

QD4310 使用手册

V2.2

目录

| | |
|-----------------------|----|
| QD4310 使用手册..... | 1 |
| 一、 产品介绍..... | 3 |
| 1.1. 产品简介..... | 3 |
| 1.2. 产品特点..... | 3 |
| 1.3. 参数详情..... | 4 |
| 1.4. 产品尺寸..... | 4 |
| 1.5. 接口定义..... | 5 |
| 二、 初次使用..... | 6 |
| 2.1. 连接电源线..... | 6 |
| 2.2. 连接上位机..... | 6 |
| 2.3. 校准与简单运行..... | 7 |
| 三、 校准、调参与 USB 控制..... | 8 |
| 3.1. USB 接线..... | 8 |
| 3.2. 电机校准..... | 8 |
| 3.3. 上位机调参..... | 9 |
| 3.4. 零点设置..... | 11 |
| 3.5. USB 控制..... | 11 |
| 3.5.1. 上位机控制..... | 11 |
| 3.5.2. 程序控制..... | 13 |

| | | |
|--------|-------------------|----|
| 四、 | 上位机命令说明 | 15 |
| 4.1. | 查询指令 | 15 |
| 4.2. | 控制指令 | 15 |
| 4.3. | 配置指令 | 15 |
| 4.4. | 其他指令 | 15 |
| 五、 | CAN 总线控制 | 16 |
| 5.1. | CAN 总线概述 | 16 |
| 5.2. | CAN 总线控制接线 | 16 |
| 5.3. | 控制方法 | 18 |
| 5.3.1. | 设置电机 CAN ID | 18 |
| 5.3.2. | CAN 数据包定义 | 18 |
| 5.3.3. | 程序编写指南 | 22 |
| 六、 | UART 控制 | 23 |
| 6.1. | UART 控制接线 | 23 |
| 6.2. | 控制方法 | 23 |
| 6.2.1. | 设置电机 ID | 23 |
| 6.2.2. | 数据包定义 | 23 |
| 6.2.3. | 程序编写指南 | 25 |
| 7. | PWM 控制 | 26 |
| 7.1. | PWM 控制接线 | 26 |
| 8. | 升级固件 | 27 |
| 9. | 常见问题与注意事项 | 30 |
| 10. | 修改记录 | 31 |

一、产品介绍

1.1. 产品简介

QD4310 是袖珍电子全新推出的无刷驱控一体伺服电机。它集成了无刷电机闭环驱动器，并融合卓越的闭环控制算法，具有极高的控制精度和极快的响应速度，以及其他更多强大的功能。QD4310 通过集成无刷驱动器，免去其他繁琐的控制流程，用户仅需通过集成的 CAN 总线、USB、UART、PWM 与电机通信，即可轻松操纵电机，享受无刷电机的优异性能。

QD4310 具有不错的安全性和可靠性，满足更广大用户的需求，适用于二维云台、机械臂末端、小车比赛等众多场景，欢迎大家使用。

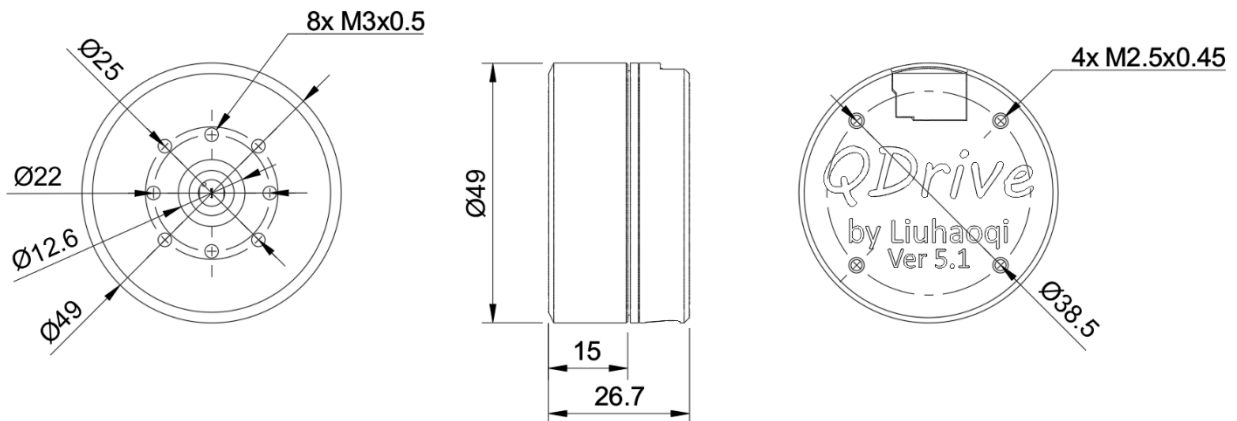
1.2. 产品特点

- ◆ QD4310 采用先进的 FOC 增量闭环控制算法，支持力矩、速度、位置三环控制；
- ◆ 搭载 15 位工业级高精度磁编码器，等效 32768 线，角度分辨率达 0.0109 度；
- ◆ 命令行配置电机，支持最大电流限制、最大转速限制，内置 PID 参数等配置；
- ◆ 定制无刷电机，齿槽转矩小，转矩波动小，转速稳定无抖动，电机运行无噪声；
- ◆ 板载精密电流传感器，实现精准电流(力矩)控制；
- ◆ 最高转速 800RPM，最低转速低至 0.1RPM，适用于高精度场合；
- ◆ 支持 7~26V 宽电压输入，完美兼容 2S~6S 电池组；
- ◆ 支持 Type-C 口直连电脑/树莓派，实现一键控制；
- ◆ 支持实时转速 RPM、相电流 mA、总线电压 mV 的回传；
- ◆ 支持 CAN 总线控制、UART 控制、PWM 控制，一条通信线路可串联多达 8 个电机；
- ◆ 支持编码器自校准、抗齿槽补偿、恢复出厂设置等功能；

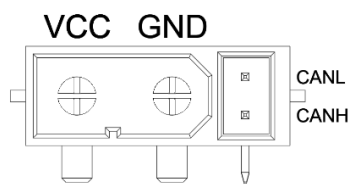
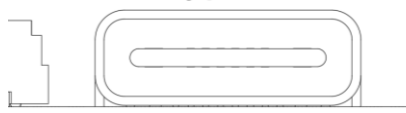
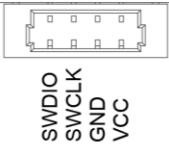
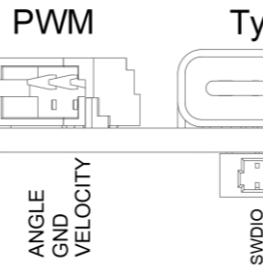
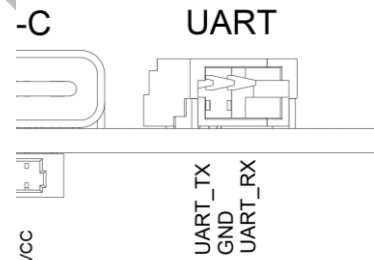
1.3. 参数详情

| 参数 | 数值 | 单位 |
|----------|----------------------------------|-----|
| 额定电压 | 7~26 | V |
| 额定电流 | 1 | A |
| 峰值电流 | 2 | A |
| 额定扭矩 | 0.2 | Nm |
| 峰值扭矩 | 0.3 | Nm |
| 额定转速 | 500 | rpm |
| 峰值转速 | 800 | rpm |
| 极对数 | 14 | |
| 编码器位数 | 15 | bit |
| 控制精度 | 0.0109 | ° |
| 控制模式 | 电流模式、速度模式、角度模式 低速大力矩模式、角度步进模式 | |
| 通信方式 | CAN、USB、UART | |
| CAN 波特率 | 1M | bps |
| UART 波特率 | 115200 (可调) | bps |

1.4. 产品尺寸



1.5. 接口定义

| 接口名称 | 接口标注 | 说明 |
|------------|---|---|
| XT30(2+2) |  | <ol style="list-style-type: none"> 1. 通过 XT30(2+2)-F 插头的电源连接线来连接电源, 额定电压为 7~26V, 为电机供电 2. 通过 CAN 通信端子连接外部控制设备, 可接收 CAN 控制命令, 反馈电机状态信息 |
| Type-C |  | <ol style="list-style-type: none"> 1. 通过上位机控制和配置电机 2. 使用 Type-C 接口的数据线连接至 PC/ K230/XX 派 等终端, 可以通过 USB 虚拟串口控制电机 |
| HC0.8-4Pin |  | <p>使用 HC0.8-4Pin 接口连接线连接至 stlink/daplink/jlink 等烧录器, 可以为电机升级固件</p> |
| SH1.0-3Pin |  | <p>使用 SH1.0-3Pin 接口连接线连接至控制器, 可以通过调整 PWM 信号的占空比控制电机转动的速度和角度</p> |
| SH1.0-3Pin |  | <p>使用 SH1.0-3Pin 接口连接线连接至控制器, 可以通过发送 UART 数据控制电机转动</p> |

二、初次使用

2.1. 连接电源线

QD4310 的供电接口为 XT30(2+2)，用户可以使用 XT30(2+2)-F、XT30U-F 为电机供电。接线图如图所示：

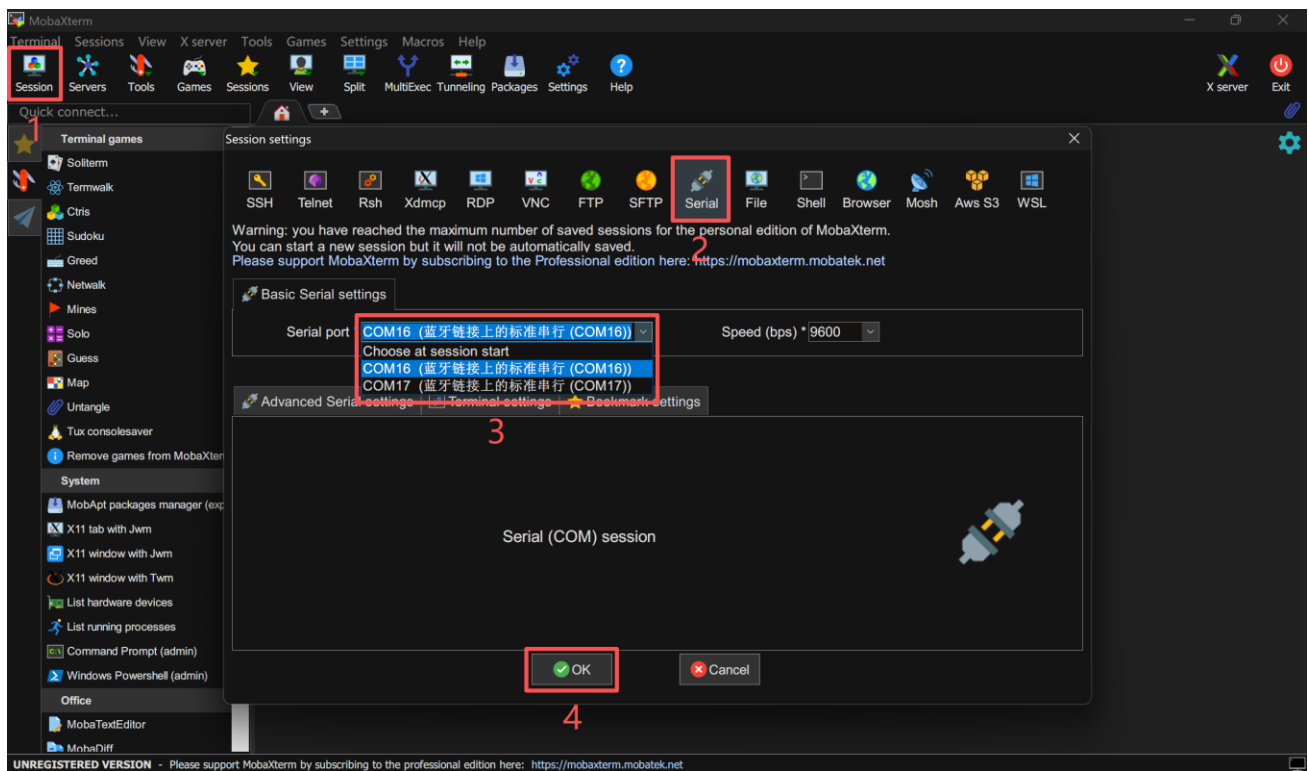


其中红色为电源正极，黑色为电源负极，XT30U-F 插头上亦有+-标注，请务必注意线序。供电电压为 7~26V，可以使用 2S~6S 电池组为电机供电。(对于有电池的应用常见，建议电池直连电机供电)。

2.2. 连接上位机

QD4310 设计有 Type-C 接口，使用 USB A to C 或 USB C to C 将其连接至电脑，即可进行简单的控制、校准及调参。

使用数据线连接电脑，打开 MobaXterm 软件，点击左上角 Session 按钮，在新弹出的页面中选择 Serial 选项，Serial Port 选择电机对应的 COM 口，波特率任意，并点击 OK。如图所示：



2.3. 校准与简单运行

新电机需执行校准后才能运行，但商品发货时为进行质检，会校准电机，故无需再次校准。如您仍需校准，请参考 [3.2.电机校准](#)。

输入“enable”命令（并回车），电机使能，绿灯亮起。输入“ctrl speed 10”命令（并回车），电机将以 10rpm 的速度逆时针旋转。

三、校准、调参与 USB 控制

3.1. USB 接线

接线顺序参考 [2.1.连接电源线](#)和 [2.2.连接上位机](#)。

3.2. 电机校准

电机校准分为基础校准和抗齿槽校准。其中，基础校准为必须步骤，新电机需执行校准后才能运行。若未进行校准，电机将无法使能。若校准出现异常，电机将会异常转动，此时需要重新进行电机校准。

校准步骤如下：

1. 保证电源连接：确保正确连接电源供电，校准推荐供电电压 8~20V，电源供电能力需大于 2A。
2. 确保电机空载：校准时需确保电机尽可能空载，几乎未连接任何结构件，没有物体阻挡电机旋转。
3. 确保电机失能：电机使能（绿灯亮起）时无法进行校准。
4. 输入校准命令：在上位机输入“calibrate”命令（并回车）进行基础校准，若一切正常将开始校准，若重复校准，请输入'y'确认。
5. 等待校准完成：基础校准持续约 10s，抗齿槽校准会持续数分钟，确保电机空载并耐心等待校准完成，校准完成后上位机会有字符提示。

抗齿槽校准可以有效削弱电机转动时的齿槽感，提升低速运行时的稳定性。但截至目前“Software version 6.1.1”（使用“version”命令查看），此功能仍处于内测阶段，暂不予开放。

3.3. 上位机调参

QD4310 支持上位机调参，可调参数包括内置速度环 PID 参数、内置角度环 PID 参数、电流限幅、速度限幅、CAN 总线 ID、UART 波特率等。调参时只需连接 Type-C 线，电源线非必要。

- 查看当前参数

输入“config --list”命令即可查看当前参数配置，如图所示：

```
QDrive:/$ config --list
Current Configuration:
pid.speed.kp = 0.00300
pid.speed.ki = 0.000390
pid.speed.kd = 0.000
pid.angle.kp = 1200
pid.angle.ki = 0.000
pid.angle.kd = 0.000
limit.speed = 1000 rpm
limit.current = 1.65 A
can.id = 000
can.baud_rate = 1'000'000
uart.baud_rate = 115200
QDrive:/$
```

- 设置参数

输入“config pid.speed.kp 0.003”或“config pid.speed.kp=0.003”即可设置速度环 kp 参数为 0.003。其他参数类似，如图所示：

```
QDrive:/$ config --help
Usage: config [--list | PARAM_PATH VALUE | key=value]

Examples:
config pid.speed.kp 0.1
config pid.speed.ki=0.1
config --help
config --list

Configuration Parameters:
pid.speed.kp      : Speed PID proportional gain
pid.speed.ki      : Speed PID integral gain
pid.speed.kd      : Speed PID derivative gain
pid.angle.kp      : Angle PID proportional gain
pid.angle.ki      : Angle PID integral gain
pid.angle.kd      : Angle PID derivative gain
limit.speed       : Speed limit in rpm
limit.current     : Current limit in A
can.id            : CAN ID of the motor (0-7)
uart.baud_rate    : UART BaudRate of the motor (50K-10M)
zero_pos         : Position zero offset in rad
QDrive:/$
```

- 储存参数

使用“config”命令设置参数后，参数只会被临时储存，断电便会恢复原始数值。此时需要运行“store”命令将参数储存进电机内部储存器。输入“store”后，会提示将要储存的参数数值，输入'y'确认储存，输入'n'中止储存。如图所示：

```
QDrive:/$ store
Current Configuration:
pid.speed.kp = 0.00300
pid.speed.ki = 0.000390
pid.speed.kd = 0.000
pid.angle.kp = 1200
pid.angle.ki = 0.000
pid.angle.kd = 0.000
limit.speed = 1000 rpm
limit.current = 1.65 A
can.id = 000
can.baud_rate = 1'000'000
uart.baud_rate = 115200
Are you sure you want to store configurations? (y/n)
Store configuration completed

QDrive:/$ █
```

- 恢复出厂设置

更改参数后若效果不理想，可以使用“restore”命令恢复出厂参数。输入“restore”命令后，输入'y'确认恢复出厂设置，输入'n'中止。如图所示：

```
QDrive:/$ restore
Are you sure you want to restore factory settings? (y/n)
QDrive factory restore completed
Current Configuration:
pid.speed.kp = 0.00300
pid.speed.ki = 0.000390
pid.speed.kd = 0.000
pid.angle.kp = 1200
pid.angle.ki = 0.000
pid.angle.kd = 0.000
limit.speed = 1000 rpm
limit.current = 1.65 A
can.id = 000
can.baud_rate = 1'000'000
uart.baud_rate = 115200

QDrive:/$ █
```

3.4. 零点设置

使用“config zero_pos”命令可以设置电机零点。输入“config zero_pos”命令，设置当前角度为零点，即当前角度变为 0rad。输入“config zero_pos xxx”，设置 xxx rad 为零点。设置零点立即生效，无需使用“store”命令保存设置。如图所示：

```
QDrive:/$ config zero_pos
Setting config [zero_pos]

QDrive:/$ config zero_pos 1.23
Setting config [zero_pos]

QDrive:/$ █
```

3.5. USB 控制

3.5.1. 上位机控制

使用上位机可以进行电机的简单控制，电机控制模式包括：电流模式、速度模式、角度模式。参考 [2.2.连接上位机](#) 章节连接上位机，若未进行校准，参考 [3.2.电机校准](#) 校准电机。输入“enable”命令使能电机，绿灯亮起。

- 电流模式

输入命令“ctrl current 0.1”或“ctrl current=0.1”，电机将以 0.1A 的 Q 轴电流沿正方向（逆时针）转动。QD4310 的转矩常数为 0.3Nm/A，即此时电机扭矩为 0.03Nm。如图所示：

```
QDrive:/$ enable
QDrive enabled

QDrive:/$ ctrl current 0.1
Setting current = 0.10 A

QDrive:/$ █
```

注意：电流值不会超过 [3.3.上位机调参](#) 中 limit.current 的限制，即使成功显示 Setting current = 10.00 A。

- 速度模式

输入命令“ctrl speed 10”或“ctrl speed=10”，电机将以 10rpm 的转速沿正方向（逆时针）转动。如图所示：

```
QDrive:/$ enable
QDrive enabled

QDrive:/$ ctrl speed 10
Setting speed = 10.00 rpm

QDrive:/$ █
```

注意：对于转速低于 5rpm 的应用场景，推荐使用 low_speed 命令代替 speed 命令，或许会有更好的控制效果。转速值不会超过 [3.3.上位机调参](#) 中 limit.speed 的限制，即使成功显示 Setting speed = 5000.00 rpm。

- 角度模式

输入命令“ctrl angle 1”或“ctrl angle=1”，电机将转动到 1rad 处停下。如图所示：

```
QDrive:/$ enable
QDrive enabled

QDrive:/$ ctrl angle 1
Setting angle = 1.00 rad

QDrive:/$ █
```

角度控制单位为弧度 rad，范围无限，电机内部会进行浮点取余。例如：“ctrl angle 1”与“ctrl angle 7.28”效果一致，“ctrl angle -3”与“ctrl angle 3.28”效果一致。同时，转动方向为沿着较近方向转动，例如当前角度为 0.0rad（使用“status”命令查看），输入“ctrl angle 3.28”将会沿着负方向转动（ $3.28-0=3.28$ ， $0+6.28-3.28=3$ ， $3<3.28$ ）。

使用角度模式控制时，电机将会以 [3.3.上位机调参](#) 中 limit.speed 的速度上限转动（前提是

能达到此速度)。例如，使用“config limit.speed 10”将最大速度限制到 10rpm，再输入“ctrl angle 3”，电机将以 10rpm 的速度转动到 3rad 的位置。类似于达妙电机的位置速度模式。

- 角度步进模式

输入命令“ctrl step_angle 1”或“ctrl step_angle=1”，电机将沿着正方向转动 1rad 后停下。如图所示：

```
QDrive:/$ enable
QDrive enabled

QDrive:/$ ctrl step_angle 1
Stepping 1.00 rad angle

QDrive:/$ step_angle
```

角度步进控制单位为弧度 rad，步进范围无限，正数表示沿着正方向步进，负数表示沿着负方向步进。例如：“ctrl step_angle 100”电机将沿着正方向转动约 16 圈后停下，“ctrl step_angle -10”电机将沿着负方向转动约 1.6 圈后停下。同时，“status”中的电机角度不会随着“ctrl step_angle”命令的递增而超过 $0\sim 2\pi$ 的范围，即若当前角度为 5.28rad，执行“ctrl step_angle 3”命令后，电机沿着正方向转动 3rad，当前角度变为 2.28rad ($5.28+3-6.28$)。

使用角度步进模式控制时，电机将会以 [3.3.上位机调参](#) 中 limit.speed 的速度上限转动（前提是能达到此速度）。例如，使用“config limit.speed 10”将最大速度限制到 10rpm，再输入“ctrl step_angle 3”，电机将以 10rpm 的速度沿着正方向转动 3rad 后停止。类似于达妙电机的位置速度模式。

3.5.2. 程序控制

对于使用树莓派等上位机的小型简单系统，可以使用程序通过 Type-C 接口对 QD4310 进行简单控制。这将尤为方便，通过直接给电机发送指令，使用电机内置速度模式、角度模式进

行控制而无需增加一块 stm32 进行电机控制。

常见的语言如 Python、C/C++，只需在代码中打开 USB 串口设备，并往 USB 串口设备中发送指令即可实现程序控制。

- Python

```
# 打开 USB 串口设备 (COM*改成电机对应 COM 口)
ser = serial.Serial('COM*', baudrate=115200, timeout=0.005)
ser.write(f'enable\n'.encode()) # 使能电机
ser.write(f'ctrl speed {10}\n'.encode()) # 以 10rpm 转速转动
```

- C/C++

代码量较大，详见参考例程

对于不需要读取电机状态的场景下，推荐在电机使能后向电机发送“silent”命令，用于静默电机返回数据，以降低 USB 带宽占用，提高控制频率。对于 Python 语言，对电机发送指令的频率不可超过 25Hz，对于 C/C++ 语言，使用“silent”静默电机后，发送指令的频率不可超过 70Hz。

详细代码示例参考例程文件。

四、上位机命令说明

4.1. 查询指令

- clear: 清空终端
- help: 显示 QD4310 命令帮助信息
- version: 显示电机硬件和软件版本信息
- info: 显示极对数、相电阻、相电感等电机固有信息
- status: 显示电压、电流、转速、控制模式等运行状态

4.2. 控制指令

- enable: 使能电机, 注意未做基础校准无法使能
- disable: 失能电机, 电机失能后控制状态一直保留, 重新使能恢复原有控制状态
- silent: 静默输出, 用于 USB 程序控制, 详见 [3.5.2.程序控制](#)
- reboot: 重启电机, 注意重启后上位机需重新连接
- ctrl: 控制电机, 包括电流模式、速度模式、角度模式, 详见 [3.5.1.上位机控制](#)

4.3. 配置指令

- store: 储存当前配置参数
- restore: 恢复出厂设置
- config: 配置电机参数, 详见 [3.3.上位机调参](#)

4.4. 其他指令

- calibrate: 校准电机
- upgrade: 进入升级模式

五、CAN 总线控制

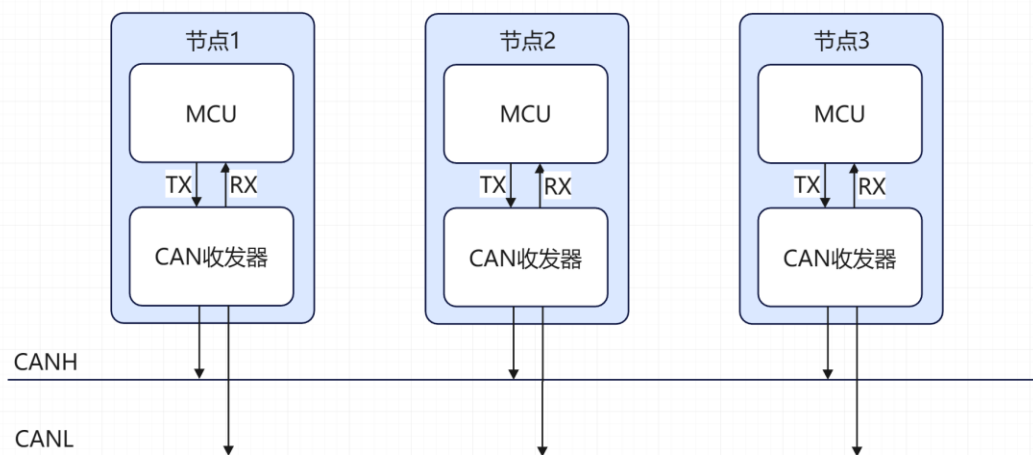
5.1. CAN 总线概述

CAN 总线是一种常用的串行通信总线协议，具有可靠性高、实时性强、支持多节点通信的特点，广泛应用于汽车、工业控制和机器人系统。CAN 总线通信支持多主多从通信，只需 CANH、CANL 两根信号线即可完成信号传输。

QD4310 使用 CAN 总线通信，通信速率为 1Mbps，暂不可调整。一条 CAN 总线上最多可以串联 8 个 QD4310，即一个主机可以同时控制 8 个电机。

5.2. CAN 总线控制接线

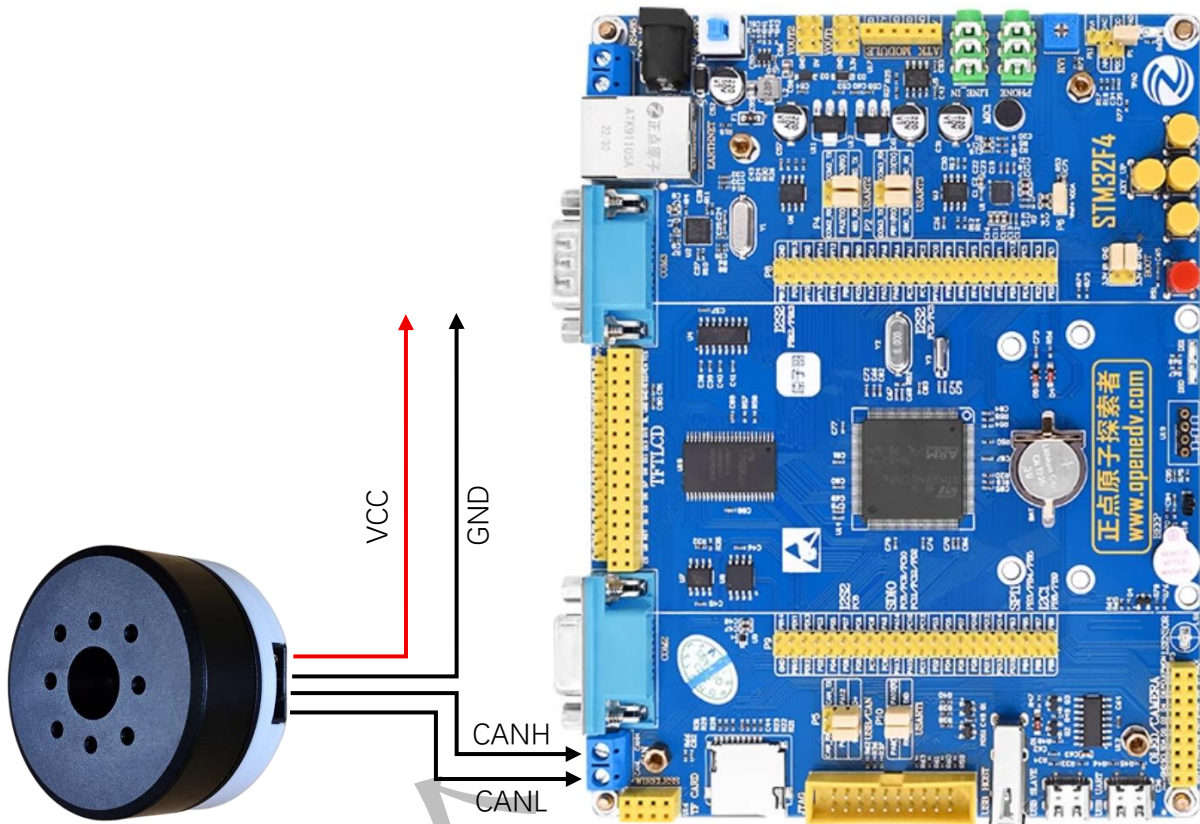
CAN 总线使用差分信号传输，MCU 在使用 CAN 总线通信时，需要通过 CAN PHY 物理层芯片（CAN 收发器）进行电平转换，常见的 CAN PHY 物理层芯片有 SIT3051TK、TJA1050T、SN65HVD251DR 等。CAN 总线通用接线图如图所示：



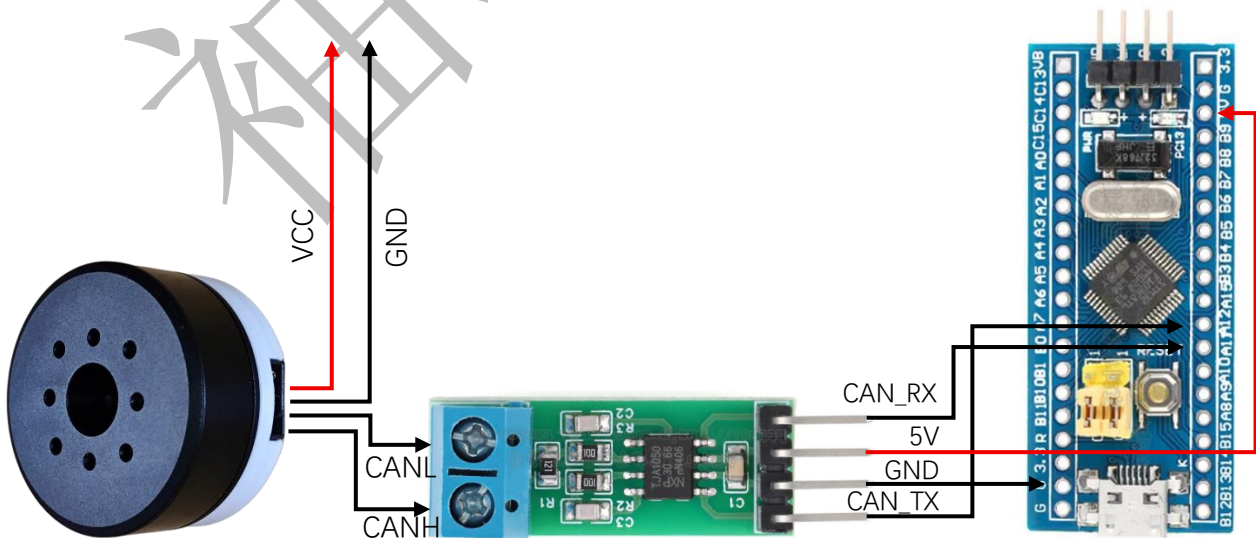
电机内部已经集成 CAN 收发器，使用 XT30(2+2)接口进行 CAN 总线通信。如[接口图](#)所示，从 XT30(2+2)接口引出 CANH、CANL 引脚，将其连接到 CAN 总线上即可进行通信。

若主控板上已经集成 CAN 收发器，如：正点原子探索者、野火霸道-V1，直接将电机 CANH、

CANL 连接到开发板即可。接线如图所示：



若主控板未集成 CAN 收发器，则需要外挂一个 CAN 收发器模块辅助电平转换，推荐[优信电子 TJA1050 模块](#)。接线如图所示：



5.3. 控制方法

5.3.1. 设置电机 CAN ID

CAN 总线支持多节点通信，将多个电机挂载在同一条总线上即可实现多个电机同时控制。同一总线上的电机使用 CAN ID 作为唯一身份标识。使用 CAN 总线控制电机时需先确定被控电机的 CAN ID，电机 CAN ID 需要通过上位机配置，参考 [3.3.上位机调参](#) 章节。

注意：同一条总线上不可挂载两个相同 ID 的 QD4310 电机。

5.3.2. CAN 数据包定义

CAN 总线上的每一次数据传输为一个 CAN 帧，根据 CAN 协议规范，CAN 总线总共有数据帧、遥控帧、错误帧、过载帧、帧间隔 5 种帧。其中，数据帧为实际传输数据所使用的帧。数据帧又分为标准数据帧和扩展数据帧。QD4310 电机仅使用标准数据帧进行通信。

CAN 标准数据帧由一个 11-bit 帧地址和 0~8byte 字节组成。

QD4310 电机 CAN 总线的默认波特率为 1Mbps。

5.3.2.1. 控制报文

主控向电机发送的标准数据帧称为控制报文。通过向电机发送控制报文，可以控制电机使能失能、运行模式和控制量。

控制报文帧地址为 $0x400+ID$ ，其中 ID 范围为 $0x00\sim 0x0F$ ，数据字段长度(DLC)为 3 bytes。

每一字节定义如下：

| | | |
|------|-----|------|
| Byte | 2-1 | 0 |
| 说明 | 控制量 | 指令类型 |

其中，指令类型定义如下：

| 指令类型 | 0x00 | 0x01 | 0x02 | 0x03 | 0x04 | 0x05 | 0x06 |
|------|-------------------|------|------|--------------|------|------|------|
| 说明 | NOP (用于获取反馈报文) | 使能 | 失能 | 电流(力矩) 控制 | 速度控制 | 角度控制 | 低速控制 |

● 使能/失能

当 Byte 0 等于 0x01 时，触发电机使能，若使能成功，绿灯亮起。当 Byte 0 等于 0x02 时，触发电机失能，绿灯熄灭。使用使能/失能指令时，Byte 1~2 将被忽略。电机转动前需要先使能电机，而使能电机前需要先至少完成一次电机校准，校准参考 [3.2.电机校准](#)。

● 电流（力矩）控制

当 Byte0 等于 0x03 时，电机进入电流（力矩）控制模式。控制量 Byte 1~2 为 int16 类型数据，代表电流-10A~10A，其中 Byte 1 为 int16 的低 8 位，Byte 2 位 int16 的高 8 位。int16 类型可表示的范围为 -32,768 到 32,767，即 若希望控制电流为 0.75A，则控制量应为 $0.75/10 \times 32767 = 2458$ 。用 C 语言可以表示为：

```
uint8_t datas[3];
datas [0] = 0x03;
datas [1] = ((int16_t)(0.75/10*32767)) & 0xFF;
datas [2] = ((int16_t)(0.75/10*32767)) >> 8;
```

发送这条指令后，电机将以 0.75A 的 Q 轴电流沿正方向（逆时针）转动。电机转矩常数为 0.3Nm/A，即电流为 0.75A 时，力矩为 0.225Nm。注意：电流值不会超过 [3.3.上位机调参](#)中 limit.current 的限制。

● 速度控制

当 Byte0 等于 0x04 时，电机进入速度控制模式，使用电机内部速度环进行速度闭环速度闭环控制。控制量 Byte 1~2 为 int16 类型数据，代表速度-1000rpm~1000rpm，其中 Byte 1 为

int16 的低 8 位，Byte 2 位 int16 的高 8 位。int16 类型可表示的范围为-32,768 到 32,767，即若希望控制转速为 250rpm，则控制量应为 $250/1000*32767=8192$ 。用 C 语言可以表示为：

```
uint8_t datas [3];
datas [0] = 0x04;
datas [1] = ((int16_t)(250/1000*32767)) & 0xFF;
datas [2] = ((int16_t)(250/1000*32767)) >> 8;
```

发送这条指令后，电机将以 250rpm 沿正方向（逆时针）匀速转动。

注意：对于转速低于 5rpm 的应用场景，推荐使用低速控制模式，或许会有更好的控制效果。转速值不会超过 [3.3.上位机调参](#) 中 limit.speed 的限制。

● 角度控制

当 Byte0 等于 0x05 时，电机进入角度控制模式，使用电机内部角度环进行角度闭环速度闭环控制。控制量 Byte 1~2 为 uint16 类型数据，代表角度 $0 \sim 2\pi$ rad，其中 Byte 1 为 uint16 的低 8 位，Byte 2 位 uint16 的高 8 位。uint16 类型可表示的范围为 0 到 65,535，即若希望控制角度为 0.5 rad，则控制量应为 $0.5/(2*3.14)*65535=5218$ 。用 C 语言可以表示为：

```
uint8_t datas [3];
datas [0] = 0x05;
datas [1] = ((int16_t)(0.5/(2*3.14)*65535)) & 0xFF;
datas [2] = ((int16_t)(0.5/(2*3.14)*65535)) >> 8;
```

发送这条指令后，电机将会转动到 0.5rad 处停止。

进行角度控制时，转动方向为沿着较近方向转动，例如当前角度为 0.0rad，下一条指令控制 3.28rad，则电机将会沿着负方向转动 ($3.28-0=3.28$, $0+6.28-3.28=3$, $3<3.28$)。同时，电

机将会以 [3.3.上位机调参](#) 中 limit.speed 的速度上限转动（前提是能达到此速度）。例如，在上位机使用“config limit.speed 10”将最大速度限制到 10rpm, 再使用 CAN 总线控制转动到 3rad, 则电机将以 10rpm 的速度转动到 3rad 的位置。类似于达秒电机的位置速度模式。

● **低速控制**

当 Byte0 等于 0x06 时，电机进入低速控制模式，与速度控制基本一致，相比速度控制，在低速工况下或许会有更好的表现。注意：低速控制频率不易过高，建议低于 50Hz。

● **角度步进控制**

当 Byte0 等于 0x07 时，电机进入角度步进模式，与角度控制基本一致，只是绝对角度控制变为增量角度控制，控制量 Byte 1~2 为 int16 类型数据，代表角度 $-2\pi \sim 2\pi$ rad。

5.3.2.2. 反馈报文

电机向主机发送的标准数据帧称为反馈报文。电机每成功接收到一个自己 ID 的数据包，将会向主机发送一个反馈报文，其中包含电机状态信息，可以使用这些信息进行外部闭环控制。

反馈报文帧地址为 0x500+ID, 其中 ID 范围为 0x00~0x0F, 数据字段长度(DLC)为 8 bytes。每一字节定义如下：

| | | | | | |
|------|-----|-----|-----|----|------|
| Byte | 7-6 | 5-4 | 3-2 | 1 | 0 |
| 说明 | 角度 | 转速 | 电流 | 预留 | 电机状态 |

其中，电机状态每一 bit 定义如下：

| | | |
|-----|-----|------|
| Bit | 7-1 | 0 |
| 说明 | 预留 | 使能标志 |

反馈报文中的角度、转速、电流信息，均与控制报文中的格式一致，通过控制报文中的格式反推即可得到电机当前状态。

5.3.3. 程序编写指南

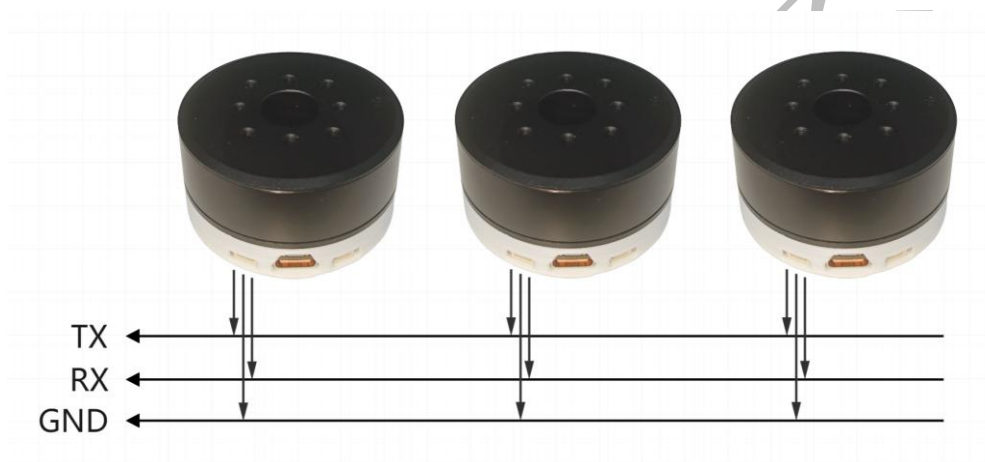
电机控制采用应答模式，即电机每接收一个控制报文，返回一个反馈报文，推荐使用定时器触发，定时向电机发送控制报文，同时在 CAN 总线接收回调函数中读取电机反馈报文，实时更新电机状态。

虽然电机具有内置的速度环和角度环控制，可以快速应用于一些简单场景。但对于负载变化、使用工况变化，常常需要调整电机内置 PID 参数才能实现良好的控制效果。另外，对于其他高级功能如：速度位置控制、速度斜坡控制、阻尼控制、加速度控制，电机内置程序无法做到通吃。推荐使用 CAN 总线进行外部控制，通过读取反馈报文获取反馈量，发送**电流**控制报文执行控制量，通过外部控制器设计 PID、LQR、MPC 等控制回路，实现高度自定义的运动效果。

六、UART 控制

6.1. UART 控制接线

通过特殊设计，QD4310 的 UART 支持一主多从通信。如[接口图](#)所示，使用 SH1.0-3Pin 接口将电机的 TX 连接至主机的 RX，电机的 RX 连接至主机的 TX，电机的 GND 连接至主机的 GND，UART 高电平为 3.3V。将电机连接到同一 UART 总线上即可进行通信，如图所示：



6.2. 控制方法

6.2.1. 设置电机 ID

通过特殊设计，QD4310 的 UART 支持一主多从通信，可以将多个电机挂载在同一条总线上即可实现多个电机同时控制。同一总线上的电机使用 ID 作为唯一身份标识。使用 UART 控制电机时需先确定被控电机的 ID，电机 ID 需要通过上位机配置，参考[3.3.上位机调参](#)章节。

注意：同一条总线上不可挂载两个相同 ID 的 QD4310 电机。

注意：硬件版本 V6.1 及以上电机支持 UART 一主多从通信，V6.0 电机 UART 控制仅支持点对点控制，即一个 UART 外设只能控制一个电机。

6.2.2. 数据包定义

UART 数据包总体上兼容 **CAN 数据包**。仅在 CAN 报文前增加 1 字节的 ID (UART ID 和

CAN ID 为同一个 ID)，后增加 1 字节的 CRC8 校验（多项式 0x07，初始值 0x00，结果异或值 0x00，不反转输入输出）。中间数据格式均保持一致，各种控制模式的控制细节也保持一致，本章不再赘述，详情查看 [5.3.2.CAN 数据包定义](#)。

QD4310 电机 UART 的默认波特率为 115200bps，波特率可以通过上位机更改，更改方式同 pid、ID 等参数，详见 [3.3.上位机调参](#)。

注意：修改 UART 波特率后需要重启电机生效。可使用“reboot”命令重启。

6.2.2.1. 控制数据包

控制数据包共 5 字节。通过向电机发送控制数据包，可以控制电机使能失能、运行模式和控制量。控制数据包格式如下定义如下：

| | | | | |
|------|------|-----|------|----|
| Byte | 4 | 3-2 | 1 | 0 |
| 说明 | CRC8 | 控制量 | 指令类型 | ID |

其中，指令类型定义如下：

| | | | | | | | |
|------|-------------------|------|------|--------------|------|------|------|
| 指令类型 | 0x00 | 0x01 | 0x02 | 0x03 | 0x04 | 0x05 | 0x06 |
| 说明 | NOP (用于获取反馈报文) | 使能 | 失能 | 电流(力矩) 控制 | 速度控制 | 角度控制 | 低速控制 |

6.2.2.2. 反馈数据包

反馈数据包共 10 字节。格式定义如下：

| | | | | | | | |
|------|------|-----|-----|-----|----|------|----|
| Byte | 9 | 8-7 | 6-5 | 4-3 | 2 | 1 | 0 |
| 说明 | CRC8 | 角度 | 转速 | 电流 | 预留 | 电机状态 | ID |

其中，电机状态每一 bit 定义如下：

| | | |
|-----|-----|------|
| Bit | 7-1 | 0 |
| 说明 | 预留 | 使能标志 |

反馈报文中的角度、转速、电流信息，均与控制报文中的格式一致，通过控制报文中的格

式反推即可得到电机当前状态。

6.2.3. 程序编写指南

由于 UART 协议本身不支持一主多从、总线仲裁等功能，要实现总线功能需要在编写代码时遵循以下规则：

1. 必须遵循一发一收的原则，发送数据包后，需要等收到电机反馈数据包或接收超时后，才能发送下一个数据包。严禁发送和接收异步操作，必须串行操作（如例程所写）！

实测在 UART 波特率 4Mbps 下，使用阻塞方式满速收发，每秒可以发送并接收超 6000 个数据包，丢包率约 0.035%。

7. PWM 控制

7.1. PWM 控制接线

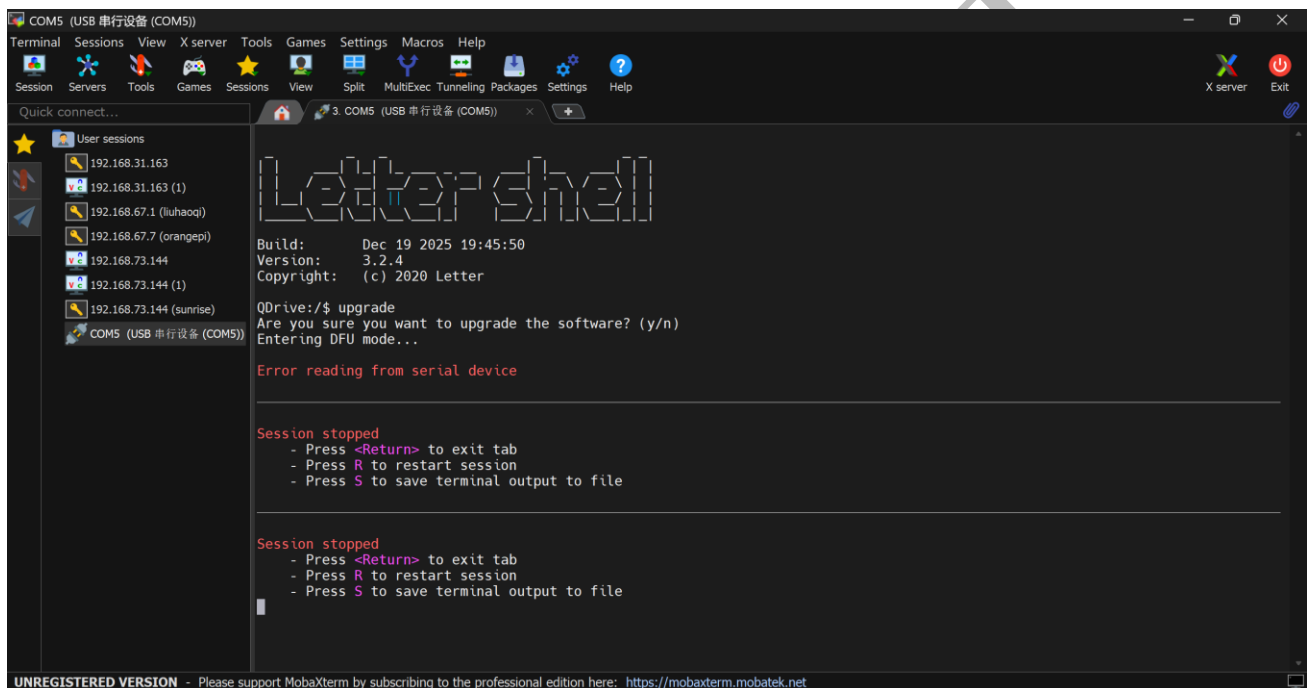
PWM 控制正在开发中，敬请期待……



8. 升级固件

自“Software version 5.2.1”（使用“version”命令查看）起，QD4310 支持通过 Type-C 接口进行固件升级，升级步骤如下。

1. 进入升级模式。使用 Type-C 数据线将电机连接至电脑，在上位机输入“upgrade”命令，并输入'y'确认，电机将进入升级模式，红绿指示灯交替闪烁，上位机与电机的连接将被断开。



```
COM5 (USB 串行设备 (COM5))
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect... 3. COM5 (USB 串行设备 (COM5))
User sessions
192.168.31.163
192.168.31.163 (1)
192.168.67.1 (liuhaogj)
192.168.67.7 (orangepl)
192.168.73.144
192.168.73.144 (1)
192.168.73.144 (sunrise)
COM5 (USB 串行设备 (COM5))

LetterShell
Build:      Dec 19 2025 19:45:50
Version:    3.2.4
Copyright:  (c) 2020 Letter

QDrive:/$ upgrade
Are you sure you want to upgrade the software? (y/n)
Entering DFU mode...

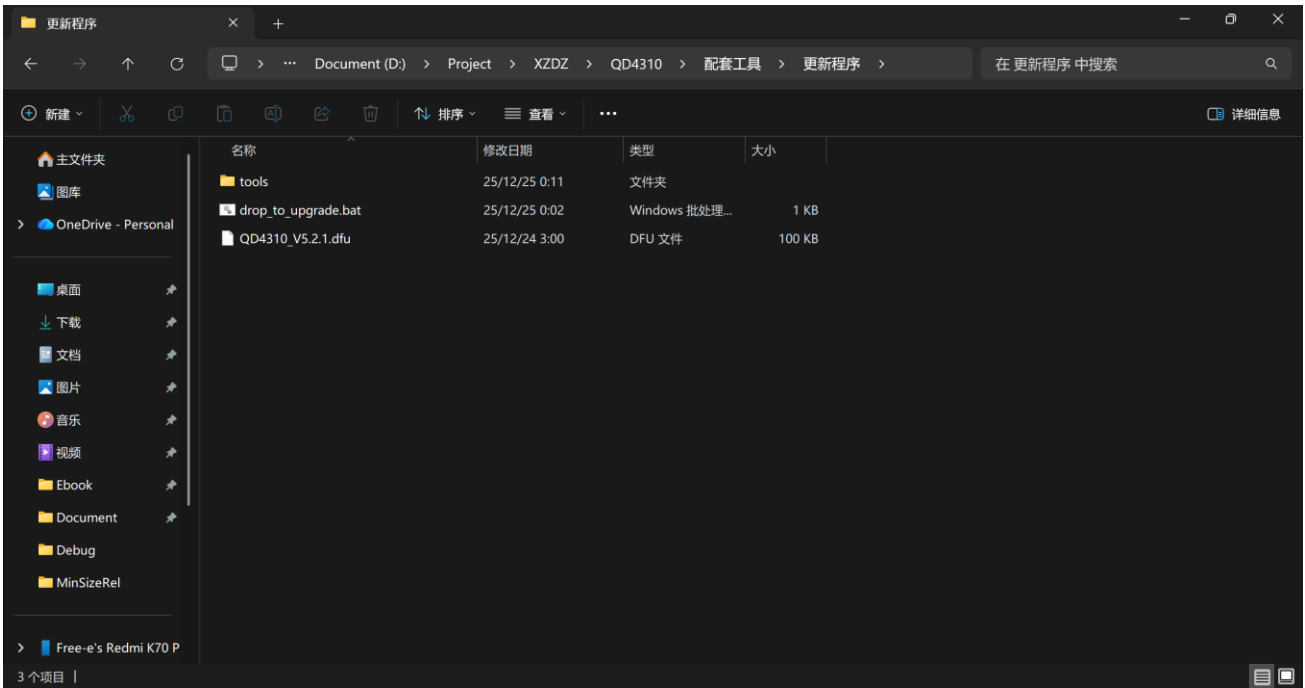
Error reading from serial device

Session stopped
- Press <Return> to exit tab
- Press R to restart session
- Press S to save terminal output to file

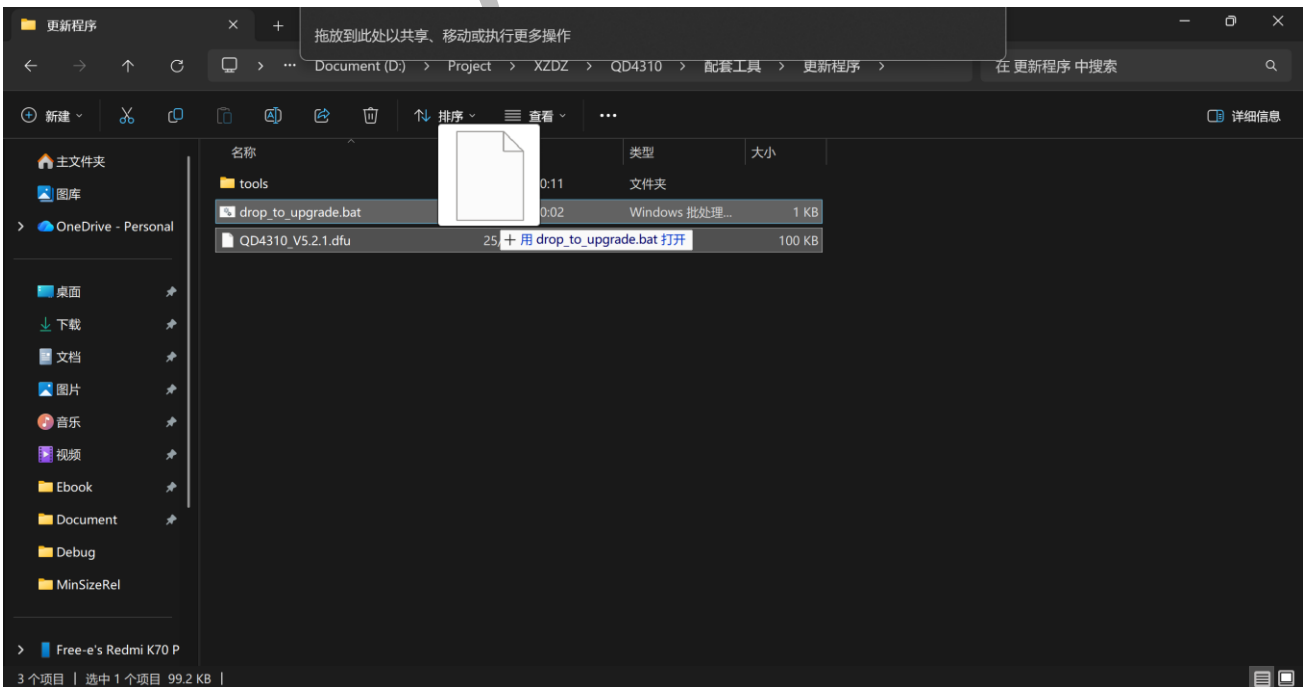
Session stopped
- Press <Return> to exit tab
- Press R to restart session
- Press S to save terminal output to file

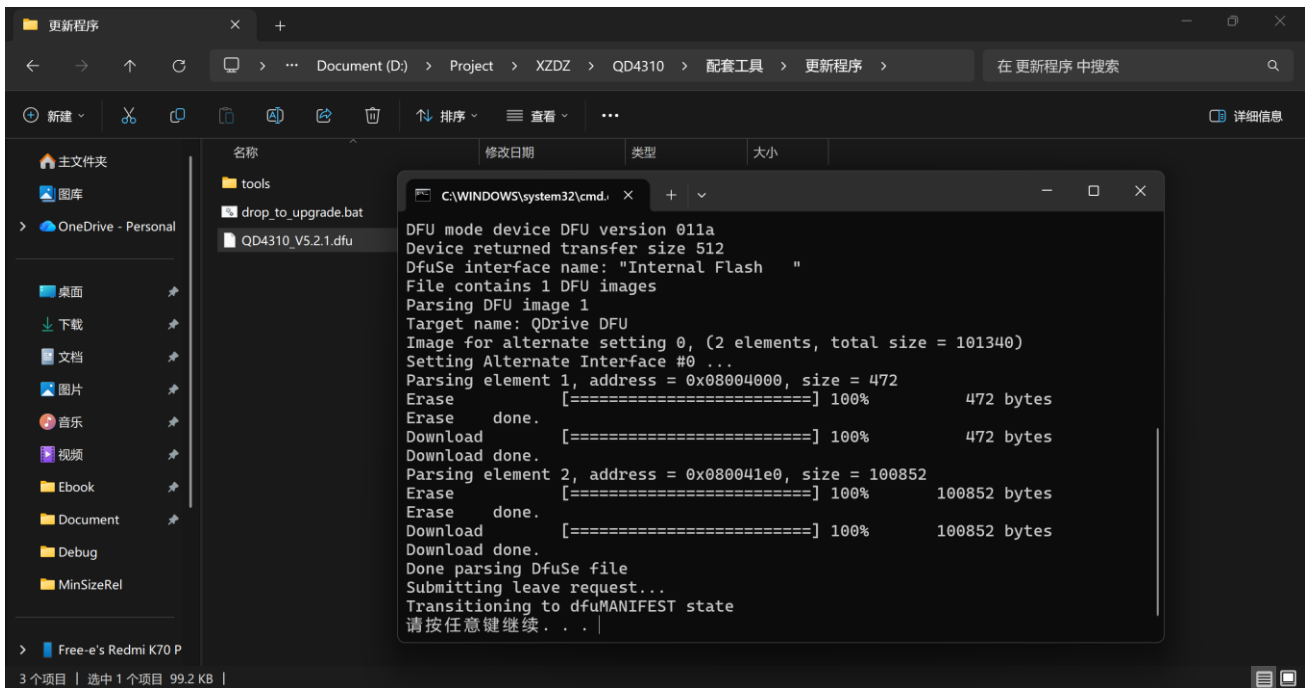
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
```

2. 准备升级工具及固件。从 [QD4310 资料](#) 下载升级工具和最新版固件。



3. 拖拽升级固件。使用鼠标拖动固件文件(QD4310_Vx.x.x.DFU)到 drop_to_upgrade.bat, 将自动完成固件升级。





注意：更新固件后需要重新校准电机，且会重置保存的用户配置参数，如 can.id、limit.speed、pid.speed.kd 等。

9. 常见问题与注意事项

1. Q: 电机无法使能。

A: 电机需要校准后才能使能。校准参考 [3.2.电机校准](#)。

2. Q: 断电重新上电后，电机仍按照上一次的指令执行。

A: 这是未完全断电导致。XT30(2+2)和 Type-C 均断电才算完全断电。

3. Q: 调整电机参数，断电后失效。

A: 调整参数后需要运行“store”命令将参数储存进内部储存器。参考 [3.3.上位机调参](#)。

4. Q: 使用电机时出现震荡。

A: 带载时使用电机内部速度模式/角度模式控制时，容易出现震荡现象。原因是对于不同的负载工况，需要有不同的 PID 参数，出厂内置的速度环、角度环 PID 参数只针对空载工况。解决方法有如下几种：1、[参考 3.3.上位机调参](#)调整合适的 PID 参数，并储存。2、外部实现速度环/角度环（使用电流模式控制，并结合 CAN 反馈报文，使用外部控制器进行闭环控制）。

5. Q: 导电滑环如何接线？

A: 导电滑环不是电机的一部分，属于电机配件，类似于汽车挡泥板，接线完全独立于电机。导电滑环上相同颜色的导线互相导通，约等于可以旋转的导线，可以用于供电、通信等各种用途，接线由用户自定义。

6. ...

10. 修改记录

| 日期 | 版本 | 内容 |
|------------|-----|--|
| 2025/10/20 | 0.3 | 草稿 |
| 2025/11/02 | 0.4 | 添加“上位机调参”、“上位机控制” |
| 2025/11/02 | 0.5 | 添加“程序控制”、“常见问题与注意事项” |
| 2025/11/24 | 0.6 | 添加 CAN 总线控制接线图 |
| 2025/11/24 | 1.0 | 完成 CAN 总线控制章节 |
| 2025/12/18 | 1.1 | 添加角度步进控制模式 |
| 2025/12/25 | 1.2 | 添加升级固件章节 |
| 2025/12/28 | 1.3 | 添加零点设置功能 |
| 2026/1/25 | 1.4 | 调整角度步进范围 |
| 2026/2/7 | 1.5 | 纠正 F103 CAN 引脚标错 |
| 2026/3/9 | 2.1 | 添加 UART 控制章节、添加 SH1.0-3Pin 接口描述、补充 CAN 波特率说明、纠正 CAN 反馈报文字节定义范围重叠 |
| 2026/3/31 | 2.2 | 添加 UART 电平描述、纠正两处转矩常数数值不一致的问题、优化 Q&A 描述、优化其他若干细节 |