

# FANUC 机器人

## User Socket Message 指导手册

修订	姓名	日期
编制	梁海坚	2022/02/23
审核		
批准		

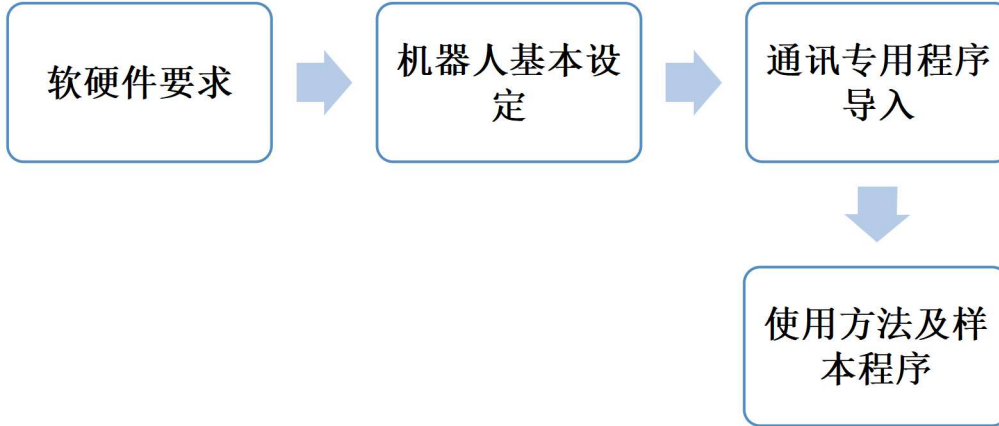
版本	更新内容	日期	姓名
V1.0	首次发布	2022/02/23	梁海坚

## 目录

1、概述.....	3
2、机器人软硬件要求.....	3
2.1 硬件要求.....	3
2.2 软件要求.....	3
3、机器人基本设定.....	4
3.1 IP 地址设定.....	4
3.2 客户端 Client 设定（机器人作为服务器时，请忽略本处设定步骤）.....	4
3.3 服务器 Servers 设定（机器人作为客户端时，请忽略本处设定步骤）.....	5
4、通讯专用程序导入.....	6
4.1 专用通讯程序说明.....	6
4.2 专用通讯程序导入.....	6
5、使用方法及样本程序.....	7
5.1 程序说明.....	7
5.2 在 TP 程序中调用通讯专用程序.....	9
5.3 实际使用案例.....	9
6、常见报警.....	10
6.1 INTP-311.....	10
6.2 INTP-366.....	10
6.3 APSH-000.....	10
6.4 APSH-000.....	10
6.5 HOST-212.....	10
6.6 HOST-213.....	10
4、附录.....	11

# 1、概述

User socket message 功能又简称为 SMG 功能，是 FANUC 机器人与工控机等外部设备通过 TCP/IP 协议进行通讯连接的一种常用功能。SMG 具有可靠性高、传输格式自由、可传输数据类型多等优点。其主要步骤如下：



# 2、机器人软硬件要求

<b>2.1 硬件要求</b>
1) 普通网线。
2) MC 卡或者 USB 存储器。

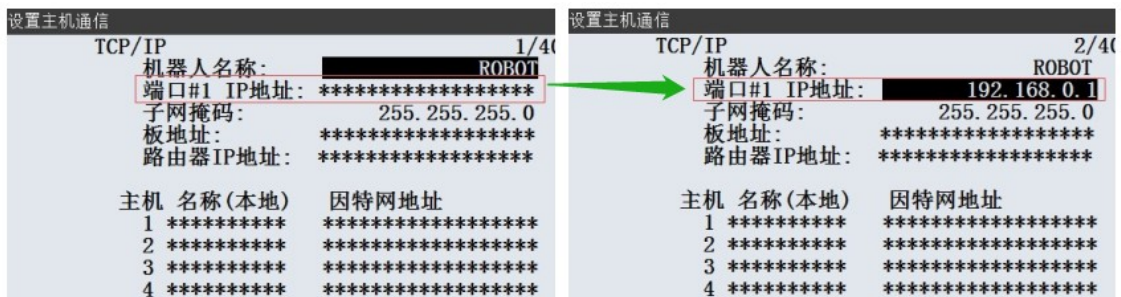
<b>2.2 软件要求</b>
<p>1) 需要选装 User Socket Msg 软件（R648）。          可通过[MENU]-[NEXT]-[状态]-[版本 ID]-[NEXT]-[订购文件]中查看，如下图所示。</p> <div style="border: 1px solid gray; padding: 10px; background-color: #f0f0f0;"> <pre> 1A05B-2600-R641 ! PC Interface 1A05B-2600-J541 ! Password Protection 1A05B-2600-J669 ! Payload Ident. 1A05B-2600-J878 ! Payload confirm 1A05B-2600-R648 ! User Socket Msg 1A05B-2600-FVRC ! Virtual Robot 1A05B-2600-H755 ! LR Mate 200iD Z  按PREV键退出           </pre> </div>

### 3、机器人基本设定

#### 3.1 IP 地址设定

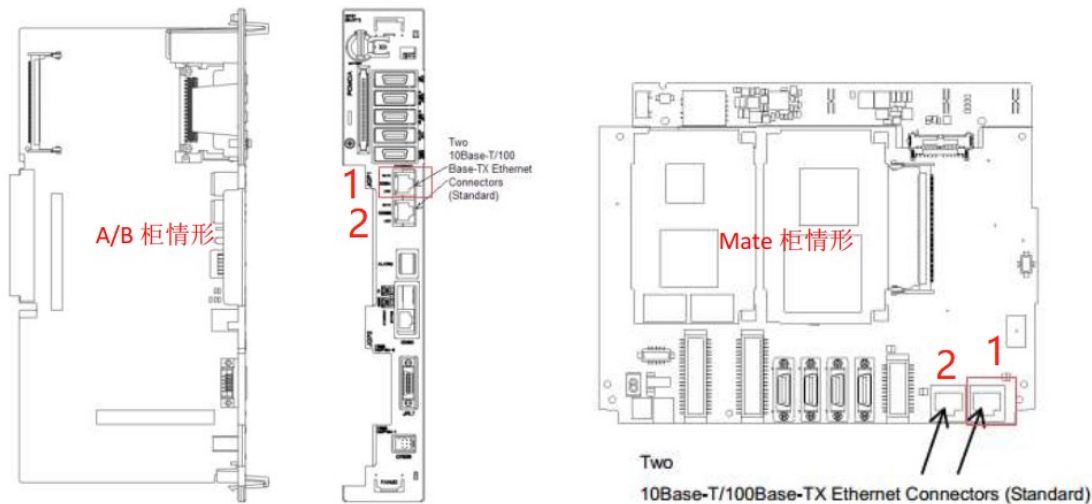
1) 操作步骤如下：

- ① 按下[MENU]；
- ② 选择[设置]；
- ③ 选择[主机通讯]；
- ④ 选择[1 TCP/IP]；
- ⑤ 在 TCP/IP 页面中，光标移至“端口#1 IP 地址：”处，按下 ENTER 键后输入机器人的 IP 地址，例如 192.168.0.1，输入完毕后以 ENTER 键结束；上位机的 IP 地址应与机器人的 IP 地址处于同一网段，例如上位机的 IP 地址可以设置为 192.168.0.2，如下图所示；



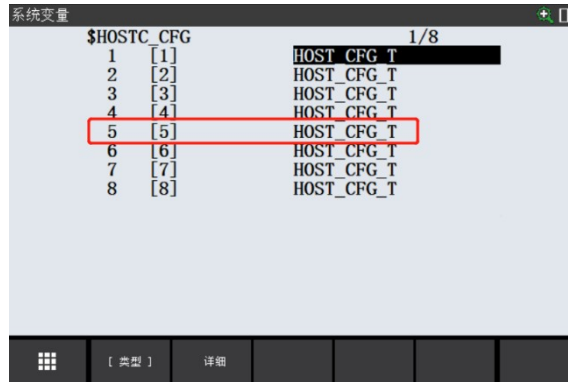
⑥ 设置完机器人 IP 地址后，重启生效；

⑦ 将网线插入机器人控制柜内，插入位置如下图所示：（端口#1 则插入 1 号网口）



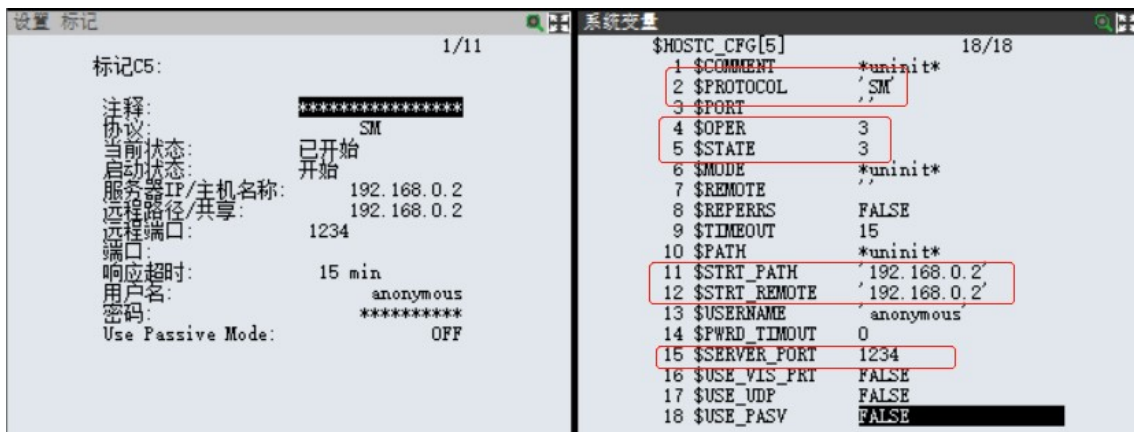
#### 3.2 客户端 Client 设定（机器人作为服务器时，请忽略本处设定步骤）

1) 找到以下变量：\$HOSTC\_CFG，在变量中可以看到，共有 8 个标签可以共用户选择使用，如下图所示，本文以[5]号标签作为举例说明；（注意不要搞错变量！）



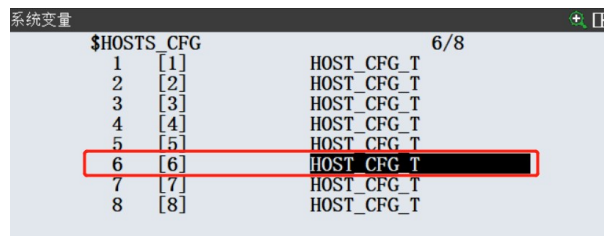
2) 光标移至[5]号标签，按下 ENTER 键进入，需要设定的地方包括以下内容：

- ①将\$PROTOCOL 改为“SM”（注意是大写）；
- ②将\$OPER 改为“3”；
- ③将\$STATE 改为“3”；
- ④将\$STRT\_PATH 改为上位机的 IP 地址，例如“192.168.0.2”；
- ⑤将\$STRT\_REMOTE 改为上位机的 IP 地址，例如“192.168.0.2”；
- ⑥将\$SERVER\_PORT 改为端口号，4 位非全 0 数字，例如 1234；
- ⑦设置完毕后重启机器人生效。



### 3.3 服务器 Servers 设定（机器人作为客户端时，请忽略本处设定步骤）

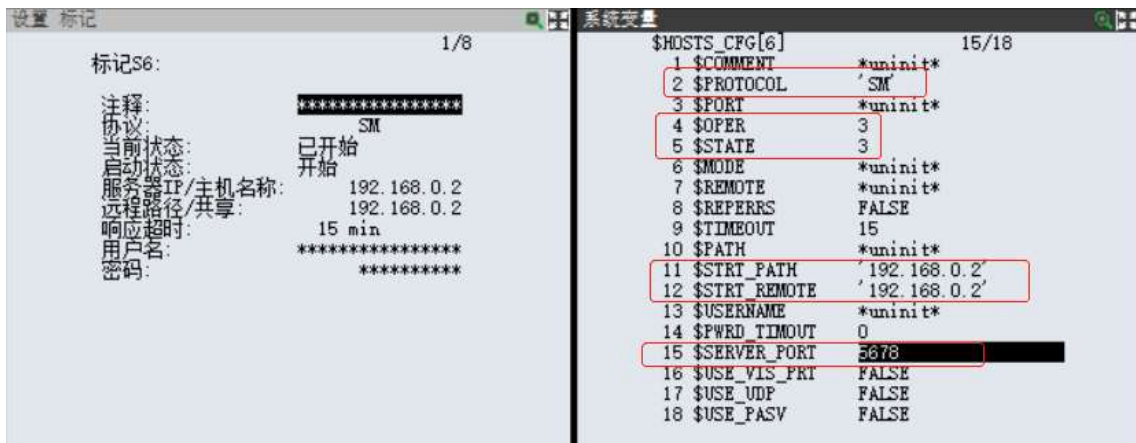
1) 找到以下变量：\$HOSTS\_CFG，在变量中可以看到，共有 8 个标签可以共用户选择使用，如下图所示，本文以[6]号标签作为举例说明；（注意不要搞错变量！）



2) 光标移至[6]号标签，按下 ENTER 键进入，需要设定的地方包括以下内容：

- ①将\$PROTOCOL 改为“SM”（注意是大写）；
- ②将\$OPER 改为“3”；
- ③将\$STATE 改为“3”；
- ④将\$STRT\_PATH 改为上位机的 IP 地址，例如“192.168.0.2”；
- ⑤将\$STRT\_REMOTE 改为上位机的 IP 地址，例如“192.168.0.2”；
- ⑥将\$SERVER\_PORT 改为端口号，4 位非全 0 数字，例如 5678；

⑦设置完毕后重启机器人生效。



## 4、通讯专用程序导入

### 4.1 专用通讯程序说明

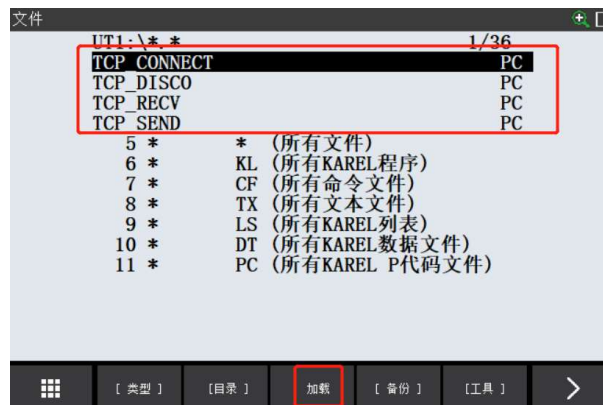
1) SMG 通讯通常会有 2 个或者 4 个专用的通讯程序，以下举例 4 个程序时的情形：

- ①通讯连接程序：tcp\_connect.pc;
- ②通讯断开程序：tcp\_disco.pc;
- ③接收上位机数据程序：tcp\_recv.pc;
- ④向上位机发送数据程序：tcp\_send.pc;

具体通讯程序名称和个数以实际为准，功能方面是一样的。

### 4.2 专用通讯程序导入

- 1) 将通讯专用程序放入 USB 存储器根目录：
- 2) 按照以下步骤进行操作：
  - ①按下[MENU];
  - ②选择[文件];
  - ③按下[F5 工具]，选择[切换设备]，选择“TP 上的 USB”；
  - ④将 U 盘插入示教器右侧 USB 插口；
  - ⑤浏览移到“\*”所在行，按下 ENTER 键，读取出 U 盘内的文件，如下如所示：

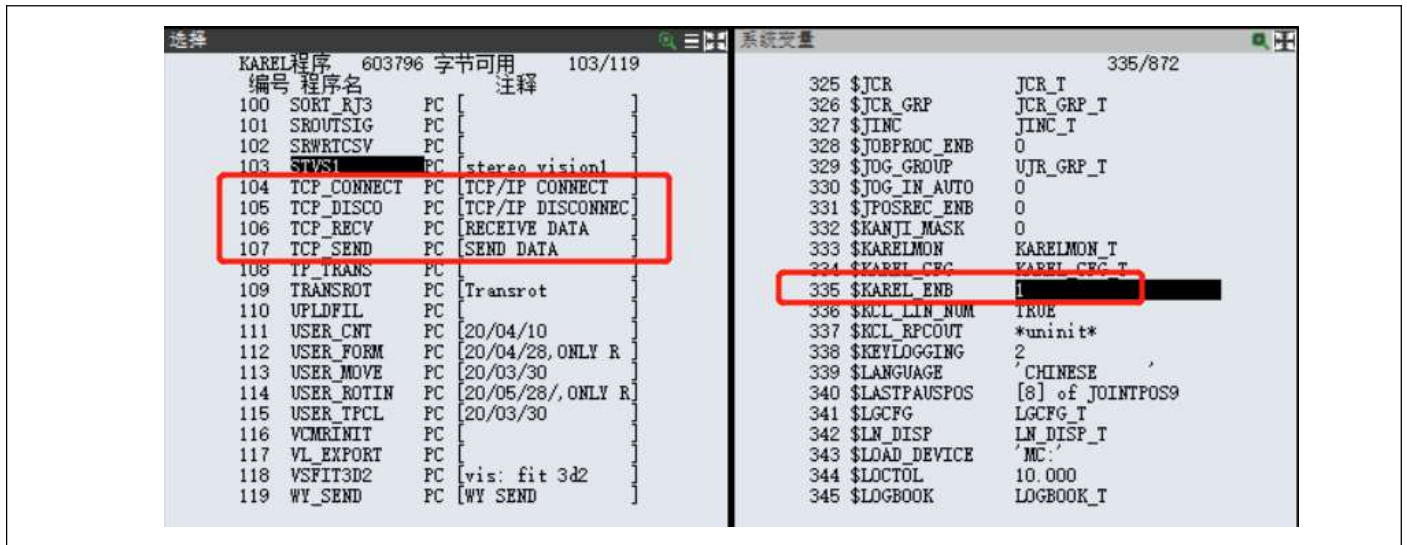


⑥依次选择每个程序，按下[F3 加载]，并确认加载；

3) 确认通讯专用 PC 程序已经加载。

按下 Select 键，按下[F1 类型]，选择“KAREL 程序”，如无此选项，则在系统-变量中将\$KAREL\_ENB 改为 1 即可，请确认一下 4 个程序已被成功导入，如下图所示：





## 5、使用方法及样本程序

### 5.1 程序说明

程序名	文件名	说明
TCP_CONNECT	TCP_CONNECT.pc	<p>机器人与上位机建立通讯连接</p> <p><b>输入参数:</b></p> <p><b>参数 1 (string)</b></p> <p>string: 通讯端口编号, 例如上述的设置中用了 C1 端口, 则此处写入 C1 即可。</p> <p>例: CALL TCP_CONNECT('C1')</p>
TCP_DISCO	TCP_DISCO.pc	<p>机器人与上位机断开通讯连接</p> <p><b>输入参数:</b></p> <p><b>参数 1 (string)</b></p> <p>string: 通讯端口编号, 例如上述的设置中用了 C1 端口, 则此处写入 C1 即可。</p> <p>例: CALL TCP_DISCO('C1')</p>
TCP_SEND	TCP_SEND.pc	<p>机器人发送给上位机字符串数据</p> <p>机器人可以给上位机发送 R、PR、SR 寄存器中的内容, 也可以发送指定值。发送的数据均以回车换行作为结束符。</p> <p><b>输入参数:</b></p> <p><b>参数 1 (int: 1-4)</b></p> <p>1: 机器人发送 R 寄存器里面的内容给上位机;</p> <p>2: 机器人发送 PR 寄存器里面的内容给上位机;</p> <p>3: 机器人发送 SR 寄存器里面的内容给上位机;</p> <p>4: 机器人发送参数 2 里面的内容给上位机。</p> <p><b>参数 2 (int 或 string)</b></p> <p>当参数 1 的值为 1 时, 参数 2 是 int 类型, 表示需要发送的 R 寄存器号码;</p>

			<p>当参数 1 的值为 2 时，参数 2 是 int 类型，表示需要发送的 PR 寄存器号码；                  当参数 1 的值为 3 时，参数 2 是 int 类型，表示需要发送的 SR 寄存器号码；                  当参数 1 的值为 4 时，参数 2 是 string 类型，表示机器人会将参数 2 的内容直接发给上位机；</p> <p><b>参数 3 (int)</b>                  当参数 1 是 1 时，表示需要发送 R 寄存器的个数，以参数 2 指定的寄存器号码开头，个数限制在 1-20 之间。<b>(数据格式注意：不同 R 寄存器值之间用逗号隔开)</b>                  当参数 1 是 2 时，表示需要发送 PR 寄存器的个数，以参数 2 指定的寄存器号码开头，个数限制在 1-5 之间。<b>(数据格式注意：同一 PR 内的 XYZWPR 用逗号隔开，不同 PR 寄存器值之间也用逗号隔开)</b>                  当参数 1 是 3 时，表示需要发送 SR 寄存器的个数，以参数 2 指定的寄存器号码开头，个数限制在 1-5 之间。<b>(数据格式注意：不同 SR 寄存器值之间用逗号隔开)</b>                  当参数 1 是 4 时，参数 3 无效。</p> <p><b>例：</b> CALL TCP_SEND(3,1,5)                  发送 SR[1]-SR[5]这 5 个字符串寄存器的内容给上位机。                  CALL TCP_SEND(4,'OK')                  直接发送 OK 给上位机。</p>	
TCP_RECV	TCP_RECV.pc		<p>机器人接收上位机字符串数据                  机器人可以把来自上位机的数据放到 R、PR、SR 寄存器中。要求上位机发来的数据以回车换行作为结束符。</p> <p><b>输入参数：</b>  <b>参数 1 (int: 1-4)</b>                  1: 把上位机发来的数据放到 R 寄存器中；上位机发来的 R 个数建议限制在 50 个以内。<b>(数据格式注意：不同数据之间用逗号隔开，结束符以回车换行结束即可)</b>                  2: 把上位机发来的数据放到 PR 寄存器中；上位机发来的 PR 个数建议限制在 5 个以内，6 个数作为一个 PR。<b>(数据格式注意：不同数据之间用逗号隔开，结束符以回车换行结束即可)</b>                  3: 把上位机发来的数据放到 SR 寄存器中；上位机发来的 SR 个数建议限制在 10 个以内。<b>(数据格式注意：不同数据之间用逗号隔开，</b></p>	



			<p>结束符以回车换行结束即可)</p> <p>4: 把上位机发来的数据同时放到 R 和 PR 寄存器中。(数据格式注意: 不同数据之间用逗号隔开, 结束符以回车换行结束即可, 上位机发来的数据顺序要求 R1,x1,y1,z1,w1,p1,r1,R2,x2,y2,z2,w2,p2,r2……以此类推, 7 个数作为一组, 建议一次性发送组数不要超过 4 个。否则可能会报警)</p> <p><b>参数 2 (int)</b> 存放的寄存器号码, 如果接收到多个, 则自动往后面的寄存器存放。</p> <p><b>例: CALL TCP_RECV(1,1)</b> 从上位机接收 R[1]-R[5]这 5 个数值寄存器的内容到机器人。</p>	
--	--	--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

## 5.2 在 TP 程序中调用通讯专用程序

步骤如下 (以通讯连接程序为例子):

- ①在 TP 程序中, 光标移至需要调用的行;
- ②按下[指令];
- ③[调用];
- ④[调用程序];
- ⑤F3[COLLECT];
- ⑥选择 “KAREL”;
- ⑦选择 “TCP\_CONNECT”;
- ⑧光标移至本条指令的最后面空白处, F4[选择];
- ⑨F4[选择]-[字符串]-F5[字符串]-[大写]-输入 “C1”, 如下图所示:



## 5.3 实际使用案例

如下图案例所示, 其能够实现的通讯功能具体是:

向上位机发送数据, 发送内容是 PR[1]和 PR[2]的 XYZWPR 的值, 共计 12 个数据。

接收来自上位机发送的数据, 将数据放在以 R[1]开始的数值寄存器内。

```
CALL TCP_CONNECT('C1')
CALL TCP_SEND(2, 1, 2)
CALL TCP_RECV(1, 1)
CALL TCP_DISCO('C1')
```

与上位机建立通讯连接, 机器人作为客户端, 使用 C1 号标签

与上位机断开通讯连接。

## 6、常见报警

### 6.1 INTP-311

故障现象：INTP-311 (tcp\_recv, \*\*\*)参数还没有设定。

故障原因：机器人与上位机之间的通讯尚未建立

处理方案：

1.建立机器人与上位机之间的通讯连接。

### 6.2 INTP-366

故障现象：INTP-366 (tcp\_send, \*\*\*)还没有清除 IO。

故障原因：机器人与上位机之间的通讯已断开的情况下运行发送数据程序

处理方案：

1.建立机器人与上位机之间的通讯连接。

### 6.3 APSH-000

故障现象：APSH-000 sending fail。

故障原因：机器人与上位机之间的通讯已断开的情况下运行发送程序

处理方案：

1.建立机器人与上位机之间的通讯连接。

### 6.4 APSH-000

故障现象：APSH-000 Open fail。

故障原因：机器人与上位机之间的通讯已断开运行接收数据程序

处理方案：

1.建立机器人与上位机之间的通讯连接。

### 6.5 HOST-212

故障现象：HOST-212 写入方向关闭。

故障原因：机器人与上位机之间的通讯已断开运行发送数据程序

处理方案：

1.建立机器人与上位机之间的通讯连接。

### 6.6 HOST-213

故障现象：HOST-213 读取方向关闭。

故障原因：机器人与上位机之间的通讯已断开运行接收数据程序

处理方案：

1.建立机器人与上位机之间的通讯连接。

## 4、附录

			
tcp_recv.pc	tcp_send.pc	tcp_disco.pc	tcp_connect.pc