

配置环境

Qt: qt-opensource-windows-x86-msvc2015_64-5.8.0

下载地址 http://download.qt.io/new_archive/qt/5.8/5.8.0/

编译环境: visual studio 2015

下载地址 <https://visualstudio.microsoft.com/zh-hans/vs/older-downloads/>

连接控制器

```
void connect_robot(char *ip);
```

ip: 控制器 ip

机器人上电

```
void servo_power_on();
```

机器人下电

```
void servo_power_off();
```

关节运动

```
void robot_movj(double *pos,int vel,int coord);
```

点位数组，长度 6

速度，参数范围： $0 < vel \leq 100$

坐标系，参数范围： $0 \leq coord \leq 3$

直线运动

```
void robot_movl(double *pos,int vel,int coord);
```

点位数组，长度 6

速度，参数范围： $0 < vel \leq 100$

坐标系，参数范围： $0 \leq coord \leq 3$

获取机器人运行状态

```
void get_robot_running_state (int &state);
```

state 机器人运行状态： 0 停止； 1 暂停； 2 运行

开始点动

```
void start_jogging (int axis,bool direction);
```

axis 轴号

direction 方向

停止点动

```
void stop_jogging(int axis);
```

axis 轴号

设置数字输出

```
void set_dout(int port,int value);
```

port 端口号

value 数值,参数范围: 0 或 1

设置模拟输出

void set_aout(int port,double value);

port 端口号

value 数值,参数范围: $0 \leq \text{value} \leq 10$

查询数字输出

void get_dout(double *dout);

dout 数字输出数组

查询数字输入

void get_din(double *din);

din 数字输入数组

查询模拟输出

void get_aout(double *aout);

aout 模拟输出数组

查询模拟输入

void get_ain(double *ain);

ain 模拟输入数组

获取机器人当前位置

void get_current_position(double *pos);

pos 点位数组, 长度 6

设置点动速度

void set_jogging_speed(int speed);

speed 速度, 参数范围: $0 < \text{speed} \leq 100$

连续运动模式

void continuous_motion_mode(int on);

on : 0 关闭; -1 开启

发送连续运动指令队列

void send_continuous_motion_queue(cmdPara cmd[],unsigned int size);

param cmd 指令数组

size 指令数组长度

指令结构体

```
typedef struct cmdPara
{
    // MOVJ 关节运动
    // MOVL 直线运动
    enum Type{MOVJ=1, MOVL};
    double m_velocity;      // 速度
    double m_acc;           // 加速度
    double m_dec;           // 减速度
    double m_pl;            // 平滑
    // m_position[0]和 m_position[1]代表坐标系
    // m_position[2]和 m_position[7]代表位置坐标
    double m_position[8];
    Type cmdType;           // 指令类型
}cmdPara;
```