

FANUC 机器人

IO 数据 karel 程序传输指导手册

| 修订 | 姓名 | 日期 |
|----|----|-----------|
| 编制 | 匡宏 | 2022/2/23 |
| 审核 | | |
| 批准 | | |

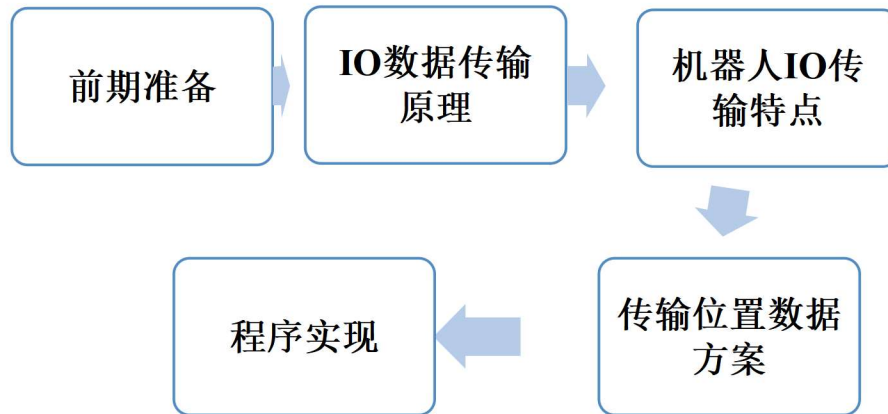
| 版本 | 更新内容 | 日期 | 姓名 |
|------|------|-----------|----|
| V1.0 | 首次发布 | 2022/2/23 | 匡宏 |
| | | | |
| | | | |

目录

| | |
|----------------------------------|---|
| 1、概述..... | 3 |
| 2、作业步骤..... | 3 |
| 2.1 确认通讯软件安装..... | 3 |
| 2.2 IO 数据传输原理..... | 3 |
| 2.3 PLC 与发那科机器人 IO 数据传输特点..... | 3 |
| 2.4 PLC 与发那科机器人 IO 传送实时位置数据..... | 3 |
| 2.5 PC 程序导入..... | 4 |
| 2.6 程序参数说明..... | 5 |
| 2.7 程序实现..... | 6 |
| 3、常见报警..... | 8 |
| 4、附录..... | 9 |
| 4.1 KAREL 程序..... | 9 |

1、概述

在 Fanuc 机器人应用中，如果通讯 IO 点位足够，可以利用机器人 IO 传输位置数据及偏移位置数据等。



2、作业步骤

2.1 确认通讯软件安装

- 1) 首先确认机器人控制柜已经安装了总线通讯软件（如 EIP,Profinet），相应的 IO 端口通讯已经配置完成，已经配置相应的 GI 与 GO，GI 用于机器人数据输入，GO 用于机器人数据输出。

2.2 IO 数据传输原理

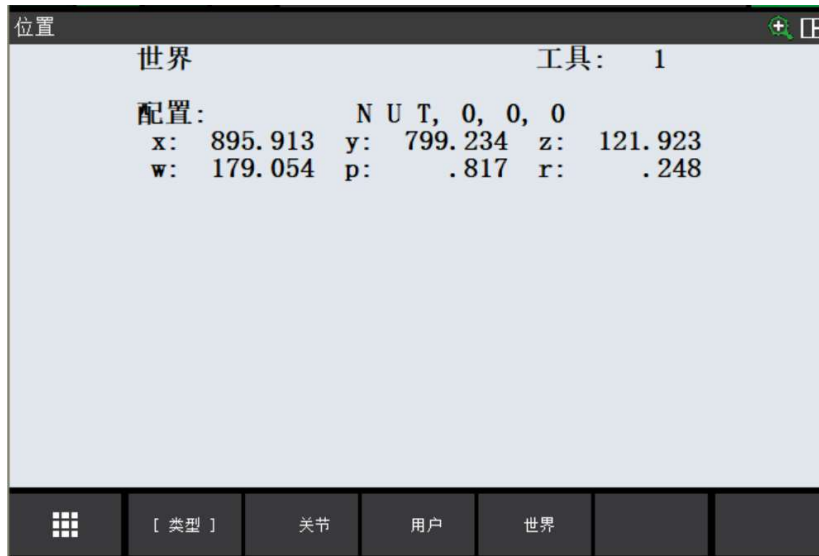
- 1) 计算机通讯中数据是以补码的形式进行传输的。正数和零的补码是其二进制表示，与原码相同。负数的补码是将其原码除符号位（首位）外的所有位取反（0 变 1，1 变 0，符号位为 1 不变）后加 1。以 8 位二进制数为例进行说明：+9 的原码是 00001001，补码也是 00001001。-9 的原码是 10001001，补码是 11110110 加 1，即 11110111。

2.3 PLC 与发那科机器人 IO 数据传输特点

- 1) PLC 与外部数据 IO 传输，无论发送与接收的数据都是补码，传输的数据必须以整数的形式发送。如果数据不是整数则要乘以一定的倍数，将小数转换成整数。机器人通过 GI/GO 传输数据，GI 接收的数据是以无符号数进行显示；而将数据寄存器 R 值赋值给 GO 时，GO 会自动转换成相应的补码，无需人为转换。因此，PLC 与发那科机器人 IO 进行数据传输时，机器人只需要将收到的 PLC 数据进行正负区分和转换。发那科机器人组信号 GI/GO 每组最多只能设置 16 个点数。16 个信号组成的 16 位数据传递的有符号数范围是-32767~+32767。

2.4 PLC 与发那科机器人 IO 传送实时位置数据

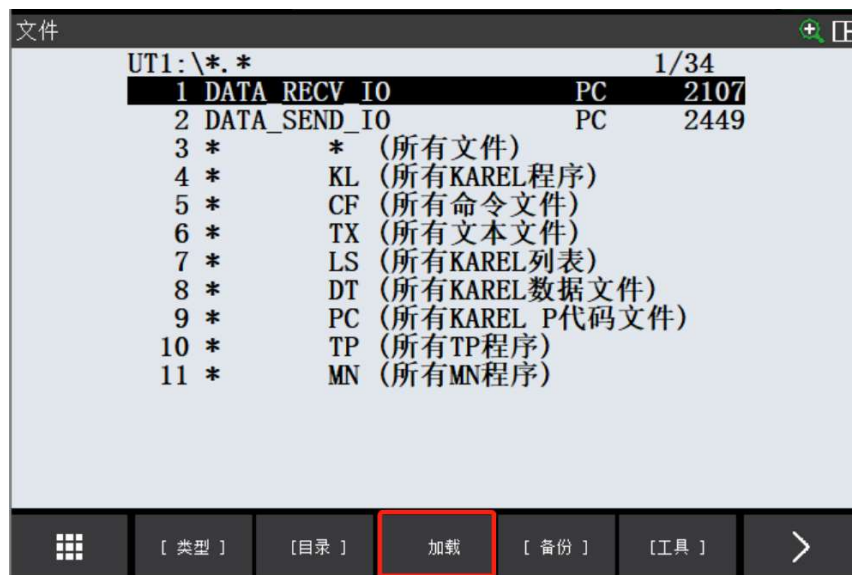
- 1) 为了保证机器人运行满足实际精度要求，机器人数据必须保留到小数点后三位，如下图所示。如果采用将机器人位置数据乘 1000 扩大 1000 倍的方法，那么每个 GI 或 GO 传输的数据实际大小范围则为-32.767~+32.767，很明显不满足实际要求。为了避免受到 GI/GO 只有 16 位点位数据的限制，可以采用 DATA_SEND_IO 与 DATA_RECV_IO 数据进行传输。



2.5 PC 程序导入

1) 将两个 PC 程序 (DATA_SEND_IO.pc, DATA_RECV_IO.pc) 存入 U 盘或者 CF 卡, 按以下操作步骤执行文件导入:

- a. 按[Menu]键
- b. 选择[文件], 进入文件页面, 继续选择[文件]
- c. 选择切换设备, 并找到 UT1(此处以 USB 存储器插在示教器上为例), 然后按 F3 对相应的程序进行加载:



2) 选择[MENU]-[下页]-[系统]-[变量], 将\$KAREL_ENB 变量置为 1。

```

系统变量
311/823
308 $KANJI_MASK      0
309 $KAREL_MON       KAREL_MON_T
310 $KAREL_CFG       KAREL_CFG_T
311 $KAREL_ENB       1
312 $KCL_LIN_NUM     TRUE
313 $KCL_RPCOUT      *uninit*
314 $KEYLOGGING      2
315 $LANGUAGE        'DEFAULT'
316 $LASTPAUSPOS     [8] of JOINTPOS9
317 $LGCFG           LGCFG_T
318 $LN_DISP         LN_DISP_T
    
```

3) 选择[SELECT]-[类型(F1 键)]-[KAREL 程序], 检查程序是否正确加载。

```

KAREL程序 695340 字节可用 3/19
编号 程序名 注释
1 AAVMMAIN PC [ ]
2 COMSET PC [ ]
3 DATA_RECV_IO PC [ ]
4 DATA_SEND_IO PC [ ]
5 EN_AROFF PC [ ]
6 EN_ARON PC [ ]
7 EN_OFFLN PC [ ]
8 EN_ONLN PC [ ]
9 EN_QCHK PC [ ]
10 EN_QCOFF PC [ ]
    
```

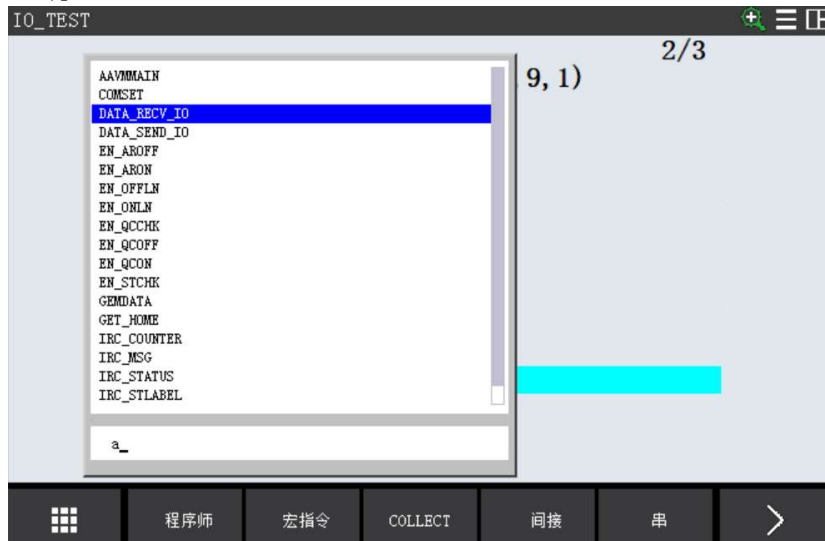
2.6 程序参数说明

| 程序名 | 文件名 | 说明 |
|--------------|-----------------|--|
| DATA_SEND_IO | DATA_SEND_IO.pc | 将数值寄存器值扩大 1000 倍然后取整向 PLC 发送 参数说明： 参数 1 (INT) :起始输出 DO 号 参数 2 (INT) :末端输出 DO 号 最多只能有 32 个 IO 进行数据传输，参数 1 是最低位，参数 2 是最高位。 参数 3 (INT) :数值寄存器 R 号，寄存器号必须在 50 以内。 举例：DATA_SEND_IO(1,32,2) 机器人将 2 号数值寄存器的值扩大 1000 倍然后取整，将取整后的数据通过 DO1-DO32 以补码的型式发送给 PLC，其中 DO[32]发送的是符号位。 |
| DATA_RECV_IO | DATA_RECV_IO.pc | 将接收的 PLC 数据缩小 1000 倍，然后将缩小 1000 倍的数据存放到相应的寄存器。 参数说明： 参数 1 (INT) :起始输入 DI 号 参数 2 (INT) :末端输入 DI 号 最多只能有 32 个 IO 进行数据传输，参数 1 是最低位，参数 2 是最高位。 |

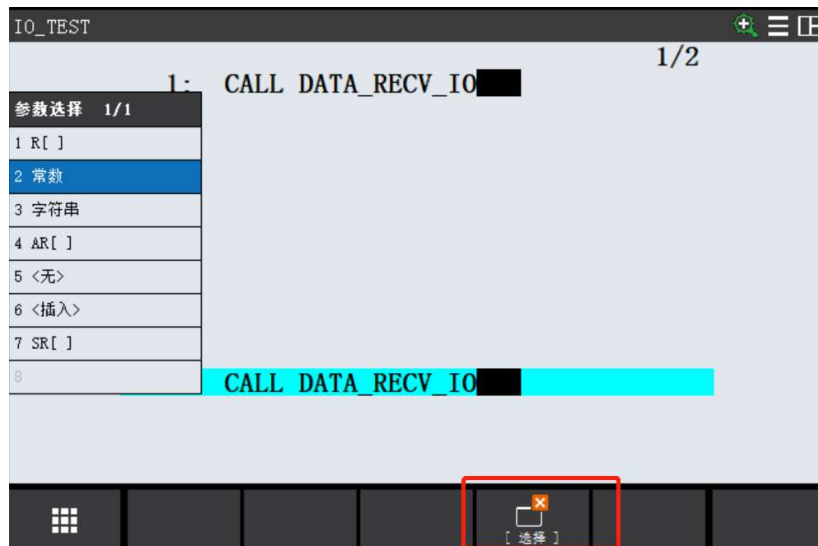
参数 3 (INT) :数值寄存器 R 号, 寄存器号必须在 50 以内。
 举例: DATA_RECV_IO(1,32,3)
 机器人将通过 DI1-DI32 收到的数据缩小 1000 倍, 然后将缩小 1000 倍的数据存放到数值寄存器 3, 其中 DI[32]是符号位

2.7 程序实现

- 1) 以调用 DATA_RECV_IO 为例, 在 TP 程序中相应位置调用程序。选择[NEXT]-[指令]-[调用]-[调用程序]-[COLLECT(F3 键)]/ENTER 键, 然后选择对应的程序。



- 2) 将光标移动到程序最后端, 然后按选择 (F4 键), 选择常数, 输入对应的参数, 重复上述步骤, 输入剩余参数。



- 3) 程序示例

```
IO_TEST 1/10
1: !将DI1-DI32收到的数据存入寄存器R1
2: !R1存储的为X坐标数据
3: CALL DATA_RECV_IO(1, 32, 1)
4: !将DI33-DI64收到的数据存入寄存器2
5: !R2存储的为Y坐标数据
6: CALL DATA_RECV_IO(33, 64, 2)
7: !将DI65-DI96收到的数据存入寄存器3
8: !R3存储的为Z坐标数据
9: CALL DATA_RECV_IO(65, 96, 3)
[End]
```

3、常见报警

| |
|--|
| |
| |

4、附录

4.1 KAREL 程序



DATA_RECV_IO.pc



DATA_SEND_IO.pc