

# 25-30mm 驱控一体使用说明手册

## 变更说明 V20260204

- 1) 支持了多控制器 can 网络互联，最多支持 60 个控制器互联。控制器的编号从 1 到 60，通讯协议的 N 代表控制器编号，通过在程序里更改
- 2) 除了原有的 mit 模式外，添加了速度模式，位置模式，两种模式
- 3) 在电机状态添加了一个 byte，用于表示，当前是什么电机。

## 1: 接口定义

颜色	定义
红色	电源+
黑色	电源-
蓝色	CanH
白色	CanL

电源默认电压 12v，范围 10- 18v

## 2: canfd 配置

Iso Canfd

仲裁域波特率：1M 80%

数据域波特率：4M 80%

注意：can 协议，需要发送端和接受端各接 1 个 120 欧姆的终端匹配电阻，保证信号质量。

## 3: 通讯协议

下面的 canid 的中的 N，表示控制器的编号，范围从 1 到 60。例如控制器的编号是 1，那么“mit 模式发送”的 canid 是  $0x501 + 1*4 = 0x505$ 。现在默认编号都是 1，如需其他编号组网，请提前告知，我们需要提前更改编号。

目前支持三种模式，上电后默认 mit 模式，通过电机控制指令切换，切换时，需要发送指令让电机停止。

- 1) 速度模式，控制电机转速，通过 mit 模式发送指令 中的 速度 指定转速
- 2) 位置模式，控制电机位置，通过 mit 模式发送指令 中的 电机位置 指定位置
- 3) Mit 模式

### 3.1 mit 模式发送

方向：上位机发送

Canid: 标准帧  $0x501 + N * 4$ 。

Dlc: 数据长度 12byte

电流环带宽：1000hz

数据：

Data0	电机位置码值 = $(s32)((u32)data3 \ll 24)   ((u32)data2 \ll 16)   ((u32)data1 \ll 8)   data0)$ - 0x80000000) ;
Data1	
Data2	电机位置角度 = 电机位置码值 / 2097152 * 360 度。
Data3	这里的位置是多圈位置，编码器是 21 位的，一圈 360 度 = 2097152 个刻度。 电机位置码值最大范围-0x7CFFFFFF 到 0x7CFFFFFF.
Data4	速度码值 = $(s16)((u16)data5 \ll 8)   ((u16)data4)$ - 0x8000);
Data5	速度 = 速度码值 / 10 rpm。 速度码值范围 -30000 到 30000
Data6	只在 MIT 模式下有效，其他模式有自己的固定参数
Data7	Kp, 位置环比例系数 $Kp = 0.00001 * ((u16)data7 \ll 8)   ((u16)data6)$ 。 建议初步调试设置为 0.008，过大容易震荡。 实际物理模型，输入单位是 rad，输出单位是 A，假设这时候的比例系数是 kact。 $Kp = kact * 0.001115$ 。
Data8	只在 MIT 模式下有效，其他模式有自己的固定参数
Data9	Kd, 速度环比例系数 $Kd = 0.01 * ((u16)data9 \ll 8)   ((u16)data8)$ 。 建议初步调试设置为 1，过大容易震荡。 实际物理模型，输入单位是 rad/s，输出单位是 A，假设这时候的比例系数是 kact。 $Kd = kact * 85.636$ 。
Data10	只在 MIT 模式下有效，其他模式下，无效
Data11	前馈电流码值 = $(s16)((u16)data11 \ll 8)   ((u16)data10)$ - 0x8000) 前馈电流值 = 电流码值 * 11 / 4096 A。

例子：

位置模式：

25mm, 30mm 电机

01 00 01 80 00 80 20 03 64 00 00 80 //

设置位置码值 = 65537

速度 = 0

Kp = 0.008

Kd = 1

电流 = 0

电流模式:

01 00 01 80 00 80 00 00 00 00 64 80

设置位置码值 = 65537

速度 = 0

Kp = 0

Kd = 0

电流码值 = 100

### 3.2 mit 模式回复

每发送一个 mit 模式发送的帧，控制器返回一个 mit 模式回复的帧

方向: 电机控制器回复

Canid: 标准帧  $0x701 + 4 * N$

Dlc: 数据长度 12byte

Data0	Canid, data0 = $0x01 + 4 * N$ , data1 = 0x05
Data1	
Data2	电机位置码值 = $(s32)((u32)data5 \ll 24)   ((u32)data4 \ll 16)   ((u32)data3 \ll 8)   data2$ ); - 0x80000000);
Data3	
Data4	电机位置角度 = 电机位置码值 / 2097152 * 360 度。 这里的位置是多圈位置，编码器是 21 位的，一圈 360 度 = 2097152 个刻度。
Data5	
Data6	速度码值 = $(s16)((u16)data7 \ll 8)   ((u16)data6)$ ); - 0x8000);
Data7	
Data8	电流码值 = $(s16)((u16)data9 \ll 8)   ((u16)data8)$ ); - 0x8000)
Data9	
Data10	0x00
Data11	0x00

### 3.3 电机状态

方向: 电机控制器发送，控制器每 20ms 发送一帧

Canid: 标准帧  $0x702 + 4 * N$

Dlc: 数据长度 20byte

Data0	Canid, data0 = $0x01 + 4 * N$ , data1 = 0x05
Data1	
Data2	电机位置码值 = $(s32)((u32)data5 \ll 24)   ((u32)data4 \ll 16)   ((u32)data3 \ll 8)   data2$ ); - 0x80000000);
Data3	
Data4	电机位置角度 = 电机位置码值 / 2097152 * 360 度。

Data5	这里的位置是多圈位置，编码器是 21 位的，一圈 360 度 = 2097152 个刻度。
Data6	速度码值 = (s16)((((u16)data7 << 8)   ((u16)data6)) - 0x8000);
Data7	速度 = 速度码值 / 10 rpm。
Data8	电流码值 = (s16)((((u16)data9 << 8)   ((u16)data8) - 0x8000)
Data9	电流值 = 电流码值 * 11 / 4096 A。
Data10	母线电压值 = (float)((((u16)data11 << 8)   ((u16)data1) - 0x8000) / 10 v
Data11	
Data12	故障状态
Data13	Fault = (((u32)data15 << 24)   ((u32)data14 << 16)   ((u32)data13 << 8)
Data14	data12))
Data15	Bit0: 驱动芯片故障 Bit1: 过压故障，母线电压超过 18v Bit2: 欠压故障，母线电压低于 8v Bit3: 过流故障，电机 10 秒内功率过大 Bit4: 编码器故障
Data16	电机状态 0: 初始化 1: 电流校准 2: 编码器零点校准 3: 停止 4: 运行 5: 错误
Data17	编码器零点校准状态 0: 空闲 1: 校准 2: 完成 3: 错误
Data18	电机当前模式 0x00: MIT 模式 0x01: 速度模式 0x02: 位置模式
Data19	电机类型 0x01: 30mm 0x02: 25mm 0x03: 20mm 0x04: 16mm

### 3.4 电机控制

方向：上位机发送

Canid: 标准帧 0x502 + 4 \* N

Dlc: 数据长度 8byte

Data0	0x01: 电机运行 0x00: 电机停止 注意: 使用 mit 模式控制电机, 要先发送命令让电机运行
Data1	0x01: 将当前电机位置设置位 0 点位置。作用于位置控制。 其他: 无效果
Data2	设置电机模式 改变电机模式的时候需要让电机停止, 即 data0 = 0x00, 才能生效。 0x00: MIT 模式 0x01: 速度模式 0x02: 位置模式
Data3	0x00
Data4	0x00
Data5	0x00
Data6	0x00
Data7	0x00

### 3.5 保留 id

0x503 + 4 \* N, 用于编码器零点校准, 用户使用时候避开此 id, 防止误操作。

0x703 + 4 \* N, 用于返回编码器零点校准结果, 用户使用时候避开此 id, 防止冲突。

## 4: 指令示意

<input type="checkbox"/> 全选	手动发送	帧ID	类型 数据长度	时间戳	备注	帧数据
<input type="checkbox"/>	发送	00000509	标准 数据 FDBRS 12	0	设置位置	01 00 00 90 00 80 20 03 64 00 00 80
<input type="checkbox"/>	发送	00000509	标准 数据 FDBRS 12	10000	设置电流	01 00 01 80 00 80 00 00 00 00 C8 80
<input checked="" type="checkbox"/>	发送	0000050A	标准 数据 FDBRS 8	20000	运行	01 00 00 00 00 00 00 00
<input type="checkbox"/>	发送	0000050A	标准 数据 FDBRS 8	30000	设置零点	00 01 00 00 00 00 00 00
<input type="checkbox"/>	发送	0000050A	标准 数据 FDBRS 8	40000	mit模式	00 00 00 00 00 00 00 00
<input type="checkbox"/>	发送	0000050B	标准 数据 FDBRS 8	50000	校准	
<input type="checkbox"/>	发送	0000050A	标准 数据 FDBRS 8	60000	速度模式	00 00 01 00 00 00 00 00
<input type="checkbox"/>	发送	0000050A	标准 数据 FDBRS 8	70000	位置模式	00 00 02 00 00 00 00 00
<input type="checkbox"/>	发送	00000509	标准 数据 FDBRS 12	80000	设置速度	01 00 00 A0 20 CE 00 00 00 00 00 80

0001: 设置 mit 模式下的位置

对于 25mm, 30mm 的电机, 建议:

Kp = 0.008

Kd = 1

01 00 00 90 00 80 20 03 64 00 00 80

0002: 设置 mit 模式下的电流

0003: 电机运行指令

0004: 电机把当前位置设置为 0 点, 电机停止, 并进入 mit 模式

0005: 电机停止, 并进入 mit 模式

0006: 校准指令

0007: 电机停止, 并进入速度模式

0008: 电机停止, 并进入位置模式

0009: 位置模式下, 速度模式下, 用于设置位置, 和速度。当前的速度是 2000rpm, 位置是 0x20000001

1) mit 模式

位置控制:

发送 0005 - 0003 - 0001。

电流控制:

发送 0005 - 0003 - 0002。

2) 速度模式

发送 0007 - 0003 - 0009, 电机以 2000rpm 运行。

3) 位置模式

发送 0008 - 0003 - 0009, 电机运行到位置 0x20000001。

4) 校准

发送 0005 - 0006, 然后等待 10s 校准完成。