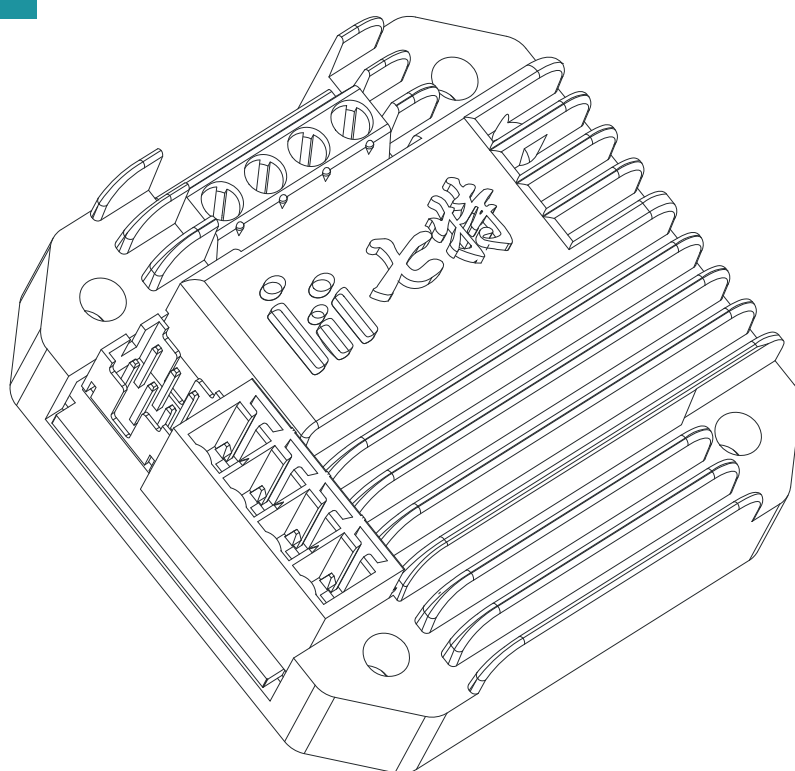




CAN 总线接口步进电机驱动器 (42 型: 7TCSM4210Q)

使用说明书



说明书修订版 (Ver3.00)



目录

前言	3
1. 产品特点	4
2. 产品参数	4
3. 电气接口	5
4. 典型使用及接线图	6
4.1 典型组网	6
4.2 传感器接口典型接线图	6
5. 运行及工作模式说明	7
5.1 步进一体机(步进电机、驱动、控制三合一)工作模式	7
5.2 定位模式	7
5.3 正反转模式	8
5.5 到位模式	9
5.6 调试步骤(供参考)	9
5.7 步进一体机参数说明	10
6. CAN 总线控制命令说明	13
6.1 CAN 总线数据帧说明	13
6.2 数据帧(8 字节)说明	13
6.3 命令码(功能码)及参数说明	14
6.4 举例说明	19
7. 包装	19
8. 买家必读	20
附件一 控制器外形尺寸图	21
附件二 控制器安装示意图	22

前言

尊敬的用户：

欢迎您使用北京瀚斯塔科技有限公司的产品，感谢您对我们的信任。

本文档和此后提及的任何图纸及计算机软件（包括描述本系统硬件部分的软件）都具有版权，并只能依据合同在北京瀚斯塔科技有限公司提供的系统及其文档的情况下使用。这些文本在事先没有得到北京瀚斯塔科技有限公司的书面许可情况下，明确的不能以任何形式（文字、口头、电子或其他形式）进行介质拷贝、编辑、翻译、出版、销售或以其他方式转让。

北京瀚斯塔科技有限公司精心的准备了本文档，当然也不免会出现错误或者印刷错误，北京瀚斯塔科技有限公司对这些错误及由此可能造成的损失不承担责任。考虑到所描述系统技术上的改进，北京瀚斯塔科技有限公司可以在任何时间没有事先通知的情况下更改原版文档到新的版本。

版本	说明
Ver1.00	建立文档
Ver1.01	传感器接口改变
Ver2.00	电源和总线接口定义改变
Ver2.01	增加变速通信协议
Ver2.02	增加软件设置相电流功能
Ver3.00	增加通信电气隔离、增加到位模式，型号从 7TCSM4210 更改为 7TCSM4210Q

北京瀚斯塔科技有限公司

Tel: 13716515078

Email: 514455707@qq.com

<http://www.bjhanstar.com>

1. 产品特点

- ✿ S 加减速曲线，运行平稳，用户可更改 S 曲线参数
- ✿ 微型设计，安装便利，可与 42 步进电机一体组装
- ✿ 网络集散控制，CAN2.0 组网
- ✿ 支持定位模式、正反转模式、速度模式、到位模式
- ✿ 停止运行时自动减电流减半
- ✿ 电气接口简洁，且接线方便
- ✿ 零位准确，有复位时的零位脱落动作
- ✿ 限位，碰到限位信号自动停止, 电平触发和边沿触发到位
- ✿ 提供计算机调试软件、DLL 和嵌入式源代码，方便调试和二次开发

2. 产品参数

表一:产品参数

产品参数		产品可更改运行参数
外观尺寸	42.2mm×42.2mm×14.5mm	可设置 电机相电流
相电流	0.3A-1.7A 连续可调	可设置 细分 1、2、4、8、16、32
工作电压	DC12V-32V	可设置 运动模式
步进细分	1、2、4、8、16、32	可设置 加速系数、启动速度和最大速度
步进频率	20Hz-20KHz 可调	可设置 电机空闲脱机
零位电气	PNP 和推挽（0-24V）	可设置 复位光电开关脱落步数
限位电气	TTL/NPN	可设置 电机方向
CAN 接口	CAN2.0A	可设置 复位、限位触发有效电平
存储温度	-20° ~85°	可设置 最大步数(找零位开关的最多步数)
保护电路 通信保护	过热、过流、过载、电源反接 电气隔离、TVS、防雷	可设置 CAN. ID

3. 电气接口

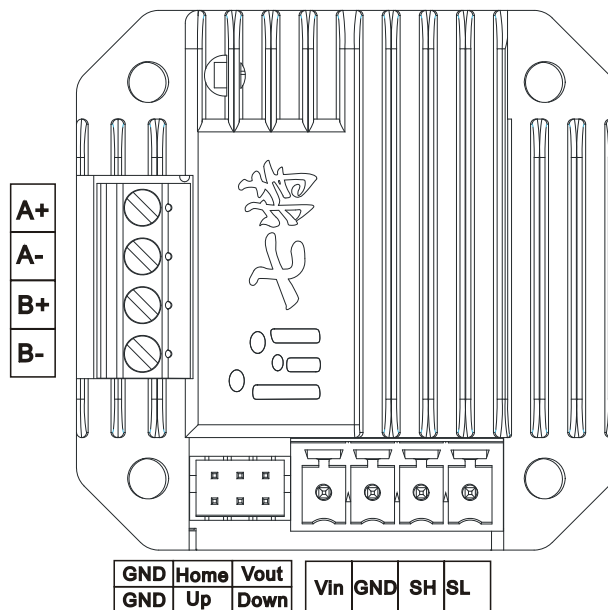


图 1: 步进一体机电气接口图

表 2: 用户接口定义说明

插座: KF2EDG-3.81	V+	电源输入 DC11V-DC32V
	GND	电源 GND
	SH	总线级联信号 CAN-H
	SL	总线级联信号 CAN-L

表 3: 传感器接口定义说明

传感器插座: PHD-2×3	Vout	默认 Vout=V+; 可定制选择 Vout= 5V@30mA
	Home	零位开关输入端, PNP 型, 支持 0-24V。推荐接光电开关
	UP	TTL 电平 (0-5V)、NPN 信号、机械开关 定位、正反转、速度模式下: 正反向限位信号 到位模式下: 正反向或者反方向的到位信号
	DW	TTL 电平 (0-5V)、NPN 信号、机械开关 定位、正反转、速度模式下: 反反向限位信号 到位模式下: 正反向或者反方向的到位信号
	GND	电源 GND, 与输入电源同一个 GND

表 4: 电机线接口

插座: kf128-3.81	A+	接步进电机的 A 相线圈的 A+。实际也可接 A-, 电机方向会反置
	A-	接步进电机的 A 相线圈的 A-。实际也可接 A+, 电机方向会反置
	B+	接步进电机的 B 相线圈的 B+。实际也可接 B-, 电机方向会反置
	B-	接步进电机的 B 相线圈的 B-。实际也可接 B+, 电机方向会反置

4. 典型使用及接线图

4.1 典型组网

☆ CAN 总线与计算机网关（以太网、USB、RS232 等转 CAN）连接使用

☆ CAN 总线与嵌入式控制器连接使用

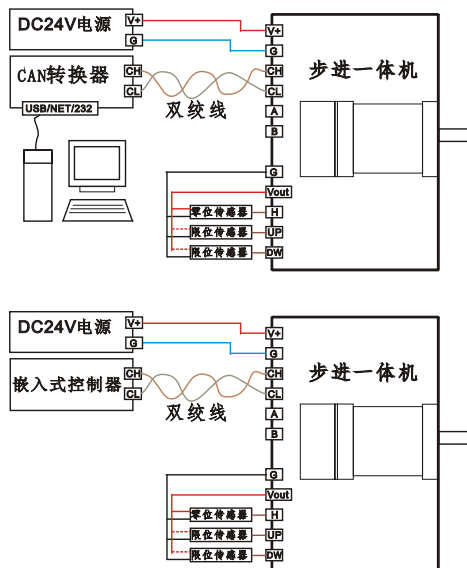


图 2：CAN 总线接线图

4.2 传感器接口典型接线图

推荐使用一个光电开关（输出 PNP 型、或者推挽输出）用于做精准的参考零位，两个限位用限位开关（行程开关、微动开关）等机械开关。如图 3、4。

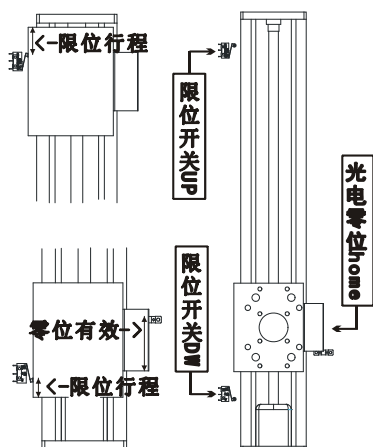


图 3：传感器安装位置参考

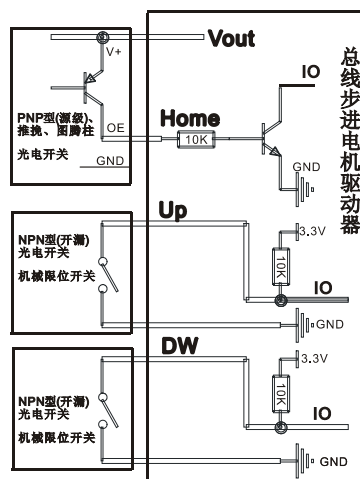


图 4：传感器接口参考接线图

在定位模式下，两个限位开关为可选，零位光电开关是必选项。

5. 运行及工作模式说明

5.1 步进一体机(步进电机、驱动、控制三合一)工作模式

工作模式：定位模式、正反转模式、速度模式、到位模式。不同的工作模式不需设置，步进一体机根据用户发送不同指令，自动切换到对应的工作模式。

步进一体机运行位置数据值变大方向，定义为步进一体机的正方向；步进一体机运行位置数据值变小方向定义为步进一体机的反方向。步进一体机运行方向可软件设定，而应对零位传感器安装在不同位置。

5.2 定位模式

定位模式：定位模式包含复位和定位。用户的机械行程，映射成一个单轴的坐标。通过标度变换，可以将用户的机械行程单位映射为一个有起点和终点的步数坐标。

复位(找零, 找 home): 找机械零点位置。复位需要一个外部零位传感器安装在机械机构上，配合步进一体机共同完成。

执行复位时，零位传感器信号是无效状态，见图 5 所示。步进一体机反方向运行找零最大步数，在运行小于找零最大步数的过程中，检测到零位传感器信号，完成复位。如果步进一体机反转找零最大步数后，没有检测到零位传感器信号则停止运行，给出找零错误的报警信号。CAN 接口可读取，同时 CAN 接口会主动返回报警信号。

执行复位时，零位传感器信号是有效状态，见图 6 所示。步进一体机正方向运行开关脱离步数，然后读取零位传感器信号。如果零位传感器信号无效状态，步进一体机则反转复位；如果零位传感器信号还是有效状态，则给出找零错误的报警信号。CAN 接口可读取，同时 CAN 接口会主动返回报警信号。因此，通过调试后，将开关脱离步数设置成稍大于零位传感器有效信号的行程步数值。

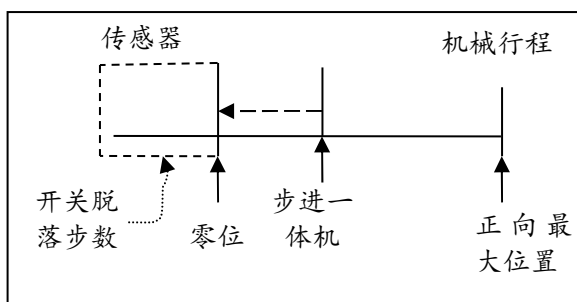


图 5: 复位时，零位信号无效

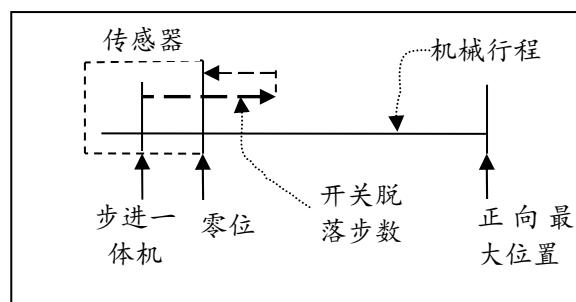


图 6: 复位时，零位信号有效

每次重新上电后，都需要一次复位操作。零位传感器的步进一体机反方向末端位置，需要覆盖用户有效的机械结构行程

定位:复位正确后才可定位。复位完成后，机械机构处于机械零位的位置。通过通信接口发送步进一体机需要去的步数坐标(目标位置)，步进一体机会根据当前位置自动计算运行方向和步数。步进一体机接收到新的定位指令，如果此时还没有完成上一次的定位指令，则将新指令的目标位置作为最终的目标位置，不再关心上一次指令的目标位置。

步进一体机在定位的过程中，还没有达到目标位置时，正方向运行碰到了 UP 限位开关或者反方向运行碰到负 DW 限位开关，步进一体机则减速停止并给出报警信号。CAN 接口可读取，同时 CAN 接口会主动返回报警信号。此时，系统硬件可能出错，需要用户的系统报警并检修处理，步进一体机不响应定位指令。

步进一体机工作在定位模式时，可以执行变速命令。接收到变速指令后，如果此时没有运行到目标位置，则自动通过加减速调整到最新速度；如果此时是在目标位置，在新的目标位置指令后，则按设置的最新速度运行。

5.3 正反转模式

正反转模式下，步进一体机不需要零位的传感器，限位传感器可选。正反转模式下，步进一体机的位置值仅能作为参考。步进一体机在接受到正转指令，则在当前位置正方向运行指定的步数。步进一体机在接受到反转指令，则在当前位置正方向运行指定的步数。步进一体机，在正反转模式下，正方向运行碰到了 UP 限位开关或者反方向运行碰到负 DW 限位开关，步进一体机则减速停止并给出报警信号。CAN 接口可读取，同时 CAN 接口会主动返回报警信号。

正反转模式下，步进一体机碰到限位信号停止后，可以继续再执行正反转指令。再继续执行新的指令时，如果对应的限位信号还存在，虽执行指令，步进一体机因限位信号的存在而不会动作。

举例，步进一体机正方向运行碰到 UP 信号后，给出报警信号。当再接收到正方向运行指令，此时如果限位信号存在，步进一体机不动作，此时如果限位信号不存在，则可继续执行正方向运行指令；当再接收到反方向运行指令，则直接反方向运行。在定位模式下，这两种情况，步进一体机都不动作。

5.4 速度模式

速度模式下，步进一体机不需要零位的传感器，限位传感器可选。速度模式和正反转模式基本一样。正反转模式下，当正转步数为 0 时，步进一体机会一直正转下去，自动切换成为速度模式下的正转；当反转步数为 0 时，步进一体机会一直反转下去，自动切换成为速度模式下的反转。速度模式下，正转碰到 UP 信号和反转碰到 DW 信号，步进一体机则减速停止并给出报警信号。CAN 接口可读取，同时 CAN 接口会主动返回报警信号。

5.5 到位模式

在定位模式、正反转模式、速度模式下，步进一体机的 UP 为正转的限位信号，DW 信号为反转的限位信号。发送定位到位、正反转到位指令后，UP 信号和 DW 信号自动切换成到位信号，不做限位信号用。

5.6 调试步骤(供参考)

第一步：接好电源和通信线后，使用调试软件或者自发指令，发送给步进一体机(不带机械负载)正转一个步数，查看电机是否转动，转动为正常。

第二步：调试电机方向。步进一体机接入机械机构，发送速度模式的正转步数，步数值先为一个小值，避免机械结构碰撞。查看零位开关是否在反转方向末端，方向相反则设置步进一体机的方向反置。查看机械结构是否能动起来，如果动不起来，则将一体机的速度设小(小于等于 200Rpm)避免扭矩不够。

第三步：调试零位传感器信号。如果需要一体机工作在定位模式下，则接上零位传感器，保证复位速度小于等于 200rpm，发送步进一体机复位指令，此时注意随时可断驱动器电源，以免出现参数不对而碰撞机械机构。发送复位指令后，人为遮挡或者模拟零位信号有效的动作(比如槽型光电开关可以用纸片挡住槽)，看步进一体机是否停下。通过通信接口读取电机状态为复位正确的状态,即空闲状态。如果不能停下，检查线路、光电开关的信号匹配和逻辑状态是否需要反置。人为遮挡或者模拟零位信号有效后，让机械机构自动完成复位功能。此处可能因为找零最大步数步进一体机的参数不合理，需要多次执行复位指令。

第四步：初步调试运行速度。发送速度模式下的正转反命令，正反转的步数由小变大，调试其速度对应是否能将机械机构运行起来，大概运行顺畅即可。可初步调试出，复位所需要的找零最大步数、开关脱落步数。

第五步：调试限位信号。在有限位开关信号情况下，发送速度模式下的正反转指令，人为遮挡或者模拟限位信号有效的动作，看步进一体机是否停下。通过通信接口读取电机状态是否为警报信号。如果不能停下，检查线路、光电开关的信号匹配和逻辑状态是否需要反置。

第六步，重新多次调整步进一体机参数，包括细分、加速步数、加速系数、找零速度、运行速度、找零最大步数、光电开关脱落步数等。

5.7 步进一体机参数说明

CAN-ID: 步进一体机可同时或者用其中之一的通信接口，CAN 通信接口为总线并联系统，并联总线的 ID 需要不同

细分：控制步进电机的线圈 A 和 B，将 A 和 B 线圈分别给不同电流，则能将步进电机输出轴矢量合力在步进电机步距角的一个夹角，最终实现一个步距角分几步走完，多少步走完一个步距角则为多少细分。理论上细分可以是任意数值，实际有限制。约定，一个脉冲走一步。如果步进电机步距角是 1.8 度，如果不细分下，200 步步进电机转动一圈；如果细分，步进电机转动一圈，需要走的步数为 $200 \times \text{细分}$ 。

限位开关信号极性：限位传感器漏极有效下为低电平，无效为悬空状态。限位开关常规低电平触发有效，特殊情况可设置为低电平为无效。设置方法详细见表 M02。

零位开关信号极性：零位传感器漏极有效下为高电平，无效为悬空状态。零位开关常规高电平触发有效，特殊情况可设置为高电平为无效。设置方法详细见表 M02。

找零速度：同时也是启动速度，对应启动速度周期(单位 us)，周期值越大，速度越慢，周期值越小，速度越快。对应的 $\text{rpm} = 300000 / (\text{周期值} \times \text{细分})$ 。

运行速度：对应运行速度周期(单位 us)，周期值越大，速度越慢，周期值越小，速度越快。对应的 $\text{rpm} = 300000 / (\text{周期值} \times \text{细分})$ 。已值 Rpm，对应的周期值 = $300000 / (\text{Rpm} \times \text{细分})$ 。

加速步数、加速系数：

电机运行过程（除找零时）的启动和停止有 S 曲线的加速和减速。加速曲线如图 7

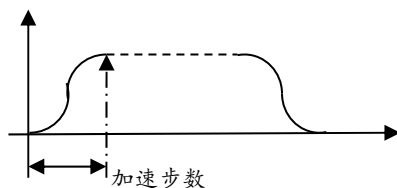


图 7：S 加速曲线

S 曲线参数:

启动速度脉冲周期(PlusStartTime) [单位 us]

运行速度对应脉冲周期(PlusConstantTime) [单位 us]

加速步数(AccSteps)

加速系数(AccCof)

根据实际负载，初次使用时，调整 S 曲线参数的启动速度脉冲周期，运行速度对应脉冲周期，加速系数。

※注意※:

S 曲线参数需要满足一定条件，否则启动后速度和设置速度会不一致。(exp: 自然对数 e 的 x 次方, log: e 为底的对数)

$$\text{PlusStartTime} - \text{PlusConstantTime} < (\exp(\text{AccSteps} * \text{AccCof} / 2) + 1) * 0.95$$
或者

$$\text{AccSteps} > \log((\text{PlusStartTime} - \text{PlusConstantTime}) / 0.95 - 1) * 2 / \text{AccCof}$$
或者

$$\text{AccCof} > \log((\text{PlusStartTime} - \text{PlusConstantTime}) / 0.95 - 1) * 2 / \text{AccSteps}$$

加速时间: $(\text{PlusStartTime} + \text{PlusConstantTime}) * \text{加速步数} * 0.5 [\text{us}]$

加速系数: 数值越小启动越缓慢，能启动更重负载或者能让负载加速到更快速度；同时启动时间变长。

加速步数: 数值越大启动越缓慢，能启动更重负载或者能让负载加速到更快速度；同时启动时间变长。

电机状态、硬件配置: 详见表 M01, M02

表 M01: 电机状态

数值	说明
0xff	开机未定义状态。当不是 0xff 时，各 bit 位表示不同意思
Bit[2:0]	0x0: 空闲状态，不属于警报状态 0x1-7: 运行状态，不属于警报状态
Bit[3]	1: 在复位中; 0: 不在复位中
Bit[7:4]	0x1: 复位状态出错，零位开关未检测，警报状态 0x6: 正转时触发 Up 开关 0x7: 反转时触发 Down 开关

表 M02: 硬件配置

数值	说明(默认值: 0x00)
Bit0	Up 开关逻辑配置。0: 低电平触发; 1: 高电平触发
Bit1	Dw 开关逻辑配置。0: 低电平触发; 1: 高电平触发
Bit2-3	备用
Bit4	空闲锁机。0: 电机状态空闲锁机; 1: 电机状态空闲不锁机
Bit5	备用
Bit6	零位信号逻辑配置。0: 高电平触发; 1: 低电平触发
Bit7	电机方向反置

6. CAN 总线控制命令说明

6.1 CAN 总线数据帧说明

CAN 总线协议为 CAN2.0A，波特率 125K。驱动器的地址为:0x0C1~0x0ff，共 63 个地址，默认为 0xC1。推荐个数小于等于 20 个，通过通信修改地址。

表 5: CAN 总线硬件帧结构(辅助信息忽略)

CAN. ID-OBJ (地址帧)	辅助帧	8 字节数据(数据帧)
-------------------	-----	-------------

CAN 总线上挂着的设备，都有一个自己的设备地址，称为“本机 CAN. ID”。下述用 CAN. ID 表示。表 5 中的 CAN. ID-OBJ：目标 CAN 设备地址，这是硬件帧结构，在 CAN2.0A 中是 11bits 位。在 CAN 总线上挂着的设备，某个设备(CAN. ID)发送一帧硬件 CAN 信息，发给谁呢？就是发给总线上挂着的设备中的地址为 CAN. ID= CAN. ID-OBJ 的 CAN 设备。

6.2 数据帧(8 字节)说明

8 字节数据(数据帧)分为四部分，表 6 所示的“CAN. ID-OBJ ”同表 5。8 字节数据(数据帧)，分段重新自定义，为软件协议帧。

表 6: CAN 总线数据帧结构

CAN. ID-OBJ	8 字节数据(数据帧) [64bits]			
	CAN. ID	功能码类型	功能码	数据码

CAN. ID：发送数据的 CAN 设备本机的 CAN 地址。举例，CAN 主控制器或者 CAN 网关（假设 CAN. ID=0x01）发送 CAN 帧给 CAN 总线步进一体机(地址为 0xC1)，则 CAN. ID-OBJ=0xC1，CAN. ID=0x01；步进一体机(地址为 0xC1)发送 CAN 帧给 CAN 主控制器或者 CAN 网关（假设 CAN. ID=0x01），则 CAN. ID-OBJ=0x01，CAN. ID=0xC1。

功能码类型：CAN 总线系统中，每个 CAN 设备均可主动发送数据，但是在使用中，一般定义 CAN 主控制器或者 CAN 网关为主机(简称:CAN 主控)，CAN 总线步进一体机和其它 CAN 设备为从机(简称:CAN 从机)。CAN 主控发送给 CAN 从机，功能码类型一般取值 0x01，详细值见表 7。CAN 从机发送给 CAN 主控，无论是 CAN 从机应答 CAN 主机还是 CAN 从机主动发送数据给 CAN 主控，功能码类型一般取值 0x02，其它值详细见表 7（比如 CAN 从机对应的步进一体机不支持 CAN 主机发送来的命令，功能码类型值取 0x05）。

功能码：也即指令码，不同的功能码，完成不同的任务。详细见 7.3，命令码(功能码)及参数说明。

数据码：数据码总共 5 个字节。根据不同的命令码，对应不一样含义，配合功能码完成一些参数的读取、动作的完成。

表 7：CAN 总线数据帧结构

字节	说明					软件码
0	本机 CAN 总线地址(11bits)的高 8bits					CAN. ID
1	Bit[7-5]: 本机 CAN 总线地址(11bits)的低 3bits					
	Bit[6-0]: 数据帧序列, 当 Bit[6-0]=0 为一个命令最后一帧数据:一般设置为 0。					
2	Bit[7:5]: 命令码(功能码)类型 0x00 广播命令(不需要返回) 0x01 请求命令(需要返回) 0x02 请求命令的正确返回 0x03 请求命令的不正确返回 0x05 无此命令码 0x06 命令参数错误					功能码 类型
	Bit[4:0]: 命令码(功能码), 详细见命令码(功能码)说明表					功能码
	字符型	半字型	字型	浮点型	数据码说明:	数据码
3	char[0]	ShortDate[0]	IntDate	FloatDate	在传输不同参数时, 用的数据类型不一致。	
4	char[1]	小端模式	小端模式	小端模式		
5	char[2]	ShortDate[1]				
6	char[3]	小端模式				
7	char[4]	字节: ByteData				

6.3 命令码(功能码)及参数说明

命令码 0x00：通信测试命令

命令码	CAN 总线说明
0x00	通信测试命令，CAN 主控发送给 CAN 从机时； ●命令码：0x00 ●命令码类型：0x01, 详见表 7 的字节 2(Bit[7:5]) ●数据码：任意值 CAN 从机返回或者主动发送给 CAN 主控时，数据帧格式： ●命令码：0x00 ●命令码类型：0x02, 详见表 7 的字节 2(Bit[7:5]) ●数据码返回为“字型”。IntDate 表示当前步数，ByteData 表示电机状态。电机状态详见表 M01

命令码 0x01: 电机复位

命令码	CAN 总线说明
0x01	<p>电机复位(电机定位模式)，CAN 主控发送给 CAN 从机时：</p> <ul style="list-style-type: none"> ●命令码：0x01 ●命令码类型：0x01, 详见表 7 的字节 2(Bit[7:5]) ●数据码 ByteData = 1, IntDate 忽略 <p>CAN 从机返回或者主动发送给 CAN 主控时，数据帧格式：</p> <ul style="list-style-type: none"> ●命令码：0x01 ●命令码类型：0x02, 详见表 7 的字节 2(Bit[7:5]) ●数据码返回为“字型”。IntDate 表示当前步数，ByteData 表示电机状态。

命令码 0x02: 电机定位

命令码	CAN 总线说明
0x02	<p>电机定位，数据码为“字型”。CAN 主控发送给 CAN 从机时：</p> <ul style="list-style-type: none"> ●命令码：0x02 ●命令码类型：0x01, 详见表 7 的字节 2(Bit[7:5]) ●数据码（如下前提：步进一体机正确复位情况下） <ul style="list-style-type: none"> ->ByteData = 1, IntDate 为定位的步数坐标, 执行定位模式的定位功能。到达目标后，主动发送命令给 CAN 主控。正向运行碰到 UP 开关，停下警报。反向运行碰到 DW 开关，停下警报。 ->ByteData = 6, IntDate 为定位的步数坐标, 执行到位模式。碰到 UP 有效边缘信号停止，碰不到信号则运行到目标坐标位置。到达目标或者碰到 UP 停止后，主动发送命令给 CAN 主控。 ->ByteData = 7, IntDate 为定位的步数坐标, 执行到位模式。碰到 DW 有效边缘信号停止，碰不到信号则运行到目标坐标位置。到达目标或者碰到 DW 停止后，主动发送命令给 CAN 主控。 <p>CAN 从机返回或者主动发送给 CAN 主控时，数据帧格式：</p> <ul style="list-style-type: none"> ●命令码：0x02 ●命令码类型：0x02, 详见表 7 的字节 2(Bit[7:5]) ●数据码返回为“字型”。IntDate 表示当前步数，ByteData 表示电机状态。

命令码 0x03: 电机正转

命令码	CAN 总线说明
0x03	<p>电机正转，数据码为“字型”。CAN 主控发送给 CAN 从机时：</p> <ul style="list-style-type: none"> ●命令码：0x03 ●命令码类型：0x01, 详见表 7 的字节 2(Bit[7:5]) ●数据码 <ul style="list-style-type: none"> ->ByteData= 3, 无论当前电机状态如何，在当前的位置正转 IntDate(IntDate!=0)步。步进一体机执行正反转模式的正转功能。到达目标后，主动发送命令给 CAN 主控。IntDate=0, 步进一体机执行速度模式下的正转。无论 IntDate 何值，正向运行碰到 UP 开关，停下警报。执行此命令一次后，步进一体机状态强制变为“正确复位”，但此时的步数值和机械机构尺寸没有对应关系。 ->ByteData = 6, 步进一体机正确复位情况下，在当前的位置正转

	<p>IntDate, 执行到位模式。碰到 UP 有效边沿信号停止, 碰不到信号则运行到目标坐标位置。到达目标或者碰到 UP 停止后, 主动发送命令给 CAN 主控。</p> <p>->ByteData = 7, 步进一体机正确复位情况下, 在当前的位置正转 IntDate, , 执行到位模式。碰到 DW 有效边沿信号停止, 碰不到信号则运行到目标坐标位置。到达目标或者碰到 DW 停止后, 主动发送命令给 CAN 主控。</p> <p>CAN 从机返回或者主动发送给 CAN 主控时, 数据帧格式:</p> <ul style="list-style-type: none"> ● 命令码: 0x03 ● 命令命类型: 0x02, 详见表 7 的字节 2(Bit[7:5]) ● 数据码返回为“字型”。IntDate 表示当前步数, ByteData 表示电机状态。
--	---

命令码 0x04: 电机正转

命令码	CAN 总线说明
0x04	<p>电机正转, 数据码为“字型”。CAN 主控发送给 CAN 从机时:</p> <ul style="list-style-type: none"> ● 命令码: 0x04 ● 命令命类型: 0x01, 详见表 7 的字节 2(Bit[7:5]) ● 数据码 <p>->ByteData= 3, 无论当前电机状态如何, 在当前的位置反转 IntDate(IntDate!=0)步。步进一体机执行正反转模式的反转动传功能。到达目标后, 主动发送命令给 CAN 主控。</p> <p>IntDate=0, 步进一体机执行速度模式下的反转。无论 IntDate 何值, 正向运行碰到 DW 开关, 停下警报。执行此命令一次后, 步进一体机状态强制变为“正确复位”, 但此时的步数值和机械机构尺寸没有对应关系。</p> <p>->ByteData = 6, 步进一体机正确复位情况下, 在当前的位置反转 IntDate, 执行到位模式。碰到 UP 有效边沿信号停止, 碰不到信号则运行到目标坐标位置。到达目标或者碰到 UP 停止后, 主动发送命令给 CAN 主控。</p> <p>->ByteData = 7, 步进一体机正确复位情况下, 在当前的位置反转 IntDate, , 执行到位模式。碰到 DW 有效边沿信号停止, 碰不到信号则运行到目标坐标位置。到达目标或者碰到 DW 停止后, 主动发送命令给 CAN 主控。</p> <p>CAN 从机返回或者主动发送给 CAN 主控时, 数据帧格式:</p> <ul style="list-style-type: none"> ● 命令码: 0x04 ● 命令命类型: 0x02, 详见表 7 的字节 2(Bit[7:5]) ● 数据码返回为“字型”。IntDate 表示当前步数, ByteData 表示电机状态。

命令码 0x05: 电机停止

命令码	CAN 总线说明
0x05	<p>电机停止当前的运行(速度模式、定位模式、正反转模式、到位模式), CAN 主控发送给 CAN 从机时:</p> <ul style="list-style-type: none"> ● 命令码: 0x05 ● 命令命类型: 0x01, 详见表 7 的字节 2(Bit[7:5])

	<p>●数据码</p> <p>->ByteData= 1: 减速停止</p> <p>->ByteData= 2: 立即停止</p> <p>CAN 从机返回或者主动发送给 CAN 主控时, 数据帧格式:</p> <p>●命令码: 0x05</p> <p>●命令命类型: 0x02, 详见表 7 的字节 2(Bit[7:5])</p> <p>●数据码返回为“字型”。IntDate 表示当前步数, ByteData 表示电机状态。</p>
--	---

命令码 0x06: 改变步进一体机当前速度

命令码	CAN 总线说明
0x06	<p>改变当前的电机速度, CAN 主控发送给 CAN 从机时:</p> <p>●命令码: 0x06</p> <p>●命令命类型: 0x01, 详见表 7 的字节 2(Bit[7:5])</p> <p>●数据码</p> <p>->ByteData= 0: 数据码为“浮点型”。FloatDate 表示目标 Rpm(1.0-1000.0)</p> <p>->ByteData= 1: 数据码为“字型”。表示速度脉冲周期[us]</p> <p>CAN 从机返回或者主动发送给 CAN 主控时, 数据帧格式:</p> <p>●命令码: 0x06</p> <p>●命令命类型: 0x02, 详见表 7 的字节 2(Bit[7:5])</p> <p>●数据码返回:</p> <p>ByteData = 0, 数据码为“浮点型”, 实际执行的 Rpm。</p> <p>ByteData = 1, 数据码为“字型”, 实际执行的脉冲周期值[us]。</p>

命令码 0x13: 修改步进一体机参数

命令码	CAN 总线说明
0x13	<p>改变当前的电机速度, CAN 主控发送给 CAN 从机时:</p> <p>●命令码: 0x13</p> <p>●命令命类型: 0x01, 详见表 7 的字节 2(Bit[7:5])</p> <p>●数据码: 详见表 8</p> <p>CAN 从机返回或者主动发送给 CAN 主控时, 数据帧格式:</p> <p>●命令码: 0x13</p> <p>●命令命类型: 0x02, 详见表 7 的字节 2(Bit[7:5])</p> <p>●数据码返回: 详见表 8</p>

命令码 0x14: 保存参数

命令码	CAN 总线说明
0x14	<p>改变当前的电机速度, CAN 主控发送给 CAN 从机时:</p> <p>●命令码: 0x14</p> <p>●命令命类型: 0x01, 详见表 7 的字节 2(Bit[7:5])</p> <p>●数据码</p> <p>->ByteData=0: 将 0x13 命令改变的参数存于内部 FLASH。</p> <p>CAN 从机返回或者主动发送给 CAN 主控时, 数据帧格式:</p> <p>●命令码: 0x14</p> <p>●命令命类型: 0x02, 详见表 7 的字节 2(Bit[7:5])</p> <p>●数据码: 忽略值</p>

表 8: 0x13 的数据码

ByteData	说明(电机参数命令码 0x13)
0	字型; 设置加速步数, IntDate 为步数参数 [1~3000]
1	字型; 读取加速步数, IntDate 忽略
4	字型; 设置找零步数, IntDate 为步数参数 [1~0x7fffffff]
5	字型; 读取找零步数, IntDate 忽略
6	字型; 设置脱落步数, IntDate 为步数参数 [1~0x7fffffff]
7	字型; 读取脱落步数, IntDate 忽略
8	字型; 设置模块 CAN. ID, IntDate 为 ID [0xC1~0xFF]
9	字型; 读取模块 ID, IntDate 忽略
10	字型; 设置细分, IntDate 为[1 2 4 8 16 32 64 128]
11	字型; 读取细分, IntDate 忽略
12	字型; 设置步进一体机的硬件参数, 详见表 M02
13	字型; 读取进一体机的硬件参数, IntDate 忽略
14	字型; 设置运行速度周期, IntDate 为周期 us[20~5000]
15	字型; 读取运行速度冲周期, IntDate 忽略
16	字型; 设置找零速度脉冲周期, IntDate 为周期 us[200~5000]
17	字型; 读取找零速度脉冲周期, IntDate 忽略
18	浮点型; 设置加速系数
19	浮点型; 读取加速系数
22	浮点型; 设置运行电流 0.3~1.7
23	浮点型; 读取运行电流

表 13 所述: 设置参数时, CAN 主控发送给 CAN 从机的 IntData/FloatData 为设置的值, CAN 从机返回给 CAN 主控的为步进一体机的实际设置值; 读取参数时, CAN 主控发送给 CAN 从机的 IntData/FloatData 值忽略, CAN 从机返回给 CAN 主控的为步进一体机的当前值。

6.4 举例说明

CAN 主控发送正转 1000 步命令给步进一体机：功能码 03、功能码类型 01、数据码前 4 字节=1000、数据码第 5 字节=3

表 9：CAN 主控发送正转 1000 步协议分解

CAN. ID-Obj	8字节数据(数据帧)						
	CAN. ID(11Bits)			功能码类型	功能码	数据码	
0xC1	0x001 (HEX)			0x01	0x03	1000	03
0xC1	00	0x20[b0010 0000]		0x23[b0010 0011]		0x000003E8(1000)	03
0xC1	00	20	23	E8	03	00	00 03

步进一体机返回 CAN 主控：功能码 03、功能码类型 03、数据码前 4 字节=1000、数据码第 5 字节=3

表 10：CAN 从机运行 1000 后的数据返回

CAN. ID-Obj	8字节数据(数据帧)					
	CAN. ID(11Bits)		类型	功能码	数据码	
0x01	0x0C1[000 1100 0001]		0x02	0x03	xxxx xxxx	xx
0x01	0x18[000 1100 0]	0x20[b0010 0000]	0x43[b0100 0011]		xxxx xxxx	xx
0x01	18 20 43 xx xx xx xx[电机当前位置] xx[电机状态]					

7. 包装

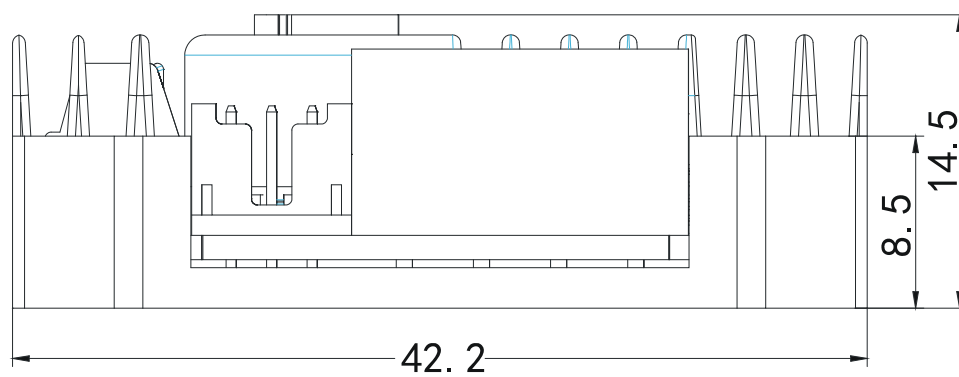
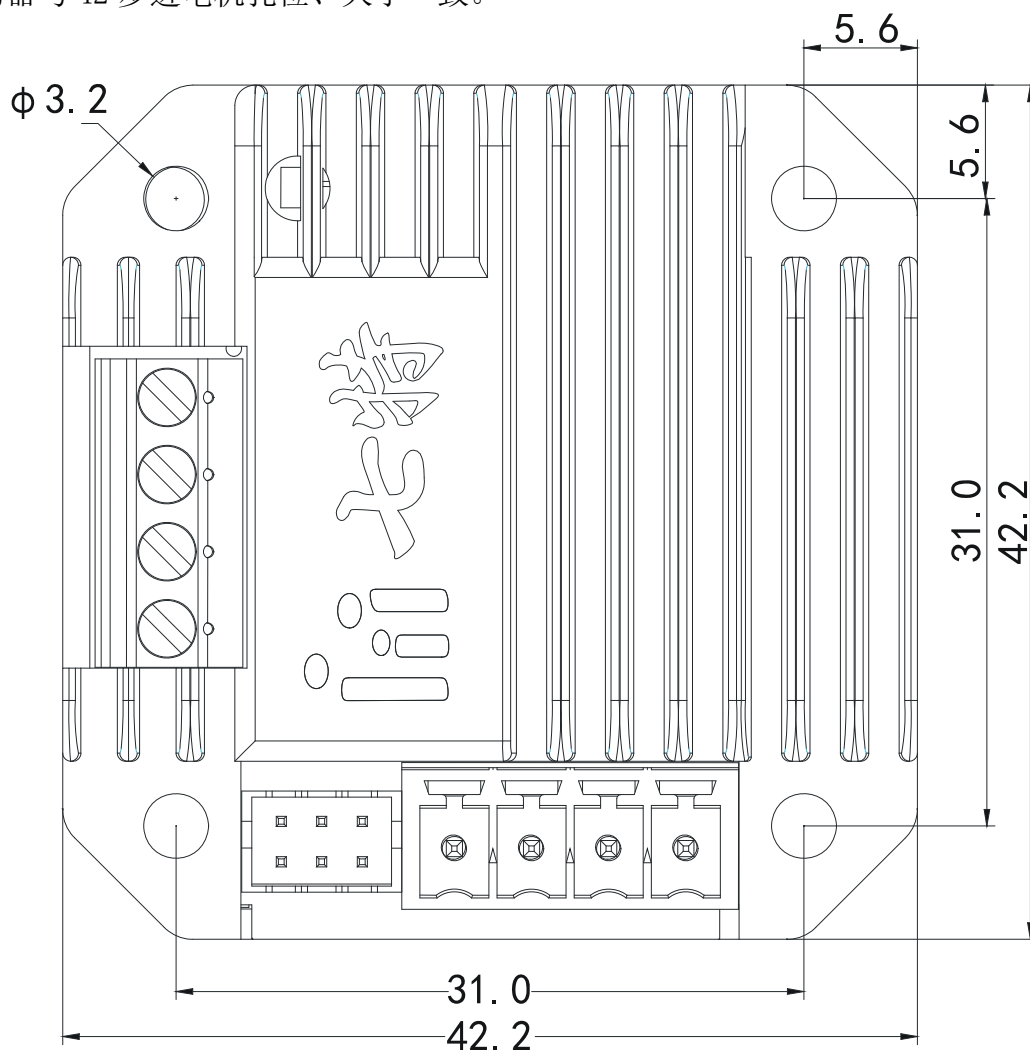
序号	数量	部件
1	1	CAN 接口 42 型步进电机驱动器 [7TCSM4210Q]
2	1	PH2.0-2×3P 带线
3	1	KF128-3.81-4P

8. 买家必读

- 1) 在使用步进电机的过程中,请确保接线正确,无虚接、压线皮等问题,否则可能造成虚、短接引起打火。
- 2) 请注意看清正负标示,避免因将电机的正负极接错出现烧毁现象,不要带电拔插。
- 3) 选取电机的电流参数,不可超过 2A,否则容易发生保护。
- 4) 电流调节电位器不能拧得过紧,容易造成过量使调节钮松动。
- 5) 扭矩请考虑充分余量,否则差异性可能引起某些情况下扭矩不够。
- 6) 请注意使用环境要求,为其提供适用且优化的环境,并在环境中注意避免金属屑、油污、粉尘及腐蚀性气体。
- 7) 批量购买前请先测试好,无质量问题,不接受批量退货。
- 8) 需要高低温要求,请订购前联系厂家做定制及高低温筛选。
- 9) 未经公司书面同意,不得自行拆装。
- 10) 确保应用符合电气相关技术规范。

附件一 控制器外形尺寸图

驱动器与 42 步进电机孔位、大小一致。



附件二 控制器安装示意图

控制器可以与 42 步进电机一体化安装,配有不同规格 42 步进电机(高度不一致)所需要的 ϕ 3 螺钉 2 个。

与电机一体化安装时: 建议保留步进电机 2 个对角螺丝。

