BEIJING-FANUC

FANUC 人机界面教材

FANUC PICTURE 篇

北京发那科培训中心 BFTC-303C/02

·本教材中的任何内容不得以任何方式复制。

·系统或实验台的外观和规格可能会随着技术改进而有所变更。

我们试图在本教材中叙述尽可能多的情况,但由于篇幅原因无法一一列举。因此,对于那些在本教材中没有列举的情况,请参考 FANUC 相关产品的说明书。

我们在教材编写过程中,不可避免的会出现一些错误。如果教材中有和实际产品或者 FANUC 产品说明书不一致的内容,请以 FANUC 产品说明书或者相关技术文档为准。

本教材中提到的我公司产品以外的程序名称和设备名称,它们包含在各制造商的注册商标中。但 是,正文中在某些情况下并没有标注 ® 和 TM 标记。

1. FANUC 人机界面软件简介 - 1 -会图栏......-6-L L 4. 使用 FP 制作画面流程图......-9-L 打开工程.....-13-L L L L L Т 鼠标操作和键盘操作.....--22-L FANUC PICTURE 符号说明.....-- 23 -L 生成 CNC 可识别 MEM 文件的步骤.....-- 25 -L L L L L L L Т L

录

	Ι	图片显示控件	- 42 -
8.	FP	函数	- 43 -
	Ι	函数指定方法	- 43 -
	Ι	CNC Window 库函数功能说明	- 44 -
	Ι	PMC Window 库函数功能说明	- 50 -
	Ι	其他类型函数	- 54 -
9.	在非	ŧ触摸屏显示器上的应用	- 55 -
	Ι	屏幕中焦点移动功能	- 56 -
	Ι	连续输入功能	- 58 -
	Ι	快捷键功能	- 59 -
	Ι	各控件焦点控制相关属性设定	- 61 -
1(). FA	NUC PICTURE 制作例子	- 63 -
	Ι	创建一个新工程	- 63 -
	Ι	创建主画面	- 64 -
	Ι	按钮子画面的制作	- 65 -
	Ι	弹出画面的制作	- 66 -
	Ι	按钮指示灯演示画面	- 68 -
	Ι	输入控件演示画面	- 72 -
	Ι	函数的使用和报警信息的显示	- 75 -

1. FANUC 人机界面软件简介

| FANUC 控制器人机界面开发应用软件主要有以下三种:

MACRO COMPILER / EXECUTOR C LANGUAGE EXECUTOR FANUC PICTURE

| 各种开发工具的适用范围、功能和对比如下:

	宏执行器	C 语言执行器	FANUC PICTURE
适用系统	0i Mate -D 0i -D 30i / 31i / 32i -A	0i -D 30i / 31i / 32i -A	0i -D * 30i / 31i / 32i -A
开发环境	类似用户宏程序 纯文本编辑器	C 语言	类似 Visual Basic
编译软件	FANUC 宏编译器	WindRiver Compiler v4.4b 或者 v5.6	FANUC PICTURE
界面显示 功能	基本绘图 基本文字接口	较多的绘图、文字显示 基本的图形显示	较强的图形显示 有具贴图功能
集成 NC 加工程序	可 **	无	无
CNC 控制功能	一般	强	一般
计算功能	一般	强	弱 ***
开发周期	较长	ĸ	短
稳定性	高	一般	高

- * 不支持 0i Mate 系列,支持触摸屏或非触摸屏,可用于 8.4 寸标配显示器。
- ** 加工程序可以作为可执行宏程序嵌入 CNC。
- *** 2.2 版之后加入脚本功能,可实现简单运算。还可搭配 PMC 或 C 语言执行器实现复杂计算功能。

本讲义重点介绍 FANUC PICTURE。

2. FANUC PICTURE 的运行环境

| 电脑软硬件要求 (FANUC 推荐配置):

CPU :	Pentium 4 - 2GHz 以上
内存:	256 MB 以上(Windows XP)
	1 GB 以上(Windows Vista)
硬盘空间:	50MB 以上
显示器:	分辨率 1024×768 以上,真彩色 65000 色以上
操作系统:	Windows XP 或者 Windows Vista
应用软件:	Internet Explorer 6.0 sp1 以上
外围设备:	PCMCIA 兼容读卡器
至少 16MB 以	人上的存储卡(CF 卡带适配器或 ATA 卡)
CD-ROM 光望	伛(仅在安装时需要)
多语言对应的	习字体文件以及输入法

- I CNC 系统要求:
 - | 30i / 31i / 32i -A 系列

对于带有触摸屏的系统,应当选择"FANUC PICTURE function"(S879)。对于 不带触摸屏的系统,应当选择"FANUC PICTURE function for non-touch panel display"(S944)。这两种功能均包含 6 MB 的用户自定义软件容量,不用再选择 用户软件容量。

如果同时使用 FANUC PICTURE 和 C 语言执行器功能,直接选择'Macro executor + C-language executor "(J734)和用户自定义软件容量"Custom software size" (J738#??),其中??表示具体软件容量大小,建议使用 4MB~6MB)。

| 0i - D 系列

当在带触摸屏的 CNC 上使用 FANUC PICTURE 时,应选择"Touch Panel C" (S881)功能,该功能包含用户 6 MB 自定义软件容量。对于非触摸屏的 0i-D 系 统,应选择"FANUC PICTURE Executor"(R644)功能,以及用户自定义软件 容量 "Custom software size "(J738#??), 其中??表示具体软件容量大小, 建议 使用 4MB~6MB)。

CNC 相关参数:

在 NC 上运行 FANUC PICTURE 须设定以下参数:

8661 = 59

S-RAM 变量存储区大小(59K)

8662 = 4

S-RAM 文件存储区大小(4K)

8781 = ??

D-RAM 大小,设定分配给 C 执行器的主 CPU 用户自定义软件容量 (每 64K 对应设定单位 1)

例如:

当分配所有的主 CPU 用户软件容量

- 6MB 96 (对于 Power Mate i, 总是设定 96)
- 5MB 80
- 4MB 64

如果同时使用宏执行器,则此参数设定值须减去被宏执行器占用的数值。比如 说,当宏执行器使用了 512K 的容量,则设定参数应减去 8,在上面例子中, 应分别设定 88、72 和 56。

8650#1

当 CNC 发生报警时, C 语言执行器制作画面不切换到 CNC 报警画面显示。 这个参数设定后,必须切断 CNC 电源。

在 FANUC PICTURE 中可通过 FP 参数 No.4 ," Screen switch rule when NC alarm occurs ", 详见《FANUC PICTURE 操作说明书》(B-66284)。

3. FANUC PICTURE 软件使用说明

当打开 FANUC PICTURE 软件后,显示界面如下。

菜单栏 工具栏 Property 绘图栏 ////////////////////////////////////	Value

| 菜单栏和工具栏

用于进行 FANUC PICTURE 基本操作。包括文件创建、打开;画面调整、对齐;工程 编译、修改等。工具栏则是菜单栏里面常用内容的快捷按钮,使用更加方便。

具体内容详见《FANUC PICTURE 操作说明书》(B-66284) 2.2.1 节。

| 窗体工具栏

窗体工具栏主要用于画面中的 Lamp、Button 状态切换、多语言显示以及比例缩放等功能,详见说明书 2.2.9.8。

Button 切换功能效果图如下:



如果在创建工程时设定了多语言,可通过下拉菜单进行切换。所有使用了多语言功能的 控件(按钮、标签等)均可实时完成切换。另外,FANUC PICTURE 还可以在画面窗口内 调整显示比例。语言切换和显示比例示意图如下:

ENG 🔽	100%
ENG	100%
CHS	200%
JPN	400%

| 绘图栏

本栏用于图形的绘制。用于绘制点、直线、矩形、圆弧、圆和椭圆,还可以写一些简单的字符。详细用法和说明请参考《FANUC PICTURE 操作说明书》(B-66284)2.2.2节。

| 控件栏

控件栏是 FANUC PICTURE 最重要的内容,制作画面所有的元素都来自控件。关于控件的介绍,可以参考《FANUC PICTURE 操作说明书》(B-66284)2.3节。另外,本讲义的第9章也将详细介绍几个常用控件。

FANUC PICTURE 3.1版软件中包含的控件列表如下:

	图标	名称	功能	参考章节
1		Coroon oot control	设定画面公共属性,该控件在CNC上不显示。	222
		Screen set control	双击画面任意位置可显示设定内容。	2.3.2
		Saraan atrustura definition	当主画面创建时,必须使用这个控件设定子	
2	8		画面属性以及是否使用弹出画面等内容。该	2.3.3
		control	控件在CNC上不显示。	

	图标	名称	功能	参考章节
3	e	Screen switch control	画面切换按钮 ,可用于切换主画面和子画面。	2.3.4
4		Lamp control	信号指示灯。	2.3.5
5	BIN	Button control	通用按钮,用于信号输出等。	2.3.6
6	ABC	Framed button control	带有指示灯和边框的信号输出按钮。	2.3.7
7		MDI key control	一个MDI按键,用于输入字母、数字以及功 能键,由控件发送键值代码。	2.3.8
8		MDI keyboard control	MDI按键阵列,单个按键功能同MDI key control。	2.3.9
9	12.34	Key input buffer control	来自MDI或者MDI相关控件的输入缓冲,可写入输入控件。	2.3.10
10	12.34	Numeral indication control with key input	显示数字,并且可以通过key input buffer控 件重新写入。	2.3.11
11	9.99	Numeral indication control for calling a ten-key pad	10键小键盘。数字输入控件,使用是可以弹 出一个带10个按键小键盘。	2.3.12
12	PMC STRG	PMC area character string indication control	显示PMC中存储的字符串,也可以将key input buffer.中的字符串写入PMC。	2.3.13
13	12.34 STRG	Numeral/character string indication control	数字和字符串显示。	2.3.14
14	Α	Label control	固定字符标签。	2.3.15
15		Text character string indication control	显示一个信息列表中的信息,信息列表由PC 机创建。	2.3.16
16		Composite message indication control	合成和显示多个信息表中的不同信息。信息 表由PC机创建 , 单个信息不超过32个字符。	2.3.17
17		History message indication control	显示历史信息(如报警等),可切换概要显示 或细节显示。	2.3.18
18	d MSG	Message indication control	简单信息显示。由PMC信号的1个字节中的8 位定义8个不同的信息。	2.3.19
19	BMP	Image display control	显示位图或者JPEG格式图形(.bmp或.jpg)。	2.3.20
20		Clock control	显示日期和时间的控件。	2.3.21

3. FANUC PICTURE 软件使用说明

	图标	名称	功能	参考章节
21		Meter control	以饼型或条型仪表形式显示数值信息。	2.3.22
22		Graph control	以条型图形式显示数值信息。	2.3.23
23	8	Line chart control	显示折线图	2.3.24
24		Screen switch and image control	功能和画面切换按钮相同 ,增加了图形显示。	2.3.25
25		Button and image control	控制按钮操作,增加了图形显示。	2.3.26
26	ВМР	Lamp and image control	带有图片的指示灯显示。	2.3.27



5. 创建工程及属性设定

用 FANUC PICTURE 制作的自定义画面程序称为工程,每个工程可以编译成一个 CNC 可识别的文件。一台 CNC 只能使用一个工程,但是一个工程可以对应多个画面。同样,在 电脑上,一个工程则对应一个文件夹(目录),工程的名字就是文件夹的名字。一个工程文 件夹中包了含该工程的一些必要信息,如 CNC 类型、语言种类等。

| 创建一个新工程

使用 FANUC PICTURE 软件创建一个新工程,就是建立一个新的文件夹。步骤如下:

7. 单击菜单栏上的 File Project New, 弹出下图所示菜单。

Creating new	project 🛛 🔀
	Restoration from F-ROM
Project Name:	
Directory Name:	D:\Bfm\FANUC PICTURE\
	OK Cancel

2. 单击 🔜 显示目录浏览文件夹菜单,选择一个目录,单击 [确定]。

浏览文件夫	? 🗙
D:\Bfm\FANUC PICTURE\Sample	
Sample AutoHmi-nc AutoHmi-nc Bng FF_Exercise_ENG FF_Exercise_JPN FF_Exercise_JPN MSO19 Multi-Language FF_Multi-Language SampleScript	
確定	以消

3. 在 Project Name 中输入新建工程名称,如:Test,单击 [OK]。

Creating new	project	\mathbf{X}
	Restoration from F-ROM	
Project Name:	Test	
Directory Name:	FANUC PICTURE \Sample	
	OK Cancel	

4. 如果从 F-ROM 中恢复已备份的文件,则需要选择 Restoration from F-ROM。

Creating new project 🛛 🗙				
Restoration from F-ROM				
Project Name:	Test			
Directory Name:	VFANUC PICTURE \Sample]		
	OK Cancel			

5. 选择备份的文件,单击打开。



 通过以上方式,均可创建一个新的工程,之后自动弹出工程属性设定画面。具体 设定内容请参考"工程属性设定"相关内容。

| 打开工程

对于一个已经存在的工程,我们可以通过 File Project Open 命令打开,并进行 相关编辑。

打开		? 🗙
查找范围(<u>I</u>):	🎦 FF_Exercise_ENG 💽 🗲 🖆 🎫	
	 ➡ FFBatchInput ➡ FFScript ➡ PhotoData ➡ StringData ➡ VtsData ➡ FF Exercise ENG.fpp 类型: FANUC PICTURE Project 修改日期: 08-2-19 15:58 大小: 67 字节 	
	文件名(R): Fr_Exercise_ENG.tpp 11 文件类型(T): FANUC FICTURE Project(*.fpp) マ 取	"

| 工程属性设定

打开属性设定有几种方法。前面提到过,创建新工程会自动进入属性设定画面;如果是 一个已经打开的工程,则可以通过菜单 File Project Setting 进入设定画面,或者右 键单击打开的工程名称,再选择 Setting of Project,如图:



工程属性设定对话框,可分为 CNC 系统设定、多语言设定、历史信息收集设定三个部分。下面介绍详细的设定画面和内容。

I CNC 系统设定

属性对话框的第一个选项卡内容就是 CNC 系统设定 ,在本页面可设定 CNC 相关内容。

Setting of CNC system Setting of Multi-language Collection of history	
CNC system FS30i/31i/32i MODEL A	
PMC type	
Display unit type	
Resolution VGA VGA	
Font Small (14inch)	
Soft key 10+2	
-Call key to input pop up screen	
Numeral Indication Control 数字键 🔽	
Numeral Indication Control for 数字键 🔽	
PMC Area Character String Indication Control	
Specify the common focus color	
Focus background color	
Focus character color	
	∃ (λ)

常用的设定例如下:

CNC system :

指定使用 FP 画面的 CNC 类型。如果使用 31i -A 系统 则需要选择 FS30i / 31i / 32i MODEL A。 PMC type :

根据系统型号选择 PMC 的类型,不同 CNC 系统的 PMC 类型可能相同也可能不同,必须根据实际情况选择。本例 31i -A 系统,只有一种 PMC 形式,无须选择。

Resolution :

设定显示器分辨率, VGA 或者 XGA。一般 10.4 寸显示器选择 VGA, 分辨率 640 × 480, 如果是 30i 系列使用的 15 寸显示器,选择 XGA,分辨率 1024 × 768。

Font :

设定字体大小, Big、Small 两种选择, 通常选择 Small (小)。

Soft key :

设定软键键样式。格式对应:基本键 + 扩展键 + 竖排键(30i系列用)。

Call key to input pop-up screen :

如果是在非触摸屏系统上使用 FANUC PICTURE 画面,通常需要设定此功能。

在 FANUC PICTURE 创建的画面中输入字符或数据,可以通过以下三种方式: Key-in buffer、Pop-up ten-key pad 和 Pop-up screen。对于后两种方式,必须在 触摸屏上点击相应控件才会生效。如果我们也想在非触摸屏上使用,则必须设定对 应的触发按键,以激活弹出画面(pop-up 功能)。

对于以下几种控件类型,我们可以选择数字键(0~9等)字母键(数字键和 a~z) 或者特定功能键(软键键、CAN、INPUT、ALTER等)触发弹出画面。 Numeric Indication Control with Key Input Numeric Indication Control for Calling Ten-Key pad PMC Area Character String Indication Control

Specify the common focus color :

一般来说,当前激活的控件可以用其他颜色显示。这些显示的颜色可以单独指定, 也可以统一指定。这里设定统一的焦点(focus)颜色,包括前景色和背景色。

| 多语言设定

FANUC PICTURE 支持多种语言显示,但是常用的 ASCII 字符只占用一个字节,而中 文每一个字符都占用两个字节,所以在显示上会有冲突。本设定页面用于设定语言种类和不 同种类语言的切换方式。

多语言设定例如下,本例包含英语和简体中文。

Setting of project 🛛 🗙				
Setting of CNC system Setting of Multi-language Collection of history				
The Multi-language display function Char. code SIMPLIFIED	×			
The Multi-language Character Type function is us				
Method of switching Multi-language display The display language parameter of CNC is used.				
○ The Multi-language display switch signal is used. (1B)				
PMC Area 0 : R 🗸 Symbol :	~			
PMC Address 0				
No Se Mul Param No. Number Comment Note Char. co 01 1 ENG Default Default 英语 01 FANUC COI	de)E			
02 1 CHS 3281 15 简体中文 02 SIMPLIFIE	2D			
Add Del	ete			
() 御定 () 取消 ()	应用 (A)			

Char. Mode :

如果没有选择多语言显示,则在这里设定默认的字符集,可选择简体中文或者日文。 该字符集主要针对多字节字符,如果使用 ASCII 字符,任何字符集都可以显示。

The Multi-language display function

选择是否使用多语言显示。

Method of switching Multi-language display

设定多语言切换方式,可以使用系统参数方式或者 PMC 信号方式。一般来说,我 们使用系统参数切换方式,并且把语言切换设置成和 CNC 系统语言切换参数一致, 以方便操作者使用。单击 add 按钮,出现图示画面:

在此画面中,可以设定语言标签(Multi-language key)注释(Comment) 切换 参数(Param No. Param bit)以及使用的字符集(Char. code)。

Language change setting					
Multi-language key	CHS				
Comment	简体中文				
🔲 Default display lang	guage				
Param No.	3281				
Param bit	0 🗸				
Number	15				
Char. code	SIMPLIFIED				
	OK Cancel				

| 历史信息收集设定

历史信息是 FANUC PICTURE 软件中最复杂也最难理解的内容,它不同于 CNC 的历 史信息显示,是一套机床厂家自己编辑开发信息记录。本页面设定这些信息存储的位置和显 示方式以及时间日期格式等内容。历史信息收集内容具有以下功能:

- . 可以选择 4 个区域选择历史信息记录的触发信号。
- . 每个区域最多记录 300 条信息,记录了 300 条信息的位置称为一个"块"。
- 每一条历史信息记录包含序号、发生时间、解除时间,以及每个块的信息删除日期和时间等。
- 支持数字指定方式和位指定方式。数字指定方式最多可以监控 32 个 PMC
 区域;位指定方式则可以监控 512 位(64 字节)连续的区域。

该功能是 FANUC PICTURE 中较为复杂的功能,需要配合 FANUC PICTURE 历史信息显示、复杂信息显示控件以及历史信息相关函数共同使用。如果一开始没有用到,可以暂时保持为空,等以后对软件逐步了解后再进行设定。历史信息收集设定例如下:

Setting of project				
Setting of CNC system Setting of Multi-language Collection of history				
History Block Number	Block 1	Block 2	Block 3	Block 4
History Deletion Area	0 : R 💙	0 : R 💙	0 : R 🗸 🗸	0 : R 💙
History Deletion	9091	0	0	0
History Deletion Bit	0 🗸	0 🗸	0 🗸	0 🗸
S-1-1 .	Symbol :	Symbol :	Symbol :	Symbol :
Symbol .	Y	×	×	~
BitIF	Bit	🗹 Bit	🗹 Bit	🗹 Bit
Search Start Area :	1 : D 🗸 🗸	0 : R 💌	0 : R 🗸 🗸	0 : R 💉
Search Start Address	4038	0	0	0
Search Start Bit :	0 🗸	0 🗸	0 🗸	0
Symbol :	Symbol :	Symbol :	Symbol :	Symbol :
Search Bit Counter	0	0	0	0
NumberIF	🔽 Number	Number	Number	Number
Message Årea :	1 : D 💌	0 : R 💌	0 : R 💌	0 : R 💙
Message Address :	4038	0	0	0
Sembol	Symbol :	Symbol :	Symbol :	Symbol :
Symbol .	~	×	×	¥
Message Number	4	1	1	1
Message Area Size VIByte VIByte VIByte VIByte 2Bytes 2Bytes 2Bytes 2Bytes				
Date & Time Display Format				
YYYY 💌	/ MO 💌 /	DD 💙 - HH	* : MI	💙 : SS 💙
		通道	定 取消	应用(2)

6. 基本操作

| 将控件放到窗口中

- 先用鼠标左键单击,选择一个控件,图中<方框 1>。只能使用 FANUC PICTURE 提供的控件。比如选择 Button 控件。
- 2. 在绘图区,鼠标将变为"+"形式,如果没有变化,需要重新选择控件。
- **3.** 按住鼠标左键,以矩形的方式,按对角线方向拖动鼠标,完成后松开鼠标左键。
- 4. 选择的控件粘贴完成。如图,在屏幕左上角出现了一个空白按钮,<方框2>。

| 属性设定

右键单击任意一个绘制好的控件,在弹出菜单中选择 Property...;或者双击任意一个控件,即可打开如上图所示的控件属性画面。不同的控件选项卡数目不同<方框 3>,通过这些选项卡,可以设定控件的功能。一般常用的是 Character、Action 和 Image。



或者,可以直接通过屏幕右边的属性设定表直接修改,如图:

| 移动 / 缩放控件



在控件上任意一个位置,按住鼠标左键拖动,即可完成移动控件的功能。

将鼠标移动到控件边缘,白点的位置,鼠标形状将变为箭头形,此时按住鼠标左键拖动 鼠标,即可放大或者缩小控件。



| 弹出菜单显示

鼠标右键单击任意一个控件,可弹出如下的菜单。通过这个菜单,可以打开属性画面, 完成复制、粘贴、剪切、删除、排序等操作。

<u>B</u> ring to Front Send to Beck	
Bring Forward Send Back <u>w</u> ard	
	Bring to Front Send to Back Bring Forward Send Backward

| 鼠标操作和键盘操作

FANUC PICTURE 可以使用鼠标加 [Ctrl]、[Shift] 或者其他按键动作,相关操作见下表:

功能	操作说明			
选择一个控件	左键单击,可以选定一个控件进行移动、缩放等操作			
显示属性对话框	左键双击一个控件			
控件弹出菜单显示	右键单击一个控件			
<u> </u>	以矩形方式,从起点到终点按住左键拖动鼠标			
近年夕十年日	如果起点在一个控件上,则使用 [Ctrl] + 拖动			
增加选择控件	[Shift] + 左键单击,在原有基础上再增加选择一个控件			
摘加式试试法 按你件	[Ctrl] + 左键单击,在原有基础上增加选择一个控件,或者减少			
墙加以减少边拜控件	一个控件。			
变更当前选择的控件	当选择一组控件时,使用方向键可以变更当前选择的控件			
移动协准	按住左键选择一个控件,并拖动鼠标			
杨刘瑶叶	[Ctrl] + 方向键,可以对多个控件同时进行移动			
<i>按</i> 动物///	将鼠标放在控件边框""的位置,按住左键并拖动鼠标			
细加灯工	[Shift] + 方向键,可以对多个控件同时进行缩放操作			

│ FANUC PICTURE 符号说明

使用 FANUC PICTURE 的时候经常要用到一些 PMC 信号,这些信号也可以像 FANUC Ladder III 软件一样进行定义,使用符号(symbol)来表示。

| 符号数据定义

在菜单栏上,选择 File Symbol 打开符号列表框,如下图所示。

一个新的工程并不包含任何符号定义。一般来说,我们通过这个列表框进行符号数据定 义、修改和删除,也可以通过旁边的按钮进行批量数据的导入导出。目前 FANUC PICTURE 支持以助记符文件 (Mnemonic)格式导入和 CSV 格式的导入和导出。

如果在梯形图中定义了符号数据,可以通过 FANUC Ladder III 软件将梯形图转换为助记符 (Mnemonic) 文件,并通过 FANUC PICTURE 导入画面工程。。



符号数据可定义为布尔型(Bool、是或非)、字节型、字型和双字型。

| 在属性对话框中使用符号代替信号地址

在每一个 FANUC PICTURE 控件的属性对话框中,凡是涉及到 PMC 地址指定的地方,都可以使用绝对地址指定和符号方式指定两种方式。以按钮控件为例,如图:

Button	\mathbf{X}
General Character Action Image	NoAction Image Interlock Image
NoAction:	
Switch Type Momentary 🔻	
Action Type Bit Set 🔻 I	Detail
Shortcut Ke FL/F1A	
Watch ON/OFF	Interlock
Watch: USER 💌	🔽 Interlock
Watch PMC Area 0 : R 💌	InterlockPMCAre 0 : R 💌
Watch Address: 0	InterlockPMCAddre: 0
Watch Bit: 0 💌	InterlockPMCBit 0
🗖 Symbol; 🔽	Symbol:
	确定 取消 应用 (2)
Watch PMC Area 5 : X	Match FML Area 5 : X
Watch Address: 8	Watch Address: 8
Watch Bit: 4 💌	Watch Bit: 4
Symbol *ESP. K	Symbol *ESP. K
使用绝对地址方式	使用符号方式

在上面两张图中,按钮的地址都是 X8.4,指定方式不同而已。

| 在 FP 函数中使用符号

在 FP 函数中,同样可以使用符号代替 PMC 绝对地址。具体格式如下:

绝对方式:

FP-function-name [PMC-area_PMC-address_bit-postion-or-type_data] 符号方式:

FP-function-name [:scope.symbol: _data]

例:

按位写入 PMC

wrpmcb [adrtype_snumber_bitpos_data] => wrpmcb [:symbol: _data]
wrpmcb [5_0035_1_1] => wrpmcb [:DoorOpen: _1]

| 生成 CNC 可识别 MEM 文件的步骤

生成 MEM 文件

在这一步,我们需要将 FANUC PICTURE 创建的画面程序文件编译为存储卡格式。步骤如下:

 先择菜单栏的 File Project Make MEM Files...或者单击工具条的 按钮,弹出如下画面:

L	aking memory	card file			
	PMC type:	SB7]	
	Start-up screen:	FPictu	ure1 🔽	$\overline{\mathbf{D}}$	
	Num of registered us	er screens:	□ Ma	aking of backup to F-R	.OM
	Form File Name	Softkey Number	Key Caption	Screen Number	Selec
	[RETURN]		FP		0
	FPicture1	0×0000	TEST	0	1
	FPicture2		FPicture2	0	0
	FPicture3		FPicture3	0	0
					>
	🔽 No use multi-lang	uage of key caption			
		ОК	Cancel	Detail FP-	PARAM

2. 选择 FANUC PICTURE 开始画面。在 Start-up screen 选项中列出的画面中选择。

- **3.** 如果希望把 FANUC PICTURE 创建的画面信息保存在 F-ROM 中,可以选择 Making of backup to F-ROM 选项。恢复备份可以参考本讲义第5章相关内容。
- 4. 双击任意一个主画面包括 [RETURN],在弹出对话框中选择启动 FANUC PICTURE 的按键,可使用 MDI 按键或系统软键键。如果使用系统软键键,则可 在 Key Caption 一栏设定软键键名称。工程中设定了多语言,那么按键的名称也 可以选择多语言设置。最后,选择 Select flag 选项,确认从 CNC 调用画面有效。

Setting user s	creen	×
🔽 Select flag		
Softkey Number:	0x0100 POSITION-RELATIVE]
Multi-language key:	CHS	
Key Caption:	test	
	OK Cancel	

5. 单击 [ok] 按钮,编译成功则弹出如下对话框。



如果编译有错误则记录在 MKMEMERR.log 文件中,如图,选择是,可自动打开错误记录文件。

FARUC	PICTURE
8	An unregistered screen name is specified. Please look at MKMEMERR.log. Do you open copies of MKMEMERR.log?
	<u> </u>

| 写入存储卡

Write to card		X		
	Size:	Last Updated Date:		
FPF0FPDT.MEM	34KB	2008/2/28 17:15		
BY27.mem	1025KB	2007/7/2 14:30		
Transfer to:	File	name:		
PICTURE\Sample\SampleScript	FPF	OFPDT.MEM		
The FP driver is transferred to the memory card.				
	[OK Cancel		

FANUC PICTURE 软件生成的存储卡程序包含两个文件,一个是用户编译画面的文件, 文件名是 FPF0FPDT.MEM 或者 CEX0FPDT.MEM;另一个是 FP 驱动文件,详细内容见下 表。第一次传输或者使用了新版 FANUC PICTURE 软件的时候,必须将两个文件全部导入 CNC,以后则只需传入用户自定义文件即可。

CNC 类型	FP 驱动文件名称
30i / 31i / 32i - A 系列	BY27.mem
0i D 系列	DY61.mem
0i C 系列	DY60.mem
16i / 18i / 21i - A / B 系列	BY17.mem
Power Mate i - H	BY18.mem
Power Mate i - D	BY19.mem
16i / 18i / 21i - A / B 系列	BY28.mem

& BY28 用于多个 C 语言执行器的情况,需要选择对应的功能。如 Macro executor + Multi C executor (S770)。

| 使用存储卡传入 CNC 的 F-ROM 中

在系统屏幕左侧插入存储卡,同时按住 CNC 屏幕下方最右边的两个软键键或者 MDI 上的 6 和 7 两个按键,接通 CNC 电源,系统进入 Boot 画面。在 Boot 画面中选择" 2. USER DATA LOADING "或者" 3. SYSTEM DATA LOADING "均可载入存储卡中的程序,和 boot 画面装入梯形图方法相同。



& 首次使用需要装入 FP 驱动文件。

7. 控件

本章讲述控件的使用方法,因篇幅限制,无法全部列举,只选择了最常用的10个控件。 关于其他控件的使用方法,可以参考《FANUC PICTURE 操作说明书》(B-66284)。

| 设定控件公共属性

所谓公共属性,就是所有控件几乎都有的常用属性,因为他们设定起来大同小异,所以, 集中到一起进行说明。

| 控件坐标位置和大小设定

控件的坐标设定在属性的 General 页面中,可以手动设定控件坐标位置。但通常情况下,坐标位置通过鼠标拖动控件即可完成自动设定。而控件占用画面空间的大小,也在此画面设定,单位和坐标位置一样。

在 FANUC PICTURE 制作的的画面中,计算基准是控件的左上角。屏幕的左上角坐标 定义为(0,0),水平方向为 X 方向,垂直方向为 Y 方向。对于 VGA 类型的显示器,分辨 率为 640×480,坐标范围是(0,0)-(639,479);而 XGA 类型的显示器,分辨率为 1024 ×768,坐标范围是(0,0)-(1023,767)。

Char				×
General	Character			
Object TD:	FPChar4			
X:	464	¥:	256	
Width:	120	Height:	40	
		确定	取消 应用	(<u>A</u>)

例如,设定字符显示控件位置:

| 设定显示的字符

在控件属性的 Character 页面中,可以输入控件显示的字符。如按钮标签等。在字符显示选项中,同样可以指定字符位置,也可以选择字符居中显示。另外,对于按钮一类的控件,还可以用不同字符表示按下状态和弹起状态,当然,也可以指定同样的字符显示。

如果字符显示需要换行,则使用和C语言一样的"\n"字符(引号以内部分)。

在 Character Type 选项中可以选择字体的大小,共有 5 种字体,ANK、X2、X4、X6 和 SMALL。ANK 是标准字体,SMALL 最小,X6 最大。另外,简体中文只能选择 ANK 标 准字体。

ChgScrn						X
Gener al	Character	Action Image	NoAction 3	Image Interlock	Image	
🔲 No u	use multi-la	anguage				
Multi-1	anguage 🛛			v.		
🔽 Cent	erin 🔽 On	caption is co	pied onto Of:	fcag		
ON Contine	循环	m启动				
	Х:	0	Y: 0			
OFF	循环	n启动				
	Х:	0	У: 0			
Charact	er Type:	2 : X4	•			
ON Capt	ion					
OFF Cap	tion					
				确定	取消	应用(A)

| 编辑多语言

如果工程设定了多语言选项,那么在 Character 页面里就可以使用多语言标签。如图:

🖵 No use multi-language	🔲 No use multi-language		
Multi-language	Multi-language		
🔽 Centerin 🔽 On caption is copied onto Off ca	🔽 Centerin 🔽 On caption is copied onto Off ca		
0N C+: 简体中文	ON English		

简体中文显示

英文显示

| 设定颜色

无论是字体还是背景色,大部分 FANUC PICTURE 控件都可以修改颜色。一般不需要 设定太复杂的颜色,从图示基本颜色中选择即可。如有特别需要,可以使用自定义颜色。



| 画面设定控件

画面设定控件的属性页面可以通过双击画面空白部分打开,或者右键单击,再从弹出菜 单选择 Property... 这个控件用于设定画面相关信息,如背景色等。

| 设定项目说明:

Background color

设定背景色

Use signal of notice screen display

设定当前基本画面显示时,是否输出一个 PMC 信号。这个功能和画面结构定义控件中的功能相同,当两者全部设定时,以画面结构定义控件中的设定为准。

Use enable signal of each action control

设定是否启用 FP 各个功能有效信号。该设定用于 "FP 功能间接指定"。当这个信 号为 1 时, FP 功能开始执行。


画面结构定义控件是画面中非常重要的控件,必须在每个基本画面中设定,而且只能使 用一个。该控件有两个有两个功能,一是设定主画面和子画面的关系,二是设定弹出型画面。

这个控件在 CNC 屏幕上不显示,通常情况下,我们把它放在画面左上角,如图:

(]	F	P	i	c	t	u	r	e	1	-	x	1						
1P															-				-	
			~	"		"	"	٩.	_											
		2						3												
		2						3												
	F	^						ι,	7											
ŀ		2						2												
ŀ		2						3												
ŀ		Ň,	~	~		~	~	3	-											
ŀ								١.												
ŀ																				
ŀ																				
ŀ																				
ŀ																				
ŀ																				
ŀ																				
ŀ																				
ŀ																				

| 关于主画面、子画面和弹出画面

一个基本画面(Base Screen)可以包含一个主画面(Main Screen)和最多5个子画面(Sub Screen);也可以只是一个主画面。子画面可以单独切换,也可以和主画面一起作为一个基本画面共同切换。在子画面中,不需要使用画面结构定义控件。

除了主画面和子画面, FP 还支持弹出画面 (popup screen)。弹出画面是在当前画面 前段再生成一个画面,原有画面不擦除。弹出画面的大小和位置,由画面结构定义控件设定。



| 设定项目说明:

Base screen Use Popup Screen

选择当前画面是否作为弹出画面显示。

Base screen Free Memory

通常画面信息在显示的时候会先保存在内存中,可以实现高速显示。但是内存的容 量是有限的,如果选择了这个选项,则该画面不保存在内存中。

Base screen UseOutPMC

这个功能和画面设定控件中的 Use signal of notice screen display 设定的功能类似,只是优先级更高一些。

Sub screen Use Sub Screen n

设定基本画面中包含的子画面名称。

Sub screen Save SubScreen n

保存之前使用过的子画面。例如,主画面 main1 包含一个子画面 sub1,若选择了 Save SubScreen1,则,当子画面 sub1 切换为 sub2 以后,再次显示主画面时保 存上次的切换记录,显示 sub2。反之,仍然显示 sub1。

Option Use Signal of Update notification

如果当前基本画面中的数据或字符串发生变化,例如被其他控件修改等,则输出一个 PMC 信号。

🗉 画面切换按钮控件 🚟

画面切换按钮用于切换 CNC 当前的屏幕显示。可以用于基本画面切换。子画面切换、 NC 画面切换以及弹出画面切换和擦除。

这个控件的按钮功能可以通过不同方式进行互锁,详细内容请参考《FANUC PICTURE 操作说明书》(B-66284)2.3.4节。

| 设定项目说明

Action Screen Type

选择切换的画面类型,如果是弹出画面,选择 base screen。如果选择 Popup Screen Clear,则按钮按下后当前弹出画面被擦除。

Action Screen Name

选择切换的画面名称,如果是 NC 画面,则无须设定。

Action Key Code

如果 Screen Type 选择的是切换到 NC 画面,则以 MDI 按键方式选择相应的 NC 画面。

Action Shortcut Key

选择按钮的快捷键,如果不想使用触摸屏或者没有触摸屏,可选择这个选项。

Action Blink PMC

设定由用户指定的 PMC 信号控制画面切换按钮以一定频率闪烁。

Action Light PMC

按钮按下,用户设定的 PMC 信号变为1,并保持。可通过梯形图将该信号置为0。

Image Style

设定按钮样式。

│ 普通按钮控件 🔤

普通按钮控件和画面切换按钮控件形式上比较类似,但功能有较大区别。普通按钮是一 种通用的按钮控件,可针对不同的控制对象,不像画面切换控件只针对画面。另外,这个控 件的按钮功能也可以通过不同方式进行互锁,和画面切换按钮控件一样,不再赘述。

| 设定项目说明

Action Switch Type

选择按钮切换类型, Momentary 和 Alternate。第一种发出一个按键脉冲,第二种 有 ON 和 OFF 两种状态。或者说,第一种是按下自动弹起,第二种是按下或弹起 自保持。

Action Action Type

选择按钮的活动方式 ,	可选择以下几种功能:
[Bit Set]	按位设定 PMC 地址信号 ;
[Data Set]	将一个数据写入 PMC ;
[Function Call]	执行 FP 功能 ;
[Change Screen]	调用一个画面(必须使用 Momentary 类型);
[History Display]	切换用户自定义历史信息显示。

Action Detail

根据 Action Type 设定的方式,选择具体的功能执行内容。

Action Watch ON/OFF

选择按钮 ON/OFF 状态改变方式,可选择自动方式(根据按钮控制的 PMC 位)或 者用户自定义 PMC 信号方式,通常选择自动方式。



指示灯控件可根据 PMC 信号状态显示不同颜色和字符。

| 设定项目说明

Character

关于字符设定,可参照前面章节的公共属性设定语言、位置、颜色、状态等。

Action Watch PMC Area

设定指示灯监控的 PMC 信号地址

Action Use Blink

当指定的 PMC 信号为 1,指示灯持续闪烁,类似按钮的 Blink 功能。

Option Use 7 Color

选择按钮 ON/OFF 状态改变方式,可选择自动方式(根据按钮控制的 PMC 位)或 者用户自定义 PMC 信号方式,通常选择自动方式。

Option Type

选择显示方式。	
[PMC (bit)]	根据 PMC 地址信号位设定 ;
[PMC (Compare large or small)]	根据 PMC 比较数据大小设定 ;
[Function (bit)]	根据功能函数返回值的位设定 ;
[Function (Compare large or small)]	根据功能函数返回值比较数据大小设定。

Option Set Up Number

当选择比较大小时,设定比较数据的个数。

Option Data Length

当选择比较大小时,设定数据长度,是否带符号。

Option Light Color #0 - #6

设定各个位对应的指示灯的颜色。

Option Character Color #0 - #6

设定各个位对应的字符颜色。

Option Change Beginning Value

设定比较数值大小,当 PMC 比较结果大于设定数值时,则显示相应颜色。该选项 在使用了 PMC (Compare large or small) 或者 Function (Compare large or small) 时有效。设定例如图:



当比较数据在 0~9 之间,输出黄色; 10~19 则为红色。

Option Function Name

设定功能函数名称,由此函数决定显示颜色。

Option Function Blink Bit

设定闪烁信号位 0~7,如果不使用闪烁功能,设定 No use blink。

□ 标签控件 A

标签控件可实现在一个可定义矩形框内显示字符或数字。

| 设定项目说明

Character

关于字符设定,可参照前面章节的公共属性设定语言、位置、颜色等。

Image

设定矩形框外形、颜色、边框等。

Ⅰ 数据 / 字符显示控件 12.34 STE6

本控件用于显示 PMC 数据或者通过 FP 功能函数读取的 NC 数据,可显示数字或者数 字对应的 ASCII 字符。

| 设定项目说明

Action Action Type

选择数据来源,PMC 或者 FP 功能函数。

Action Display Data Type

选择数据源类型,字节型(Byte),浮点型(Double)或者字符串(String),

Action Display Format

选择数据源格式,十进制(Decimal),二进制(Binary)或者 ASCII 代码形式。

Action Display Use Decimal Point

在十进制数据中使用小数点。

Action Display Decimal Digit

十进制小数点位数。

Action Display Non zero surpress

是否显示小数点前无意义的 0 位。

Action Display Decimal digit

十进制小数点位数。

Action Display Number of digits specification 指定小数位数,只有选择十进制有效。

Action PMC

当 ActionType 选择 PMC 时有效,用于设定 PMC 相关内容。

Action Function

当 ActionType 选择 Function 时有效,用于设定 FP 功能函数相关内容。

Action Function Display

选择是否显示函数内容,可以实现只计算而不显示。

Action Option

在 Option 选项中,也可以选择类似指示灯控件类型的7种不同颜色的显示方式。

| 字符显示 / 输入控件 🚟

本控件主要用于字符的输入,各种字符以ASCII代码格式存储在 PMC 地址中。在输入的时候可以自定义画面或者键盘输入缓冲控件。

| 设定项目说明

Action PMC Char Count

设定字符位数。一般一个字符占用一个字节。

Action Popup Screen Name

选择自定义输入画面名称,该画面必须是弹出画面形式。

Action Use Key-in-buffer

使用键盘输入缓冲控件。如果仅用于显示字符,本选项和 Popup Screen Name 选项都不要选择,这样就不会触发写入功能。

Action Look Signal

设定一个 PMC 信号,只有当这个信号为1时,可以进行字符写入。

| 数值输入控件 🛃

数值输入控件用于对 PMC 进行数字的输入或显示,在用法上与字符输入控件类似,但 增加了一些功能。

| 设定项目说明

Action In / Out Target

设定输入输出目标, PMC 地址或者 FP 功能函数。并根据所选内容设定相关参数。

Action Set max and min value

选中后,可以设定输入数据的最大值和最小值,超过此范围无法输入。



键盘输入缓冲控件,有些类似 NC 的输入提示栏,用于显示输入字符的缓冲,检测输入数据是否正确。当输字符时,先显示在缓冲控件中,按 [INPUT] 键后才会写入 NC。每个 画面只能创建一个键盘输入缓冲控件。

| 设定项目说明

Action Interlock Message

设定当键盘输入缓冲无效时,输入框的提示信息。

Action Number of digits specification

选中后可指定输入数据的位数,包括符号和小数点。



图片显示控件用于在 CNC 屏幕上显示一个图片,支持 BMP 格式和 JPEG 格式。因为 CNC 系统只支持 256 色,而电脑屏幕通常都是 16 ~ 32 位真彩色,所以 CNC 上图片显示 效果可能没有电脑上好。

另外,在 CNC 上显示 BMP 图片比 JPEG 图片速度稍快,但是 JPEG 图片容量小,可 以节省存储空间。

| 设定项目说明

General Image File Name

在此文本框内输入图片文件名。或者单击后面的" 🔜"图标,选择一个文件。

以上内容是关于 FANUC PICTURE 软件最基本控件的介绍,很多不同控件中重复的或 者有类似含义的功能都没有详细介绍,具体内容可以参考《FANUC PICTURE 操作说明书》 (B-66284)。随着软件版本的更新,FANUC 会提供更多的控件给设计人员使用。

8. FP 函数

FANUC PICTURE 软件中的函数功能,简称 FP 函数,可以用于读取 CNC 状态或者 PMC 状态。这些函数主要包含 CNC 库函数(CNC window library), PMC 库函数(PMC window library),以及其他类型函数(矩阵形式变量控制、FP 脚本控制函数)。

| 函数指定方法

FP 函数可以在 FANUC PICTURE 软件的控件属性对话框中设定。如图:

NDisp			
General Character	Action Image Option N	oAction Image	,
TNOAction: Action Type:	1: Function		指定 Function 方式
		Display Data Type: -	0 : 1 Byte(Signe 💌
PMC		Format:	0 : Decimal 💌
Watch PMC Area:	0 : R 💌	🔽 Use Decimal F	oi
Watch Address:	0	Decimal digi	0
🔲 Symbol:			用的 FP 函数
Function Function Name:			
			_
		确定	取消 应用 (4)

在数据/字符显示控件中,数据来源可以指定 PMC 或者 Function,当我们选择 Function 后,可以在最下面的输入框中设定使用的 FP 函数。

大部分 FP 函数都有自己的参数,参数范例中括号"[]"里面的部分。这些参数在设定的时候必须指定,并以下划线"_"分隔,不可以使用任何形式的空格。关于参数的含义和格式可以查操作书或者帮助文件。

对于多通道的系统,则在参数最后增加一个后缀"_(通道号)"进行区分。

例如:读取第2通道绝对位置坐标,对应函数为 abslute[3_2]

I CNC Window 库函数功能说明

- | 读取实际进给速度(F)
 - [函数名称] actf
 - [数据类型] 5:4 bytes

[参数格式] actf[(path)]

[参数说明] (path) CNC 通道号指定,可选

[返回值]

实际轴进给速度

[相关设定] CNC 参数 3105#0=1(是否显示实际进给速度)

| 读取各轴绝对坐标

- [函数名称] absolute
- [数据类型] 4:4 bytes (带符号)
- [参数格式] absolute[axis(_path)]
- [参数说明]

 axis
 轴号(1 到最大控制轴数)

 (path)
 CNC 通道号指定,可选

[返回值]

指定轴的绝对位置坐标。

| 读取用户宏变量

[函数名称]

rdmacro

[数据类型]

7:Floating-Point

[参数格式]

rdmacro[number_pmc_address(_path)]

[参数说明]

pmc

number 用户宏变量号

- 用于输出处理结果的 PMC 类型
 - 5: R
 - 9: D
 - 12: E (PMC-SB7 , PMC for Series 30i)
 - 99: 不输出处理结果

address 用于输出处理结果的 PMC 地址 (需占用 2 字节)

如果不输出处理结果,指定0

(path) CNC 通道号指定,可选

[功能说明]

读取 CNC 中用户宏变量的数值

[使用例]

rdmacro[100_9_200] 读取用户宏变量#100,并把处理结果写入 D200。 rdmacro[1000_99_0_2] 读取第 2 通道#1000,不输出处理结果

[输出的处理结果]

- 0 正常
- 2 不正确的块长度
- 3 用户宏变量号不正确
- 5 宏变量数值超出允许的范围
- 6 缺少相应的系统选择功能
 - 例如:
- 用户宏程序功能
- 用户宏变量号扩展功能
- -1 读取失败,可能是其他程序正在写入用户宏变量或者其他窗口功 能正在执行

| 写入用户宏变量

[函数名称]

wrmacro

[数据类型]

2:2 bytes (signed)

[参数格式]

wrmacro[number_data1_data2_pmc_address(_path)]

[参数说明]

_	
number	用户宏变量号
data1	不含小数点的整数型数据(使用 ten-key 控件设定 0)
data2	小数位数(使用 ten-key 控件设定 0)
pmc	用于输出处理结果的 PMC 类型

- 5: R
- 9: D
- 12: E (PMC-SB7, PMC for Series 30i)
- 99: 不输出处理结果

 address
 用于输出处理结果的 PMC 地址(需占用 2 字节)

 如果不输出处理结果,指定 0

(path) CNC 通道号指定,可选

[功能说明]

在 CNC 中写入用户宏变量的数值 data1 和 data2 的设定规则如下:

实际写入数值	data1	data2
空	0	-1
0.000	0	3
0.0000001	1	7
00000.100	100	3
00001.000	1000	3
999999999.	99999999	0
-00123.000	-123000	3

当写入 "123. " 时,可以设定 " data1=123000 ; data2=3 " 也可以设 定 " data1=123 ; data2=0 " 或者 " data1=1230 ; data2=1 " 等等, 并不是唯一的。另外,这个 FP 功能函数还可以写入一个空变量。

[使用例]

wrmacro[100_0_0_9_10]

使用 ten-key 控件将输入数值写入#100,处理结果输出到 D10 wrmacro[1000_123456_2_99_0_2] 将 1234.56 写入第 2 通道的宏变量#1000,不输出处理结果

[输出的处理结果]

- 0 正常
- 2 不正确的块长度
- 3 用户宏变量号不正确
- 5 宏变量数值超出允许的范围
- 6 缺少相应的系统选择功能

例如:

- 用户宏程序功能

- 用户宏变量号扩展功能
- -1 读取失败,可能是其他程序正在写入用户宏变量或者其他窗口功 能正在执行

| 读取操作信息

[函数名称]

rdopmsg

[数据类型]

6:String

[参数格式]

rdopmsg[type_pmc_address(_path)]

[参数说明]

pmc

type 操作信息类型(通常设定0)	
--------------------	--

- 用于输出处理结果的 PMC 类型
 - 5: R
 - 9: D

12: E (PMC-SB7 , PMC for Series 30i)

99: 不输出处理结果

address 用于输出处理结果的 PMC 地址 (需占用 2 字节)

如果不输出处理结果,指定0

(path) CNC 通道号指定 (30i), 可选

[功能说明]

读取操作履历信息并显示在 CNC 画面上。 对于 0i-C、16i/18i/21i、Power Mate i , 设定 CNC 参数 3207#0 = 0. 对于 30i 系列,根据实际情况设定 CNC 参数 6310#1 和 6301。

[使用例]

rdopmsg[0_9_10] 读取操作信息完成,信息内容显示在屏幕上,处理结果输出到 D10.

[输出的处理结果]

- 0 正常
- 4 操作信息类型错误
- 6 没有外部信息显示或外部信息输入功能

│ 读取 CNC 报警

[函数名称]

rdalminfo

[数据类型]

- 6: Character string (只能显示一行)
- 8: Rectangular character string

[参数格式]

rdalminfo[type_almtype_num_pmc_address(_path)]

[参数说明]

type 显示类型

0: 只显示报警号 报警号和报警信息 1: 报警类型 almtype Series 30i /31i /32i 0: 参数写保护 (SW) 1: 输入参数需要断电 (PW) 2: I/O 错误 (IO) 3: 前台 P/S 报警 (PS) 超程报警 / 外部数据输入报警 (OT) 4: 5: 过热报警 (OH) 6: 伺服报警 (SV) 7: 数据 I/O 错误 (SR) 8: 宏报警 (MC) 9: 主轴报警 (SP) 10: 其他 DS 报警 (DS) 11: 防止故障相关报警 (IE) 12: 后台 P/S 报警 (BG) 13: 同步超差错误 (SN) 14: (Reserved) 15: 外部报警信息 (EX) 16: (Reserved) 17: (Reserved) 18: (Reserved) 19: PMC 错误 (PC) 99: 所有类型 Series 0i-TTC, Series 16i /18i /21i, Power Mate I 0: P/S 100 1: P/S 000 2: P/S 101 3: 其他 P/S 报警 4: 超程报警 5: 过热报警 6: 伺服报警 8: APC 报警 9: 主轴报警 10: P/S 5000 99: 所有类型 获取的报警数量(1-50) num 如果指定数据超过范围,则默认为1 用于输出处理结果的 PMC 类型 pmc 5: R 9: D 12: E (PMC-SB7, PMC for Series 30i) 99: 不输出处理结果

address	用于输出处理结果的 PMC 地址(需占用 2 字节)
	如果不输出处理结果,指定 0
(path)	CNC 通道号指定,可选

[返回值]

- 0 正常
- 3 显示类型指定错误
- 5 报警类型指定错误

[功能说明]

用于获取 CNC 发生的报警信息。

对于 0i-C、16i /18i /21i、Power Mate i,系列,这个函数每次最多可以读取 5 条报警信息。

信息内容是字符串形式,和当前 CNC 显示语言相同。

| PMC Window 库函数功能说明

Ⅰ PMC 读取(位)

[函数名称]

rdpmcb

[数据类型]

1:1byte

[参数格式]

rdpmcb[adrtype_snumber_bitpos]

[参数说明]

adrtype PMC 地址类型

0:

1: F

G

- 2: Y
- 3: X
- 4: A
- 5: R
- 6: T
- 7: K
- 8: C
- 9: D
- 12: E (PMC-SB7 , PMC for Series 30i)

snumber 需要读取的 PMC 地址

bitpos 位(0-7)

[功能说明]

读取 PMC 的 1 个指定位

[使用例]

rdpmcb[9_100_0] 读取 D100 的第 1 位

Ⅰ PMC 写入(位)

[函数名称]

wrpmcb

[数据类型]

[参数格式]

wrpmcb[adrtype_snumber_bitpos_data]

[参数说明]

adrtype	PMC 地址类型
	0: G
	1: F
	2: Y
	3: X
	4: A
	5: R
	6: T
	7: K
	8: C
	9: D
	12: E (PMC-SB7 , PMC for Series 30i)
snumber	需要读取的 PMC 地址
bitpos	位(0-7)
data	写入数据 0 或 1(当使用 ten-keys 输入控件时 ,设定 0)

[功能说明]

在指定 PMC 位写入 0 或 1

[使用例]

wrpmcb[9_0_0_1] 在 D0 第 1 位写入 1

Ⅰ PMC 读取(1/2/4 字节)

[函数名称]

rdpmc

[数据类型]

- 0: 1 byte(Signed)
- 1: 1 byte
- 2: 2 bytes(Signed)
- 3: 2 bytes
- 4: 4 bytes(Signed)
- 5: 4 bytes

[参数格式]

rdpmc[adrtype_snumber_datatype]

- [参数说明]
 - adrtype PMC 地址类型
 - 0: G
 - 1: F
 - 2: Y
 - 3: X
 - 4: A
 - 5: R

- 6: T
- 7: K
- 8: C
- 9: D

12: E (PMC-SB7 , PMC for Series 30i)

snumber 需要读取的 PMC 地址

datatype PMC 数据类型(和数据类型设定相同)

- 0: 1 byte
- 1: 2 bytes
- 2: 4 bytes

[功能说明]

读取指定的 PMC 地址

[Example]

rdpmc[5_2500_1] 读取 R2500 开始的 2 字节数据

PMC 写入(1/2/4 字节)

[函数名称]

wrpmc

[数据类型] ------

[参数格式]

wrpmc[adrtype_snumber_datatype_data]

[参数说明]

adrtype PMC 地址类型

- 0: G
- 1: F
- 2: Y
- 3: X
- 4: A
- 5: R
- 6: T
- 7: K
- 8: C
- 9: D
- 12: E (PMC-SB7 , PMC for Series 30i)

snumber 需要读取的 PMC 地址

datatype PMC 数据类型(和数据类型设定相同)

- 0: 1 byte
- 1: 2 bytes
- 2: 4 bytes

data 写入数据(当使用 ten-keys 输入控件时,设定 0)

[功能说明]

在指定 PMC 地址写入数据

[使用例]

wrpmc[9_2000_0_127] 在 D2000 开始的 1 个字节写入 127

| 其他类型函数

其他类型的函数主要包括矩阵形式变量的读取、写入、复制等功能,以及专门用于 FP 脚本功能控件属性设定函数等。这些函数通常不会使用,在本讲义中不做详细 介绍。

9. 在非触摸屏显示器上的应用

一般来说,FANUC 推荐在触摸屏上使用 FANUC PICTURE 软件,以达到最好的效果。 但是,考虑的各种用户的需要,FANUC PICTURE 也可以在非触摸屏系统上使用,但显示 单元必须具有图形显示功能。

在非触摸屏上制作画面时,基本方法和触摸屏相同,也是通过设定控件属性的方式来实现。大多数针对触摸屏的画面都可以通过修改控件属性设定来适应非触摸屏,并不需要重新制作。对于一个工程来说,在触摸屏或非触摸屏环境下使用,他们并不矛盾。也就是说,我们可以实现同一个工程既可以在触摸屏上使用,又可以应用于非触摸屏。

控件名称	属性名称	简要说明				
屏幕切换按钮控件		将屏幕上需要点击的部分 ,通过快捷键的方式映射				
普通按钮控件	快捷键	到软键键或者 MDI 按键上。以实体按键代替屏幕				
带边框的按钮控件		触摸按键。				
同草识宣协供	焦点规则	设定焦点移动规则。使用方向键可以移动这些焦				
併希设定控件 	标签	点,以选取不同的输入控件。				
数值输入控件		焦点标签用于指定各个可输入数据控件的索引号,				
Ten-key 数值输入控件	焦点标签	也就是光标的移动顺序。在一个画面中每个索引号				
PMC 字符显示控件		都是唯一的。				

FANUC PICTURE 软件针对非触摸屏的属性如下表所示:

& 对于非触摸屏,MDI输入单元是必须选择的,所以在 FANUC PICTURE 软件中不需 要使用 MDI 按键控件和 MDI 键盘控件,但如果使用也没有影响。

当使用非触摸屏时,画面上每个按钮、开关都可以对应一个 MDI 按键或者一个软键键。 在触摸屏上,通常靠点击屏幕对应的位置来选择一个控件,而对于非触摸屏则使用方向键移 动屏幕焦点(focus)。如果当前焦点选择了一个输入型控件,那么可以通过弹出 Ten-key 键 盘、弹出自定义画面或者输入缓冲控件三种方式进行数据和字符的输入。同时,每一个画面 中可选择的控件都有一个唯一的索引号(Index Number),通过这个索引号和焦点移动规则, 就能够实现屏幕光标(焦点)按照方向键指定的方式移动了。

| 屏幕中焦点移动功能

在用户自定义画面第一次显示时,没有任何焦点。之后,当按下光标键,焦点会按照"主画面 子画面 1 子画面 2 …… 子画面 5"的顺序移动。如果当前焦点的索引号是主画面或者子画面的最大值时,如果按下了增加焦点的按键,则光标跳转到下一个画面。

例如,该画面的主画面包含10个控件,子画面1包含6个控件,子画面2包含4个。 画面结构如图:



首先,设定每个可选控件的索引号,以按钮为例,如图。同一个画面中,索引号不能重复,这一点需要特别注意。

Input 🛛 🛛
General Character Action Focus Interlock Image
Focusing index © Specify the focusing index of this contri 1 © Use Tabladex property for focusing index
Focus movement rule UP Key O DOWN Key
Don t move rocus to the next control after data input convetion

其次,要在主画面中设定焦点移动规则。双击屏幕背景任何地方,打开画面属性设定控件,翻到 Focus rule 页面。由图可知,按方向键""或者"",当前焦点 -1;按"" 或者"",当前焦点数值 +1。

ScrnSet	×
General Option Focus rule	
DOWN Key Continuous inp Direction C UP C LEFT RLGHT C C DOWN	
	3)

所以,如果当前光标在主画面的 10 号控件上,我们按下了方向键"""或者"", 那么当前焦点将变为子画面 1 的 1 号控件。

| 连续输入功能

当一个画面中包含有两个或多个输入型控件,如数字输入控件、ten-key型数字输入控件、PMC字符输入控件,可以使用连续输入功能。通常情况,输入一个数据以后,要先按方向键移动光标,再输入下一个数据。使用连续输入功能,可以省去移动方向键的步骤,当输入完成后,光标直接跳到下一控件。

在屏幕属性设定控件中可以选择使用该功能以及设定光标移动的方向,如图:

ScrnSet	×
General Option Focus rule Common focus movement rule UF Key -1 DOWN Key I Continuous inp Direction C UF C LEFT RIGHT DOWN DOWN	
确定 取消 应用 (4)	

连续输入功能也可以应用于触摸屏。

| 快捷键功能

关于快捷键的用法,在前面有过简略介绍,用于实体按钮替代触摸屏幕。快捷键在以下 控件的 Action 页面可以使用:

> 屏幕切换控件 按钮控件 带边框的按钮控件

以普通按钮控件为例,选中 Shortcut Key 选项,如图。当按键按下时,不管是否有触 摸屏、或者和是否屏幕被点击,都会执行相应控件的功能。

Button				
General Charac	ter Action Ima	ge NoAct	ion Image Interlock	: Image
🔲 NoAction:				
Switch Type	Momentary 💌]		
Action Type	Bit Set 💌	Detail	1	
🔽 Shortcut Ke	FL/F1A 💌]		
Watch ON/	OFF		Interlock	
Watch:	AUTO 💌]	TINT Interlock	
Watch PMC Area	0 : R 💌]	InterlockPMCAre	0 : R 💌
Watch Address:	0	1	InterlockPMCAddre:	0
Watch Bit:	0 💌]	InterlockPMCBit	0 💌
🔲 Symbol:	<u>_</u>]	Symbol:	
			确定 取消	当 应用(A)

可选择的按钮列表如下:

L

软键键和扩展键(包括水平和垂直)
FL、F1-F10、FR 对应 10 + 2 的排列
FL、F1-F5、FR 对应 5 + 2 的排列
VF1-VF8, and VF9 for the vertical soft keys)
方向键和翻页键(和 Page-UP、Page-DOWN)
编辑键(CAN、INPUT、ALTER、INSERT、DELETE)
帮助键(HELP)

I 软键键的排列形式:



| 各控件焦点控制相关属性设定

以下控件具有焦点控制附加属性页面:

数值输入控件 Ten-Key 数值输入控件 PMC 字符串输入控件

以 PMC 字符串输入控件为例,属性设定页面如图:

PICStrDisp	×
General Character Action Focus Image	
Focusing index © Specify the focusing index of this conting 1 C Use TabIndex property for focusing index Focus movement rule UF Key DOWN Key DOWN Key DOWN Key Down t move focus to the next control after data imput energies	

在这个设定页面中,可以指定焦点索引(Focus index),通常使用第一个选项,也就是 "Specify the focusing index of this control as",直接根据需要设定索引号。设定后,即可 由方向键控制焦点按照由小到大的顺序移动。如果设定0,则当前控件没有焦点。

另外,在"Focus movement rule"栏中,还可以设定特殊的移动规则,方法和画面属性设定控件中类似,详见"屏幕焦点移动规则"。

最后,选择 "Don't move focus to the next control after data input operation."可停止连续输入功能(配合"连续输入"功能使用)。

10. FANUC PICTURE 制作例子

本章介绍一个简单的例子制作方法,进一步的研究可以参考 FANUC PICTURE 自带例 子" FP_Exercise_ENG",进行实际操作练习。例子的路径是:"...\FANUC PICTURE\Sample\FP_Exercise_ENG\"。

| 创建一个新工程

系统类型可以选择 18i 或 30i 系列,本例使用 18i 系列,设定项目如下。如果要在非触摸屏上使用,请注意设定方框中的内容。

ion
_
ł

再选择多语言界面,设定简体中文和英语。如图:

Setting of project	X
Setting of CNC system Setting of Multi-language Collection of history	
The Multi-language display function Char. code SIMPLIFIED	
The Multi-language Character Type f	
Method of switching Multi-language display The display language parameter of CNC is used.	
The Multi-language display switch signal is used. (1B)	
PMC Area 0 : R 🗸 Symbol:	
PMC Address 0	
No Select Mu Param No. Number Comment Note Char. code	
02 1 CHS 3190#6 0 02 SIMPLIFIED	

| 创建主画面

选择新建一个画面,可自定义名称或使用默认名称 FPicture1,使用该画面作为工程的 启动画面。整个画面包括一张图片、一个方框、一行文字、一个时间显示控件以及底部的一 排按钮。考虑到按钮和时钟在每个画面中都会使用,可以使用子画面方式制作。因此,我们 把这个画面分为两部分,主画面和一个子画面,其中子画面可以在整个工程中可以多次使用。 合成后的结构框图如下:



制

| 设定画面结构控件

通常在左上角使用结构定义控件,在该控件中设定子画面,如图:

	PScrn 🛛
	General Base screen Sub screen Option
· · · · · · · · · · · · · · · · · · ·	Sub Screen 1
	♥ Use Sub Screen 1: Save SubScreen1: Free Memory:
	SubScreenName1: sub1
· · · · · · · · · · · · ·	Sub Screen 2

| 设定其他部分

其他部分使用了标签控件 🗚 定义换颜色的外框,图片显示控件 📝	定义中间显
示的图标,以及字符显示工具 🔼 定义画面中的文字。	

| 按钮子画面的制作

子画面的制作方式和主画面相同,子画面不需要画面结构定义控件,直接新建一个画面 修改名称即可。如图:



作。切换按钮1的控件设定示例如下,如果需要在非触摸屏使用,需要设定方框中的内容:

	Action Image	NoAction	Image Interlock	s Image	
NoAction:			Blink PM	IC	
ScreenType:	Base Screen	*	Blink:		
Screen Noor	FPicturel		BlinkPMCArea:	0 : R	~
Key Code.	CAN		BlinkPMCAddres	0 23	
🗹 Shortcut Kej	F1/F2B	~	BlinkPMCBit:	0	V
	-		Symbol:		~
Interlock			Light PN	IC	
Interlock			🗌 Light:		
InterlockPMCAre	0 : R	~	LightPMCArea:	0 : R	\sim
InterlockPMCAddre	0		LightPMCAddres	s 0	
	D	~	LightPMCBit:	0	V
InterlockPMCBit			Symbol ·		V
InterlockPMCBit		~			

弹出画面的制作

L

弹出画面使用的前面讲义中的例子,在按钮和指示灯演示画面的弹出效果如图:

		2008/08/07 11:56
		· · · · · · · · · · · · · · · · · · ·
· · · · · · · · · · · · · · · · · · ·		· · · · · · · · · · · · · · · · · · ·
· · · · · · · · · · · · · · · · · · ·		
	Deserves Hitselawa	
	l von-up window	
		.
		· · · · · · · · · · · · · · · · · · ·
	C1	
	Llear	
w p		<mark> </mark>
		📕
		• • • • • • • • • • • • • • • • • • •
		• • • • • • • • • • • • • • • • • • •
	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·
	· · · · · · · · · · · · · · · · · · ·	
(1 2	3 4 5 6 7 8	POPUP POS >

| 设定画面结构控件属性

弹出画面的位置和大小就是画面结构属性覆盖的位置和大小,而且需要选择"Use Popup Screen"属性。如下:

		Pop	-up W	ind	low			
			Clea	r				
PScrn			Y				[3
Genera Scree	1 Base scr en Name: - OUT PMC	een Sub	screen Optio		✔ Free Memor	y:		
Us 🗌	eOutPMC:				🕑 Use Popup :	Screen		· · · ·
OutPh	ICArea:	0 : R		~				
OutPh	ICAddress:	0						
OutPh	ICBit:	0		~				
Sy	mbol:			V				

设定弹出画面擦除按钮

L



| 设定弹出画面启动按钮

启动按钮放到了 10.3 制作的子画面中,具体设定如下:

2 3	4 5 6	7 8	POPUP	POS
hgScrn				
General Character	r Action Image NoAction	Image Interlock	Image	
NoAction:		Blink PMC		
ScreenType:	Base Screen 💌	Blink:		
Screen	popup	BlinkPMCArea:	0 : R	*
Key Code:	SYSTEM	BlinkPMCAddress	0	
Shortcut Kej	F9/F10J 💙	BlinkPMCBit:	0	~
		Symbol:		~

| 按钮指示灯演示画面

这个画面同样使用主画面和按钮子画面合成完成,子画面部分和 10.3 部分的一样,不 再重复,主画面部分如下:

· · · · · · · · · · · · · · · · · · ·	
· · · · · · · · · · · · · · · · · · ·	
	按钮、指示灯便不
N	
• • • • • • • • • • • • • • • • • • •	· · · · · · · · · · · · · · · · · · ·
M	
	nonmentary
· · · · · · · · · · · · · · · · · · ·	•••••••••••••••••••••••••••••••••••••••
	· · · · · · · · · · · · · · · · · · ·
· · · · · · · · · · · · · · · · · · ·	
\sim	
	Alternate
• • • • • • • • • • • • • • • • • • •	• • • • • • • • • • • • • • • • • • • •
	· · · · · · · · · · · · · · · · · · ·
e e e e e e e e e e e e e e e e e e e	
In the M	
	[:::: Watch ON / OFF [:::: (*)]::::::::::::::::::::::::::::::::::::
errererere in the second in the second se	
· · · · · · · · · · · · · · · · · · ·	• • • • • • • • • • • • • • • • • • • •
	· · · · · · · · · · · · · · · · · · ·
••••••••••••••••••••••••••••••••••••••	
270	
	Color Color
· · · · · · · · · · · · · · · · · · ·	• • • • • • • • • • • • • • • • • • • •
	· · · · · · · · · · · · · · · · · · ·
I Monmentary 类型按钮

	Button					
	General Charact	er Action	Image	NoAction I	mage Interlock	(Image
	NoAction:					
	Switch Type	Momentary	*			
M	Action Type	Bit Set	v D	etail		
	Shortcut Ke	FL/F1A	~	Detail[1	omentary]-[Bit Set] 🔀
A	Watch ON/O	FF		PMC Area:	12:E 🗸 🗸	ОК
: : :	Watch:	AUTO	~	Address:	20	Cancel
	Watch PMC Area	12 : E	~	Bit:	0 🗸	
M	Watch Address:	23		Sumbol		
	Watch Bit:	0	~	Bit Pattern:	2. Payment and	
220	Symbol:		~		2: Hevers V	

│ Alternate 类型按钮

	Button						
	General Charact	er Action	Image	NoAction Imag	ze Interloci	k Image	
· · · · · · · · · · · · · · · · · · ·	NoAction:						
M	Switch Type	Alternate	*				
	Action Type	Bit Set	v D	etail			
	Shortcut Ke	FL/F1A	~				
A	Watch ON/O	FF		1	interlock		
	Watch:	AUTO	~	Int	erlock		
M	Watch PMC Area	0 : R	~	Interl	.ockPMCAre	0 : R	~
:f :	Detail[Alter	nate]-[B	it Set	:]			
anol	ON Action			OFF Action		- (OK
220	PMC Area:	12:E	~	PMC Area:	12 : E	▼ (Cancel
	Address:	20		Address:	20		
	Bit:	1	~	Bit	1	~	
	Symbol:		~	Symbol:		~	
	Bit Pattern:	1:1	~	Bit Pattern:	0:0	~	

| 指示灯的制作

本例中的指示灯用于显示检测按钮状态,地址对应均为按钮地址,以 Monmentary 类型 按钮为例,如图:

Monmentary	- 0	Y Width Height No use	t e m
Lanp			
General Charac	ter Action Image	Option NoAction Image	
NoAction:			191
Watch PMC Area	: 12 : E	~	
Watch Address:	20		
Watch Bit:	0	~	
Symbol:		~	

│ Watch ON/OFF 用法

"W"按钮通过 Alternate 方式设定 PMC 地址 E23.0,再通过 E23.0 控制相邻 "M"按钮的 ON/OFF 状态。"W"按钮示例如下:

	Button
	General Character Action Image NoAction Image Interlock Image
	NoAction:
· · · · · · · · · · · · · · · · · · ·	Switch Type Alternate 💙
	Action Type Bit Set 💙 Detail
• W •	Shortcut Ke: FL/F1A
Metter Metter	Detail[Alternate]-[Bit Set]
· · · · · · · · · · · · · · · · · · ·	ON Action OFF Action OK
	PMC Area: 12: E Cancel
· · · · · · · · · · · · · · · · · · ·	Address: 23 Address: 23
	Bit: 0 💌 Bit: 0 💌
	Symbol: Symbol:
	Bit Pattern: 1:1 V Bit Pattern: 0:0 V

" M " 按钮和 10.4.1 中的 " M " 差不多 , 只是设定地址不同 , 检测 ON/OFF 的方式不同 , AUTO 方式变为 USER 方式。

	Button				
M	General Charact	er Action Image	NoAction I	mage Interlock	t Image
	NoAction:				
:f:	Switch Type	Momentary 💟			
A	Action Type	Bit Set 🔽	Detail		
	Shortcut Ke	FL/F1A 🗸	Detail 🔳	omentary]-[)	Bit Set] 🔀
M	Watch ON/0	FF	PMC Area:	12:E 🗸 🗸	ОК
Annual Contraction of the second s	Watch:	USER 💌	Address:	20	Cancel
220	Watch PMC Area	12 : E 💌	Bit:	2 💌	
	Watch Address:	23	Symbol:		~
· · · · · · · · · · · · · · · · · · ·	Watch Bit:	0 💌	Bit Pattern:	2 : Revers 🗸	
	Symbol:	~			

I 7 Color 用法

首先,使用十键控件输入一个字节的数据到 E21。

12.34	TInput
· · · · · · · · · · · · · · · ·	General Character Action Action2 Focus Image Option
	NoAction:
· · · · · · · · · · · · ·	Select 0: Tenkey1
	✓ No use multi-language
M	Multi-language ENG
	Ten-key
;f:	ReadFunction: rdpmc[12_21_0]
A	WriteFunction: wrpmc[12_21_0_0]
:ł :	CheckFunction:
M	In / Out
	Data Type: 1 : 1 Byte 💌 🗌 Left Align:
	In/Out Format 🛛 : Decimal 💌 🗌 Non zero suppress:
• ZU •	Decimal digi 🛛 🗌 Use Decimal Point
2 anno anno 2	Max. value: 60 🗹 Set max. and min. value
· · · · · · · · · · · · · · · · · · ·	Min. value: 0
	✓ Number of digits specif: 2
	Identification 0
	Use focus index for the identificatio:

然后,使用指示灯控件,选择 7 Color 方式,即可根据 E21 中不同位的状态显示 7 种不同的颜色。

amp	2
General Character Action Image Option	NoAction Image
✓ Use 7Colors:	DataLength: 1 : 1 Byte
Type: 1 : PMC (Compare large an 💙	UseBlinkSignal:
SignalArea 12 : E 💙	BlinkSignalArea: 0 : R 🗸
SignalAddr(21	BlinkSignalAddre: 0
SetUpNumber: 7	BlinkSignalBit: 0
Symbol:	Symbol:
BitPosition: #6: #5: #	#4: #3: #2: #1: #0:
LightColor:	
Character Color:	
ChangeBeginningValu60 50 40	0 30 20 10 0
FunctionName	~
FunctionBlink No use blink 💌	

| 输入控件演示画面

这个画面同样使用主画面和按钮子画面合成完成,子画面部分和 10.3 部分的一样。这 个画面包括了各种输入控件(普通、十键),以及不同输入内容(数字、字符)等。主画面 部分如下:

 输入控件演示							
数字 / 字符显示	E32 一个字节	ZZØ	0				
TEN-KEY 输入控件	ZZ 0	ZZ0	220.000				
数值输入控件	220	字符输入控件	HHH				
 Input Bu	ffer						
	1234	56789					

| 数字/字符显示控件

这个控件仅用于显示,修改 Display 中的 Data Type 可显示数字或 ASCII 字母。该控件 没有输入功能。

	NDisp			
	General Character	Action Image Option	NoAction Image	
	NoAction:			
770	Action Type:	0 : PMC 💙	Display	
220			Data Type:	1 : 1 Byte 💌
	PMC		Format:	0 : Decimal 💌
220	Watch PMC Area:	12 : E 💌	🗌 Use Decimal Poix	
	Watch Address:	36	Decimal digi	0
	Symbol:	~	Left Align	Non zero suppress:
字符输入控件	The stress		✓ Number of digits	specificatio3

| 十键输入控件

输入一个 1 字节带符号的整数到 PMC 地址 E31,如图:

	TInput							
12.34 9.99 🗛 🛛	General Character	Action .	Action2 Fo	cus Ima	age	Option		
	NoAction:							
	Select	0: Tenkeyl		*				
输入控	No use multi-l	language						
	Multi-language	CHS		*				
	Ten-key							
E32 一个字节	ReadFunction:	rdpmc[12_3	1_0]					~
	WriteFunction:	wrpmc[12_3	1_0_0]					*
220	CheckFunction:							~
<u></u>	In / Out							
220	Data Type:	0 : 1 By	te (Sig 💙	Left	Alig	n:		
	In/Out Format	O : Deci	mal 💙	Non	zero	suppress:		
	Decimal digi	0		🗌 Vse 🛛	Decim	al Point		
ffer	Max. value:	127		🖌 Set	max.	and min. v	alue	
12345	Min. value:	-128						
12313	✔ Number of digi	its specif:	3					
	Identification		0					
	Use focus inde	ex for the :	identificat:	io:				

输入一个带 3 位小数的数值到变量#500,如图:

10. FANUCPICTURE 制作例

IInput	×	
General Characte	r Action Action2 Focus Image Option	
NoAction:		
Select	2: Tenkey3 💌	
🗌 No use multi-	language	
Multi-language	CHS	
Ten-key		
ReadFunction:	rdmacro[500_99_0]	0
WriteFunction:	wrmacro[500_0_0_99_0]	
CheckFunction:		770 000
In / Out		220.000
Data Type:	7 : Double 🔽 🗌 Left Align:	
In/Out Format	0 : Decimal 🔽 Non sere suppress:	HHH
Decimal digi	3 Use Decimal Point	
Max. value:	127 Det mex end min. value	
Min. value:	-128	
✓ Number of dig	its specif: 8	
Identification	0	
Use focus ind	ex for the identificatio:	· · · · · · · · · · · · · · · · · · ·

| 使用输入缓冲方式

这种方式可以输入数值或者 ASCII 字符。首先需要创建一个输入缓冲控件,用于显示 当前来自 MDI 面板输入的信息。

12.34 PMC 12.34 J	KyInBf				×
	General Chara	acter Action	Image		
	Screen Int	erlock			
	PMC			Message	
· · · · · · · · · · · · · · · · · · ·	PMC Area:	0 : R	~	No use multi-language	
数字 / 字符显示 E32 一个	Address:	0		Multi-language key:	
	Bit:	0	*	ENG 💟	
ен-кеү 输入控件 乙	Symbol:	digits speci	fica	Interlock Message:	
	9			Character Col	
数值输入控件 220					
T (D 00					
Input Buffer					
	L <mark>23456</mark> 7	789		Bo Dis	splay s

例 1:输入一个数值到 PMC 地址 E30,如图:

	Input
	General Character Action Focus Interlock Image
	NoAction:
示 E32 一个字节	In/Out 0 : PMC V ReadFunction:
	PMC Area: 12 : E 💙 WriteFunction
	Address: 30 CheckFunction:
	Symbol:
ZZO	Data Type: 0 : 1 Byte(Signv 🛩 In/Out Format 0 : Decimal 🗸
	Use Decimal Point 0 Left Align:
Buffer	✓ Number of digits specifi 3 Non zero suppress:
422	Identification 0 Set max. and min. value
123	Use focus index for the identification m Max. value: 127
	Popup Screen Min. value: -128
	· · · · · · · · · · · · · · · · · · ·

例 2: 输入 3 个字符到 PMC 地址 E34、E35、E36。

PECStrDisp		
General Character	Action Focus Image	
PMCMessageArea:	12 : E V Interlock:	
PMCMessageAddress:	34 InterlockPMCArea: 0 : R 🗸	
PMCMessageBit:	0 InterlockPMCAddres 0	0
Symbol:	InterlockPMCBit: 0	
PMCCharCount:	3 Symbol:	220 000
LookSignal:	Popup Screen Name:	
SignalArea:	0 : R 🗸 Vse Key-in-buffer	HHH
SignalAddress:	0	
SignalBit:	0	
Symbol:	\checkmark	
Identification	0	
Use focus index	for the identificatic	
	· · · · · · · · · · · · · · · · · · ·	

| 函数的使用和报警信息的显示

关于 FP 函数的用法,前面的例子中已经使用过了许多,在使用中需要注意设定控件的 数据类型和函数定义要求的数据类型一致。例如:读取系统坐标,每一个轴的坐标值都是4 字节带符号;读取变量要求浮点型数据;读取报警信息则要求字符串型数据。主画面如下:

	· · · · · · · · · · · · · · · · · · ·
FP 函数演示、报警信息	
	#100 2222220 000
第1路径 X轴绝对位置写入#100	#101 22222220.000
X_{1} 22220.000 22220.000	#102 22222220.000 #103 22222220.000
第2路径 X轴绝对位置	#104 22222220.000 #105 22222220.000
X 22220.000	#106 22222220.000
· · · · · · · · · · · · · · · · · · ·	#110 22222220.000 #111 22222220.000
报警信息	
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS	S55555555565555

| 读取绝对位置坐标

如果是第1路经,命令 "absolute[1_2]"结尾的 "_2"可以省略。

	NDisp			
	General Character	Action Image Option	NoAction Image	
第1路径 X轴绝对位置	NoAction:			
	Action Type:	1: Function 💌	Display	
X_1 ZZZZO.			Data Type:	4 : 4 Byte (Signe 💙
· · · · · · · · · · · · · · · · · · ·	PMC		Format:	O : Decimal 💌
第2路径:X轴绝对位置	Watch PMC Area:	0 : R 💌	🔽 Use Decimal Poin	
V OCCCC V	Watch Address:	0	Decimal digi	3
$\mathbf{X}_2 \mathbf{Z} \mathbf{Z} \mathbf{Z} \mathbf{Z} \mathbf{U}$	Symbol:	~	Left Align	Non zero suppress:
			Number of digit:	s specificatio 10
报警信息	Function			
555555555155555	Function Name:	Display:		
	absolute[1_2]			~

| 读取系统变量

读取变量的小数点位数应当根据系统实际情况设定。

General Character	Action Image Option	NoAction Image		
NoAction:				#199 2222229 99
Action Type:	1: Function 💌	Display		#101 27777720.00
		Data Type:	7 : Double 💌	#102 22222220,00
PMC		Format:	O : Decimal 🛛 💙	#103 22222220.00
Watch PMC Area:	0 : R 😽	🗸 Use Decimal Poi	1	#104 22222220.00
Watch Address:	0	Decimal digi	3	#105 22222220.00
Symbol:	~	Left Align	Non zero suppress:	#106 22222220.00
		Wumber of digit	r manification	#110 22222220.00
		Mumber of digit	s specificatio 1	#111 22222220.00
Function Function Name:	Display:			· · · · · · · · · · · · · · · · · · ·
rdmacro[100 99 (0]		~	5555555555565555

| 读取报警信息

报警信息的数据类型应设为 String,可显示简体中文。

Disp				
General Character	Action Image Option	NoAction Image		
NoAction:				
Action Type:	1: Function 💙	Display		
		Data Type:	6 : String	*
PMC		Format:	0 : Decimal	~
Watch PMC Area:	0 : R 🗸	🗌 Use Decimal Poi	i.	
Watch Address:	0	Decimal digi	0	
Symbol:	~	🗹 Left Align	Non zero suppr	ess
		Number of digit	ts specificatio:1	
Function				
Function Name:	Display:			
rdalminfo[1_99_1_	99_0]			*
		一時中		
		uil Z		ti (A
SSSSSSSS155	555555525555555555555555555555555555555	35555555554555555	555555555555555555555555555555555555555	555