

# 接口说明

## 版本记录

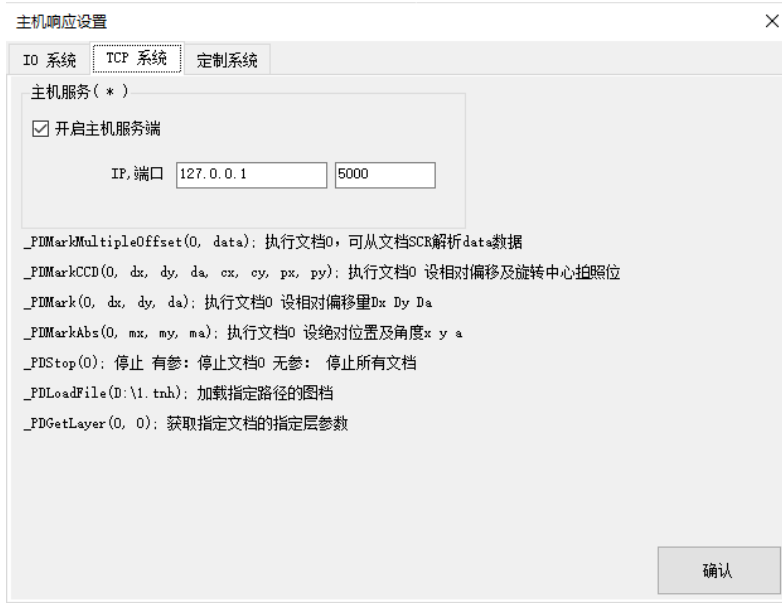
| 版本号    | 作者   | 日期         | 备注          |
|--------|------|------------|-------------|
| V1.0.0 | 鹏鼎智控 | 2021.11.18 |             |
| V1.0.1 | 鹏鼎智控 | 2023年6月13日 | 软件版本 8.1.4  |
| V1.0.2 | 鹏鼎智控 | 2023年6月30日 | 软件版本 8.1.6  |
| V1.0.3 | 鹏鼎智控 | 2023年7月20日 | 软件版本 8.2.03 |
| V1.0.4 | 鹏鼎智控 | 2024年7月10日 | 软件版本 8.3.2C |
|        |      |            |             |
|        |      |            |             |

# 外部触发(TCP/IP 系统)

## 1. 通讯协议及方式

1.1 . 通过 TCP/IP 的方式建立连接,激光软件为服务器。

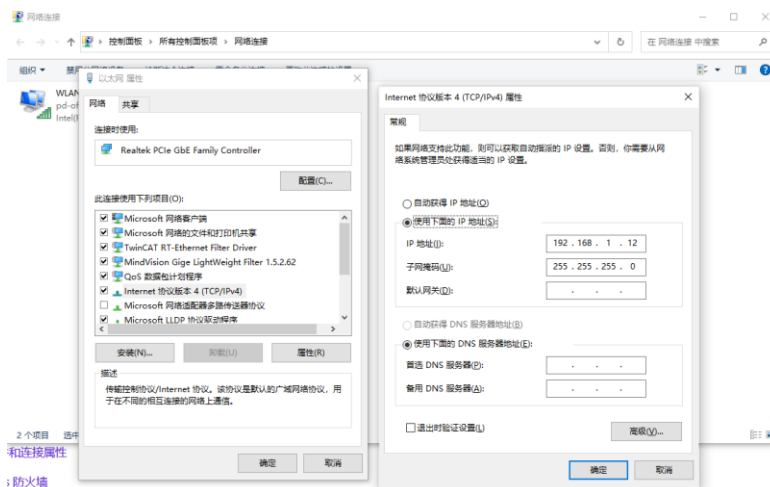
软件设置：设置>主机响应设置>TCP 系统



注：勾选开启后重启软件即可启动 TCP/IP 服务器

## 1.2 . IP 设置

- 1.如果是本机连接，就保持默认设置，IP 地址就设置 127.0.0.1 端口 5000
- 2.如果是外部连接，就设置静态 IP 地址，这边是服务端软件上设置成网口配置里设置的静态 IP 地址。



## 2. 通讯格式

收发消息均使用字符串格式收发报文，编码方式为 UTF-8。如果不是 UTF-8 中文会显示乱码

发送格式：\_Command(arg1,arg2,arg3,...);

返回格式：\_Command(index,result,data);

### 一、 外部触发通讯指令

#### 1. 触发文档，设置相对偏移量

##### 1.1 接口说明

功能：触发指定的文档并设置相对偏移量（旋转中心默认为 0,0）

客户端发送：\_PDMark(tnhIndex,dx,dy,da);

| 序号 | 参数名称     | 参数说明      | 备注               |
|----|----------|-----------|------------------|
| 1  | tnhIndex | 文档号 (int) | 已打开后缀为 tnh 的文档索引 |
| 2  | dx       | X 偏移量     |                  |
| 3  | dy       | Y 偏移量     |                  |
| 4  | da       | 角度偏移量     |                  |

服务器：文档运行完成后反馈：\_PDMark(0,0,"");

| 序号 | 参数名称     | 参数说明      | 备注               |
|----|----------|-----------|------------------|
| 1  | tnhIndex | 文档号 (int) | 返回发送的 tnh 文档号    |
| 2  | result   | 执行结果      | 0 = OK /非 0 = NG |
| 3  | data     | 错误内容      | 正常为空，错误为具体信息     |

##### 1.2 示例

发送：\_PDMark(0,1,2,3);

返回：\_PDMark(0,0,"");

#### 2. 触发文档，设置绝对偏移量

##### 2.1 接口说明

功能：触发指定的文档并设置绝对偏移量

客户端发送：\_PDMarkAbs(tnhIndex,x,y,a);

| 序号 | 参数名称     | 参数说明      | 备注               |
|----|----------|-----------|------------------|
| 1  | tnhIndex | 文档号 (int) | 已打开后缀为 tnh 的文档索引 |
| 2  | x        | 坐标 x      |                  |
| 3  | y        | 坐标 y      |                  |
| 4  | a        | 角度 angle  |                  |

服务器：文档运行完成后反馈：\_PDMarkAbs(0,0,"");

| 序号 | 参数名称 | 参数说明 | 备注 |
|----|------|------|----|
|----|------|------|----|

|   |          |           |                  |
|---|----------|-----------|------------------|
| 1 | tnhIndex | 文档号 (int) | 返回发送的 tnh 文档号    |
| 2 | result   | 执行结果      | 0 = OK /非 0 = NG |
| 3 | data     | 错误内容      | 正常为空, 错误为具体信息    |

2.2 示例

发送: `_PDMarkAbs(0,1,2,3);`

返回: `_PDMarkAbs(0,0,"");`

### 3. 终止文档执行

3.1 接口说明

功能: 终止指定文档或所有运行文档

客户端发送: `_PDStop(tnhIndex);`

| 序号 | 参数名称     | 参数说明      | 备注               |
|----|----------|-----------|------------------|
| 1  | tnhIndex | 文档号 (int) | 已打开后缀为 tnh 的文档索引 |

服务器: 文档运行完成后反馈: `_PDStop(tnhIndex, result);`

| 序号 | 参数名称     | 参数说明    | 备注               |
|----|----------|---------|------------------|
| 1  | tnhIndex | 文档号(选填) | 未指定文档号即终止所有文档    |
| 2  | result   | 执行结果    | 0 = OK /非 0 = NG |

3.2 示例

发送: `_PDStop(0);`

返回: `_PDStop(0,0);`

### 4. 触发文档, 设置相对偏移量 (指定旋转中心和拍照位)

4.1 接口说明

功能: 触发指定的文档并设置相对偏移量

客户端发送: `_PDMarkCCD(tnhIndex,x,y,a,cx,cy,px,py);`

| 序号 | 参数名称     | 参数说明          | 备注               |
|----|----------|---------------|------------------|
| 1  | tnhIndex | 文档号 (int)     | 已打开后缀为 tnh 的文档索引 |
| 2  | x        | 坐标 x          |                  |
| 3  | y        | 坐标 y          |                  |
| 4  | a        | 角度 angle      |                  |
| 5  | cx       | 旋转中心 center x |                  |
| 6  | cy       | 旋转中心 center y |                  |
| 7  | px       | 拍照位 photo x   | 默认为 0, 可做手动叠加量 x |
| 8  | py       | 拍照位 photo y   | 默认为 0, 可做手动叠加量 y |

服务器: 文档运行完成后反馈: `_PDMarkCCD(0,0,"");`

| 序号 | 参数名称     | 参数说明 | 备注               |
|----|----------|------|------------------|
| 1  | tnhIndex | 文档号  | 返回发送的 tnh 文档号    |
| 2  | result   | 执行结果 | 0 = OK /非 0 = NG |
| 3  | data     | 错误内容 | 正常为空, 错误为具体信息    |

4.2 示例

发送: \_PDMarkCCD(0,1.5,2.5,0.25,0,0,0,0);

返回: \_PDMarkCCD(0,0,"");

## 5. 触发文档，暂存解析数据内容

### 5.1 接口说明

功能: 触发指定的文档并暂存解析发送内容(如多组偏移量组成的字符串)

客户端发送: **\_PDMarkMultipleOffset(tnhIndex,data);**

| 序号 | 参数名称     | 参数说明   | 备注  |
|----|----------|--|---|
| 1  | tnhIndex | 文档号 (int)  | 已打开后缀为 tnh 的文档索引  |
| 2  | data     | 暂存内容<br>x1%y1%a1%<br>cx1%cy1%p<br>x1%py1%<br>result#X2<br>%y2%a2%cx<br>2%cy2%px2<br>%py2%<br>result... | 偏移量系统固定格式(X%Y%角度%旋转中心 X%<br>旋转中心 Y%拍照位置 X%拍照位置 Y%结果),<br>result 0 是 NG, 1 是 OK。每组数据用#分<br>隔, 每个数据用%分隔, 可用流程工具<br>RecordBox 解析。<br>例如:<br>_PDMarkMultipleOffset(0,10%20%30%40%50%60<br>%70%1#1%2%3%4%5%6%7%0)中<br>10,20,30,40,50,60,70 为第一组偏移量, 1 表示<br>结果 OK, 1,2,3,4,5,6,7 为第二组偏移量, 0 表<br>示结果 NG。分隔符不可使用逗号与分号。<br>注: 可以自定义格式通过脚本解析。 |

服务器: 文档运行完成后反馈: **\_PDMarkMultipleOffset(0,0,"");**

| 序号 | 参数名称     | 参数说明      | 备注               |
|----|----------|-----------|------------------|
| 1  | tnhIndex | 文档号 (int) | 返回发送的 tnh 文档号    |
| 2  | result   | 执行结果      | 0 = OK /非 0 = NG |
| 3  | data     | 错误内容      | 正常为空, 错误为具体信息    |

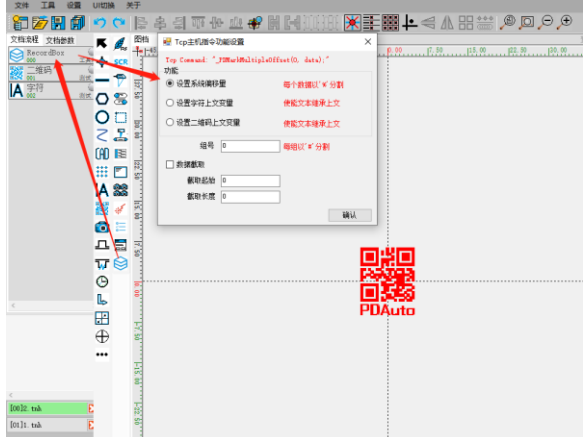
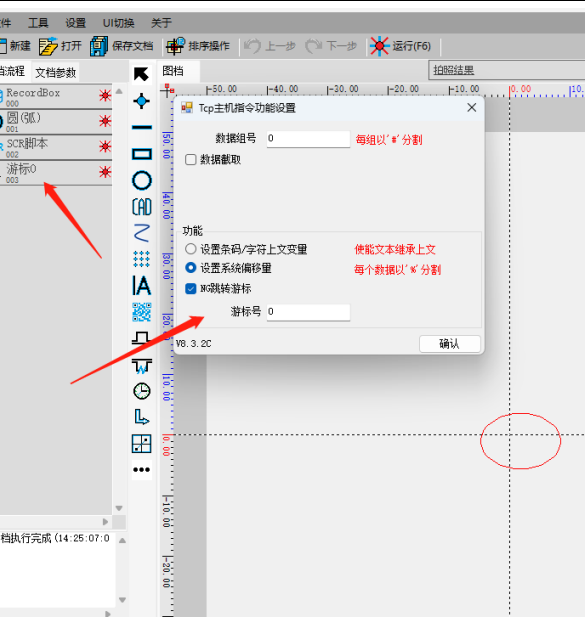
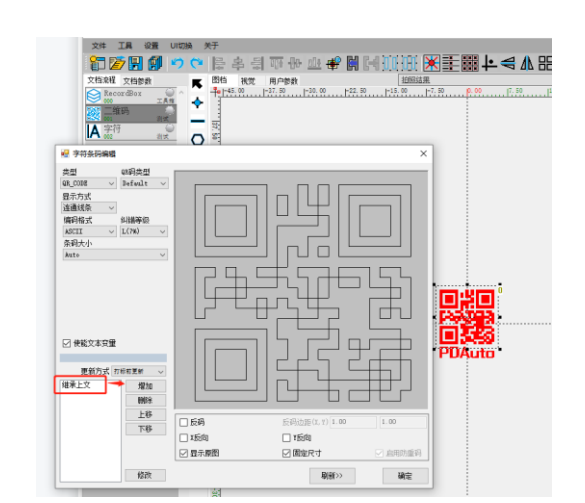
### 5.2 示例

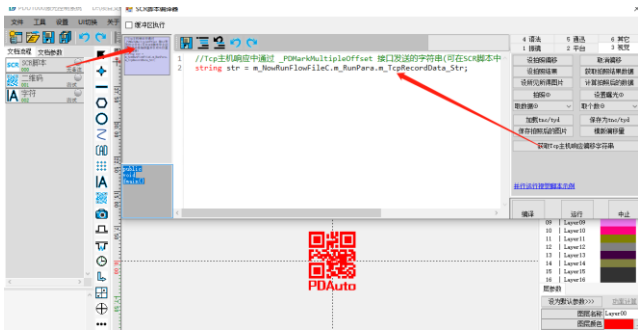
发送: \_PDMarkMultipleOffset(0,10%20%30%40%50%60%70%1#1%2%3%4%5%6%7%0);

返回: \_PDMarkMultipleOffset (0,0,"");

### 5.3 解析数据

| 解析方式 | 说明 | 图示 |
|------|----|----|
|------|----|----|

|  |   |  |
|--|---|--|
| <p>RecordBox 工具解析设置偏移量 (8.3.2C 版本之前软件)</p> | <p>软件文档流程中添加工具 RecordBox，可以将收到数据解析按照固定格式，每组偏移量用#分隔，每组偏移量中每个数据用%分隔通过 RecordBox 工具解析，解析数据按序列填入工具组号中，可实现多组偏移量赋值。</p> |    |
| <p>(8.3.2C 软件) RecordBox 工具新增偏移量结果处理</p>   | <p>可以将收到数据解析判断结果 result 是 1 还是 0。0 是 NG，1 是 OK，按以前的没有结果 result 发送默认就是 OK，然后 NG 的可以跳转游标。</p>                       |   |
| <p>RecordBox 工具解析设置二维码/字符</p>              | <p>软件文档流程中添加工具 RecordBox，可以将收到数据解析按照固定格式，每组二维码/字符内容用#分隔，通过 RecordBox 工具解析，解析数据按序列填入工具组号中。</p>                     |  |

|                           |   |  |
|---------------------------|---|--|
| <p>脚本工具解析设置偏移量/二维码/字符</p> | <p>在文档流程里添加一个脚本，脚本中点击，获取 TCP 主机响应偏移量字符串，会自动生成获取脚本变量，通过脚本变量来解析设置偏移量/二维码/字符，这里会涉及到 C# 语法，需要有一定编程基础的工程师才能完成。</p> |  |
|---------------------------|---|--|

## 6. 导入文档（若文档存在，返回文档所在位置）

### 6.1 接口说明

功能：从指定路径导入焊接软件后缀 tnh 的文档

客户端发送：\_PDLoadFile(filePath);

| 序号 | 参数名称     | 参数说明 | 备注          |
|----|----------|------|-------------|
| 1  | filePath | 文档路径 | 焊接软件后缀为.tnh |

服务器：文档运行完成后反馈：\_PDLoadFile(result,fileIndex);

| 序号 | 参数名称      | 参数说明      | 备注                |
|----|-----------|-----------|-------------------|
| 1  | result    | 执行结果      | 0 = OK /非 0 = NG  |
| 2  | fileIndex | 文档号 (int) | 文档导入后在已打开文档列表中的序号 |

### 6.2 示例：

发送：\_PDLoadFile("D:\\Test.tnh");

返回：\_PDLoadFile(0,0);

## 7. 获取指定文档的指定层参数

### 7.1 接口说明

功能：获取指定文档中指定图层的相关工艺参数

客户端发送：\_PDGetLayer(fileIndex,layerIndex);

| 序号 | 参数名称       | 参数说明      | 备注           |
|----|------------|-----------|--------------|
| 1  | fileIndex  | 文档号 (int) | 已打开文档列表中的序号  |
| 2  | layerIndex | 图层号       | 操作文档中图层参数的序号 |

服务器：文档运行完成后反馈：\_PDGetLayer(result,fileIndex,...);

| 序号 | 参数名称      | 参数说明         | 备注                       |
|----|-----------|--------------|--------------------------|
| 1  | fileIndex | 文档号 (int)    | 已打开文档列表中的序号              |
| 2  |           | 图层名          |                          |
| 3  |           | 加工数目         |                          |
| 4  |           | 速度           |                          |
| 5  |           | 空程速度         |                          |
| 6  |           | 功率百分比        |                          |
| 7  |           | 波形号          |                          |
| 8  |           | 频率           |                          |
| 9  |           | 占空比          |                          |
| 10 |           | 开光延时         |                          |
| 11 |           | 关光延时         |                          |
| 12 |           | 结束延时         |                          |
| 13 |           | 拐角延时         |                          |
| 14 |           | 打标延时         |                          |
| 15 |           | 最大跳转延时       |                          |
| 16 |           | 最少跳转延时       |                          |
| 17 |           | 跳转距离         |                          |
| 18 |           | 单点延时         |                          |
| 19 |           | 红光速度         |                          |
| 20 |           | SPI 阈值电流     |                          |
| 21 |           | 第 2 路功率      |                          |
| 22 |           | 第 2 路频率      |                          |
| 23 |           | 第 2 路占空比     |                          |
| 24 |           | 第 2 路开光延时    |                          |
| 25 |           | 第 2 路关光延时    |                          |
| 26 |           | 第 1 路点波形是否启动 | 1: 启动 0: 不启动             |
| 27 |           | 第 1 路点波形参数   | 时间%功率%时间%功率%时间%功率%时间%功率% |
| 28 |           | 第 1 路线波形是否启动 | 1: 启动 0: 不启动             |
| 29 |           | 第 1 路线波形渐进参数 | 时间%功率%时间%功率%时间%功率%时间%功率% |



|    |  |              |                          |
|----|--|--------------|--------------------------|
| 30 |  | 第 1 路线波形渐出参数 | 时间%功率%时间%功率%时间%功率%时间%功率% |
| 31 |  | 第 2 路点波形是否启动 | 1: 启动 0: 不启动             |
| 32 |  | 第 2 路点波形参数   | 时间%功率%时间%功率%时间%功率%时间%功率% |
| 33 |  | 第 2 路线波形是否启动 | 1: 启动 0: 不启动             |
| 34 |  | 第 2 路线波形渐进参数 | 时间%功率%时间%功率%时间%功率%时间%功率% |
| 35 |  | 第 2 路线波形渐出参数 | 时间%功率%时间%功率%时间%功率%时间%功率% |

## 7.2 示例

发送: `_PDGetLayer(0,0);`

返回:

```
_PDGetLayer(0,Layer00,1,800.000,2000.000,20.000,0,10.000,50.000,50,50,50,50,1000,1000,500,10.000,1.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,#1#0.000%50.000%10.000%60.000%10.000%10.000%#1#0.000%30.000%20.000%50.000%#10.000%30.000%5.000%0.000%#0#0.000%10.000%10.000%10.000%#0#0.000%10.000%#10.000%0.000%);
```

(`_PDGetLayer`(文档号, 图层名, 加工数目, 速度, 空程速度, 功率百分比, 波形号, 频率, 占空比, 开光延时, 关光延时, 结束延时, 拐角延时, 打标延时, 最大跳转延时, 最少跳转延时, 跳转距离, 单点延时, 红光速度, SPI 阈值电流, 第 2 路功率, 第 2 路频率, 第 2 路占空比, 第 2 路开光延时, 第 2 路关光延时 # 第 1 路点波形是否启动 # 第 1 路点波形参数 # 第 1 路线波形是否启动 # 第 1 路线波形渐进参数 # 第 1 路线波形渐出参数 # 第 2 路点波形是否启动 # 第 2 路点波形参数 # 第 2 路直线波形是否启动 # 第 2 路线波形渐进参数 # 第 2 路线波形渐出参数))

# 8. 振镜跳转

## 3.1 接口说明

功能: 控制振镜跳转到指定位置,跳转后延时多少 ms

客户端发送: `_PDMoveAbs(Card,X,Y, Delay);`

| 序号 | 参数名称  | 参数说明     | 备注      |
|----|-------|----------|---------|
| 1  | Card  | 卡号 (int) | 打开卡的索引号 |
| 2  | X     | 坐标 x     |         |
| 3  | Y     | 坐标 y     |         |
| 4  | Delay | 延时       |         |

服务器: 指令运行完成后反馈: `_PDMoveAbs (Card, result);`

| 序号 | 参数名称   | 参数说明     | 备注      |
|----|--------|----------|---------|
| 1  | Card   | 卡号 (int) | 打开卡的索引号 |
| 2  | result | 结果       |         |

### 3.2 示例

发送: `_PDMoveAbs(0,1,2,100);`

返回: `_PDMoveAbs(0,0,);`