

TCP/IP通讯常见问题点总结

所属: KRCT-FA
作成: 何君
Version:1.0

一. 通讯程序的问题

1、TCP) 套接字ID非法



一. 通讯程序的问题

1、TCP) 套接字ID非法

-34057 (E4057)	TCP) 套接字 ID 非法。	在执行 TCP_SEND/TCP_RECV 的过程中，通信错误产生。 或，当执行 TCP_CLOSE/TCP_END_LISTEN 命令时，未能关闭套接字。 可能已指定未创建的套接字号或端口号。或，程序的执行顺序错误。	检查套接字号或端口号是否正确，如果必要的话，则进行修改。检查程序的执行顺序是否错误，如果必要的话，则进行修正。
-------------------	-----------------	---	---

错误原因：

- (1)打开的套接字已经关闭的情况下，重复执行TCP_CLOSE/TCP_END_LISTEN命令，套接字关闭失败。
- (2)打开的套接字已经关闭的情况下，执行了TCP_SEND/TCP_RECV命令，通讯错误产生。

一. 通讯程序的问题

2、TCP) 不能创建套接字



一. 通讯程序的问题

2、TCP) 不能创建套接字

-34055 (E4055)	TCP) 不能创建套接字。	当执行 TCP_LISTEN/TCP_CONNEC 命令时，未能打开套接字。已打开太多的套接字。或，多次使用同一端口。	避免一次执行多个命令。修改参数。以前使用的端口用 TCP_CLOSE/TCP_END_LISTEN 命令关闭后，再使用。
-------------------	---------------	---	--

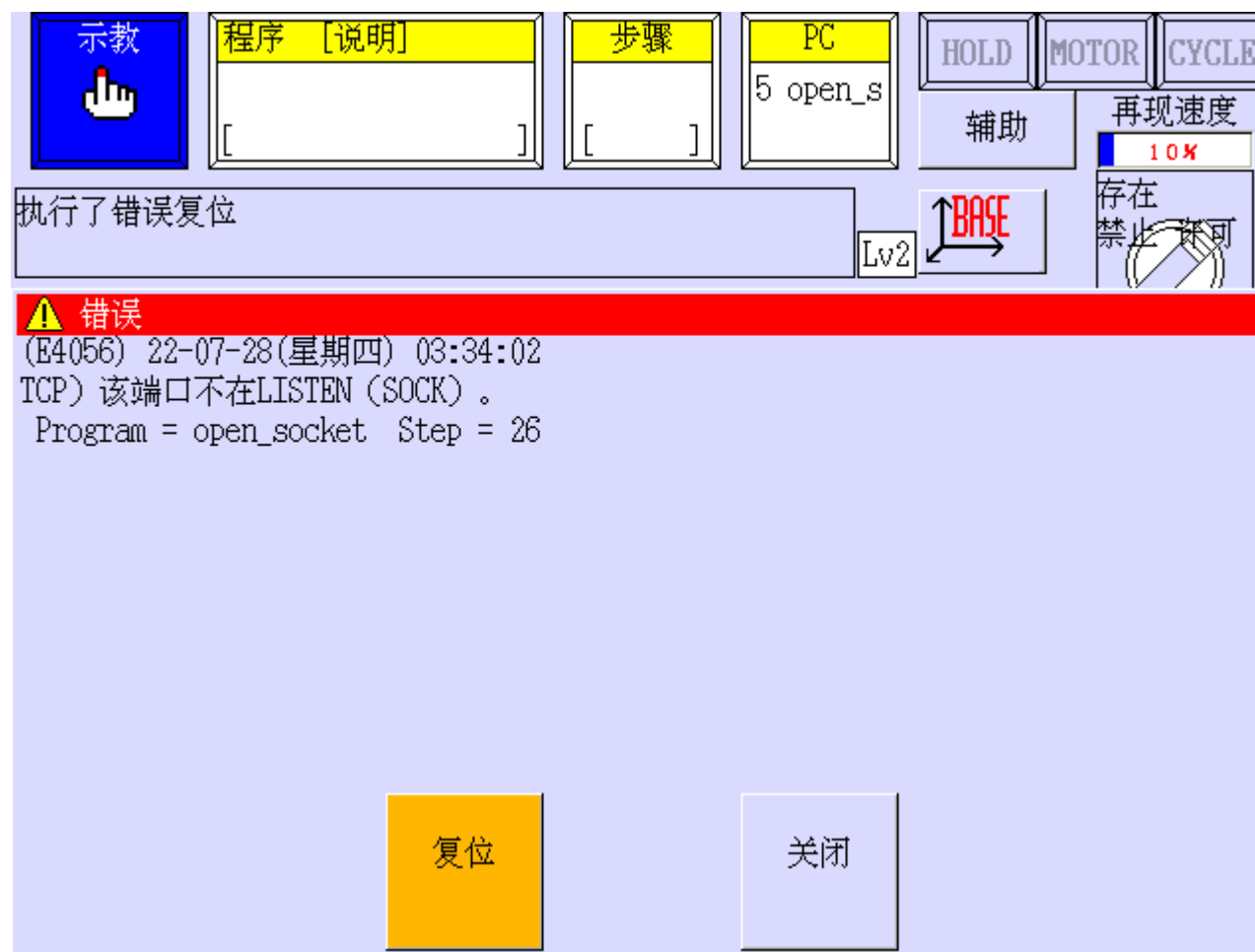
错误原因：

一个端口，同时只能打开一个套接字。

- (1)同一端口，已经打开套接字的情况下，重复执行TCP_LISTEN/TCP_CONNECT命令，多次使用同一个端口尝试打开套接字。
- (2)打开太多的套接字。

一. 通讯程序的问题

3、TCP) 该端口不在LISTEN (SOCK)



一. 通讯程序的问题

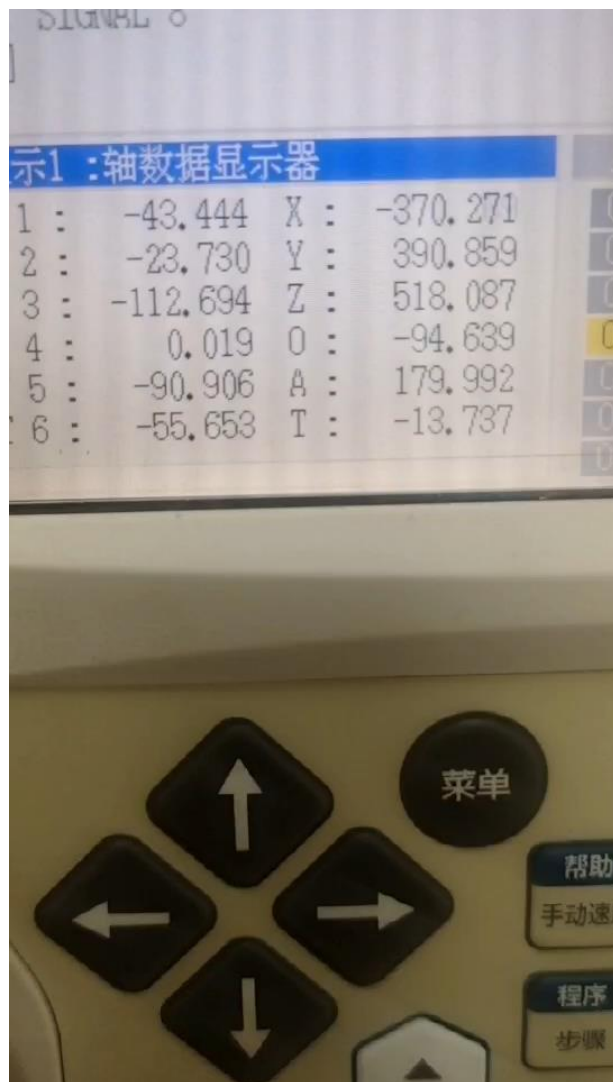
3、TCP) 该端口不在LISTEN(SOCK)

-34056 (E4056)	TCP) 该端口不在LISTEN(SOCK)。	当执行 TCP_ACCEPT 命令时，未能打开套接字。或当执行 TCP_END_LISTEN 命令时，未能关闭套接字。指定了用 TCP_LISTEN 命令未创建的套接字。	检查端口号是否正确，如果必要的话，则进行修改。
-------------------	-------------------------	---	-------------------------

错误原因：

- (1)打开的套接字，已经用TCP_END_LISTEN关闭了的情况下，重复执行TCP_END_LISTEN命令，未能关闭套接字。
- (2)打开的套接字，用TCP_END_LISTEN命令关闭了的情况下，再次通讯连接的时候，跳过了TCP_LISTEN步骤，执行TCP_ACCEPT命令，未能打开套接字。

二. 视觉标定过程中出现的问题



1、机器人在同一位置OT角变化，表现形式有两种：

(1)机器人按住使能键或者在自动状态下，机器人还没动的情况下OT值不停的跳动。

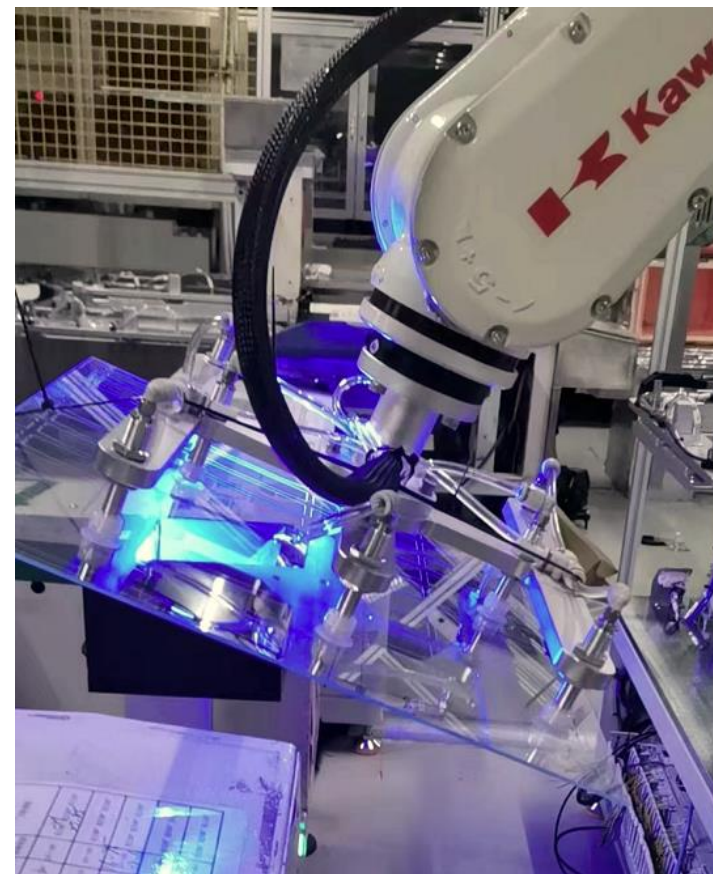
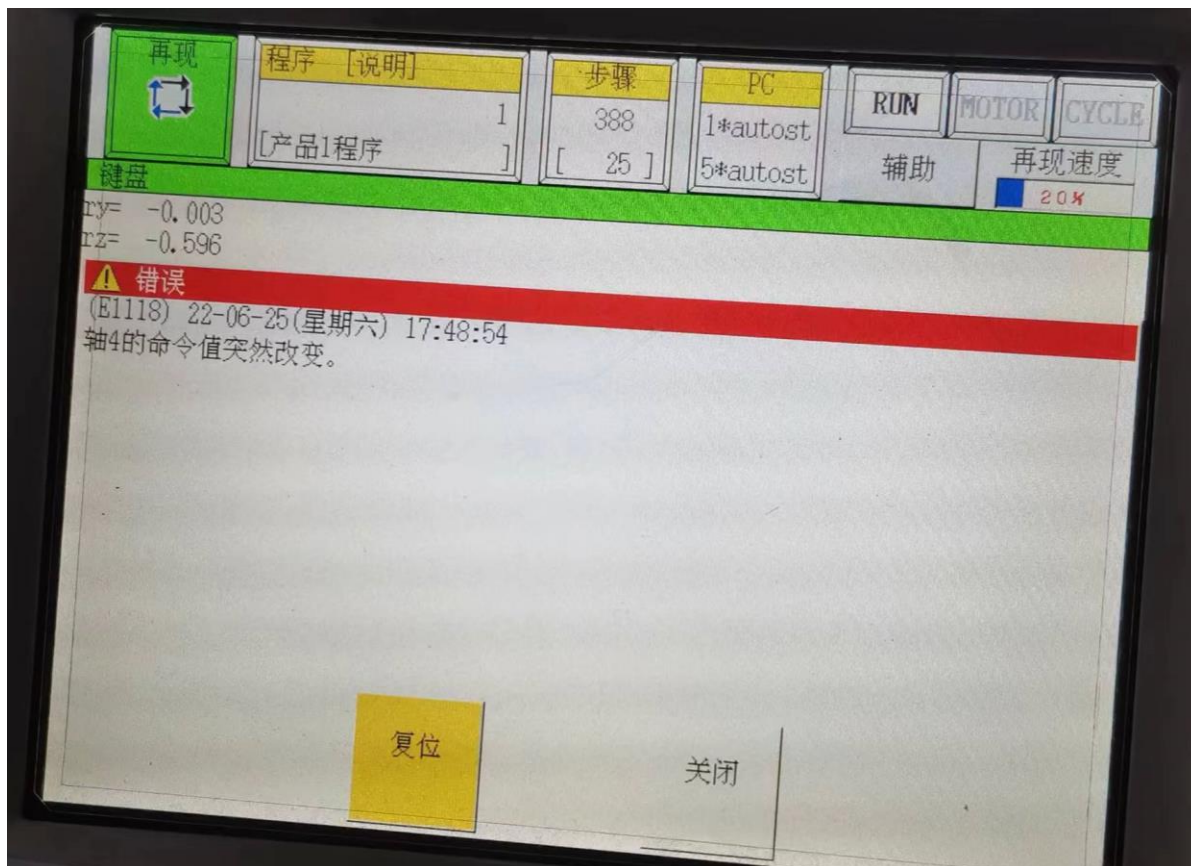
(2)机器人来回移动到同一位置，OT值与上一次不同。

问题原因：川崎机器人姿态采用的是欧拉角表达的，在同一位置可以用不同的OT值组合来表达同一姿态。

解决方法：把欧拉角换成笛卡尔坐标系的角度，即把O/A/T转换成R/P/Y，然后再把Y值发给视觉标定。

二. 视觉标定过程中出现的问题

2、轴命令值突然改变



二. 视觉标定过程中出现的问题

2、轴命令值突然改变

表现形式：机器人抓着玻璃，标定过程中机器人姿态需要保持固定。从标定起始位连续走11个标定点，在移动过程中姿态突然发生倾斜变化，机器人姿态突变报警。

问题原因：示教的起始位是变换值，机器人每次标定完成要回起始位，由于欧拉角的原因，实际上机器人回到的起始位并不是示教的起始位，有可能是多转一圈的，在下一次标定的时候就可能发生轴命令值突然变化的错误。

解决方法：把起始位的变量用各轴值表示，每次先回到各轴值表示的起始位，在起始位下面HERE一下标定用的变换值起始位。

二. 视觉标定过程中出现的问题

2、轴命令值突然改变

```
SVALUE "jz01"+$CHR(13)+$CHR(10): ;1 #相机标定基准位  
CALL safe;回到安全高度  
JAPPRO jz01,50 ;移动到上方点50  
SPEED 10 ;速度10  
LMOVE jz01;移动到p3  
BREAK  
$send_data = "OK"+$CHR(13)+$CHR(10) ;移动到位  
SIGNAL 2002 ;数据发送  
SWAIT -2002 ;等待数据发送完成
```

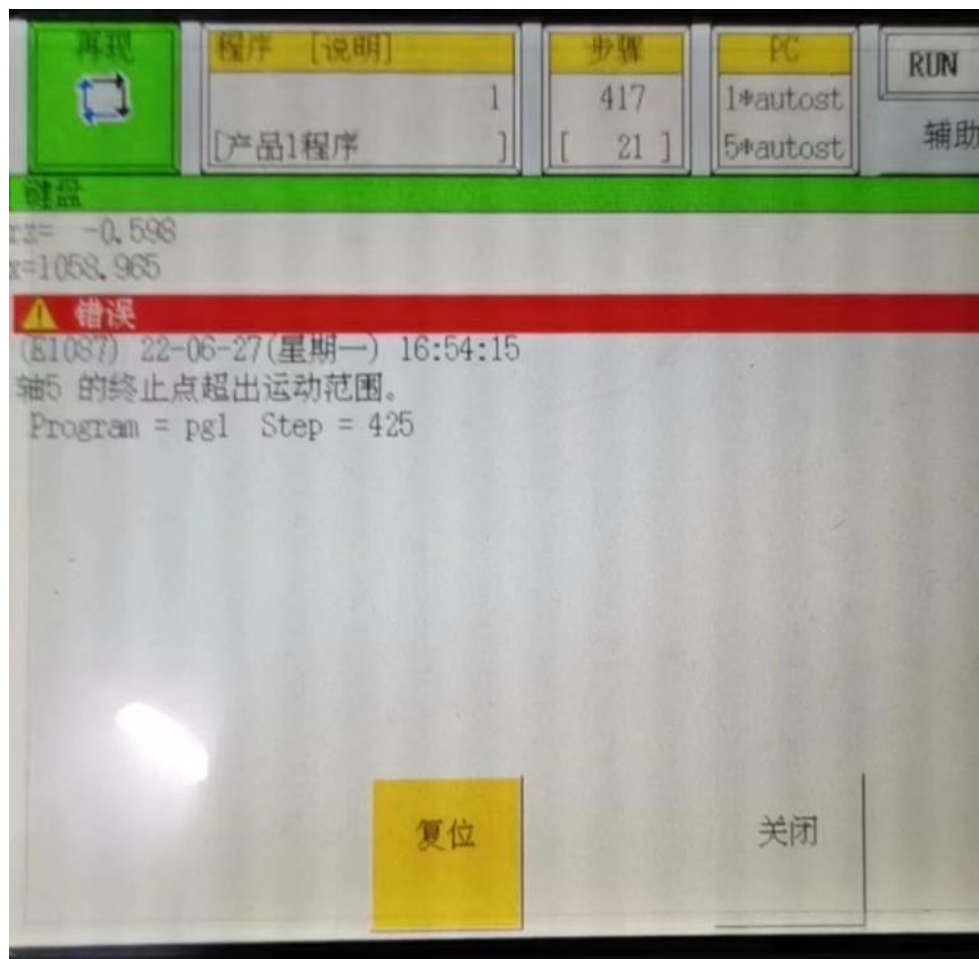
修改前程序

```
SVALUE "jz01" + $CHR (13) + $CHR (10): ;1 #相机标定基准位  
CALL safe;回到安全高度  
JAPPRO #jz01, 50 ;移动到上方点50  
SPEED 10 ;速度10  
LMOVE #jz01;移动到 #p3  
BREAK  
$send_data = "OK" + $CHR (13) + $CHR (10) ;移动到位  
SIGNAL 2002 ;数据发送  
SWAIT -2002 ;等待数据发送完成
```

修改后程序

二. 视觉标定过程中出现的问题

3、轴*的终止点超出运动范围



表现形式：标定过程中机器人接收视觉发送的绝对值坐标数据，转换成欧拉角后再移动到新的点位，同一个点位重复执行可能会出现轴*的终止点超出运动范围报错。

错误原因：RPY转换成OAT后，新的OT值组合可能与上一次的OT值不同，新的OT值组合可能会发生轴超限的错误。

解决方法：视觉发偏移值坐标，机器人不转换坐标系，用偏移值重组新的点位。

二. 视觉标定过程中出现的问题

3、轴*的终止点超出运动范围

```
SVALUE "bd01"+$CHR(13)+$CHR(10): ;#1相机开始标定
$send_data = "OK"+$CHR(13)+$CHR(10)
SIGNAL 2002 ;数据发送
SWAIT -2002 ;等待数据发送完成
SIGNAL 2001,2102,-2103 ;数据接收,解析字符
SWAIT -2001 ;等待数据接收完成
IF i<3 THEN
    $send_data = "?"+$CHR(13)+$CHR(10) ;返回PLC接受的程序号
    SIGNAL 2002 ;数据发送
    SWAIT -2002 ;等待数据发送完成
    GOTO loop ;跳转到标签loop
END
IF value[1]>100 OR value[2]>100 OR value[3]>90 THEN ;数据超出范围
    $send_data = "?"+$CHR(13)+$CHR(10) ;返回?号
    SIGNAL 2002 ;数据发送
    SWAIT -2002 ;等待数据发送完成
    GOTO loop ;跳转到标签loop
END
CALL mm
CALL keyrpy2oat(anglersltr,anglersltp,value[3]) ;笛卡尔转换成欧拉角
HERE .pos|
DECOMPOSE gg[1] = .pos ;坐标分解
POINT bd01 = TRANS(value[1],value[2],gg[3],anglerslto,anglerslta,anglersltp) ;pz01点位重组
SPEED 10 ;速度10
LMOVE bd01 ;移动到bd01
BREAK ;最高速度
$send_data = "OK"+$CHR(13)+$CHR(10) ;移动到位
SIGNAL 2002 ;数据发送
SWAIT -2002 ;等待数据发送完成
```

修改前程序

二. 视觉标定过程中出现的问题

3、轴*的终止点超出运动范围

```
SVALUE "bd01" + $CHR (13) + $CHR (10): ;#1相机开始标定
$send_data = "OK" + $CHR (13) + $CHR (10)
SIGNAL 2002 ;数据发送
SWAIT -2002 ;等待数据发送完成
SIGNAL 2001, 2102, -2103 ;数据接收, 解析字符
SWAIT -2001 ;等待数据接收完成
IF i < 3 THEN
    $send_data = "?" + $CHR (13) + $CHR (10) ;返回 PLC 接受的程序号
    SIGNAL 2002 ;数据发送
    SWAIT -2002 ;等待数据发送完成
    GOTO loop ;跳转到标签 loop
END
IF value[1] > 100 OR value[2] > 100 OR value[3] > 90 THEN ;数据超出范围
    $send_data = "?" + $CHR (13) + $CHR (10) ;返回?号
    SIGNAL 2002 ;数据发送
    SWAIT -2002 ;等待数据发送完成
    GOTO loop ;跳转到标签 loop
END
HERE .pos
DECOMPOSE gg[1] = .pos ;坐标分解
POINT bd01 = TRANS (gg[1] + value[1], gg[2] + value[2], gg[3], gg[4], gg[5], gg[6] + value[3])
SPEED 10 ;速度 10
LMOVE bd01 ;移动到 bd01
BREAK ;最高精度
$send_data = "OK" + $CHR (13) + $CHR (10) ;移动到位置
SIGNAL 2002 ;数据发送
SWAIT -2002 ;等待数据发送完成
```

修改后程序