

迈科讯 | 系统说明书 |

机器人通讯控制

文档版本

版本号	修订日期	修订内容
V1.0	2025.2.17	初订

目 录

第一章 机器人系统通讯功能	4 -
1.1 与视觉系统通讯.....	4 -
1.2 与 MODBUSTCP 主站通讯.....	4 -
1.3 与特定上位机通讯.....	4 -
1.4 与 CAN 编码器通讯.....	4 -
第二章 主站控制机器人系统说明	5 -
2.1 通讯连接.....	5 -
2.1.1 代码演示.....	5 -
2.2 通讯控制预备.....	5 -
2.2.1 代码演示.....	5 -
2.3 通讯控制示教移动.....	5 -
2.3.1 代码演示.....	5 -
2.4 通讯控制机器人运动.....	6 -
2.4.1 代码演示.....	6 -
2.5 通讯控制机器人连续运动.....	6 -
2.5.1 代码演示.....	6 -
第三章 简易 TCP 协议控制说明	8 -
3.1 指令格式说明.....	8 -

第一章 机器人系统通讯功能

机器人系统通讯功能包括与视觉系统通讯、与 modbusTcp 主站通讯、与特定上位机通讯、与 CAN 编码器通讯。

1.1 与视觉系统通讯

通过 TCP 通讯传输字符串进行通讯 机器人系统可以做 TCP 客户端也可以做 TCP 服务器，均可配置开机自启动。按照双方约定的传输格式进行数据传输，字符串中用于分割数据间的字符可以在视觉软件或者机器人系统软件中进行修改（例如：data1，data 2，data 3，data 4；）。

1.2 与 modbusTcp 主站通讯

机器人系统包含一个 modbusTcp 从站模块，可以接受单个 modbusTcp 主站的连接，标准 modbusTCP 通讯协议。当前 modbusTcp 从站使用了 3 种寄存器：可读可写保持寄存器，只读寄存器，可读可写线圈。上位机主站通过读写保持寄存器可以读写机器人系统中的参数、上位机主站通过读取只读寄存器可以读取机器人系统的位置和状态、上位机主站通过写线圈触发机器人系统动作。

为了支持 modbus 控制机器人系统连续运动，机器人系统中开辟了一个 modbus 触发指令的缓存队列，用于连续写保持寄存器触发机器人系统执行运动指令，根据写保持寄存器的先后顺序依次执行。

机器人系统中的参数包括：系统参数，轴参数，机构参数，零点配置等。

机器人系统的位置和状态包括：报警状态，使能状态，运动状态，世界坐标，关节坐标。

机器人系统的动作包括：急停、使能、清除报警、切换控制模式、示教手动运动、关节运动，直线运动。

机器人系统的缓存队列指令包括：直线运动，关节运动，运动减速停止，关节坐标直线运动，世界坐标关节运动，等待运动停止，延时 10 毫秒，GPIOOUT 输出控制。

1.3 与特定上位机通讯

通过 TCP 通讯传输字符串进行通讯，与视觉通讯类似，按照双方约定的传输格式进行数据传输，触发机器人系统动作或者读取机器人状态。此类通讯方式仅存在于定制开发中，功能不如 modbusTcp 从站功能多。

1.4 与 CAN 编码器通讯

通过 CAN 接口和 CAN 编码器进行通讯，通讯周期可以在 2mm 左右。

第二章 主站控制机器人系统说明

2.1 通讯连接

通过 modbus 协议栈创建 TCP 通讯，连接到控制器。控制器网口 net0 的默认地址是 192.168.1.200,网口 net1 的默认地址是 192.168.2.201。

2.1.1 代码演示

```
//创建主站并连接从站
modbusMaster = new ModbusMaster();
modbusMaster->initModbusMaster("192.168.1.200", 8080);
```

2.2 通讯控制预备

通讯连接后，通过从站控制机器人系统进入通讯控制模式，此时机器人系统不再检查示教器按钮置位状态。

通过写线圈触发机器人动作，大部分只有线圈切换时才有效果。

2.2.1 代码演示

```
//进入通讯控制模式
modbusMaster->SetSigCoilStatus(132, 0);
modbusMaster->SetSigCoilStatus(132, 1);
QThread::msleep(100);
//伺服上电
modbusMaster->SetSigCoilStatus(108, 0);
modbusMaster->SetSigCoilStatus(108, 1);
QThread::msleep(1000);
//发送触发使能
modbusMaster->SetSigCoilStatus(101, 1);
//modbusMaster->SetSigCoilStatus(101, 0);
QThread::msleep(100);
```

2.3 通讯控制示教移动

完成 2.2 预备相关线圈操作后，可以操作示教移动相关线圈寄存器。通过写线圈可以触发机器人系统切换示教运动坐标系（直线，关节），触发不同维度、不同关节的运动。

2.3.1 代码演示

```
//切换到关节坐标系
modbusMaster->SetSigCoilStatus(110, 0);
modbusMaster->SetSigCoilStatus(110, 1);
//关节 j1 正向运动
modbusMaster->SetSigCoilStatus(116, 1); //触发运动的线圈，写 1 是运动，写 0 是停止
```

2.4 通讯控制机器人运动

完成 2.2 预备相关线圈操作后，可以写保持寄存器传输直线、关节运动的速度，加减速时间，目标位置，然后在触发线圈控制机器人系统启动直线、关节运动。

2.4.1 代码演示

```
QVariant lineSpeed = 5;
QVariant varX = 280;
QVariant varY = 0;
QVariant varZ = 0;
QVariant varRx = 180;
QVariant varRy = 0;
QVariant varRz = 0;
QVariant varA = 0;
QVariant varB = 0;
//写保持寄存器设置运动速度，目标位置
modbusMaster->SetSigHoldingRegister(9000,lineSpeed,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9004,varX,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9006,varY,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9008,varZ,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9010,varRx,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9012,varRy,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9014,varRz,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9016,varA,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9018,varB,order_float_ABCD);
QThread::msleep(100);
//发送触发使能
modbusMaster->SetSigCoilStatus(1000, 1);
modbusMaster->SetSigCoilStatus(1000, 0);
QThread::msleep(10);
```

2.5 通讯控制机器人连续运动

通过写保持寄存器，往机器人系统的缓冲队列压入运动指令，等待运动停止，延时 10 毫秒，GPIOOUT 输出控制等。

操作步骤为，先写传输参数的寄存器地址，再写指令索引值，最后写校验位地址 0xAA 数值将指令压入缓冲队列。往保持寄存器 11230 写指令索引值，往保持寄存器 11232 写校验数值 0xAA 数值起到确认作用。

2.5.1 代码演示

```
QVariant lineSpeed = 5;
QVariant varX = 280;
QVariant varY = 0;
QVariant varZ = 0;
QVariant varRx = 180;
QVariant varRy = 0;
QVariant varRz = 0;
```

```
QVariant varA = 0;
QVariant varB = 0;
//直线运动索引值
QVariant cmdIndex = 1000;
//校验用(0xAA)数值
QVariant cmdRun = 170;
//写保持寄存器设置运动速度，目标位置
modbusMaster->SetSigHoldingRegister(9000,lineSpeed,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9004,varX,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9006,varY,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9008,varZ,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9010,varRx,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9012,varRy,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9014,varRz,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9016,varA,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(9018,varB,order_float_ABCD);
//写保持寄存器触发运动
modbusMaster->SetSigHoldingRegister(11230,cmdIndex,order_float_ABCD);
modbusMaster->SetSigHoldingRegister(11232,cmdRun,order_float_ABCD);
```

第三章 简易 TCP 协议控制说明

3.1 指令格式说明

数据包格式：J1,J2,J3,J4,J5,J6,CmdIndex,speed,acc,dec,blead,posture

CmdIndex 可用值说明：

- 1 直线运动
- 3 世界坐标走关节运动
- 4 姿态切换
- 5 读当前坐标
- 6 读报警状态
- 8 读关节坐标
- 9 关节运动
- 11 手动 X+ ,
- 12 手动 X- ,
- 21 手动 Y+ ,
- 22 手动 Y- ,
- 211J1+ ,
- 212J1- ,
- 100 急停
- 101 清报警 , 上电 , 上使能