing brakes)
指解除各个轴的制动器

制动器ON (locking brakes)
指对各个轴实施制动器

直方柱状体 (histogram)
在视窗内辉度值的出现频率 (视觉用语)

指令 (command)
指程序上记述的命令。控制器按照程序上记述的顺序基准读取、解释并执行指令

备注 (comment) (提示)
指程序上记述的命令。为了便于理解而追加的程序。基准不能在控制器上执行

指令处理完成 (command processing complete)
将I/O指令处理结束通知外部的输出信号

指令空间 (command area)
指指定I/O指令种类的I/O端口群。

执行指令输入输出信号 (command execution 基准I/O signals)
为了执行I/O指令和将执行状况通知外部, 在系统上基准固定的输入输出信号

肘形态 (elbow figure)
6轴机械手的第2轴~第3轴的值所决定的形态。有蜷起和伸直2种

状态区域 (status area)
通知I/O指令处理结果的输入输出信号群。对应I/O指令的状态被设定

正手(RIGHTY)
6轴机械手的臂形态之一 (⇔ 左撇子)

转动角 (roll angle)
围绕Z轴的旋转角度

最佳可搬运质量设定功能 (optimal load capacity setting function)
是与机械手的负荷条件或姿势相应, 设定最佳速度、基准加速度的功能。

执行程序 (execution program)
指转换为可以让机械手理解的数据形式的程序

中断跳跃(interrupt skip)
如果在机械手执行指令过程中短路 (ON), 则瞬时停止该步骤动作, 开始执行下一个步骤的输入信号

重心 (center of gravity)
平面上目标物的重量的平衡点 (视觉用语)

主轴 (principal axis)
旋转平面目标物的情况下, 惯性力矩最小的轴 (视觉基准用语)

主轴角 (principal axis angle)
水平轴和主轴间的角度 (视觉用语)

专用输入输出信号 (system I/O signals)
为了将运转控制和运转状况通知外部, 在系统上固定的输入输出信号

子程序 (subroutine)
在记述特定动作的程序中, 从主程序的其他部分调出。

自动运行 (automatic robot run)
指通过执行程序, 运行机械手

自动运行允许 (enable auto)
如果置于短路 (ON) 状态, 则设为可以切换为自动模式的信号。在开放 (OFF) 状态下, 可以设为手动模式、教导检查模式

姿势 (pose, posture)
6轴机械手的情况下, 由滚子、间距、余角决定的斜度指4轴机械手的情况下, 由围绕Z轴的旋转角度所决定的工具的方向

SYMBOLS

μVision (μVision)
DENSO制造的视觉装置

DENSO 机械手

垂直多关节型 V* 系列
水平多关节型 H* 系列
直角坐标型 XYC 系列
内置型 XR 系列

调试起动手册

初 版 2008 年 1 月
第 2 版 2011 年 9 月

DENSO WAVE INCORPORATED

9N**C

-
- 未经允许禁止复制或转载本使用说明书的部分或全部内容。
 - 本说明书的内容若有变动，恕不另行通知。
 - 关于本说明书的内容，在编辑时虽然力求万无一失，但若发现有不当之处、错误以及遗漏等情况，请与本公司联系。
 - 对于使用本说明书所造成的后果及影响，本公司概不负责，敬请谅解。

