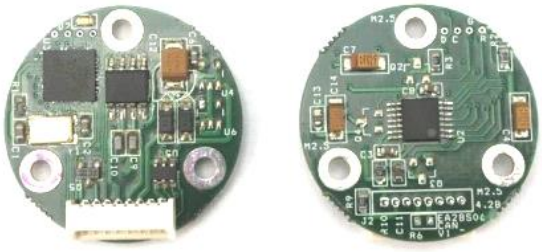
绝对值编码器 EA28S06C



接口

接口	CAN BUS(TJA1050)
通讯协议	CAN2.0b / CANopen
诊断	内存、接口自诊断
编程功能（部分可选）	子地址、波特率、预设(初始位置)，计数方向
传输波特率	最高 1Mbps
数据刷新时间	<100us
输出驱动器	差分电压

电气数据

电源电压	5VDC±5%（电源符合EN 50178）
电流消耗	< 100mA
功耗	< 0.5W
启动时间	10ms
EMC:发射干扰	EN 61000-6-4
EMC:抗干扰	EN 61000-6-2
MTTF：平均无故障时间	10 年，在40 ℃下

传感器

技术	磁电
最高圈数	65536 r（外接后备电池）
单圈分辨率	16384 cpr
精度(INL)	±0.1°
码制	二进制码

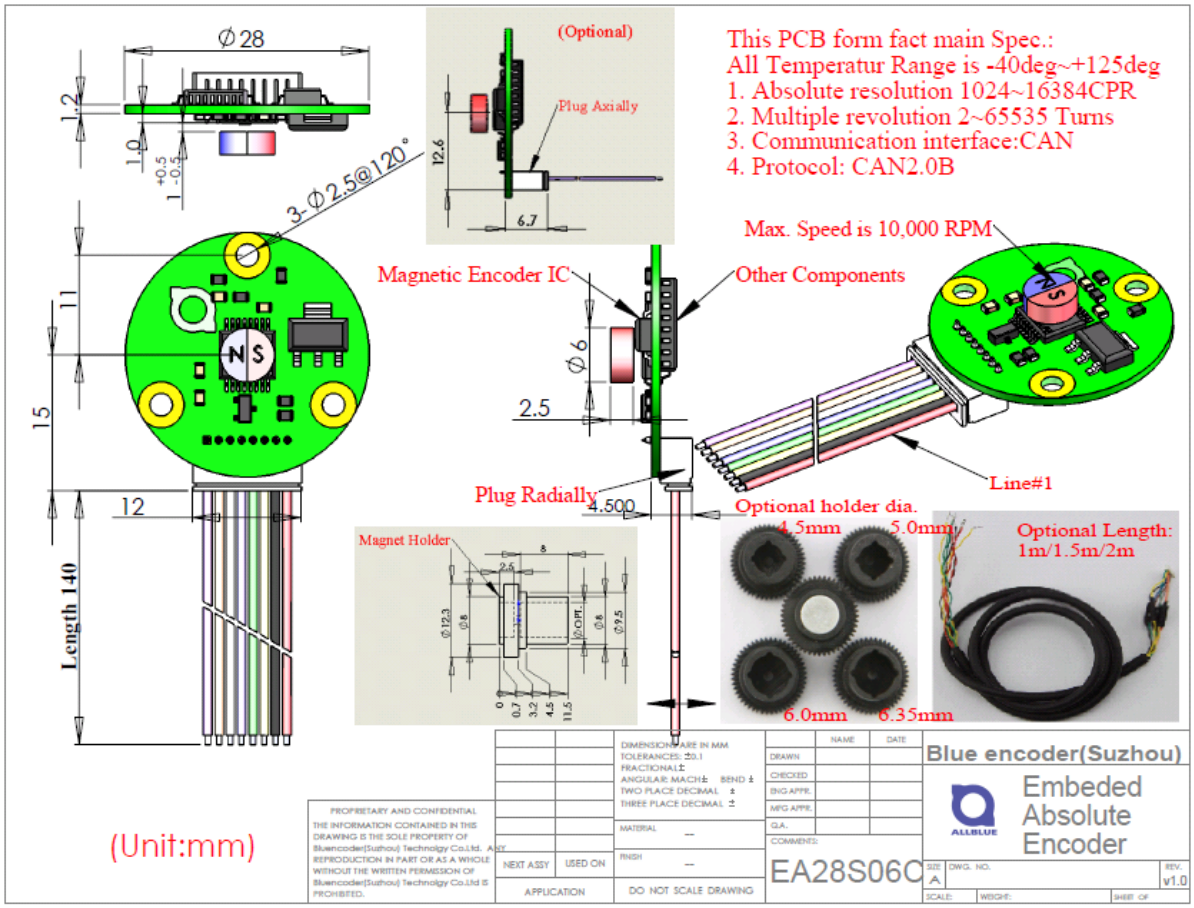
环境规格

工作温度	-40 ℃～+105 ℃
存储温度	-40 ℃～+125 ℃

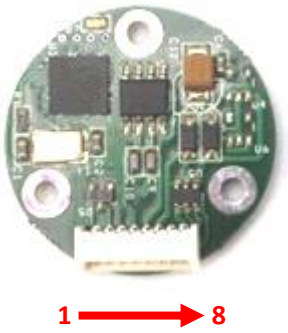
机械数据

产品直径	Ø28mm
产品厚度	3.5mm
最高机械速度	10000rpm
重量	3g（不包含磁铁和线）

尺寸图、插脚号定义图



信号电气接口示意图

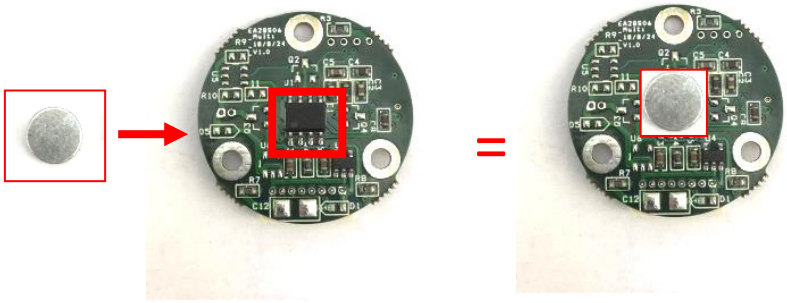


引脚号	引脚定义	说明
1	GND	电源地
2	STATUS	状态指示输出 1 - Ready 0 - Busy
3	SETPOS	位置设置输入(低电平有效,正常使用时悬空或高电平)
4	CAN_L	CAN 接口低电平接口
5	CAN_H	CAN 接口高电平接口
6	VBAT	备用电池
7	GND	电源地
8	VCC	电源 DC5V

SETPOS 是外部复用输入引线，具备两种功能：
1. 恢复出厂默认设置：
编码器上电前，预先将 SETPOS 与 GND 短接，上电 3 秒后，将 SETPOS 与 GND 断开并悬空。
2.设置原始位置(预设值)：
编码器通电时，将 SETPOS 与 GND 短接 1 秒后，将 SETPOS 与 GND 断开并悬空。

STATUS 是编码器状态输出引线： 1--Ready 0—Busy

离轴磁铁安装方式：
将磁铁对中(偏差<0.2mm)编码器背面红框内芯片，并位于其上方 1mm(偏差<0.5mm)位置处。



注意：由于编码器上电时会检测磁铁的位置，确认磁场正常后才能进行数据通讯

板载 LED 指示灯：
1. 快速闪烁：未检测到磁铁正确位置
2. 熄灭：正常待机中
3. 高频亮灯：数据通讯进行中

CAN 总线通讯方式（CAN2.0b）：

出厂默认通讯参数为:
设备 NodeID：0x03
从模式标准帧：id = 0x580+NodeID
波特率：1Mbps

要求主控端标准帧 id = 0x80(广播)或 0x600+NodeID(单点)

一、编码器设置解锁命令：
除编码器位置数据读出命令外，其他设置命令都需经此解锁命令后才可执行，编码器每次上电后处于默认锁定状态。
1. 主控端发送 CAN 命令帧：
cansndbuf[0] = 0x55;　　//解锁命令标志 1

```
cansndbuf[1] = 0xAA;        //解锁命令标志 2
cansndbuf[2] = 0xF7;        //解锁命令标志 3
cansndbuf[3] = 0x7F;        //解锁命令标志 4
cansndbuf[4] = 0x55;        //解锁命令标志 4
cansndbuf[5] = 0x0D;        //解锁命令标志 5
cansndbuf[6] = 0x0A;        //解锁命令标志 6
cansndbuf[7] = NodeID;      // NodeID 为该编码器从设备地址
CAN1_Send_Msg(cansndbuf,8);
```

2. 主控端接收编码器返回的 CAN 应答帧：

```
unsigned char getdata[2];

CAN_Receive(CAN1, CAN_FIFO0, &RxMessage);
if((RxMessage.StdId==NodeID)&&(RxMessage.DLC==2)) //NodeID 是为该编码器设置的从设备地址
  {
    getdata[0] = RxMessage.Data[0];  // getdata[0]应该等于 0xAA
    getdata[1] = RxMessage.Data[1];  // getdata[1]应该等于 NodeID
  }
//并且编码器从端 LED 闪烁 3 次，表示解锁成功
```

二、编码器从地址设置命令(注：此命令只可单机运行)：

1. 主控端发送 CAN 命令帧：

```
cansndbuf[0] = 0xA0;        //从设备地址设置命令标志
cansndbuf[1] = NodeID;      //将该编码器的从设备地址设置为 NodeID，允许设置范围 0x00~0x7F
CAN1_Send_Msg(cansndbuf,2);
```

2. 主控端接收编码器返回的 CAN 应答帧：

```
unsigned char getdata[2];

CAN_Receive(CAN1, CAN_FIFO0, &RxMessage);
if((RxMessage.StdId==NodeID)&&(RxMessage.DLC==2)) // NodeID 是为该编码器设置的从设备地址
  {
    getdata[0] = RxMessage.Data[0];  // getdata[0]应该等于 0xA0
    getdata[1] = RxMessage.Data[1];  // getdata[1]应该等于 NodeID
  }
//并且编码器从端 LED 闪烁 3 次，表示从地址设置成功
```

三、编码器位置数据读出命令(仅此命令不受设置锁定命令影响，任何时候都可以有效执行)：

1. 主控端发送 CAN 命令帧：

主控端标准帧 ID 设置为 0x80 或 0x600+NodeID，发送一帧空数据即可

2. 主控端接收编码器发出的 CAN 位置数据帧：

```
unsigned int Turns,Angle;

CAN_Receive(CAN1, CAN_FIFO0, &RxMessage);
if((RxMessage.StdId==NodeID)&&(RxMessage.DLC==4))
   {
      Turns = RxMessage.Data[0];
      Turns = (Turns << 8)|RxMessage.Data[1];    //对于单圈型编码器，圈数读数恒为 0
      Angle = RxMessage.Data[2];
      Angle = (Angle << 8)|RxMessage.Data[3];
   }
```

四、编码器内部参数读出命令：

1. 主控端发送 CAN 命令帧：

```
cansndbuf[0] = 0xA2;    //参数读出命令标志
cansndbuf[1] = NodeID; // NodeID 为该编码器从设备地址
CAN1_Send_Msg(cansndbuf,2);
```

2. 主控端接收编码器发出的 CAN 参数数据帧：

```
unsigned char    NodeID,Baudrate,Direction;
unsigned int PreTurns,PreAngle;

CAN_Receive(CAN1, CAN_FIFO0, &RxMessage);
if((RxMessage.StdId== NodeID)&&(RxMessage.DLC==7))
   {
      NodeID = RxMessage.Data[0];
      Baudrate = RxMessage.Data[1];
      Direction = RxMessage.Data[2];
      PreTurns = RxMessage.Data[3];
      PreTurns = (PreTurns << 8)|RxMessage.Data[4]; //对于单圈型圈数读数恒为 0
      PreAngle = RxMessage.Data[5];
      PreAngle = (PreAngle << 8)|RxMessage.Data[6];
   }
//并且编码器从端 LED 闪烁 3 次，表示参数读出成功
```

五、编码器内部参数组一设置命令：

1. 主控端发送 CAN 命令帧：

```
cansndbuf[0] = 0xA3;          //第一组参数设置命令标志
cansndbuf[1] = NodeID;        // NodeID 指定需要设置参数的编码器
cansndbuf[2] = Baudrate;      //设置范围 0x01~0x12,对应 18 个不同波特率（附表 1）
cansndbuf[3] = Direction;     //旋转方向  0x00--CW , 0x01—CCW
cansndbuf[4] = PreTurns_H; //圈数预设值高字节(对于单圈型设置为0)
```

```
cansndbuf[5] = PreTurns_L; //圈数预设值低字节(对于单圈型设置为0)
cansndbuf[6] = PreAngle_H; //角度预设值高字节
cansndbuf[7] = PreAngle_L; //角度预设值低字节
CAN1_Send_Msg(cansndbuf,8);
```

2. 主控端接收编码器返回的 CAN 应答帧：

```
unsigned char getdata[2];

CAN_Receive(CAN1, CAN_FIFO0, &RxMessage);
if((RxMessage.StdId==NodeID)&&(RxMessage.DLC==2)) //NodeID 是为该编码器设置的从地址
  {
    getdata[0] = RxMessage.Data[0];  // getdata[0]应该等于 0xA3
    getdata[1] = RxMessage.Data[1];  // getdata[1]应该等于 NodeID
  }
//并且编码器从端 LED 闪烁 3 次，表示参数设置成功
```

六、设置参考位置命令(与外部 SETPOS 引脚功能相同**)：**

1. 主控端发送 CAN 命令帧：

```
cansndbuf[0] = 0xA4;          //设置参考位置命令标志
cansndbuf[1] = NodeID;        //将该地址编码器的当前位置，设置为 PreTurns &PreAngle 中的值
CAN1_Send_Msg(cansndbuf,2);
```

2. 主控端接收编码器返回的 CAN 应答帧：

```
unsigned char getdata[2];

CAN_Receive(CAN1, CAN_FIFO0, &RxMessage);
if((RxMessage.StdId==NodeID)&&(RxMessage.DLC==2)) /
  {
    getdata[0] = RxMessage.Data[0];  // getdata[0]应该等于 0xA4
    getdata[1] = RxMessage.Data[1];  // getdata[1]应该等于 NodeID
  }
//并且编码器从端 LED 闪烁 3 次，表示参考位置设置成功
```

附表 1：（波特率代号对应值）

设置值	对应值	设置值	对应值	设置值	对应值
1	1M	7	300K	13	90K
2	900K	8	250K	14	80K
3	800K	9	200K	15	60K
4	600K	10	150K	16	50K
5	500K	11	125K	17	40K
6	400K	12	100K	18	30K

CANopen 通讯协议：

1.1 出厂设置

编码器出厂时初始参数设置如下：
波特率：250kbps；
节点号（Node-ID）：07H；
工作模式：循环发送模式，发送周期为 200 毫秒；
心跳报文：上电发送心跳报文，发送周期为 1000 毫秒；
旋转方向：沿轴向看顺时针旋转数据增加；
CAN 总线终端电阻：无。

1.2 CAN 桢 ID 说明

编码器 CAN 总线数据桢采用标准桢格式，CAN 桢 ID（COB-ID）为 11 位，根据 CANopen 协议规范将报文 ID（COB-ID）划分为功能码（Function-ID）和节点号（Node-ID）。如图 2 所示，功能码占用 COB-ID 的高 4 位，节点号占用 COB-ID 的低 7 位，节点号可设置的范围为 1~127。

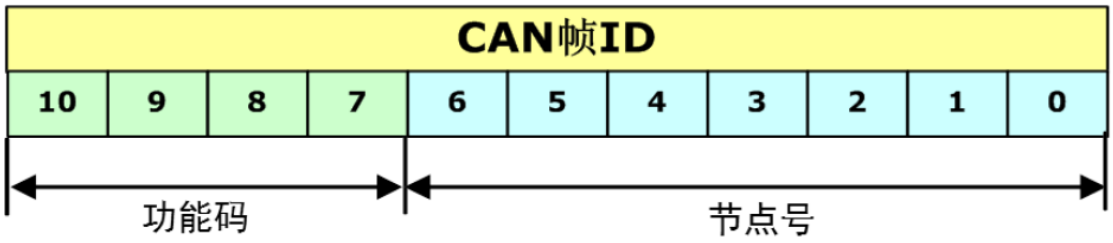


图 2 COB-ID 说明图

1.3 角度数据格式

编码器发送的角度数据为其当前绝对位置。角度位置的发送形式有循环和同步两种模式，循环模式下角度数据桢的 CAN 桢 ID 为 180H+ Node-ID,同步模式下角度数据桢的 CAN 桢 ID 为 280H+ Node-ID。

角度数据帧的长度均为 4 个字节，低字节在前，高字节在后，不足 4 个字节时高位补 0。例如一台多圈分辨率为 12 位，单圈分辨率为 12 位的编码器，返回的角度数据帧为 16 87 CF 00 时，表示编码器当前所在位置的多圈值为 0xCF8，单圈值为 0x716，因总分辨率为 24 位（12 位多圈+12 位单圈），只用到 3 个字节，因此最高字节为 0x00。

1.4 心跳报文

编码器工作时会定时向总线发送心跳报文，以方便总线上其他设备确认其工作状态并对其进行网络管理。心跳报文默认每隔 1 秒钟发一次，用户可根据需要通过参数更改指令关闭心跳报文功能或者更改其发送周期。心跳报文的 CAN 桢 ID 为 700H+ Node-ID，数据为 1 个字节，该数据代表当前编码器的工作状态：04H 为停止状态；05H 为操作状态；7FH 为预操作状态。

编码器上电或者复位后会向总线发送一次节点上线报文，提示总线上其他设备其已经加入到总线上。节点上线报文的 CAN 桢 ID 为 700H+ Node-ID，数据为 1 个字节 00H。

1.5 网络管理 NMT

网络上其他设备可通过发送节点状态切换命令来控制编码器启动、重启或停止。节点状态切换命令的 CAN 桢 ID 为 000H，CAN 总线优先级为最高，数据为两个字节。节点状态切换命令的 CAN 桢格式见表 3

表 3 节点状态切换命令桢格式

CAN 桢 ID	数据长度	数据 1	数据 2
000H	2	命令符	Node-ID/00H

数据 1 代表命令类型，有效命令共有 5 种，分别为 01H,02H,80H,81H,82H，不同命令的含义如下所示：

01H 为启动命令。编码器接收到启动命令后，心跳报文变为 05H，进入操作状态。在此状态下，编码器根据不同的工作模式，循环或者同步向总线发送当前角度数据；此外，编码器响应节点状态切换命令和参数更改指令。

02H 为停止命令。编码器接收到停止命令后，心跳报文变为 04H，进入停止状态。在此状态下，编码器只接收节点状态切换命令，停止向总线发送当前角度数据并且不响应网络上其他设备对其参数的更改指令。

80H 为预操作命令。编码器接收到预操作命令后，心跳报文变为 7FH，进入预操作状态。在此状态下，编码器停止向总线发送当前角度数据，但响应节点状态切换命令和参数更改指令。上电后编码器发送一次节点上线报文，然后进入预操作状态。

81H 为系统复位命令。编码器接收到系统复位命令后，产生内部系统复位，让编码器重新初始化。复位后编码器先发送一次节点上线报文，然后进入预操作状态。

82H 为 CAN 总线复位命令。编码器接收到 CAN 总线复位命令后，编码器重新初始化 CAN 模块和 CANopen 通讯。复位后编码器先发送一次节点上线报文，然后进入预操作状态。

数据 2 代表要控制的编码器的节点号，如果要控制总线上所有的编码器，则第二个字节的值为 00H。

1.6 参数更改指令

1.6.1 参数更改指令数据帧格式

网络上其他设备可通过 CANopen 协议中的快速 SDO 协议对编码器进行参数更改，参数更改指令的 CAN 帧 ID 为 600H+ Node-ID。编码器成功接收到参数更改指令后，会响应该指令，响应帧的 CAN 帧 ID 为 580H+ Node-ID，配置帧和响应帧的数据帧长度均为 8 个字节，数据帧格式见表 4

表 4 参数更改指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
命令符	索引	索引	子索引	参数值 1	参数值 2	参数值 3	参数值 4

命令符分为写命令和读命令。写命令用于将新的配置参数写入编码器，读命令用于读取编码器当前的参数设置。命令符的有效值如下所示：

22H：写入　　40H：读取
2FH：写入 1 个字节　　4FH：读响应 1 个字节
2BH：写入 2 个字节　　4BH：读响应 2 个字节
27H：写入 3 个字节　　47H：读响应 3 个字节
23H：写入 4 个字节　　43H：读响应 4 个字节
60H：写成功应答　　80H：异常应答

索引与子索引用于区分编码器参数的类型和功能，编码器的每个参数对应一个索引和子索引值。索引值为 2 个字节，低字节在前，高字节在后，子索引为 1 个字节。

参数值共 4 个字节，低字节在前，高字节在后，若参数值不足 4 个字节则高位补 0。

1.6.2 方向指令

写命令

网络上其他设备对编码器更改方向指令的数据帧见表 5，其中 XX=00H 时，编码器沿轴向看顺时针角度增加；XX=01H 时，编码器沿轴向看顺时针角度减小。

表 5 方向写指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H/2FH	00H	21H	00H	XXH	00H	00H	00H

编码器成功接收到指令后响应帧见表 6

表 6 方向写指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
60H	00H	21H	00H	00H	00H	00H	00H

读命令

网络上其他设备读取编码器方向参数的数据帧见表 7

表 7 方向读指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
40H/4FH	00H	21H	00H	00H	00H	00H	00H

编码器成功接收到指令后响应帧见表 8

表 8 方向读指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
4FH	00H	21H	00H	XXH	00H	00H	00H

1.6.3 节点号指令

写命令

网络上其他设备对编码器更改节点号指令的数据帧见表 9，其中节点号的值=XX+1，节点号的范围为 1-127。节点号更改成功后不会立即生效，需再发送保存命令后才能生效。

表 9 节点号写指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H/2FH	00H	30H	00H	XXH	00H	00H	00H

编码器成功接收到指令后响应桢见表 10

表 10 节点号写指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
60H	00H	30H	00H	00H	00H	00H	00H

读命令

网络上其他设备读取编码器节点号参数的数据帧见表 6.8

表 11 节点号读指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
40H/4FH	00H	30H	00H	00H	00H	00H	00H

编码器成功接收到指令后响应桢见表 6.9

表 12 节点号读指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
4FH	00H	30H	00H	XXH	00H	00H	00H

1.6.4 循环模式指令

写命令

网络上其他设备对编码器更改角度发送周期指令的数据帧见表 13，角度数据发送周期=YYXXH 毫秒，范围为 1-65535 毫秒。当 XX 和 YY 都为 00H 时，编码器退出循环模式进入同步模式，同步模式下，编码器停止循环向总线发送角度数据，只有在收到一定数量的同步帧后才向总线发送一帧角度数据。

表 13 循环周期写指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H/2BH	00H	18H	05H	XXH	YYH	00H	00H

编码器成功接收到指令后响应桢见表 14

表 14 循环周期写指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
60H	00H	18H	05H	00H	00H	00H	00H

读命令

网络上其他设备读取编码器发送周期参数的数据帧见表 15

表 15 循环周期读指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
40H/4BH	00H	18H	05H	00H	00H	00H	00H

编码器成功接收到指令后响应桢见表 16

表 16 循环周期读指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
4BH	00H	18H	05H	XXH	YYH	00H	00H

1.6.5 同步模式指令

写命令

同步模式指令用于设置编码器在接收到多少次同步帧后，向总线返回一帧角度数据。网络上其他设备对编码器更改同步模式指令的数据帧见表 17，XX 值为同步帧发送的次数，例如 XX=03H 时表示编码器每接收到 3 次同步桢返回 1 桢角度数据。默认情况下，编码器每接收到 1 次同步桢返回 1 桢角度数据。同步桢的 CAN 桢 ID 为 80H，数据长度为 1 个字节，值为 00H。

表 17 写同步模式指令的数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H/2FH	01H	18H	02H	XXH	00H	00H	00H

编码器成功接收到指令后响应桢见表 18

表 18 写同步模式指令返回的数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
60H	01H	18H	02H	00H	00H	00H	00H

读命令

网络上其他设备读取编码器同步模式参数的数据帧见表 19

表 19 读同步模式指令的数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
40H/4FH	01H	18H	02H	00H	00H	00H	00H

编码器成功接收到指令后响应桢见表 20

表 20 读同步模式指令返回的数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
4FH	01H	18H	02H	XXH	00H	00H	00H

1.6.6 位置预设指令

写命令

网络上其他设备更改编码器当前角度值指令的数据帧见表 21，该指令可将编码器当前位置设置为 DDCCBBAAH，例如：AA=12H，BB=5FH，CC=01H，DD=00H，则表示把编码器当前位置设置为 0x015f12。此处要注意编码器的分辨率，例如编码器多圈分辨率为 12，单圈分辨率为 13 时，则当前位置最大只能设置为 0x01FFFFFF（DD 的值最大为 01H）。

表 21 位置预设写指令数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H	03H	21H	00H	AAH	BBH	CCH	DDH

编码器成功接收到指令后响应桢见表 22

表 22 位置预设写指令返回数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
60H	03H	21H	00H	00H	00H	00H	00H

读命令

网络上其他设备读取编码器前一次位置预设的数据帧见表 23

表 23 位置预设读指令数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
40H/4BH	03H	21H	00H	00H	00H	00H	00H

编码器成功接收到指令后响应桢见表 24，响应帧中的数据 1 根据编码器的分辨率（多圈分辨率+单圈分辨率）不同，有所变化。若分辨率≤8 位，AA 为 4FH；若分辨率≤16 位，AA 为 4BH；若分辨率≤24 位，AA 为 47H；若分辨率≤32 位，AA 为 43H。

表 24 位置预设读指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
AAH	03H	21H	00H	AAH	BBH	CCH	DDH

1.6.7 波特率指令

写命令

网络上其他设备对编码器更改波特率指令的数据帧见表 26，XX 的范围为 00H~07H，XX 与波特率对应关系见表 25。波特率更改成功后不会立即生效，需再发送保存命令后才能生效。

表 25 XX 与波特率对应关系表

XX	波特率（bps）
0	20K
1	50K
2	100K
3	125K
4	250K
5	500K
6	800K
7	1000K

表 26 波特率写指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H/2FH	01H	30H	00H	XXH	00H	00H	00H

编码器成功接收到指令后响应帧见表 27

表 27 波特率写指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
60H	01H	30H	00H	00H	00H	00H	00H

读命令

网络上其他设备读取编码器波特率参数的数据帧见表 28

表 28 波特率读指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
40H/4FH	01H	30H	00H	00H	00H	00H	00H

编码器成功接收到指令后响应帧见表 6.26

表 29 波特率读指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
4FH	01H	30H	00H	XXH	00H	00H	00H

1.6.8 CAN 总线终端电阻指令

写命令

网络上其他设备对编码器终端电阻指令的数据帧见表 30，其中 XX=00H 时，编码器无终端电阻；XX=01H 时，编码器有终端电阻。终端电阻大小为 120 欧姆。

表 30 终端电阻写指令数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H/2FH	02H	30H	00H	XXH	00H	00H	00H

编码器成功接收到指令后响应帧见表 31

表 31 终端电阻写指令返回数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
60H	02H	30H	00H	00H	00H	00H	00H

读命令

网络上其他设备读取编码器终端电阻的数据帧见表 32

表 32 终端电阻读指令数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
40H/4FH	02H	30H	00H	00H	00H	00H	00H

编码器成功接收到指令后响应桢见表 33

表 33 终端电阻读指令返回数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
4FH	02H	30H	00H	XXH	00H	00H	00H

1.6.9 心跳报文发送周期指令

写命令

网络上其他设备对编码器更改心跳报文发送周期指令的数据帧见表 34，心跳报文数据发送周期=YYXXH 毫秒，范围为 1-65535 毫秒。当 XX 和 YY 都为 00H 时，编码器关闭心跳报文功能，不再向总线上发送心跳报文。

表 34 心跳报文发送周期写指令数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H/2BH	17H	10H	00H	XXH	YYH	00H	00H

编码器成功接收到指令后响应桢见表 35

表 35 心跳报文发送周期写指令返回数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
60H	17H	10H	00H	00H	00H	00H	00H

读命令

网络上其他设备读取编码器发送周期参数的数据帧见表 36

表 36 心跳报文发送周期读指令数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
40H/4BH	17H	10H	00H	00H	00H	00H	00H

编码器成功接收到指令后响应桢见表 37

表 37 心跳报文发送周期读指令返回数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
4BH	17H	10H	00H	XXH	YYH	00H	00H

1.6.10 保存指令

保存命令用于将修改的参数保存在编码器中。要想使更改的参数在断电再重新上电后仍然有效，则必须发送保存指令。保存指令无返回值且不可读。

编码器的保存指令有两种。一种是保存不复位指令，另一种为保存并复位指令。编码器成功接收到保存不复位指令后，将修改的参数保存在编码器的内部铁电存储器中，仍按接收到保存指令之前的状态运行，只有发生复位或者重新上电后才会加载新的参数，保存不复位指令数据帧见表 38；编码器成功接收到保存并复位指令后，将修改的参数保存在编码器的内部铁电存储器中，并立即复位，加载新的参数，保存并复位指令数据帧见表 39

表 38 保存不复位数据桢格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H	10H	10H	01H	73H	61H	76H	65H

表 39 保存并复位数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H	00H	23H	00H	55H	AAH	AAH	55H

1.6.11 恢复出厂设置

恢复出厂设置指令能够将编码器的参数恢复成编码器出厂时的初始参数。恢复出厂设置指令的数据帧见表 40

表 40 恢复出场设置数据帧格式

数据 1	数据 2	数据 3	数据 4	数据 5	数据 6	数据 7	数据 8
22H	11H	10H	01H	6CH	6FH	61H	64H

基于 STM32 CAN 时钟 36MHz 的内部时序参数配置参考:

SJW	BS1	BS2	Prescaler	
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	6	// 1M
CAN_SJW_1tq	CAN_BS1_4tq	CAN_BS2_3tq	5	// 900K
CAN_SJW_1tq	CAN_BS1_5tq	CAN_BS2_3tq	5	// 800K
CAN_SJW_1tq	CAN_BS1_6tq	CAN_BS2_3tq	6	// 600K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	12	// 500K
CAN_SJW_1tq	CAN_BS1_5tq	CAN_BS2_3tq	10	// 400K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	20	// 300K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	24	// 250K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	30	// 200K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	40	// 150K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	48	// 125K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	60	// 100K
CAN_SJW_1tq	CAN_BS1_4tq	CAN_BS2_3tq	50	// 90K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	75	// 80K
CAN_SJW_1tq	CAN_BS1_6tq	CAN_BS2_3tq	60	// 60K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	120	// 50K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	150	// 40K
CAN_SJW_1tq	CAN_BS1_6tq	CAN_BS2_3tq	120	// 30K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	300	// 20K
CAN_SJW_1tq	CAN_BS1_3tq	CAN_BS2_2tq	600	// 10K
CAN_SJW_2tq	CAN_BS1_6tq	CAN_BS2_4tq	600	// 5K
CAN_SJW_2tq	CAN_BS1_6tq	CAN_BS2_4tq	1000	// 3K
CAN_SJW_2tq	CAN_BS1_10tq	CAN_BS2_6tq	1000	// 2K

www.docin.com

订货选择信息：

订货代码	描述
EA28S06C	28mm外径,CAN2.0b总线接口，单圈
EA28S06CB	28mm外径,CAN2.0b总线接口,带电池多圈
EA28S06CN	28mm外径,CANopen总线接口，单圈
EA28S06CBN	28mm外径,CANopen总线接口,带电池多圈

联系方式:

苏州傲蓝电子科技有限公司
地址：上海浦东新区金皖路 399 号 5 楼 508 室
电话：021-50373236 / 13901668071
公司网址：www.bluencoder.com

所有尺寸均为毫米。此信息和图纸的目的旨在于一般性的介绍,请参阅“下载”部分技术图纸进行了解,傲蓝 Allblue 版权所有,对于技术性误差或疏漏,我们不承担责任,产品规格的变更恕不另行通知。