

VSMD116_025T
步进电机驱动控制器
CAN2.0 协议



北京伟恩斯技术有限公司

www.vincetech.com

【序言】

感谢您购买本公司微型步进电机驱动器，本使用说明书将详细介绍该产品的各项功能和操作方法，让您充分感受本产品带给您的方便、快捷和安全。

【安全使用说明】

- 使用前请务必仔细阅读本使用说明书，按照说明书要求进行接线，以免损坏产品；
- 请不要将本产品暴露在潮湿过高的地方；
- 请不要将接线端子短路，否则会毁坏产品；
- 如果步进电机额定电流大于 2.5A，请将驱动器电流调整到 2.5A 以下，以免损坏电机；

【联系方式】

北京伟恩斯技术有限公司

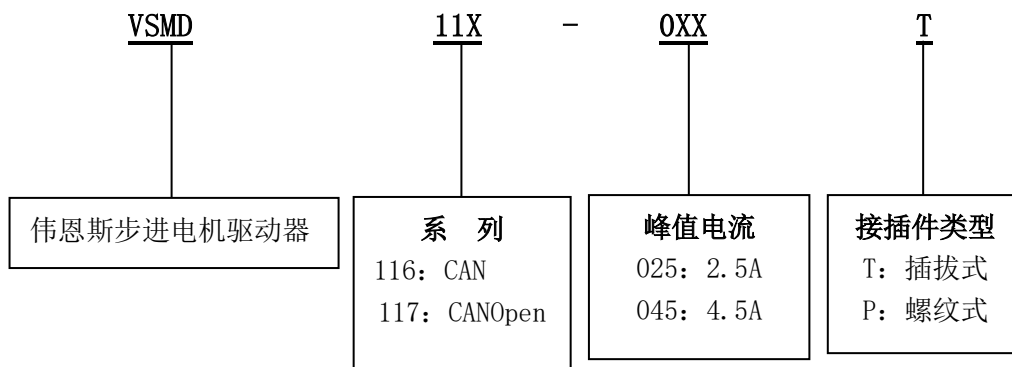
地址：北京市昌平区科技园区生命园路 29 号 1 幢 B316

电话：18612497280

邮箱：xu_guoen@vincetech.com

网址：www.vincetech.com

【命名规则】



目录

1.	简介.....	1
2.	接线方式.....	2
2.1.	接线端口.....	2
2.2.	级联接线.....	3
2.3.	恢复出厂设置.....	3
2.4.	传感器接线.....	3
2.4.1.	NPN 型传感器接线方式.....	3
2.4.2.	PNP 型传感器接线方式.....	4
2.4.3.	机械开关接线方式.....	5
3.	CAN 通讯指令和反馈.....	6
3.1.	帧结构.....	6
3.1.1.	标准帧结构.....	6
3.1.2.	扩展帧结构.....	6
3.2.	TARGET ID.....	7
3.3.	控制字 C1C0.....	7
3.3.1.	读取状态寄存器.....	7
3.3.1.1.	状态位信息.....	7
3.3.2.	数据寄存器.....	8
3.3.2.1.	设备 ID 号寄存器.....	9
3.3.2.2.	通讯波特率寄存器.....	9
3.3.2.3.	微步细分地址寄存器.....	9
3.3.2.4.	目标速度寄存器.....	10
3.3.2.5.	加速度寄存器.....	10
3.3.2.6.	减速度寄存器.....	10
3.3.2.7.	加速电流寄存器.....	11
3.3.2.8.	运行电流寄存器.....	11
3.3.2.9.	保持电流寄存器.....	12
3.3.2.10.	传感器 S1/S2 功能设置寄存器.....	12
3.3.2.11.	传感器 S3/S4 功能设置寄存器.....	12
3.3.2.12.	传感器 S5/S6 功能设置寄存器.....	13
3.3.2.13.	S1/S2/S3/S4/S5/S6 输入输出设置寄存器.....	13
3.3.2.14.	归零模式设置寄存器.....	14
3.3.2.15.	归零传感器常开常闭设置寄存器.....	14
3.3.2.16.	归零传感器编号设置寄存器.....	14
3.3.2.17.	归零速度寄存器.....	14
3.3.2.18.	归零后停止位置寄存器.....	15
3.3.2.19.	离线运行模式寄存器.....	15
3.3.2.20.	离线运行启动时间寄存器.....	15
3.3.2.21.	正负极限传感器设置寄存器.....	15

3.3.2.22.	自动使能寄存器.....	16
3.3.2.23.	指令一问一答设置寄存器.....	16
3.3.2.24.	加电自动归零寄存器.....	16
3.3.3.	读取数据寄存器.....	16
3.3.4.	写入数据寄存器.....	17
3.4.	控制指令.....	18
3.4.1.	电机使能.....	19
3.4.2.	电机失能.....	19
3.4.3.	设置原点.....	19
3.4.4.	电机停止.....	20
3.4.5.	电机连续运行.....	20
3.4.6.	绝对位移指令.....	20
3.4.7.	保存指令.....	21
3.4.8.	输出指令.....	21
3.4.9.	启动归零.....	22
3.4.10.	停止归零.....	22
3.4.11.	相对位移指令.....	23
3.4.12.	启动离线运行.....	23
3.4.13.	停止离线运行.....	23
3.4.14.	清除离线节点.....	24
3.4.15.	增加离线归零节点.....	24
3.4.16.	增加速度模式节点.....	24
3.4.17.	增加位置模式节点.....	24
3.4.18.	增加延时节点.....	25
3.4.19.	读取状态寄存器.....	25
3.4.20.	读取数据寄存器.....	26
4.	运行模式.....	27
5.	归零功能设置.....	28
5.1.	归零功能简介.....	28
5.2.	归零参数设置.....	29
5.3.	执行归零动作.....	29
5.4.	停止归零动作.....	29
6.	指示灯.....	29
7.	PC 端控制/配置软件.....	30
7.1.	主界面.....	30
7.2.	状态、控制和配置.....	31
7.2.1.	状态栏.....	31
7.2.2.	控制栏.....	32
7.2.3.	参数设置.....	33
7.3.	命令行.....	35
8.	性能指标.....	36
8.1.	电气性能.....	36
8.2.	使用环境.....	36

8.3.	尺寸及重量.....	36
9.	附件.....	37
9.1.	CAN 总线概述.....	37
9.2.	外形尺寸图.....	38

1. 简介

VSMD116_025T 是运动控制和电机驱动一体化的步进电机控制驱动模块。符合标准 CAN2.0（包括 CAN2.0A 和 CAN2.0B）通讯协议，用户通过发送指令就可以方便的控制电机的精确运动，无需关心步进电机驱动的底层知识。

【基本参数】

- 输入电压：12 ~ 40VDC
- 峰值电流：2.5A
- 微步细分：1/2/4/8/16/32

【通讯】

- 通讯方式：CAN2.0
- 通讯协议：CAN 2.0A、CAN2.0B
- 通讯速率：20K ~ 1M

【结构】

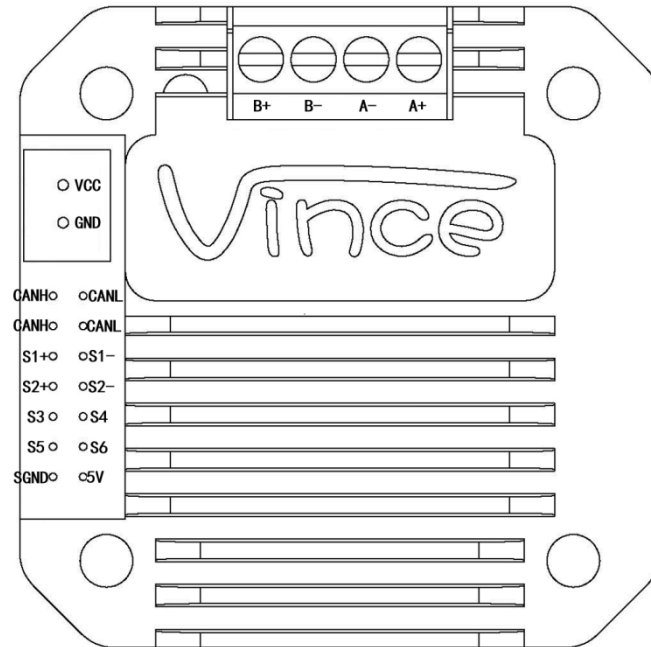
- 铸铝外壳：坚固、散热好
- 外形尺寸：42.5mm × 42.5mm × 16.8mm

【特点】

- 采用 32 位微处理器控制，运动控制更精确。
- 根据指令，完成各种复杂运动的控制（平滑加减速、平滑转向等）。
- 可运行在速度模式、位置模式，并能自由切换。
- 独特的电流控制模式，能在保持平稳的前提下，减小噪声，降低发热量。
- 内置归零功能，简单设置归零参数后，归零过程由驱动器完成，减轻用户工作量。
- 内置离线模式，可以脱离上位机运行。
- 传感器接口 S1、S2 支持 3.3V ~ 24V 信号电压，并支持共阳/共阴两种连接方法。
- 传感器接口 S3、S4、S5、S6 支持 3.3V ~ 5V 信号电压，可配置为输入/输出。
- 支持 CAN2.0 通讯协议，最多可级联 110 个驱动器。

2. 接线方式

2.1. 接线端口



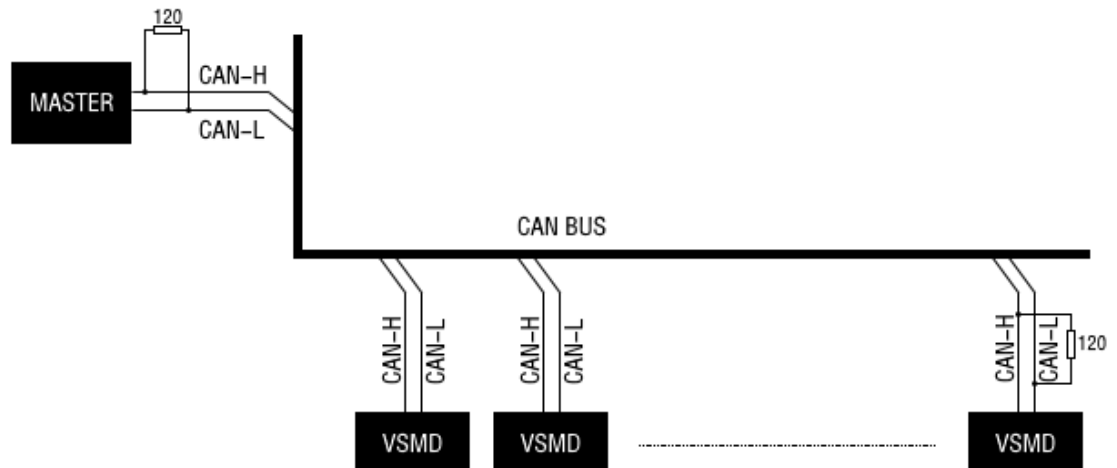
接口	说明
A+ A- B- B+	电机接口
VCC	电源（12~40VDC）
GND	电源地
CANH CANL	CAN 接口（两组）
S1+ S1-	传感器 1（3.3~24V 兼容）
S2+ S2-	传感器 2（3.3~24V 兼容）
S3 S4 S5 S6	传感器 3~6（3.3~5V 兼容）
5V	5V 输出（<100mA）
SGND	信号地

※ 两组 CAN 接口用于驱动器级联。

※ 5V 可给外部提供 5V 供电。

2.2. 级联接线

CAN 总线支持级联方式连接，通过两条 CAN 总线可以控制 110 台驱动器，典型连接方式如下图所示：



※CAN 总线首尾，各需要一个 120 欧姆的终端电阻。

2.3. 恢复出厂设置

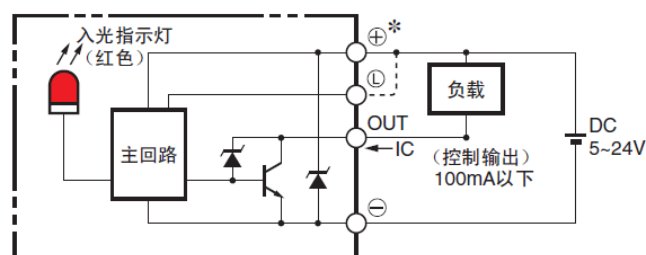
将 S3 和 S4 端子信号线短接，然后上电，这时 ID 号和波特率就是 1 和 100K，重新设置 ID 和波特率并保存后，重新上电生效。

2.4. 传感器接线

传感器从性质上来看，主要分为有源和无源的。常用的光电开关，是有源的，微动开关是无源的。如何选择开关，以及如何连接，需要根据实际情况来决定。

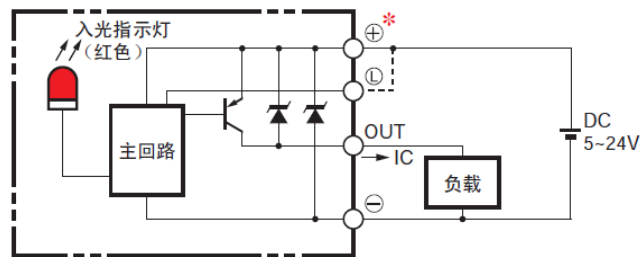
类型	传感器信号（开放状态）	传感器信号（触发状态）
常通/常开	1	0
常闭	0	1

2.4.1. NPN 型传感器接线方式

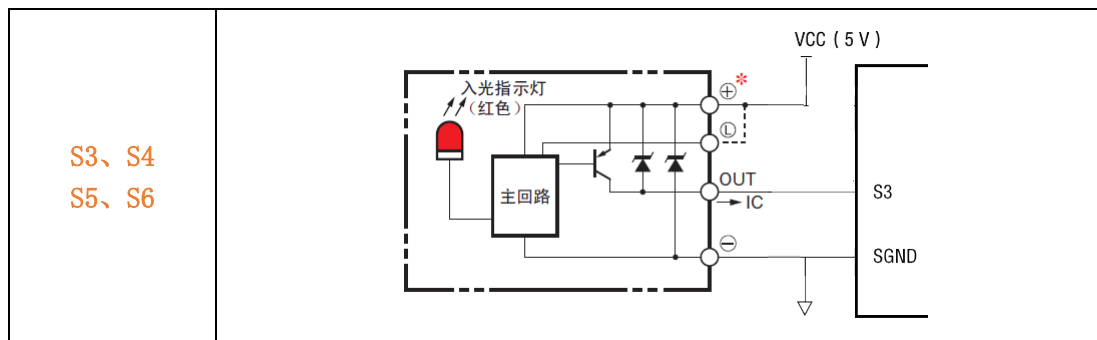


传感器	接线方式
S1、S2	
S3、S4 S5、S6	

2.4.2. PNP 型传感器接线方式



传感器	接线方式
S1、S2	



2.4.3. 机械开关接线方式

机械开关属于无源传感器，把传感器的 COM 端连接 GND，S1/S2/S3 根据需要（常开，常闭）连接到传感器的 NO/NC 管脚即可。

传感器	接线方式
S1、S2	
S3、S4 S5、S6	

3. CAN 通讯指令和反馈

3.1. 帧结构

VSMD1X6 系列驱动器采用 CAN 2.0 通讯，支持 CAN2.0A 和 CAN2.0B 协议。CAN2.0A 使用标准帧，CAN2.0B 使用扩展帧。

3.1.1. 标准帧结构

标准数据帧由 11 位标识符 + 8 字节数据组成，如下所示：

10	9	8	7	6	5	4	3	2	1	0
SID										
TARGET ID				C1	C0	CMD / REG ADDR				
0	0	0	0	X	X	X	X	X	X	X

位	定义
TARGETID (BIT10:BIT7)	目标设备号
C1 C0	控制字
CMD / REG ADDR (BIT4:BIT0)	指令 / 寄存器地址

3.1.2. 扩展帧结构

扩展数据帧由 29 位扩展标识符 + 8 字节数据组成，如下所示：

28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
SID											EID																				
0	TARGET ID										0	SOURCE ID										C1	C0	CMD / REG ADDR							
	0	0	0	0	0	0	0	0	0	0	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x			

位名称	说明
27BIT 28BIT	固定 0
TARGETID (BIT27:BIT19)	目标设备号
18BIT	固定 0
SOURCE ID (BIT17:BIT9)	源设备号
C1 C0	控制字
CMD / REG ADDR (BIT6:BIT0)	指令 / 寄存器地址

※ 设备号支持范围为 0x001~0x1FE，目标设备号为 0x000，并且 18BIT 为 1 时，为广播。

3.2. TARGET ID

TARGET ID 代表 CAN 总线节点的标识符。在两条 CAN 总线上，会有多个节点相连，每一个节点都有一个互不相同的 ID 号，这个 ID 号就是 TARGET ID。

3.3. 控制字 C1C0

控制字描述总线上数据传送的类型和方式。

C1	C0	说明
0	0	读取状态寄存器：驱动器返回给主机
0	1	读取数据寄存器：驱动器返回给主机
1	0	写入数据寄存器：主机发送给驱动器
1	1	指令：主机发送命令给驱动器

3.3.1. 读取状态寄存器

当控制字 C1=0 C0=0 时，驱动器向主机发送状态寄存器值，每一个寄存器长度是 32 位，状态寄存器地址范围是 0x00~0x10，具体内容如下所示：

地址	类型	说明
0x00	32 位有符号浮点数	当前运行速度
0x01	32 位有符号整数	当前运行位置
0x02	32 位无符号整数	当前状态位
0x03 ~ 0x09	保留	保留
0x0A ~ 0x10	字符型	软件版本号

※ 每一个数据帧最多传送 8 个字节数据，及传送 2 个寄存器数据。

3.3.1.1. 状态位信息

地址 0x02 寄存器是一个 32 位无符号整数，存储当前状态位信息，每一个 bit 代表状态如下所示：

状态位	说明	值
0	传感器 S1 状态位	0：低电平；1：高电平
1	传感器 S2 状态位	0：低电平；1：高电平
2	传感器 S3 状态位	0：低电平；1：高电平
3	传感器 S4 状态位	0：低电平；1：高电平
4	当前位置与目标位置是否相等	0：不相等；1：相等
5	当前速度与目标速度是否相等	0：不相等；1：相等
6	硬件错误状态位	0：正常；1：硬件错误

7	原点标志位	0: 不在原点; 1: 在原点
8	停止标志位	0: 运行状态; 1: 停止状态
9	指令错误标志位	0: 指令正确; 1: 指令错误
10	存储器读写错误标志位	0: 读写正常; 1: 读写异常
11	离线运行标志位	0: 离线运行停止; 1: 离线运行
13	电机使能标志位	0: 电机失能; 1: 电机使能
14	归零结束标志位	0: 归零中; 1: 归零结束
16	传感器 S5 状态位	0: 低电平; 1: 高电平
17	传感器 S6 状态位	0: 低电平; 1: 高电平
20	过热保护状态位	0: 正常; 1: 过热保护
21	过流保护状态位	0: 正常; 1: 过流保护
22	低压保护状态位	0: 正常; 1: 低压保护

3.3.2. 数据寄存器

数据寄存器的地址范围是 0x00~0x19，每一个寄存器的数据长度是 32 位，如下所示：

地址	说明
0x00	设备 ID 号
0x01	通讯波特率，范围是（20K ~ 1M）
0x02	微步细分，范围是（0 ~ 5）
0x03	运行目标速度（-192000 ~ 192000）单位是脉冲频率
0x04	运行加速度（0 ~ 192000000）单位是脉冲频率
0x05	运行减速度（0 ~ 192000000）单位是脉冲频率
0x06	加速电流（0 ~ 2.5A）
0x07	工作电流（0 ~ 2.5A）
0x08	保持电流（0 ~ 2.5A）
0x09	传感器 1 和传感器 2 功能设置
0x0A	传感器 3 和传感器 4 功能设置
0x0B	传感器 5 和传感器 6 功能设置
0x0C	保留
0x0D	S1 ~ S6 输入输出配置。0: 输入; 1: 输出
0x0E	归零模式设置
0x0F	归零传感器常开常闭设置。0: 常闭; 1: 常开
0x10	归零传感器编号设置
0x11	归零速度设置
0x12	归零后停止位置设置
0x13	离线模式设置。0: 普通模式; 1: 离线开始前先归零
0x14	上电离线模式启动设置。0: 不自动运行; 1~60 : 时间（秒）
0x15	保留
0x16	保留
0x17	正负极限传感器设置

0x18	上电自动使能设置
0x19	是否一问一答设置
0x1a	上电自动归零设置

3.3.2.1. 设备 ID 号寄存器

地址 0x00 数据寄存器存储的是设备 ID 号，对应帧结构里面的 Target ID 字段，用于区分总线上的不同设备。

3.3.2.2. 通讯波特率寄存器

地址 0x01 数据寄存器存储的是通讯波特率，范围是 20K ~ 1M，如下所示：

通讯速率	设置参数值
20K	0x1C2132
25K	0x168132
30K	0x12C132
40K	0xE1132
50K	0xB4132
60K	0x5A163
75K	0x78132
80K	0x4B153
90K	0x4B143
100K	0x5A132
125K	0x48132
150K	0x3C132
200K	0x2D132
250K	0x24132
300K	0x1E132
400K	0xF153
500K	0x12132
600K	0x9163
750K	0x8153
900K	0x6163
1000K	0x9132

3.3.2.3. 微步细分地址寄存器

地址 0x02 数据寄存器存储的是驱动器的微步细分，范围是 0 ~ 5，对应如下所示：

数值	对应细分
0	整步

1	2 细分
2	4 细分
3	8 细分
4	16 细分
5	32 细分

3.3.2.4. 目标速度寄存器

地址 0x03 数据寄存器存储的是运行目标速度，范围是（-192000 ~ 192000）；单位是脉冲频率。速度的数据类型是 32 位有符号浮点数，设置的时候需要转换成 16 进制。示例程序如下所示：

```
bool motor_cfg_spd(int tid, float spd)
{
    uint8_t data[4];
    uint32_t val = *((uint32_t*)(&spd));

    data[0] = (val >> 24) & 0xff;
    data[1] = (val >> 16) & 0xff;
    data[2] = (val >> 8) & 0xff;
    data[3] = (val >> 0) & 0xff;

    return can_write_dat_regs(tid, CAN_DATA_SPD, data, 4);
}
```

3.3.2.5. 加速度寄存器

地址 0x04 数据寄存器存储的是加速度值，范围是（0 ~ 192000000）；单位是脉冲频率。加速度的数据类型是 32 位有符号浮点数，设置的时候需要转换成 16 进制。示例程序如下所示：

```
bool motor_cfg_acc(int tid, float acc)
{
    uint8_t data[4];
    uint32_t val = *((uint32_t*)(&acc));

    data[0] = (val >> 24) & 0xff;
    data[1] = (val >> 16) & 0xff;
    data[2] = (val >> 8) & 0xff;
    data[3] = (val >> 0) & 0xff;

    return can_write_dat_regs(tid, CAN_DATA_ACC, data, 4);
}
```

3.3.2.6. 减速度寄存器

地址 0x05 数据寄存器存储的是减速度值，范围是（0 ~ 192000000）；单位是脉冲频率。加速度的数据类型是 32 位有符号浮点数，设置的时候需要转换成 16 进制。示例程序如下所示：

```
bool motor_cfg_dec(int tid, float dec)
{
    uint8_t data[4];
    uint32_t val = *((uint32_t*)(&dec));

    data[0] = (val >> 24) & 0xff;
    data[1] = (val >> 16) & 0xff;
    data[2] = (val >> 8) & 0xff;
    data[3] = (val >> 0) & 0xff;

    return can_write_dat_regs(tid, CAN_DATA_DEC, data, 4);
}
```

3.3.2.7. 加速电流寄存器

地址 0x06 寄存器存储的是加速电流,加速电流是指电机在加速过程中使用的电流,范围是 (0 ~ 2.5A)。加速电流的数据类型是 32 位有符号浮点数,设置的时候需要转换成 16 进制。示例程序如下所示:

```
bool motor_cfg_cra(int tid, float cra)
{
    uint8_t data[4];
    uint32_t val = *((uint32_t*)(&cra));

    data[0] = (val >> 24) & 0xff;
    data[1] = (val >> 16) & 0xff;
    data[2] = (val >> 8) & 0xff;
    data[3] = (val >> 0) & 0xff;

    return can_write_dat_regs(tid, CAN_DATA_CRA, data, 4);
}
```

3.3.2.8. 运行电流寄存器

地址 0x07 数据寄存器存储的是运行电流,运行电流是指电机到达设置速度后使用的电流,范围是 (0 ~ 2.5A)。运行电流的数据类型是 32 位有符号浮点数,设置的时候需要转换成 16 进制。示例程序如下所示:

```
bool motor_cfg_crn(int tid, float crn)
{
    uint8_t data[4];
    uint32_t val = *((uint32_t*)(&crn));

    data[0] = (val >> 24) & 0xff;
    data[1] = (val >> 16) & 0xff;
    data[2] = (val >> 8) & 0xff;
    data[3] = (val >> 0) & 0xff;

    return can_write_dat_regs(tid, CAN_DATA_CRN, data, 4);
}
```


3.3.2.9. 保持电流寄存器

地址 0x08 数据寄存器存储的是保持电流。保持电流是指电机在使能的状态下，电机停止运动时使用的电流，范围是（0 ~ 2.5A）。保持电流的数据类型是 32 位有符号浮点数，设置的时候需要转换成 16 进制。示例程序如下所示：

```
bool motor_cfg_crh(int tid, float crh)
{
    uint8_t data[4];
    uint32_t val = *((uint32_t*)(&crh));

    data[0] = (val >> 24) & 0xff;
    data[1] = (val >> 16) & 0xff;
    data[2] = (val >> 8) & 0xff;
    data[3] = (val >> 0) & 0xff;

    return can_write_dat_regs(tid, CAN_DATA_CRH, data, 4);
}
```

3.3.2.10. 传感器 S1/S2 功能设置寄存器

地址 0x09 数据寄存器存储的是传感器 S1 和 S2 功能。类型是无符号 32 位整数，每一个字节从高位到低位分别代表：S1 下降沿功能、S1 上升沿功能、S2 下降沿功能、S2 上升沿功能，如下所示：

31 ~ 24	23 ~ 16	15 ~ 8	7 ~ 0
s1f	s1r	s2f	s2r

传感器功能如下所示：

功能号	说明
0	无动作（只有状态位变化通知）
1	重新设置原点位置
2	减速停止
3	减速停止，并在停止后重新设置原点位置
4	立刻停止
5	立刻停止，并在停止后重新设置原点位置
6	正向连续运转（正速度）
7	反向连续运转（负速度）
8	启动离线功能
9	停止离线功能

3.3.2.11. 传感器 S3/S4 功能设置寄存器

地址 0x0a 数据寄存器存储的是传感器 S3 和 S4 功能。类型是无符号 32 位整数，每一个字节从高位到低位分别代表：S3 下降沿功能、S3 上升沿功能、S4 下降沿功能、S4 上升沿功能，如下所示：

31 ~ 24	23 ~ 16	15 ~ 8	7 ~ 0
s3f	s3r	s4f	s4r

传感器功能如下所示：

功能号	说明
0	无动作（只有状态位变化通知）
1	重新设置原点位置
2	减速停止
3	减速停止，并在停止后重新设置原点位置
4	立刻停止
5	立刻停止，并在停止后重新设置原点位置
6	正向连续运转（正速度）
7	反向连续运转（负速度）
8	启动离线功能
9	停止离线功能

3.3.2.12. 传感器 S5/S6 功能设置寄存器

地址 0x0b 数据寄存器存储的是传感器 S5 和 S6 功能。类型是无符号 32 位整数，每一个字节从高位到低位分别代表：S5 下降沿功能、S5 上升沿功能、S6 下降沿功能、S6 上升沿功能，如下所示：

31 ~ 24	23 ~ 16	15 ~ 8	7 ~ 0
s5f	s5r	s6f	s6r

传感器功能如下所示：

功能号	说明
0	无动作（只有状态位变化通知）
1	重新设置原点位置
2	减速停止
3	减速停止，并在停止后重新设置原点位置
4	立刻停止
5	立刻停止，并在停止后重新设置原点位置
6	正向连续运转（正速度）
7	反向连续运转（负速度）
8	启动离线功能
9	停止离线功能

3.3.2.13. S1/S2/S3/S4/S5/S6 输入输出设置寄存器

地址 0x0d 数据寄存器用于存储 S3/S4/S5/S6 输入输出设置，0 为输入，1 为输出，

如下所示：

31 ~ 24	23 ~ 16	15 ~ 8	7 ~ 0
00000000	00000000	00000000	00S6S5S4S3S2S1

※ vsmd116 系列，S1 和 S2 只能输入，不能输出。

3.3.2.14. 归零模式设置寄存器

地址 0x0e 数据寄存器用于存储归零模式，范围（0 ~ 4）归零模式如下所示：

参数值	说明
0	归零功能关闭
1	一次归零
2	一次归零+安全位置
3	二次归零
4	二次归零+安全位置

3.3.2.15. 归零传感器常开常闭设置寄存器

地址 0x0f 数据寄存器用于存储归零用传感器在未触发状态的电平值，0 为低电平，1 为高电平。

3.3.2.16. 归零传感器编号设置寄存器

地址 0x10 数据寄存器用于存储归零用传感器编号，范围（0 ~ 5），如下所示：

参数值	说明
0	设置传感器 1 为归零用传感器
1	设置传感器 2 为归零用传感器
2	设置传感器 3 为归零用传感器
3	设置传感器 4 为归零用传感器
4	设置传感器 5 为归零用传感器
5	设置传感器 6 为归零用传感器

3.3.2.17. 归零速度寄存器

地址 0x11 数据寄存器用于存储归零速度，范围是（-192000 ~ 192000），单位是脉冲频率，正负号表示方向。速度的数据类型是 32 位有符号浮点数，设置的时候需要转换成 16 进制。示例程序见 3.4.4 目标速度寄存器。

3.3.2.18. 归零后停止位置寄存器

地址 0x12 数据寄存器用于存储归零后停止在的安全位置，范围是 (-2147483647 ~ 2147483647)，类型是有符号 32 位整数。

3.3.2.19. 离线运行模式寄存器

地址 0x13 数据寄存器用于存储离线运行模式，范围是 0 和 1，如下所示：

参数值	说明
0	普通模式，按照设定站点运行
1	运行前先执行归零程序

3.3.2.20. 离线运行启动时间寄存器

地址 0x14 数据寄存器用于存储加电后离线模式运行的等待时间，范围是 0 ~ 60，单位是秒。如下所示：

参数值	说明
0	开机不自动执行离线运行模式
1~60	单位：秒。1 ~ 60 秒内，无握手信号，自动启动离线运行模式

3.3.2.21. 正负极限传感器设置寄存器

地址 0x17 数据寄存器用于存储正负极限传感器设置参数，32 个 bit 如下所示：

31 ~ 24	23 ~ 16	15 ~ 8	7 ~ 0
负极限传感器编码	负极限触发电平	正极限传感器编码	正极限触发电平

传感器触发电平如下所示：

参数值	说明
0	低电平触发
1	高电平触发

极限传感器编码器如下所示：

参数值	说明
0	无负极限
1	S1 为负极限
2	S2 为负极限
3	S3 为负极限

4	S4 为负极限
5	S5 为负极限
6	S6 为负极限

3.3.2.22. 自动使能寄存器

地址 0x18 数据寄存器用于存储上电自动使能设置，范围是 0 和 1，如下所示：

参数值	说明
0	上电不使能
1	上电使能

3.3.2.23. 指令一问一答设置寄存器

地址 0x19 数据寄存器用于存储指令是否需要一问一答，范围是 0 和 1，如下所示：

参数值	说明
0	指令不支持一问一答
1	指令支持一问一答

3.3.2.24. 加电自动归零寄存器

地址 0x1a 数据寄存器用于存储加电是否自动归零设置，范围是 0 和 1，如下所示：

参数值	说明
0	加电不自动归零
1	加电自动归零

3.3.3. 读取数据寄存器

当控制字 C1=0 C0=1 的时候，上位机读取驱动器数据寄存器内容，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	01	00111

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	01	0000111

3.3.4. 写入数据寄存器

当控制字 C1=1 C0=0 的时候，上位机读取驱动器数据寄存器内容，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	10	00111

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	10	0000111

3.4. 控制指令

当控制字 C1=1 C0=1 时,上位机向驱动器发送指令,参数最多 8 个字节数据(D1 ~ D8)。控制指令如下所示:

指令码	名称	说明
0x01	enable	电机使能
0x02	disable	电机失能
0x03	org	设置当前位置为原点
0x04	stop	电机停止
0x05	mov	电机连续转动
0x06	pos	绝对位移
0x07	save	保存
0x08	output	S1 ~ S6 输出指令
0x09	zero start	执行归零
0x0A	zero stop	归零终止
0x0B	rmv	相对移动
0x0C	-	保留
0x0D	-	保留
0x0E	-	保留
0x0F	-	保留
0x10	action start	启动离线
0x11	action stop	停止离线
0x12	action clear	清除离线节点
0x13	action zero	增加归零节点
0x14	action speed	增加速度模式节点
0x15	action position	增加位置模式节点
0x16	action delay	增加延时节点
0x17	-	保留
0x18	-	保留
0x19	-	保留
0x1A	-	保留
0x1B	-	保留
0x1C	-	保留
0x1D	-	保留
0x1E	read status registers	读取状态寄存器 ※ D1 : 寄存器地址 (0 ~ 7Fh) D2 : 数量 (1 ~ 80h)
0x1F	Read data registers	读取数据寄存器 ※ D1 : 寄存器地址 (0 ~ 7Fh) D2 : 数量 (1 ~ 80h)

3.4.1. 电机使能

电机使能指令码是 0x01，指令格式如下所示：

标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	00001

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0000001

3.4.2. 电机失能

电机失能指令码是 0x02，指令格式如下所示：

标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	00010

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0000010

3.4.3. 设置原点

设置当前位置为零位指令码是 0x03，指令格式如下所示：

标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	00011

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0000011

3.4.4. 电机停止

电机停止指令码是 0x04，当参数 D1=0 时，电机减速停止；当参数 D1=1 时，电机立即停止。指令格式如下所示：

标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	00100

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0000100

参数定义如下：

D1	说明
0	减速停止
1	立即停止

3.4.5. 电机连续运行

电机连续转动指令码是 0x05，当参数 D1 ~ D4 设置运行速度，类型是有符号 32 位浮点数。指令格式如下所示：

标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	00101

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0000101

参数定义如下：

D1 ~ D4	说明
float	运行速度

3.4.6. 绝对位移指令

电机绝对位移指令码是 0x06，参数 D1 ~ D4 设置相对于原点移动的脉冲数量，类型是有符号 32 位整数。指令格式如下所示：

标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	00110

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0000110

参数定义如下：

D1 ~ D4	说明
int32	移动脉冲数量

3.4.7. 保存指令

参数保存指令码是 0x07，将当前设置参数值保存到驱动器中，指令格式如下所示：

标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	00111

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0000111

3.4.8. 输出指令

输出指令码是 0x08，参数 D1 分别对应 S1 ~ S6 口，D2 代表输出高低电平，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	01000

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
----	-------------------	----	------------------	-------	------------

0	000000001	1	000000000	11	0001000
---	-----------	---	-----------	----	---------

参数 D1 定义如下：

D1	说明
0	无效
1	无效
2	设置 S3 端口
3	设置 S4 端口
4	设置 S5 端口
5	设置 S6 端口

参数 D2 定义如下：

D2	说明
0	输出低电平
1	输出高电平

3.4.9. 启动归零

启动归零指令码是 0x09，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	01001

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0001001

3.4.10. 停止归零

停止归零指令码是 0x0a，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	01010

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0001010

3.4.11. 相对位移指令

电机相对位移指令码是 0x0b, 参数 D1 ~ D4 设置相对于当前位置移动的脉冲数量, 类型是有符号 32 位整数。指令格式如下所示:

标准帧命令如下所示:

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	01011

扩展帧命令如下所示:

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0001011

参数定义如下:

D1 ~ D4	说明
int32	移动脉冲数量

3.4.12. 启动离线运行

启动离线模式运行指令码是 0x10, 标准帧命令如下所示:

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	10000

扩展帧命令如下所示:

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0010000

3.4.13. 停止离线运行

停止离线模式运行指令码是 0x11, 标准帧命令如下所示:

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	10001

扩展帧命令如下所示:

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
----	-------------------	----	------------------	-------	------------

0	000000001	1	000000000	11	0010001
---	-----------	---	-----------	----	---------

3.4.14. 清除离线节点

清除离线节点指令码是 0x12，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	10010

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0010010

3.4.15. 增加离线归零节点

增加离线模式归零节点指令码是 0x13，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	10011

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0010011

3.4.16. 增加速度模式节点

增加速度模式节点指令码是 0x14，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	10100

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0010100

3.4.17. 增加位置模式节点

增加位置模式节点指令码是 0x15，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	10101

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0010101

3.4.18. 增加延时节点

增加延时节点指令码是 0x16，标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	10110

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0010110

3.4.19. 读取状态寄存器

读取状态寄存器指令码是 0x1e，参数 D1 代表读取状态寄存器地址 (0x00 ~ 0x7f)，参数 D2 代表读取寄存器数量 (0x01 ~ 0x80)，指令格式如下所示：

标准帧命令如下所示：

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	11110

扩展帧命令如下所示：

28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0011110

参数定义如下：

参数	说明
D1	读取状态寄存器起始地址 (0x00 ~ 0x7f)
D2	读取状态寄存器数量 (0x01 ~ 0x80)

3.4.20. 读取数据寄存器

读取数据寄存器指令码是 0x1f, 参数 D1 代表读取状态寄存器地址(0x00 ~ 0x7f), 参数 D2 代表读取寄存器数量 (0x01 ~ 0x80), 指令格式如下所示:

标准帧命令如下所示:

目标 ID (target ID)	C1 C0	ADDR / CMD
0001	11	11111

扩展帧命令如下所示:

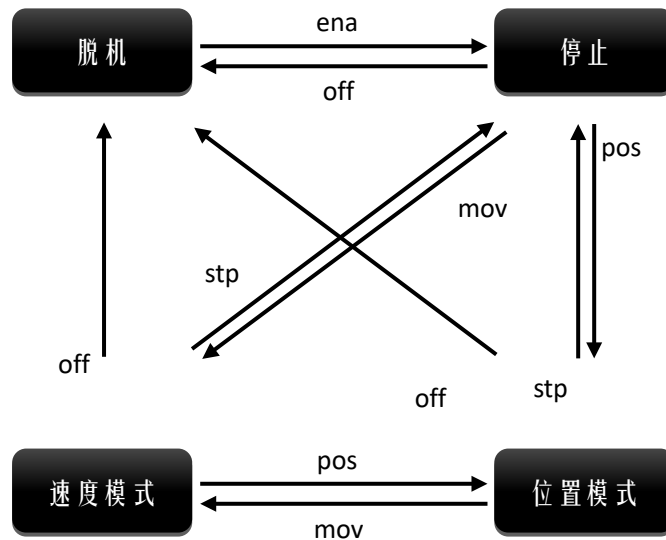
28	目标 ID (target ID)	18	源 ID (source ID)	C1 C0	ADDR / CMD
0	000000001	1	000000000	11	0011111

参数定义如下:

参数	说明
D1	读取数据寄存器起始地址 (0x00 ~ 0x7f)
D2	读取数据寄存器数量 (0x01 ~ 0x80)

4. 运行模式

VSMD 驱动有四个运行模式：脱机、停止、速度模式、位置模式。并且，速度模式和位置模式可以随意切换，并立刻执行，不需要等待前一个指令运行结束。



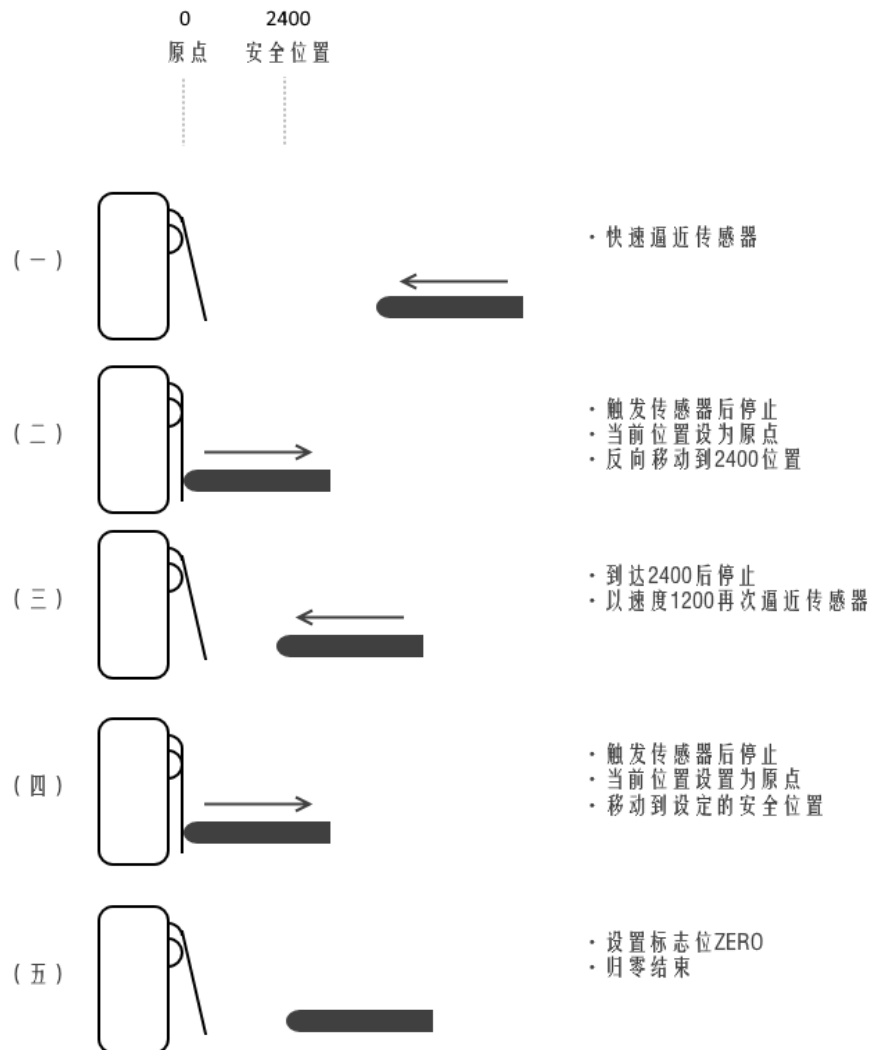
※ 模式内的目标速度或者目标位置变化，停止、速度模式以及位置模式模式之间的切换，只要是当前速度跟目标速度不一致，或者位置方向与速度反向，都会自动启动加减速，平滑运转到目标速度或者目标位置。整个过程都会进行平滑的加减速运动，避免急停或者突然转向。并且在整个运动过程中，电流会根据当前的运行状况自动匹配，让扭矩，噪声以及电机发热得以很好的控制。

5. 归零功能设置

5.1. 归零功能简介

VSMD 驱动器内置归零功能，设定好归零参数后，通过指令，驱动器自动完成整个归零过程。归零结束后，会将状态位 ZERO 置位，通过查询状态位可以判断归零动作是否完成。

下图用二次逼近归零来说明，其它模式只是这个模式的一个子集：



※ 如果归零开始时，传感器已经处于触发状态，则从（二）开始运行。

※ 归零的速度，以及合适的安全位置，要根据实际情况来设置。

※ 请注意归零速度以及安全位置的方向（正负）

5.2. 归零参数设置

VSMD 驱动器归零功能，需要设置五个数据寄存器，如下图所示：

数据寄存器地址	说明
0x0e	归零模式设置
0x10	归零用传感器编号
0x0f	传感器类型。1：常开；0：常闭
0x11	归零速度
0x12	归零后停止位置

5.3. 执行归零动作

参见 3.4.9 启动归零

5.4. 停止归零动作

参见 3.4.10 停止归零

6. 指示灯

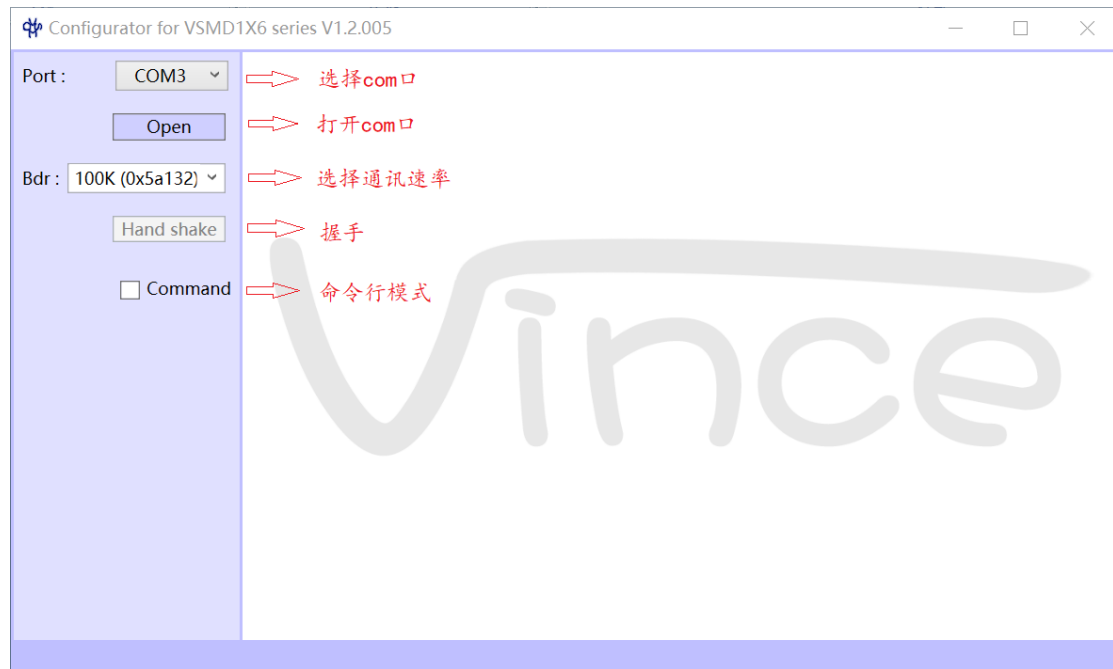
蓝色 LED 指示当前驱动器的工作状态。

序号	LED 指示	状态
1	长亮/长灭	位置故障
2	慢闪	停止
3	快闪	运行
4	双闪	驱动器硬件故障

7. PC 端控制/配置软件

PC 端控制/配置软件，提供了方便的为每个驱动器配置参数的功能。也可以控制驱动器运行。在没有编写上位机程序之前，可以让用户系统了解/测试 VSMD 驱动器的特性，以及可以了解 VSMD 指令的用法和反馈的数据格式解析。

7.1. 主界面



第一步：选择 com 口；

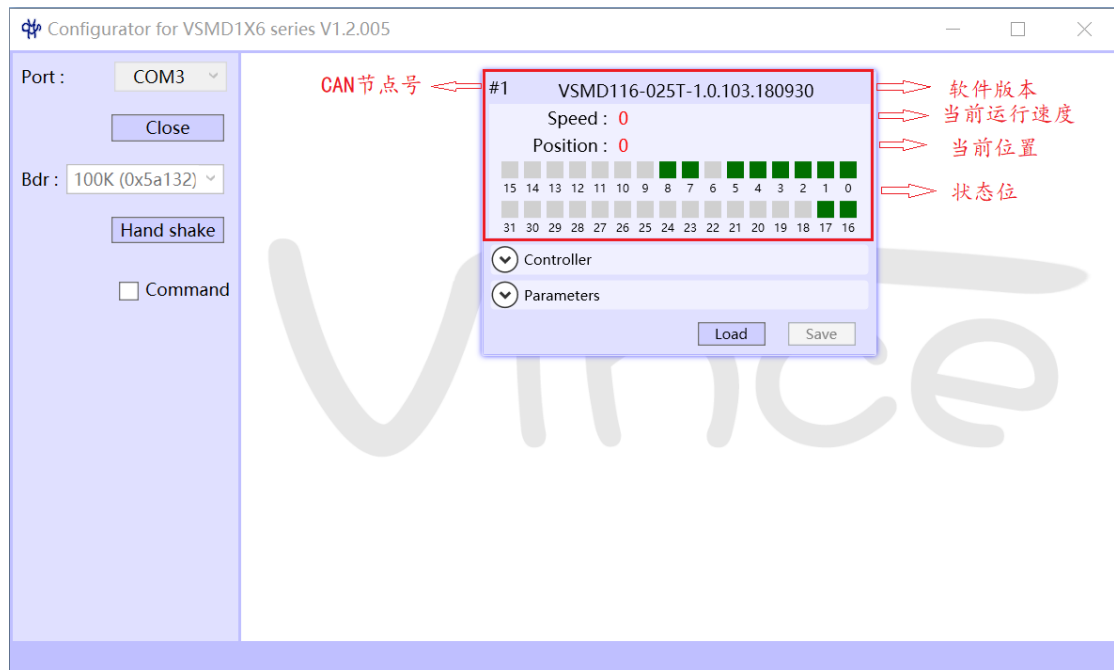
第二步：选择通讯速率，默认速率是 100K；

第三步：打开 com 口；

第四步：点击 Hand Shake 按钮，开始搜索 32 个 can 节点。

7.2. 状态、控制和配置

7.2.1. 状态栏

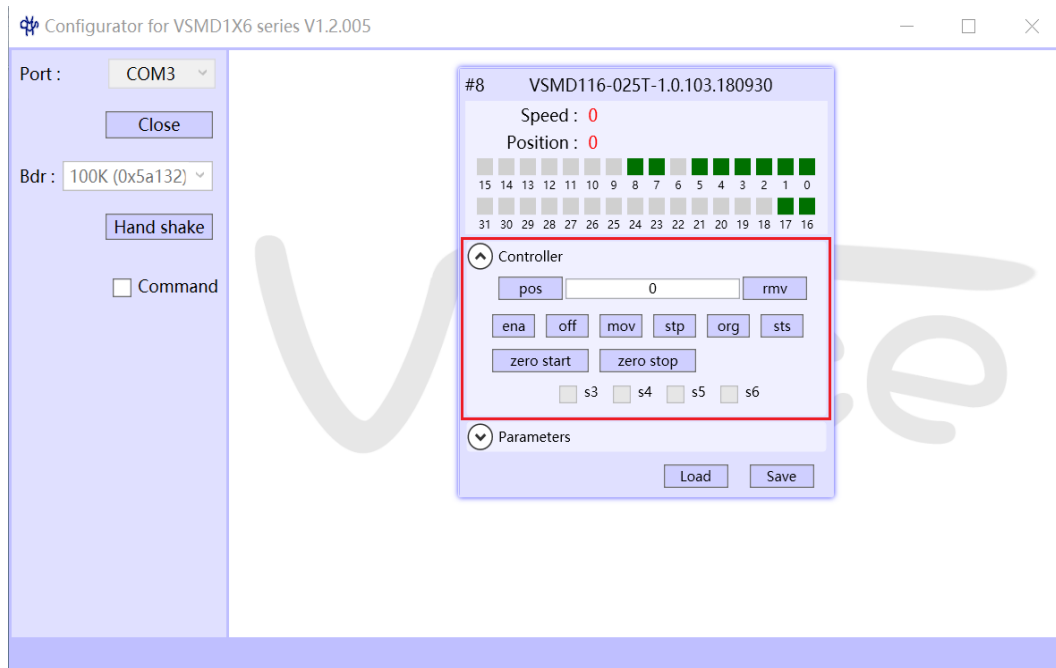


【状态位】

状态位	说明（绿色代表高电平 1，灰色代表低电平 0）
0	传感器 1 状态信息
1	传感器 2 状态信息
2	传感器 3 状态信息
3	传感器 4 状态信息
4	是否到达设置位置，1 代表到达，0 代表没有到达
5	是否到达设置速度，1 代表到达，0 代表没有到达
6	硬件错误报警，1 代表出现硬件错误，需要重新上电
7	是否停止在零点标价，1 代表停止在零点
8	电机运行停止标记位，1 代表停止，0 代表运行
9	命令正确与否标记位，1 代表发送命令错误，0 代表正确
10	读写存储器错误标记位，1 代表读写存储器错误，0 代表正确
11	离线模式运行标记位，1 代表离线模式运行，0 代表没有运行
12	握手状态信息，1 代表有握手信号，0 代表没有握手信号
13	使能标记位，1 代表使能，0 代表失能
14	归零结束标记，1 代表归零结束，0 代表归零没有结束
16	传感器 5 状态信息
17	传感器 6 状态信息

7.2.2. 控制栏

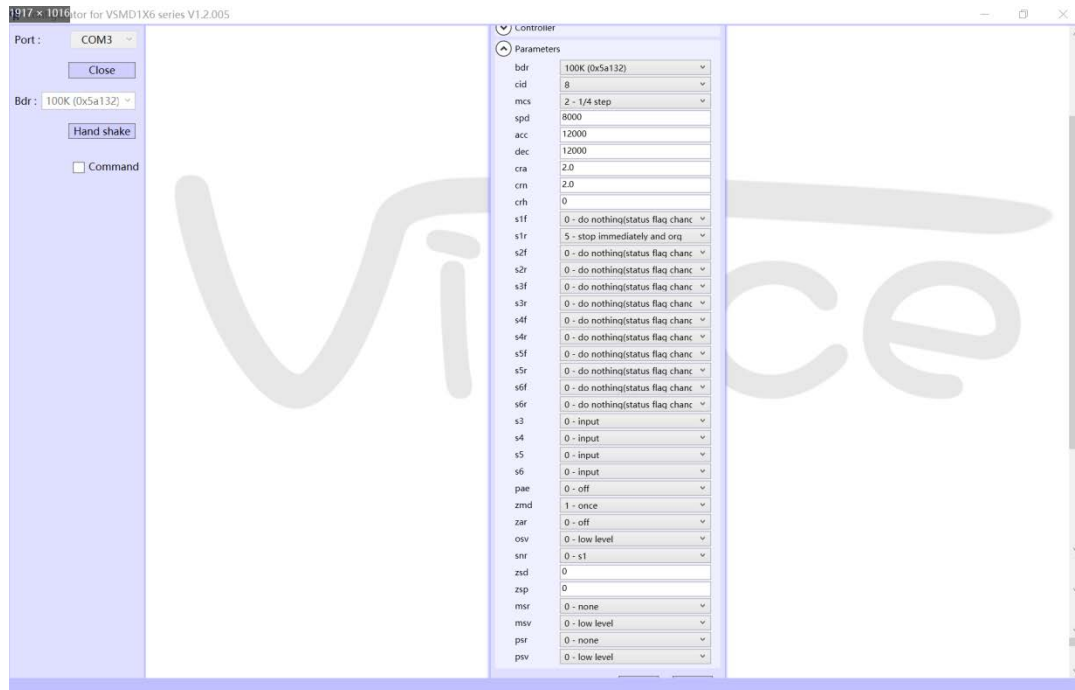
点击 Controller 会展开控制栏，如下图所示：



命令	说明
pos	绝对目标位置，单位是脉冲数，点击 pos 电机运行到设置位置
rmv	相对目标位置，单位是脉冲数，点击 rmv 电机相对于当前位置运行
ena	电机使能，Bit13 设置为 1
off	电机失能，Bit12 设置为 0
mov	速度模式，在使能状态下，根据设置速度一直运行
stp	电机减速停止
org	设置当前位置为零位
sts	得到当前状态信息，包括速度，位置和状态位
zero start	归零开始运行
zero stop	归零过程停止
s3、s4、s5、s6	设置 s3、s4、s5、s6 输出高低电平

7.2.3. 参数设置

点击 Parameters 会展开参数设置栏，如下图所示：



项目	说明
bdr	通讯波特率，默认是 100K
cid	can 总线电机的站点号，范围是 1~32，默认是 1
mcs	细分，0~5 对应 1/2/4/8/16/32 细分
spd	运行速度，单位是脉冲频率，-192000~192000Hz
acc	加速度，单位是脉冲频率，0~192000000
dec	减速带，单位是脉冲频率，0~192000000
cra	加速电流，电机在加速过程中的电流，范围是 0~2.5
crn	运行电流，电机达到设置速度运行中的电流，范围是 0~2.5
crh	保持电流，电机在使能并且停止运动的时候的电流，范围 0~2.5
s1f	传感器 1 在下降沿触发的动作
s1r	传感器 1 在上升沿触发的动作
s2f	传感器 2 在下降沿触发的动作
s2r	传感器 2 在上升沿触发的动作
s3f	传感器 3 在下降沿触发的动作
s3r	传感器 3 在上升沿触发的动作
s4f	传感器 4 在下降沿触发的动作
s4r	传感器 4 在上升沿触发的动作
s5f	传感器 5 在下降沿触发的动作
s5r	传感器 5 在上升沿触发的动作
s6f	传感器 6 在下降沿触发的动作

s6r	传感器 6 在上升沿触发的动作
s3	设置 s3 为输入还是输出
s4	设置 s4 为输入还是输出
s5	设置 s5 为输入还是输出
s6	设置 s6 为输入还是输出
pae	上电自动使能
zmd	归零模式，on 开启，off 关闭，二次接近归零模式
osv	归零用传感器是常开还是常闭，常开设置 1，常闭设置 0
snr	选择归零用传感器，0~5 分别代表 S1~S6
zsd	归零速度
zsp	归零结束后停止的位置，单位是脉冲数
zar	上电自动归零
msr	设置负极限传感器
msv	负极限传感器是常开还是常闭
psr	设置正极限传感器
psv	正极限传感器是常开还是常闭

7.3. 命令行

勾选主界面左侧的 Command，在窗口底部会出现命令行栏。用户输入指令，按下回车键，软件会自动增加换行符。



8. 性能指标

8.1. 电气性能

工作电压 (DC)	12~40VDC
峰值电流	2.5A, 实际电流可调
驱动方式	CAN 2.0
励磁方式	1/2/4/8/16/32
最大脉冲频率	192KHz
波特率	20KHz - 1MHz
绝缘电阻	常温常压下>100MΩ
绝缘强度	常温常压下 0.5KV, 1 分钟

8.2. 使用环境

冷却方式	自然冷却
工作温度	-30℃~80℃
工作湿度	≤80%

8.3. 尺寸及重量

外形尺寸	42.5mm × 42.5mm × 16.8mm
重量	0.07Kg

9. 附件

9.1. CAN 总线概述

CAN 是 Controller Area Network 的缩写（以下称为 CAN），是 ISO 国际标准化的串行通信协议。在汽车产业中，出于对安全性、舒适性、方便性、低公害、低成本的要求，各种各样的电子控制系统被开发了出来。由于这些系统之间通信所用的数据类型及对可靠性的要求不尽相同，由多条总线构成的情况很多，线束的数量也随之增加。为适应“减少线束的数量”、“通过多个 LAN，进行大量数据的高速通信”的需要，1986 年德国电气商博世公司开发出面向汽车的 CAN 通信协议。此后，CAN 通过 ISO11898 及 ISO11519 进行了标准化，在欧洲已是汽车网络的标准协议。

CAN 的高性能和可靠性已被认同，并被广泛地应用于工业自动化、船舶、医疗设备、工业设备等方面。现场总线是当今自动化领域技术发展的热点之一，被誉为自动化领域的计算机局域网。它的出现为分布式控制系统实现各节点之间实时、可靠的数据通信提供了强有力的技术支持。

9.2. 外形尺寸图

