# Unit-Roller485

## RS485 控 制 协 议

# 目录


.....1
1、通信协议结构.....3
1.1 通信协议参数.....3
1.1.1 部通信接口的通信参数.....3
1.2 数据包格式.....3
1.2.1 配置指令发送帧.....3
1.2.2 配置指令响应帧.....3
1.2.3 CRC8.....4
2、配置指令集.....4
2.1 Mode Switch.....4
2.2 Mode Setting.....5
2.3 Remove Protection.....6
2.4 Save to Flash.....7
2.5 Encoder.....8
2.6 Button Switch Mode.....9
2.7 RGB LED Control.....10
2.8 RS485 Baud Rate.....11
2.9 Device ID.....11
2.10 Motor Jam Protection.....12
2.11 Motor Position Over Range Protection.....13
3、速度环控制指令集.....14
3.1 Speed Control.....14
3.2 Speed PID Configuration.....15
4、位置环控制指令集.....16
4.1 Position Control.....16
3.2 Position PID Configuration.....17
5、电流环控制指令集.....18
5.1 Current Control.....18
6、 状态读取指令集.....18
6.1 Motor Status Readback.....19
6.2 Other Status Readback.....21
7、 RS485 to I2C 转发控制指令集.....22
7.1 I2C Read Register.....22
7.2 I2C Write Register.....23
7.3 I2C Read Raw.....24
7.3 I2C Write Raw.....25

# 1、通信协议结构

## 1.1 通信协议参数

### 1.1.1 部通信接口的通信参数

采用半双工异步串行通讯。
默认波特率为 115200bps，可定制其他波特率。
数据格式：8 位数据位（低位 在前）和 1 位停止位，无校验位。

## 1.2 数据包格式

发送帧与响应帧都为 15 bytes 固定长度，其中响应帧返回消息以 0xAA 0x55 开头， 0xAA 0x55 不参与 CRC 校验。

### 1.2.1 配置指令发送帧

<table>
<tr><th>命令</th><th>设备 ID</th><th>数据 1</th><th>数据 2</th><th>数据 3</th><th>CRC8</th></tr>
<tr><td>1 byte</td><td>1 byte</td><td>4 bytes</td><td>4 bytes</td><td>4 bytes</td><td>1 byte</td></tr>
<tr><td rowspan="4">参数说明</td><td>命令</td><td colspan="5">配置指令</td></tr>
<tr><td>设备 ID</td><td colspan="5">设备 ID 默认为 0x00，支持设置范围为 0x00-0xFF</td></tr>
<tr><td>数据 1-3</td><td colspan="5">每组数据 4 bytes，未使用的数据位可置 0。</td></tr>
<tr><td>CRC8</td><td colspan="5">参考下方代码进行校验值计算，所有返回的消息以 0xAA 0x55 开头，0xAA 0x55 不参与 CRC 校验。</td></tr>
</table>

### 1.2.2 配置指令响应帧

| 响应命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|

| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
|---|---|---|---|---|---|
| 参数说明 | 命令 | 响应指令 | | | |
| | 设备 ID | 设备 ID 默认为 0x00，支持设置范围为 0x00-0xFF | | | |
| | 数据 1-3 | 每组数据 4 bytes，未使用的数据位可置 0。 | | | |
| | CRC8 | 参考下方代码进行校验值计算，所有返回的消息以 0xAA 0x55 开头，0xAA 0x55 不参与 CRC 校验 。 | | | |

### 1.2.3 CRC8

```
uint8_t crc8(uint8_t *data, uint8_t len) {
    uint8_t crc, i;
    crc = 0x00;
    while (len--) {
        crc ^= *data++;
        for (i = 0; i < 8; i++) {
            if (crc & 0x01) {
                crc = (crc >> 1) ^ 0x8c;
            } else
                crc >>= 1;
        }
    }
    return crc;
}
```

# 2、配置指令集

## 2.1 Mode Switch

- 功能说明： 电机使能开关。
- 输入参数：

Motor ID(1byte)：设备地址

Status(1byte)：

| 参数 | 功能 | 描述 |
|---|---|---|
| 0x00 | Motor Disable | 电机关闭。 |
| 0x01 | Motor Enable | 电机使能。 |

- 指令代码： 00H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x00 | Motor ID | Status | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x10 | Motor ID | Status | Reserve | Reserve | CRC8 |

- 发送案例：

00 00 01 00 00 00 00 00 00 00 00 00 00 00 68

- 响应案例：

10 00 01 00 00 00 00 00 00 00 00 00 00 00 9A

## 2.2 Mode Setting

- 功能说明： 电机工作模式设置。
- 输入参数：

Motor ID(1byte)：设备地址

Mode(1byte)：

| 参数 | 功能 | 描述 |
|---|---|---|
| 0x01 | Speed Mode | 速度环模式： 控制电机运行在指定目标速度。 |

| | | |
|---|---|---|
| 0x02 | Position Mode | 位置环模式： 控制电机转动至指定位置。 |
| 0x03 | Current Mode | 电流环模式：控制电机运行在指定目标工作电流。 |
| 0x04 | Encoder Mode | 编码器模式：设备作为输入设备，采集当前旋转编码器数值。 |

- 指令代码： 01H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x01 | Motor ID | Mode | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x11 | Motor ID | Mode | Reserve | Reserve | CRC8 |

- 发送案例：

01 00 01 00 00 00 00 00 00 00 00 00 00 00 44

- 响应案例：

11 00 01 00 00 00 00 00 00 00 00 00 00 00 B6

## 2.3 Remove Protection

- 功能说明： 解除堵转 Jam 保护，当发生堵转锁定保护后，发送该指令进行解锁。
- 输入参数：

Motor ID(1byte)：设备地址

Status (1byte)：

1：解除堵转保护

- 指令代码： 06H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x06 | Motor ID | Reserve | Status | Reserve | CRC8 |

应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x16 | Motor ID | Reserve | Status | Reserve | CRC8 |

- 发送案例：

06 00 00 00 00 00 01 00 00 00 00 00 00 00 AB

- 响应案例：

16 00 00 00 00 00 00 00 00 00 00 00 00 00 1A

## 2.4 Save to Flash

- 功能说明： 保存配置参数到内部 Flash
- 输入参数：

  Motor ID(1byte)：设备地址

  Status (1byte)：

  1：执行保存

- 指令代码： 07H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x07 | Motor ID | Status | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x17 | Motor ID | Status | Reserve | Reserve | CRC8 |

- 发送案例：

07 00 01 00 00 00 00 00 00 00 00 00 00 00 AC

- 响应案例：

17 00 01 00 00 00 00 00 00 00 00 00 00 00 5E

## 2.5 Encoder

- 功能说明： 设置当前编码器数值
- 输入参数：

  Motor ID(1byte)：设备地址

  Encoder(4byte)：

  Encoder value(int32_t) Encoder = Encoder-byte0 + Encoder-byte1 * 256 + Encoder-byte2 * 65536 + Encoder-byte3 * 16777216

- 指令代码： 08H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x08 | Motor ID | Encoder | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x18 | Motor ID | Encoder | Reserve | Reserve | CRC8 |

- 发送案例：

08 00 64 00 00 00 00 00 00 00 00 00 00 00 06

- 响应案例：

18 00 64 00 00 00 00 00 00 00 00 00 00 00 F4

## 2.6 Button Switch Mode

- 功能说明： 启用按键模式切换功能
- 输入参数：

  Motor ID(1byte)：设备地址

  Status (1byte)：

  0：禁用按键模式切换功能

  1：启用按键模式切换功能(长按 5s 切换电机工作模式)

- 指令代码： 09H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x09 | Motor ID | Status | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x19 | Motor ID | Status | Reserve | Reserve | CRC8 |

- 发送案例：

09 00 01 00 00 00 00 00 00 00 00 00 00 00 3D

- 响应案例：

19 00 01 00 00 00 00 00 00 00 00 00 00 00 CF

## 2.7 RGB LED Control

- 功能说明： RGB LED 灯颜色，亮度，工作模式控制
- 输入参数：

  Motor ID(1byte)：设备地址

  Status (4byte)：

  Byte0：RGB-R value

  Byte1：RGB-G value

  Byte2：RGB-B value

  Byte3：RGB Mode:

  0:系统默认状态显示

  1:用户自定义控制

  Brightness (1byte)：

  0-100:亮度值

- 指令代码： 0AH
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x0A | Motor ID | Status | Brightness | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x1A | Motor ID | Status | Brightness | Reserve | CRC8 |

- 发送案例：

0A 00 FF 32 32 01 C8 00 00 00 00 00 00 00 3C

- 响应案例：

1A 00 FF 32 32 01 C8 00 00 00 00 00 00 00 CE

## 2.8 RS485 Baud Rate

- 功能说明： RS485 通信接口速率配置
- 输入参数：

  Motor ID(1byte)：设备地址

  Baud (1byte)：

  0：115200bps

  1：19200bps

  2：9600bps

- 指令代码： 0BH
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x0B | Motor ID | Baud | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x1B | Motor ID | Baud | Reserve | Reserve | CRC8 |

- 发送案例：

0B 00 00 00 00 00 00 00 00 00 00 00 00 00 0D

- 响应案例：

1B 00 00 00 00 00 00 00 00 00 00 00 00 00 FF

## 2.9 Device ID

- 功能说明： 配置设备 ID
- 输入参数：

Motor ID(1byte)：设备地址

ID (1byte)：

0-255：设备新 ID

- 指令代码： 0CH
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x0C | Motor ID | ID | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x1C | Motor ID | ID | Reserve | Reserve | CRC8 |

- 发送案例：

0C 00 01 00 00 00 00 00 00 00 00 00 00 00 A1

- 响应案例：

1C 00 01 00 00 00 00 00 00 00 00 00 00 00 53

## 2.10 Motor Jam Protection

- 功能说明： 设备堵转 Jam 保护启用设置，当发生堵转锁定保护后，电机将锁定停止转动，此时需要发送该 Remove Protection 指令进行解锁
- 输入参数：

Motor ID(1byte)：设备地址

Protection (1byte)：

0：禁用堵转保护

1：启用堵转保护

- 指令代码： 0DH
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x0D | Motor ID | Protection | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x1D | Motor ID | Protection | Reserve | Reserve | CRC8 |

- 发送案例：

0D 00 01 00 00 00 00 00 00 00 00 00 00 00 8D

- 响应案例：

1D 00 01 00 00 00 00 00 00 00 00 00 00 00 7F

## 2.11 Motor Position Over Range Protection

- 功能说明： 设置电机旋转范围保护。开启后当电机编码数值<-2100000000 或>2100000000 时，电机将停止旋转进入保护状态。

- 输入参数：

  Motor ID(1byte)：设备地址

  Protection (1byte)：

  0：禁用旋转范围保护

  1：启用旋转范围保护

- 指令代码： 0EH
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x0E | Motor ID | Protection | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x1E | Motor ID | Protection | Reserve | Reserve | CRC8 |

- 发送案例：

0E 00 01 00 00 00 00 00 00 00 00 00 00 00 F9

- 响应案例：

1E 00 01 00 00 00 00 00 00 00 00 00 00 00 0B

# 3、速度环控制指令集

## 3.1 Speed Control

- 功能说明： 配置目标运行速度和最大电流值限制
- 输入参数：

Motor ID(1byte)：设备地址

Speed (4byte)：

Speed Setting(int32_t) = Speed Setting-byte0 + Speed Setting-byte1 * 256 + Speed Settingbyte2 * 65536 + Speed Setting-byte3 * 16777216

Actual Speed Setting(RPM) = Speed Setting/100

支持输入范围：-2100000000-2100000000

Current (4byte)：

Max Current = Max Current-byte0 + Max Current-byte1 * 256 + Max Current-byte2 * 65536 + Max Current-byte3 * 16777216 Actual Max Current(mA) = Max Current/100

支持输入范围：-120000-120000

- 指令代码： 20H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |

| | | | | | |
|---|---|---|---|---|---|
| 0x20 | Motor ID | Speed | Current | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x30 | Motor ID | Speed | Current | Reserve | CRC8 |

- 发送案例：

20 00 80 A9 03 00 C0 D4 01 00 00 00 00 00 7C

- 响应案例：

30 00 80 A9 03 00 C0 D4 01 00 00 00 00 00 8E

## 3.2 Speed PID Configuration

- 功能说明： 速度环 PID 参数配置
- 输入参数：

  Motor ID(1byte)：设备地址

  Speed (4byte)：

P/I/D(uint32_t): PID = PID-byte0 + PID-byte1 * 256 + PID-byte2 * 65536 + PID-byte3 * 16777216

For example: P=0.0000001, P setting value=0.0000001*10000000=1, P-byte0=0x01, P-byte1=0, Pbyte2=0, P-byte3=0Current (4byte)：

- 指令代码： 21H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x21 | Motor ID | P | I | D | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|

| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
|---|---|---|---|---|---|
| 0x31 | Motor ID | P | I | D | CRC8 |

- 发送案例：

21 00 60 E3 16 00 E8 03 00 00 00 5A 62 02 D8

- 响应案例：

31 00 60 E3 16 00 E8 03 00 00 00 5A 62 02 2A

# 4、位置环控制指令集

## 4.1 Position Control

- 功能说明： 配置脉冲数来控制目标旋转至指定位置，同时配置最大电流值限制
- 输入参数：

  Motor ID(1byte)：设备地址

  Position (4byte)：

Position Setting(int32_t) = Position Setting-byte0 + Position Setting-byte1 * 256 + Position Setting-byte2 * 65536 + Position Setting-byte3 * 16777216 Actual Position Setting = Position Setting/100

输入范围：-2100000000-2100000000

Current (4byte)：

Max Current = Max Current-byte0 + Max Current-byte1 * 256 + Max Current-byte2 * 65536 + Max Current-byte3 * 16777216 Actual Max Current = Max Current/100

支持输入范围：-120000-120000

- 指令代码： 22H

- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x22 | Motor ID | Position | Current | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x32 | Motor ID | Position | Current | Reserve | CRC8 |

- 发送案例：

22 00 60 E3 16 00 C0 D4 01 00 00 00 00 00 67

- 响应案例：

32 00 60 E3 16 00 C0 D4 01 00 00 00 00 00 95

## 3.2 Position PID Configuration

- 功能说明： 位置环 PID 参数配置
- 输入参数：

  Motor ID(1byte)：设备地址

  Speed (4byte)：

P/I/D(uint32_t): PID = PID-byte0 + PID-byte1 * 256 + PID-byte2 * 65536 + PID-byte3 * 16777216

For example: P=0.0000001, P setting value=0.0000001*10000000=1, P-byte0=0x01, P-byte1=0, Pbyte2=0, P-byte3=0Current (4byte)：

- 指令代码： 23H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x23 | Motor ID | P | I | D | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x33 | Motor ID | P | I | D | CRC8 |

- 发送案例：

23 00 60 E3 16 00 1E 00 00 00 00 5A 62 02 73

- 响应案例：

33 00 60 E3 16 00 1E 00 00 00 00 5A 62 2 81

# 5、电流环控制指令集

## 5.1 Current Control

- 功能说明： 配置目标运行电流
- 输入参数：

  Motor ID(1byte)：设备地址

  Current (4byte)：

Current Setting(int32_t) = Current Setting-byte0 + Current Setting-byte1 * 256 + Current Setting-byte2 * 65536 + Current Setting-byte3 * 16777216 Actual Current Setting = Current Setting/100

输入范围：-120000-120000

- 指令代码： 24H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |
| 0x24 | Motor ID | Current | Reserve | Reserve | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | CRC8 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte |

| | | | | | |
|---|---|---|---|---|---|
| 0x34 | Motor ID | Current | Reserve | Reserve | CRC8 |

- 发送案例：

24 00 C0 D4 01 00 00 00 00 00 00 00 00 00 54

- 响应案例：

34 00 C0 D4 01 00 00 00 00 00 00 00 00 00 A6

# 6、 状态读取指令集

状态读取的请求帧与响应帧与配置指令集的数据格式有所不同，额外加入了 3bytes 用于表示工作模式，电机状态，错误码。

## 6.1 Motor Status Readback

- 功能说明： 电机当前速度值，编码器位置值，电流值，工作模式，工作状态，错误代码信息读取。
- 输入参数：

  Motor ID(1byte)：设备地址

  Read (1byte)：0

- 返回参数：

  Speed (4byte)：当前电机速度(RPM)

  Speed Readback = Speed Readback-byte0 + Speed Readback-byte1 * 256 + Speed Readback-byte2 * 65536 + Speed Readback-byte3 * 16777216

  Actual Speed Readback = Speed Readback/100

  Position(4byte)：当前电机位置

  Position Readback = Position Readback-byte0 + Position Readback-byte1 * 256 + Position Readback-byte2 * 65536 + Position Readback-byte3 * 16777216

Actual Position Readback = Position Readback/100

Current(4byte)：当前电机电流(mA)

Current Readback = Current Readback-byte0 + Current Readback-byte1 * 256 + Current Readback-byte2 * 65536 + Current Readback-byte3 * 16777216

Actual Current Readback = Current Readback/100

Mode(1byte)：

1: Speed Mode

2: Position Mode

3: Current Mode

4: Encoder Mode

Status(1byte)：当前电机状态

0: Standby

1: Running

2: Error

Error(1byte)：

1: Overvoltage

2: Stalled

4: Over Range

- 指令代码： 40H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | CRC8 |
|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte |
| 0x40 | Motor ID | Read | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | 数据 4 | 数据 5 | 数据 6 | CRC8 |
|---|---|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 byte | 4 bytes | 1 byte | 1 byte | 1 byte | 1 byte |
| 0x50 | Motor ID | Speed | Position | Current | Mode | Status | Error | CRC8 |

- 发送案例：

40 00 00 31

- 响应案例：

50 00 01 00 00 00 78 FB FF FF F7 FF FF FF 01 00 00 8B

## 6.2 Other Status Readback

- 功能说明： 电机当前输入电压，温度值，电流值，工作模式，工作状态，错误代码信息读取。
- 输入参数：

  Motor ID(1byte)：设备地址

  Read (1byte)：0

- 返回参数：

  VIN (4byte)：输入电压值(V)

  VIN X100 = VIN X100-byte0 + VIN X100-byte1 * 256 + VIN X100-byte2 * 65536 + VIN X100-byte3 * 16777216

  Actual VIN = VIN X100/100

  Temp(4byte)：内部温度参考值(° C)

  Temp = Temp-byte0 + Temp-byte1 * 256 + Temp-byte2 * 65536 + Temp-byte3 * 16777216

  Encoder Counter(4byte)：Encoder 模式下，用于读取当前编码数值

  Encoder Counter = Encoder Counter-byte0 + Encoder Counter-byte1 * 256 + Encoder Counter-byte2 * 65536 + Encoder Counter-byte3 * 16777216

  RGB Mode(1byte)：RGB 工作模式

  0:系统默认状态显示

  1:用户自定义控制

  RGB Brightness(1byte)：RGB 亮度

  0-100:亮度值

- 指令代码： 41H

- 指令包格式：

| 命令 | 设备 ID | 数据 1 | CRC8 |
|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte |
| 0x41 | Motor ID | Read | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | 数据 4 | 数据 5 | 数据 6 | CRC8 |
|---|---|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 4 bytes | 4 bytes | 4 bytes | 1 byte | 1 byte | 1 byte | 1 byte |
| 0x51 | Motor ID | VIN | Temp | Encoder Counter | RGB Mode | RGB Brightness | Reserve | CRC8 |

- 发送案例：

41 00 00 9A

- 响应案例：

51 00 1D 05 00 00 2B 00 00 00 00 00 00 00 01 64 00 CD

# 7、 RS485 to I2C 转发控制指令集

该指令集用于发送控制指令，通过 RS485 接口指令的方式，实现 Roller485 电机 I2C 端口的数据读写。

## 7.1 I2C Read Register

- 功能说明： I2C 读取寄存器指令
- 输入参数：

Motor ID(1byte)：设备地址

I2C Address (1byte)：从机地址

Register Address Length (1byte)：

0：请求寄存器地址为 1 byte 长度

1：请求寄存器地址为 2 byte 长度

Register (2byte)：寄存器地址

Data Length (1byte)：请求读取数据长度(最大支持 16byte)

- 返回参数：

  Read Status (1byte)：

  1：读取成功

  0：读取失败

  Data Length (1byte)：请求读取数据长度

  Data(16byte)：请求数据

- 指令代码： 60H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | 数据 4 | CRC8 |
|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte | 2 bytes | 1 byte | 1 byte |
| 0x60 | Motor ID | I2C Address | Register Address Length | Register Address | Data Length | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | 数据 4 | 数据 5 | CRC8 |
|---|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte | 1 byte | 3 bytes | 16 bytes | 1 byte |
| 0x70 | Motor ID | Read Status | Reserve | Data Length | Reserve | Data | CRC8 |

- 发送案例：

60 00 29 00 14 00 0C 54

- 响应案例：

70 00 01 00 0C 00 00 00 5F 06 05 00 FF FF 0B A6 00 00 00 4E 00 00 00 00 DC

## 7.2 I2C Write Register

- 功能说明： I2C 写入寄存器指令
- 输入参数：

  Motor ID(1byte)：设备地址

  I2C Address (1byte)：从机地址

  Register Address Length (1byte)：

0：写入寄存器地址为 1 byte 长度

1：写入寄存器地址为 2 byte 长度

Register (2byte)：寄存器地址

Data Length (1byte)：请求读取数据长度(最大支持 16byte)

Data (16byte)：写入数据

- 返回参数：

Write Status (1byte)：

1：写入成功

0：写入失败

- 指令代码： 61H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | 数据 4 | 数据 5 | 数据 6 | CRC8 |
|---|---|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte | 2 bytes | 1 byte | 3 bytes | 16 bytes | 1 byte |
| 0x61 | Motor ID | I2C Address | Register Address Length | Register Address | Data Length | Reserve | Data | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | CRC8 |
|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte |
| 0x71 | Motor ID | Write Status | CRC8 |

- 发送案例：

61 00 26 00 11 00 01 00 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 15

- 响应案例：

71 00 01 1A

## 7.3 I2C Read Raw

- 功能说明： I2C 读取从机数据指令

- 输入参数：

  Motor ID(1byte)：设备地址

  I2C Address (1byte)：从机地址

  Data Length (1byte)：请求读取数据长度(最大支持 16byte)

- 返回参数：

  Read Status (1byte)：

  1：读取成功

  0：读取失败

  Data Length (1byte)：请求读取数据长度

  Data(16byte)：请求数据

- 指令代码： 62H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | CRC8 |
|---|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte | 1 byte |
| 0x62 | Motor ID | I2C Address | Data Length | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | 数据 4 | 数据 5 | CRC8 |
|---|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte | 1 byte | 3 bytes | 16 bytes | 1 byte |
| 0x72 | Motor ID | Read Status | Reserve | Data Length | Reserve | Data | CRC8 |

- 发送案例：

62 00 57 03 6C

- 响应案例：

72 00 01 00 03 00 00 00 00 B3 08 00 00 00 00 00 00 00 00 00 00 00 00 00 F1

## 7.3 I2C Write Raw

- 功能说明： I2C 写入从机数据指令
- 输入参数：

  Motor ID(1byte)：设备地址

I2C Address (1byte)：从机地址

Data Length (1byte)：请求读取数据长度(最大支持 16byte)

Stop Bit(1byte)：

0：不带停止位

1：带停止位

Data (16byte)：写入数据

- 返回参数：

Write Status (1byte)：

1：写入成功

0：写入失败

- 指令代码： 63H
- 指令包格式：

| 命令 | 设备 ID | 数据 1 | 数据 2 | 数据 3 | 数据 4 | 数据 5 | CRC8 |
|---|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte | 1 byte | 3 bytes | 16 bytes | 1 byte |
| 0x63 | Motor ID | I2C Address | Data Length | Stop Bit | Reserve | Data | CRC8 |

- 应答包格式：

| 命令 | 设备 ID | 数据 1 | CRC8 |
|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte |
| 0x73 | Motor ID | Write Status | CRC8 |

- 发送案例：

63 00 57 02 01 00 00 00 01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 C1

- 响应案例：

73 00 01 55