



深圳市正元电机有限公司

伺服驱动器 技术指南

发行日期: 2020 年 02 月 25 日

当前版本: V1.0.8

修订日期: 2020 年 05 月 25 日

深圳市正元电机有限公司..

深圳市龙岗区布吉街道甘坑社区同福路 9 号永平工业园 3 号厂房 6-8 楼

电话: + 86-755-83688818

电话: + 86-755-89585358

传真: (86) 755 89585354

电子邮件: info@assunmotor.com.cn

一、调试

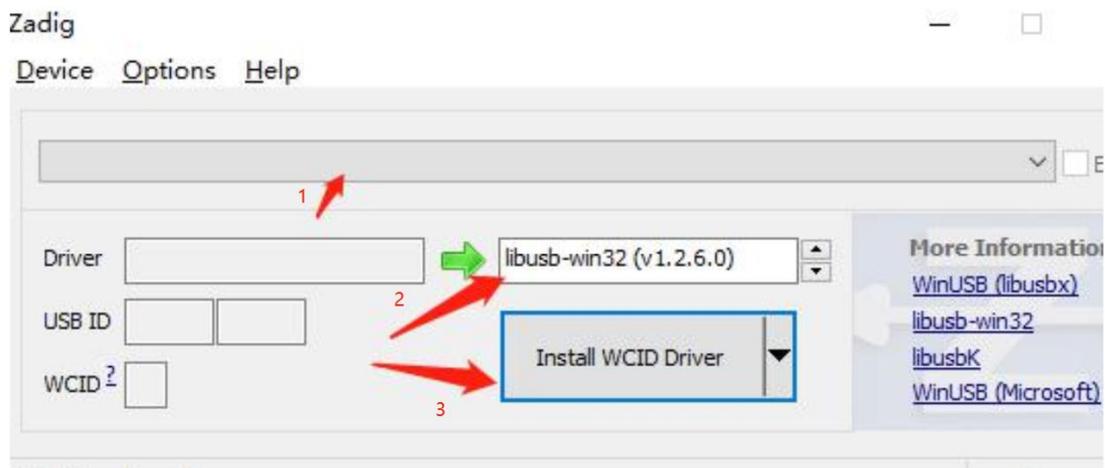
对于伺服驱动器，基本标配 USB 2.0 全速通讯接口；可以用此通讯对伺服驱动器进行基本调试，同时也可以进行三环控制。

1.1、软件驱动安装

调试软件使用 micro usb 通讯接口，因此用户需自行准备一条通讯线首次使用，需要安装对应的 usb 驱动程序，在安装目录找到《**zadig(首次 USB 驱动安装点我).exe**》

 zadig(首次USB驱动安装点我).exe	2012/11/15 7:48	应用程序	6,886 KB
--	-----------------	------	----------

双击运行，如下图



根据如图所示，在使用 usb 通讯线连接好驱动器(已上电)后，在第一个箭头的下拉框中找到 **AssunMotor FS USB** 的设备驱动之后，然后根据上图依次选择 **libusb-win32 (v1.2.6.0)**，再点击“**Install WCID Driver**”，耐心等待 1~2 分钟，之后会弹出安装成功提示，之后就可以打开调试软件（自动连接）。

安装成功后，可以在**设备管理器**查看到已挂载的驱动。如下图所示：



1.2、软件调试

在软件目录中，找到下图的应用程序 exe，双击运行

General_Servo_2.0.A.exe	2020/3/2 14:37	应用程序	5,71
-------------------------	----------------	------	------

二、硬件物理参数配置

通用总线型伺服驱动器：RS422、RS485、CAN

通用传统型伺服驱动器：2 通道模拟量输入接口、1 组脉冲输入接口

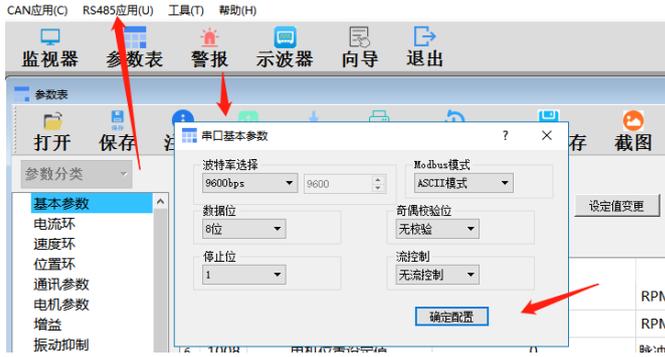
EtherCAT 总线型伺服驱动器：EtherCAT

2.1、串口 RS422/RS485

默认通讯参数

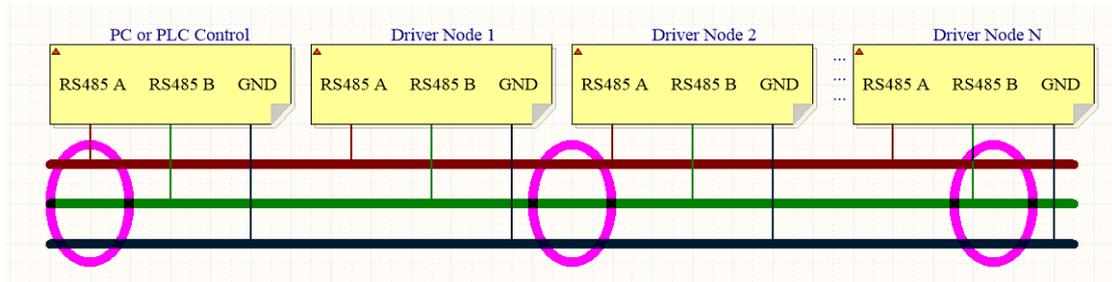
参数：57600bps、8 个数据位、1 个停止位、无校验、无流控制，Modbus 模式：RTU 模式，节点 ID：0x08。

可以在调试软件进行配置：



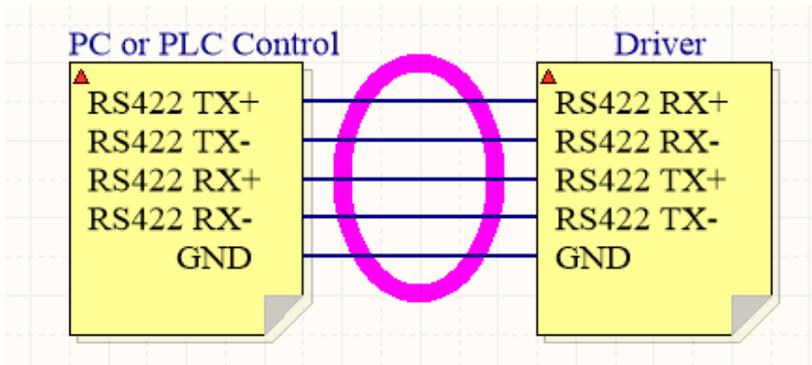
注意：RS422 和 RS485 公用一套串口基本参数，意味着二者的通讯参数是相同的。

RS485 的参考接线方式：



建议使用带屏蔽层的双绞线

RS422 的参考接线方式：



建议使用带屏蔽层的双绞线



2.2、CAN

默认参数：500kbps、节点 ID：0x08

Can 总线可以通过并联方式最高支持 127 个从机联机

目前仅支持：CIA301 通讯 + DS402 应用协议

注意：硬件没有焊接 120 欧姆终端匹配电阻

当然，可以在调试软件进行配置：

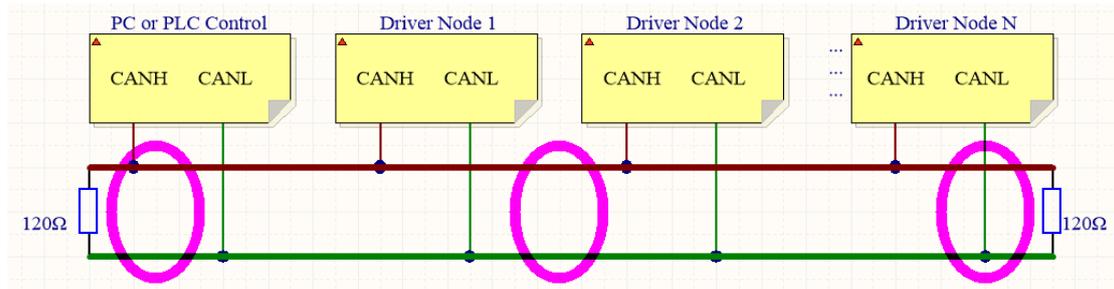


注意:如果没有特殊说明，默认不激活 CAN 总线应用，必须通过调试软件配置

波特率与通信长度和通信介质的关系，请参见下表：

波特率	最大传距离	2000h_03h(Cia301)
1M	40m	0
800k	80	1
500k	130m	2
250k	270m	3
125k	530m	4
100k	800	5
50k	1300m	6
20k	3300m	7

CAN 总线的参考接线方式：



建议使用带屏蔽层的双绞线



2.3、模拟量输入、方向脉冲

延长线接线建议：在使用的某路模拟量输入，需要用带屏蔽的双绞线做延长媒介

传统型伺服驱动器的硬件接线需参考选型手册（有详细说明，这里不再描述）

延长线接线建议：在使用的某路方向+脉冲，需要用带屏蔽的双绞线做延长媒介

2.4、电机接线

通常情况下，出厂的伺服驱动器会把电机参数匹配好，并测试完成，但由于伺服驱动器到电机的延长线一般小于1米，如果客户需要增加延长线，建议使用带屏蔽的绝缘双绞线



对于电机相线，建议根据电机功率的范围尽量合理使用直径大的双绞线。

三、三环模式应用

通常情况下，每一个环的绝大多数应用均可以在调试软件---> **配置向导** 完成。

3.1、RS422/RS485 通讯控制

3.1.1、配置工作模式

首次使用该通讯方式进行三环应用控制，需要配置部分参数

- (1)、确保该参数的值不为 3 (即不能在激活 CAN402 应用下修改)

通过选择上方的项目，与右下方的子项目来显示相关参数；
 点击每一行都会在下方显示厂家注释，可以通过点击上方“注释”
 或者点击右下角的“添加注释”可以为该行添加自定义注释；
 双击参数值单元格，可以修改数值，修改完后，
 按回车或点击“设定值变更按钮”可修改完成

地址	参数名	值	单位
1 1000	启动方式	1	(0----10)
2 1001	电机电流给定	0	0.1A(-32768----32768)
3 1002	电机速度给定	0	rpm(0-3147)

*启动方式
 0: DI控制(使能由数字IO输入控制)
 1: 停止 (自定义Can、串口RS485 关使能)
 3: 开启CANOPEN模式控制 (如需激活CAN402功能，必须写3，仅供CAN402使用，开关使能由“控制字(0x6040_00)”决定)
 10: 运行 (自定义Can、串口RS485 开使能)

为了方便修改，可以通过“快捷指令 1”进行修改，根据以下注释写不同的值即可实现切换 (支持实时切换工作模式)

参数名称	地址(H)	数据类型
快捷指令 1	1024	UINT16

参数描述(部分):

每次写入均自动清零

////////////////////////////////////

(有时候, 虽然切换到指定了工作模式, 但是无法在调试软件上去控制, 因为一些内部参数需要切换过来, 比如电流给定选择、限制选择、速度给定选择、限制选择、位置给定选择等参数)

以下是配置 USB/串口 485、422 模式下控制电机每一个环路的便捷切换方式

(必须在指定的工作模式下, 写入对应模式的值, 否则无效)

(CAN402 应用下, 激活后 (启动方式写 3), 控制器端只需要对索引 (6060_00) 写入对应模式的值即可实现自动切换)

10: 绝对位置环便捷切换 (直接对地址电机位置设定值 (1008) 写入)

11: 绝对位置环便捷切换 (内部位置指令 POV 0-7 (10DA--10EA))

12: 相对位置环便捷切换 (内部位置指令 POV 0-7 (10DA--10EA))

13: 速度模式便捷切换 (速度设定值 (1002))

14: 电流(力矩)模式便捷切换 (电流设定值 (1001))

15: 工厂测试模式电流环 (内部) 便捷切换 (电流设定值 (1001))

16: 工厂电角度对零模式 (内部) 便捷切换 (电流设定值 (1001))

注意:该配置仅需要配置一次即可, 然后进行“断电保存”即可。

3.1.2、控制参数简要介绍

点击每一行都会在下方显示厂家注释, 可以通过点击上方“注释”或者点击右上角的“添加注释”可以为该行添加自定义注释; 双击参数值单元格, 可以修改数值, 修改完后, 按回车或点击“设定值变更按钮”可修改完成

地址	参数名	值	单位
1 1000	启动方式	1	(0----10)
2 1001	电机电流给定值	0	0.1A(-32768-----)
3 1002	电机速度设定值	0	rpm或0.1rpm(-2147
4 1004	速度环控制加速度	2000	RPM/分/秒(0----429
5 1006	速度环控制减速度	5000	RPM/分/秒(0----429
6 1008	电机位置设定值	0	脉冲个数(-21474836
7 100A	工作模式	0001	H(0000----00
8 100B	命令0	0000	H(0000----FF
9 100C	命令1	0000	H(0000----FF

如上图, 为常用的应用控制参数 (可以在调试软件参数表查看详细注释)

(1)、启动方式

参数名称	地址(H)	数据类型
启动方式	1000	UINT16

参数描述(部分):

***启动方式**

0: DI 控制(使能由数字 IO 输入控制)

1: 停止 (自定义 Can、串口 RS485 关使能)

3:开启 CANOPEN 模式控制 (如需激活 CAN402 功能, 必须写 3, 仅供 CAN402 使用, 开关使能由“控制字(0x6040_00)”决定)

10: 运行 (自定义 Can、串口 RS485 开使能)

对于串口应用, 如果关使能写 1 即可实现, 如果**开**使能写 10 即可实现

(2)、电机电流给定值

参数名称	地址(H)	数据类型
电机电流给定值	1001	INT16

参数描述(部分): (单位 0.1A)

该参数的值仅仅在电流环模式或者电流限制的情况下才生效

地址	参数名	值	单位
1 1062	电流给定方式选择	2	(0---65535)
2 1063	电流 电流给定方式选择	0	(-3.5---3.5)
3 1064	电流环给定滤波常数	0	ms(0---65535)
4 1065	电流环限制选择	000F	H(0000---FF)

参数名称	地址(H)	数据类型
电流给定方式选择	1062	UINT16

写 2 表示使用 USB/RS485/RS422 给定 (写入其他值, 则无效)

参数描述(部分):

***电流环给定**

0:CAN402PT 模式给定

1:用户自定义给定

2:USB/RS485/RS422 给定

3:模拟量 0 通道给定

4:自定义 CAN

15:最大电流

其他值:强制保留为 0

参数名称	地址(H)	数据类型
电流环限制选择	1065	UINT16

表示给定的电流受到此项的限制

*Bit0~Bit3:

0:CAN402PT 模式给定

1:用户自定义给定

2:USB/RS485/RS422 给定

3:模拟量 0 通道给定

4:自定义 CAN

15:最大电流

其他值:强制保留为 0

(3)、电机速度设定值

参数名称	地址(H)	数据类型
电机速度设定值	1002	INT32

参数描述(部分):

单位 (rpm 或 0.1rpm 单位由参数 108C bit4 选定)

该参数受到以下 2 个参数的影响

地址	参数名	值	单位
1 108C	速度给定方式选择	0002	H(0000----FF)
2 108D	转速限制	000F	H(0000----FF)

(4)、加速度、减速度

地址	参数名	值	单位
1 1000	启动方式	0	(0----10)
2 1001	电机电流给定值	0	0.1A(-32768----32768)
3 1002	电机速度设定值	0	rpm或0.1rpm(-2147483648~2147483648)
4 1004	速度环控制加速度	50000	RPM/秒(0----4294967296)
5 1006	速度环控制减速度	50000	RPM/秒(0----4294967296)
6 1008	电机位置给定值	0	脉冲个数(-2147483648~2147483648)

参数名称	地址(H)	数据类型
速度环控制加速度	1004	UINT32
速度环控制减速度	1006	UINT32

这是常用的加、减速度，这个参数仅仅在“运动曲线类型”为 2 的情况下生效

速度环

地址	参数名	值	单位
4 108F	速度反馈平滑滤波时间常数	0	0.1
5 1090	S形平滑曲线中的加速平滑...	0	n
6 1091	S形平滑曲线中的减速平滑...	0	n
7 1092	运动曲线类型	2	
8 1093	速度设定比例	1	
9 1094	速度反馈比例	1	
10 1095	保留	0	(-3)

自定义注释

*运动曲线类型
0: CAN402梯形曲线;
1: S型曲线(仅限内部使用);
2: 常用梯形曲线

(5)、电机位置设定值

双击参数值单元格，可以修改数值；修改完后，按回车或点击“设定值变更按钮可修改完成

地址	参数名	值	单位
4 1004	速度环控制加速度	50000	RPM/秒(0----4294
5 1006	速度环控制减速度	50000	RPM/秒(0----4294
6 1008	电机位置设定值	0	脉冲个数(-21474836

单位：脉冲个数

参数名称	地址(H)	数据类型
电机位置设定值	1008	INT32

该参数受“位置环给定方式选择”影响

双击参数值单元格，可以修改数值；修改完后，按回车或点击“设定值变更按钮可修改完成

地址	参数名	值	单位
1 10D0	位置环给定方式选择	2	(0
2 10D1	电子齿轮分子0	1	(1
3 10D2	电子齿轮分子1	1	(1
4 10D3	电子齿轮分子2	1	(1
5 10D4	电子齿轮分母	1	(0
6 10D5	位置到达检测时间	1000	ms(
7 10D6	位置到达误差范围	10	(0----

自定义注释

*位置环给定
0: A/B脉冲 (增量式位置环)
1: 方向+脉冲 (增量式位置环)
2: USB/RS485/RS422绝对给定
3: POSx设定 (增量式位置环)
4: POSx设定 (绝对位置环)

(6)、工作模式

或者点击右上角的“添加注释”可以为该行添加自定义注释，
双击参数值单元格，可以修改数值，修改完后，
按回车或点击“设定值变更按钮”可修改完成

设定值变更

地址	参数名	值	单位
4 1004	速度环控制加速度	50000	RPM/秒(0----4294967296)
5 1006	速度环控制减速度	50000	RPM/秒(0----4294967296)
6 1008	电机位置设定值	0	脉冲个数(-2147483648~2147483647)
7 100A	工作模式	0000	H(0000----65535)
8 100B	命令0	0000	H(0000----65535)
9 100C	命令1	0000	H(0000----65535)
10 1024	快捷指令1	0	(0----65535)

只读参数

设置默认参数表

*Bit0~Bit3 :
0: 位置环;
1: 速度环;
2: 电流环;
6: 测试模式电流环;
7: 电机零位校对;

自定义注释

添加注释 添加完成 清空当前文本

参数名称	地址(H)	数据类型
工作模式	100A	UINT16

参数描述(部分):

*Bit0~Bit3 :

0: 位置环;

1: 速度环;

2: 电流环;

6: 测试模式电流环(内部调试)

7: 电机零位校对(内部调试)

(7)、命令 0

参数名称	地址(H)	数据类型
命令 0	100B	UINT16

参数描述(部分):

- Bit0 1: 伺服启动 (SON)
- Bit1 1: 位置环脉冲误差清除 (CCLR)
- Bit2 1: 零速箝位 (ZCLAMP)
- Bit3 1: 位置命令触发 (CTRG)
- Bit4 1: 转矩限制 (TRQLM)
- Bit5 1: 速度限制 (SPDLM)
- Bit6 1: 位置命令 0 (POS0)
- Bit7 1: 位置命令 1 (POS1)
- Bit8 1: 位置命令 2 (POS2)
- Bit9 1: 速度命令 0 (SPD0)
- Bit10 1: 速度命令 1 (SPD1)
- Bit11 1: 转矩命令 0 (TCM0)
- Bit12 1: 转矩命令 1 (TCM1)
- Bit13 1: 模式切换命令 (S_P)
- Bit14 1: 模式切换命令 (S_T)
- Bit15 1: 模式切换命令 (T_P)

(8)、命令 1

参数名称	地址(H)	数据类型
命令 1	100C	UINT16

参数描述(部分):

- Bit0 1: 急停 (EMGS)
- Bit1 1: 反转禁止极限 (CCWL)
- Bit2 1: 正向禁止极限 (CWL)
- Bit3 1: 正向点动 (JOGD)
- Bit4 1: 反向点动 (JOGU)
- Bit5 1: 复归原点 (ORGP)
- Bit6 1: 原点搜寻指令 (SHOM)
- Bit7 1: 脉冲输入禁止 (INHP)
- Bit8 1: 异常重置 (ARST)
- Bit9 1: 速度环增益切换 (SPEEDKP1)
- Bit10 1: 位置环增益切换 (POSKP1)
- Bit11 1: 输入反向命令 (CMDINV)
- Bit12 1: 电子齿轮分子 0 (GNUM0)
- Bit13 1: 电子齿轮分子 1 (GNUM1)
- Bit14 1: 通讯协议 (X14)
- Bit15 1: 曲线记录 (GRAPH)

(9)、快捷指令 1

参数名称	地址(H)	数据类型
快捷指令 1	100D	UINT16

参数描述(部分):

每次写入均自动清零

- 1: 位置误差清除 (位置反馈、位置给定均不清零)
- 2: 位置误差清除 (位置反馈、位置给定均清零)
- 3: 位置反馈清零
- 4、异常重置 (复位故障) (Can402 模式下不能使用)

////////////////////////////////////
 (有时候, 虽然切换到指定了工作模式, 但是无法在调试软件上去控制, 因为一些内部参数需要切换过来, 比如电流给定选择、限制选择、速度给定选择、限制选择、位置给定选择等参数)

以下是配置 USB/串口 485、422 模式下控制电机每一个环路的便捷切换方式
 (必须在指定的工作模式下, 写入对应模式的值, 否则无效)
 (CAN402 应用下, 激活后 (启动方式写 3), 控制器端只需要对索引 (6060_00) 写入对应模式的值即可实现自动切换)

- 10: 绝对位置环便捷切换 (直接对地址电机位置设定值 (1008) 写入)
- 11: 绝对位置环便捷切换 (内部位置指令 POV 0-7 (10DA--10EA))
- 12: 相对位置环便捷切换 (内部位置指令 POV 0-7 (10DA--10EA))
- 13: 速度模式便捷切换 (速度设定值 (1002))
- 14: 电流(力矩)模式便捷切换 (电流设定值 (1001))
- 15: 工厂测试模式电流环 (内部) 便捷切换 (电流设定值 (1001))
- 16: 工厂电角度对零模式 (内部) 便捷切换 (电流设定值 (1001))

3.1.3、MODBUS RTU 协议介绍

3.1.3.1、数据帧格式

在通讯线上传输的数据帧分为下传帧和返回帧两种。

- 1) 下传帧: 是由上位机向下位机发送的数据帧。

Address	Function	Data	
8-Bits	8-Bits	N x 8-Bits	

图 2 下传帧格式

2) 返回帧：是由下位机向上位机发送的数据帧。

Address	Function	Data
8-Bits	8-Bits	N x 8-Bits

图 3 返回帧格式

注意：该模式为大端格式，但对于 32 位的数据，它的低 16 位在前，高 16 位在后，但是低 16 位和高 16 位的 2 个字节也是大端格式。

RTU 模式下的传输前与传输完成后，至少需有 1ms 的静止时段。

3.1.3.2、CRC 校验说明

CRC 域是两个字节，包含一 16 位的二进制值。它由传输设备计算后加入到消息中。接收设备重新计算收到消息的 CRC，并与接收到的 CRC 域中的值比较，如果两值不同，则有误。CRC 是先调入一值是全“1”的 16 位寄存器，然后调用一过程将消息中连续的 8 位字节和当前寄存器中的值进行处理。仅每个字符中的 8Bit 数据对 CRC 有效，起始位和停止位以及奇偶校验位均无效。

CRC 产生过程中，每个 8 位字符都单独和寄存器内容相异或（XOR），结果向最低有效位方向移动，最高有效位以 0 填充。LSB 被提取出来检测，如果 LSB 为 1，寄存器单独和预置的值换一下，如果 LSB 为 0，则不进行。整个过程要重复 8 次。在最后一位（第 8 位）完成后，下一个 8 位字节又单独和寄存器的当前值相异或（XOR）。最终寄存器中的值，是消息中所有的字节都执行之后的 CRC 值。

参考校验源码：

```

unsigned short GET_RS485_MODBUS_RTU_CRC(unsigned char* data ,
unsigned char length)
{
    unsigned short j;
    unsigned short reg_crc=0xffff;
    while(length--)
    {
        reg_crc^=*data ++;
        for(j=0;j<8;j++)
        {
            if(reg_crc&0x01)
            {
                reg_crc=(reg_crc>>1)^0xa001;
            }
            else
            {
                reg_crc=reg_crc>>1;
            }
        }
    }
}

```

```

    }
}

return reg_crc;
}

```

3.1.3.3、通讯描述

(1)、工作过程:

上位机主动发送“下传帧”访问下位机，下位机收到符合自身地址且合法“下传帧”后立即上传“返回帧”，经上位机确认“返回帧”合法后，本轮通讯结束。但如果下传帧携带的地址为广播地址（见 2.2 的地址定义）时，所有从机都可接收到，但不发回返回帧。

(2)、地址定义

下位机地址为一个 8bit 的字节，地址范围从 0~250，地址 0 和 255 为广播地址，251~254 保留。下位机只有收到与自身地址相同的下传帧才作出反应。广播地址可以使所有下位机都作出反应，但不发回返回帧。

只有当接收的帧地址与此下位机地址码相符时才允许接受(如接收的帧地址为 0，则表示该帧为广播帧，此时不论本机地址是多少均可接收该广播帧)

(3)、目前支持的功能代码定义:

0x03	读取寄存器
0x06	写入单笔寄存器
0x10	写入多笔寄存器

3.1.3.4、读寄存器：命令码 03

在一个远程设备中，使用该功能码读取保持寄存器连续块的内容。请求 PDU 说明了起始寄存器

地址和寄存器数量。从零开始寻址寄存器。因此，寻址寄存器 1-16 为 0-15。

将响应报文中的寄存器数据分成每个寄存器有两字节，在每个字节中直接地调整二进制内容。对于每个寄存器，第一个字节包括高位比特，并且第二个字节包括低位比特

请求

功能码	1 个字节	0x03
起始地址	2 个字节	0x0000 至 0xFFFF
寄存器数量	2 个字节	1 至 125 (0x7D)

响应

功能码	1 个字节	0x03
字节数	1 个字节	2×N*
寄存器值	N*×2 个字节	

*N=寄存器的数量

错误

差错码	1 个字节	0x83
异常码	1 个字节	01 或 02 或 03 或 04

1) 下传帧:

地址	功能码	数据首地址高 8 位	数据首地址低 8 位	个数 N 高 8 位	个数 N 低 8 位	CRC 校验低 8 位	CRC 校验高 8 位
1 字节	(03) 1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	1 字节

2) 返回帧:

地址	功能码	数据数 2N	数据	LRC 校验低 8 位	LRC 校验高 8 位
1 字节	(03) 1 字节	1 字节	2*N 字节	1 字节	1 字节

注: RTU 模式下的传输前与传输完成后, 至少需有 1ms 的静止时段。

应用举例: 向 01H 地址驱动器在寄存器地址 0x1000 处读取 2 个数据

主站命令信息：

从站节点地址	01H
功能码	03H
起始数据位置	10H (高字节)
	00H (低字节)
数据个数	00H (高字节)
	02H (低字节)
CRC 校验低 8 位	(低字节)
CRC 校验高 8 位	CBH (高字节)

从站响应消息：

从站节点地址	01H
功能码	03H
数据数 (以字节计算)	04H
起始数据地址 1000H 的内容	00H
	00H
起始数据地址 1000H 的内容	FFH
	F1H
CRC 校验低 8 位	7AH
CRC 校验高 8 位	47H

注：RTU 模式下的传输前与传输完成后，至少需有 1ms 的静止时段。

3.1.3.5、写单个寄存器：命令码 06

在一个远程设备中，使用该功能码写单个保持寄存器。

请求 PDU 说明了被写入寄存器的地址。从零开始寻址寄存器。因此，寻址寄存器 1 为 0。正常响应是请求的应答，在写入寄存器内容之后返回这个正常响应

请求

功能码	1 个字节	0x06
寄存器地址	2 个字节	0x0000 至 0xFFFF
寄存器值	2 个字节	0x0000 至 0xFFFF

响应

功能码	1 个字节	0x06
寄存器地址	2 个字节	0x0000 至 0xFFFF
寄存器值	2 个字节	0x0000 至 0xFFFF

错误

差错码	1 个字节	0x86
异常码	1 个字节	01 或 02 或 03 或 04

1) 下传帧:

地址	功能码	数据首地址高 8 位	数据首地址低 8 位	数据高 8 位	数据低 8 位	CRC 校验低 8 位	CRC 校验高 8 位
1 字节	(06) 1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	1 字节

2) 返回帧:

地址	功能码	数据首地址高 8 位	数据首地址低 8 位	数据高 8 位	数据低 8 位	CRC 校验低 8 位	CRC 校验高 8 位
地址	功能码	数据首地址高 8 位	数据首地址低 8 位	数据高 8 位	数据低 8 位	CRC 校验低 8 位	CRC 校验高 8 位

应用举例：向 01H 地址驱动器在寄存器地址 0x105A 处处写 000AH 值

主站命令信息：

从站节点地址	01H
功能码	06H
起始数据地址	10H (高字节)
	5AH (低字节)
数据内容	00H (高字节)
	0AH (低字节)
CRC 校验低 8 位	2DH (低字节)
CRC 校验高 8 位	1EH (高字节)

从站响应消息：

从站节点地址	01H
功能码	06H
起始数据地址	10H
	5AH
数据内容	00H
	0AH
CRC 校验低 8 位	2DH
CRC 校验高 8 位	1EH

注：RTU 模式下的传输前与传输完成后，至少需有 1ms 的静止时段。

3.1.3.6、写多个寄存器：命令码 10

在一个远程设备中，使用该功能码写连续寄存器块(1 至约 120 个寄存器)。在请求数据域中说明了请求写入的值。每个寄存器将数据分成两字节。正常响应返回能码、起始地址和被写入寄存器的数量

请求 PDU

功能码	1 个字节	0x10
起始地址	2 个字节	0x0000 至 0xFFFF
寄存器数量	2 个字节	0x0001 至 0x0078
字节数	1 个字节	2×N*
寄存器值	N×2 个字节	值

*N=寄存器数量

响应 PDU

功能码	1 个字节	0x10
起始地址	2 个字节	0x0000 至 0xFFFF
寄存器数量	2 个字节	1 至 123 (0x7B)

错误

差错码	1 个字节	0x90
异常码	1 个字节	01 或 02 或 03 或 04

应用举例：向 **01H** 地址驱动器在寄存器地址 **0x1156** 处写 **12345678H** 值，假设 **0x1156** 地址对应的是一个 **32 位** 的数据

主站命令信息：

从站节点地址	01H
功能码	10H
起始数据地址	11H (高字节)
	56H (低字节)
数据个数	00H (高字节)
	02H (低字节)
数据个数 (以字节数计算)	00H (高字节)
	04H (低字节)
数据的低 16 位	56H (高字节)
	78H (低字节)
数据的高 16 位	12H (高字节)
	34H (低字节)
CRC 校验低 8 位	2AH (低字节)

从站响应消息：

从站节点地址	01H
功能码	10H
起始数据地址	11H
	56H
数据个数	00H
	02H
CRC 校验低 8 位	A4H
CRC 校验高 8 位	E4H

CRC 校验高 8 位	OFH (高字节)
-------------	-----------

注：RTU 模式下的传输前与传输完成后，至少需有 1ms 的静止时段。

1) 下传帧：

地址	功能码	数据首地址高 8 位	数据首地址低 8 位	N 高 8 位	N 低 8 位	N*2 个数据	数据 1 高 8 位	数据 1 低 8 位	数据 N 高 8 位	数据 N 低 8 位	CRC 校验低 8 位	CRC 校验高 8 位
1 字节	(10)1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	N*2 个字节	1 字节	1 字节

2) 返回帧：

地址	功能码	数据首地址高 8 位	数据首地址低 8 位	N 高 8 位	N 低 8 位	CRC 校验低 8 位	CRC 校验高 8 位
1 字节	(10)1 字节	1 字节	1 字节	1 字节	1 字节	1 字节	1 字节

3.1.4、MODBUS ASCII 协议介绍

3.1.4.1、数据帧格式

在通讯线上传输的数据帧分为下传帧和返回帧两种。

1) 下传帧：是由上位机向下位机发送的数据帧。

起始符	地址	功能码	数据	LRC 校验	结束符
1 字节(:)	2 字节	2 字节	详见功能码	2 字节	2 字节(CR+LF)

图 2 下传帧格式

2) 返回帧：是由下位机向上位机发送的数据帧。

起始符	地址	功能码	数据	LRC 校验	结束符
1 字节(:)	2 字节	2 字节	详见功能码	2 字节	2 字节(CR+LF)

返回帧格式

注意：ASCII 模式出现的字符如果是字母，必须转换成大写字母。

该模式为大端格式，但对于 32 位的数据，它的低 16 位在前，高 16 位在后，但是低 16 位和高 16 位的 2 个字节也是大端格式。

3.1.4.2、LRC 校验

```
//功能：LRC 校验
unsigned int RS485_LRC(char*v, int n)//LRC 校验
{
    unsigned int  lrc;
    int temp;
    char strtemp[4];
    lrc=0;
    while(n--!=0)
    {
        strtemp[0]=*v;
        v++;
        strtemp[1]=*v;
        strtemp[2]=0;
        temp=HEXTOLONG(strtemp);

        lrc=(unsigned int ) temp+lrc;
        lrc=lrc&0x00ff;
        v++;
    }
    lrc=~lrc+1;
    lrc=lrc&0x00ff;
    return(lrc);
}
```

3.1.4.3、通讯应用说明

工作过程：

上位机主动发送“下传帧”访问下位机，下位机收到符合自身地址且合法“下传帧”后立即上传“返回帧”，经上位机确认“返回帧”合法后，本轮通讯结束。但如果下传帧携带的地址为广播地址（见 2.2 的地址定义）时，所有从机都可接收到，但不发回返回帧。

地址定义

下位机地址为一个 8bit 的字节，地址范围从 0~250，地址 0 和 255 为广播地址，251~254 保留。下位机只有收到与自身地址相同的下传帧才作出反应。广播地址可以使所有下位机都作出反应，但不发回返回帧。

只有当接收的帧地址与此下位机地址码相符时才允许接受（如接收的帧地址为 0，则表示该帧为广播帧，此时不论本机地址是多少均可接收该广播帧）。

目前支持以下功能码

功能码	说明
0x03	读取寄存器
0x06	写入单笔寄存器
0x10	写入多笔寄存器

3.1.4.4、功能码详细介绍

(1)、读寄存器：命令码 03

1) 下传帧：

起始符	地址	功能码	数据首地址	个数 N	LRC 校验	结束符
1 字节(:)	2 字节	(03) 2 字节	4 字节	4 字节	2 字节	2 字节 (CR+LF)

2) 返回帧：

起始符	地址	功能码	数据数 2N	数据	LRC 校验	结束符
1 字节(:)	2 字节	(03) 2 字节	2 字节	4*N 字节	2 字节	2 字节 (CR+LF)

(2)、写单个寄存器：命令码 06

1) 下传帧：

起始符	地址	功能码	数据首地址	数据	LRC 校验	结束符
1 字节(:)	2 字节	(06) 2 字节	4 字节	4 字节	2 字节	2 字节 (CR+LF)

2) 返回帧：

起始符	地址	功能码	数据首地址	数据	LRC 校验	结束符
1 字节(:)	2 字节	(06) 2 字节	4 字节	4 字节	2 字节	2 字节 (CR+LF)

(3)、写多个寄存器：命令码 10

1) 下传帧：

起始符	地址	功能码	数据首地址	数据数 N	数据字节数 N*4	数据	LRC 校验	结束符
1 字节(:)	2 字节	(10) 2 字节	4 字节	4 字节	2 字节	4*N 字节	2 字节	2 字节 (CR+LF)

2) 返回帧：

起始符	地址	功能码	数据首地址	数据数 N	LRC 校验	结束符
1 字节(:)	2 字节	(10) 2 字节	4 字节	4 字节	2 字节	2 字节

					节	(CR+LF)
--	--	--	--	--	---	---------

3.1.5、应用举例

3.1.5.1、位置环

位置环有多种控制方式：直接对位置给定寄存器写值的绝对位置给定，也有使用内部位置指令的 8 个缓冲区的绝对位置和相对位置的应用。

参数名称	地址(H)	数据类型
位置环给定方式选择	10D0	UINT16

部分描述：

位置环给定

0: A/B 脉冲（增量式位置环）

1: 方向+脉冲（增量式位置环）

2: USB/RS485/RS422 绝对给定

3: POSx 设定（增量式位置环）

4: POSx 设定（绝对位置环）

5: USB/RS485/RS422 增量式给定(需要通过命令 0 位置触发)

6: CAN402 PP CSP 模式给定

7:保留

8:保留

其他值:保留

3.1.5.1.1、USB/RS485/RS422 绝对给定

“位置给定方式选择”写 2 表示激活了该种给定

绝对位置给定：直接对下图所示的寄存器地址写值即可

双击参数值单元格，可以修改数值，修改完后，按回车或点击“设定值变更按钮”可修改完成

地址	参数名	值	单位
1 1000	启动方式	0	(0----10)
2 1001	电机电流给定值	0	0.1A(-32768----32768)
3 1002	电机速度设定值	0	rpm或0.1rpm(-2147483648~2147483648)
4 1004	速度环控制加速度	50000	RPM/秒(0----4294)
5 1006	速度环控制减速度	50000	RPM/秒(0----4294)
6 1008	电机位置设定值	0	脉冲个数(-2147483648~2147483648)
7 100A	工作模式	0000	H(0000----0000)

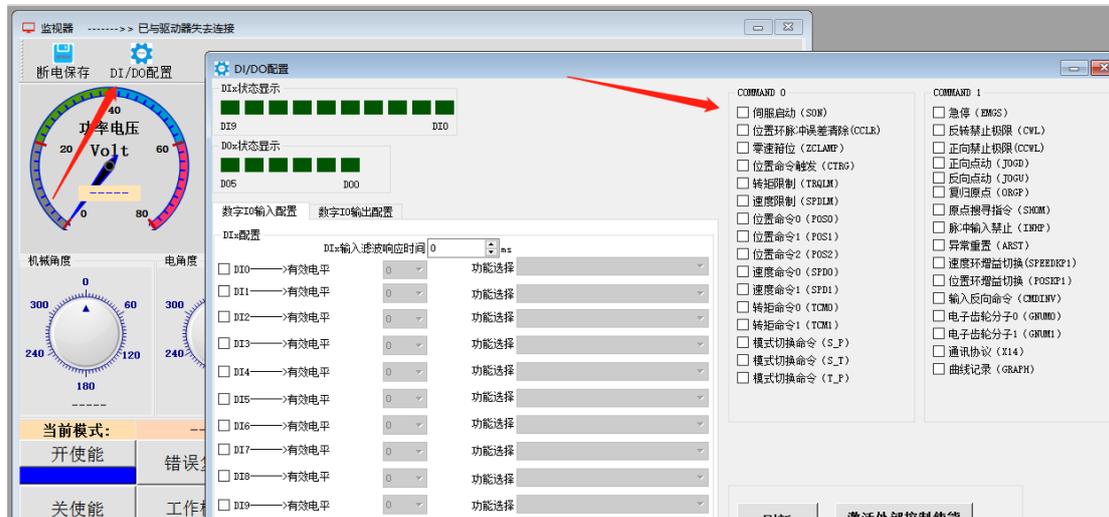
3.1.5.1.2、POSx 设定 (增量式位置环)

“位置给定方式选择”写 3 表示激活了该种给定

地址	参数名	值	单位
10	10DA 电机内部位置指令0	0	(-2147483648----21
11	10DC 电机内部位置指令1	0	(-2147483648----21
12	10DE 电机内部位置指令2	0	(-2147483648----21
13	10E0 电机内部位置指令3	0	(-2147483648----21
14	10E2 电机内部位置指令4	0	(-2147483648----21
15	10E4 电机内部位置指令5	0	(-2147483648----21
16	10E6 电机内部位置指令6	0	(-2147483648----21
17	10E8 电机内部位置指令7	0	(-2147483648----21

如何使触发位置指令 生效呢?

使用命令 0 触发 (地址 0x100B) , Uint16)
可以在下图所示的界面做测试



这个参数均为无符号 16 位，并且每一位均有其特殊功能，这里介绍常用功能，[详情参看调试软件-->“监视器”-->“DI/DO 配置”](#)，该功能参数每一位不用的时候需要清理，不能随意置 1。

命令 0:

Bit1 1:位置环脉冲误差清除 (CCLR)

Bit2 1:零速箝位 (ZCLAMP)

Bit3 1:位置命令触发 (CTRG)

Bit4 1:转矩限制 (TRQLM)

Bit5 1:速度限制 (SPDLM)

Bit6 1:位置命令 0 (POS0)

Bit7 1:位置命令 1 (POS1)

Bit8 1:位置命令 2 (POS2)

简要介绍:

(1)、Bit1 1:位置环脉冲误差清除 (CCLR) :

当在调试软件配置地址 0x10D9 脉冲清除模式写 0 (“参数表”--->“位置环”)，当该位为 1，驱动器内部将把位置反馈赋值给位置给定。

当在调试软件配置地址 0x10D9 脉冲清除模式写 1 (“参数表”---> “位置环”)，当该位实现一次上升沿触发时，驱动器内部将把回零偏移赋值给位置反馈。

当在调试软件配置地址 0x10D9 脉冲清除模式写 2 (“参数表”---> “位置环”)，当该位为 1，驱动器内部将把位置给定、位置反馈清零。

(2)、Bit2 1:零速箝位 (ZCLAMP)

当该位为 1 的时候，即使电机开使能，电机也将零速，不能转动

(3)、Bit3 1:位置命令触发 (CTRG)

当该位实现一次上升沿触发时，即 让该位先写 0，再写 1，将执行一次内部位置给定指令响应。

(4)、Bit6 1:位置命令 0 (POS0) 、Bit7 1:位置命令 1 (POS1) 、Bit8 1:位置命令 2 (POS2)

这 3 位 中 POS0 为 bit0 POS1 为 bit1 POS0=2 为 bit2 组合成 8 个内部位置指令选择

如果需要使用内部位置指令 0 则 POS2 为: 0; POS1 为: 0; POS0 为: 0;

如果需要使用内部位置指令 1 则 POS2 为: 0; POS1 为: 0; POS0 为: 1;

如果需要使用内部位置指令 2 则 POS2 为: 0; POS1 为: 1; POS0 为: 0;

如果需要使用内部位置指令 3 则 POS2 为: 0; POS1 为: 1; POS0 为: 1;

如果需要使用内部位置指令 4 则 POS2 为: 1; POS1 为: 0; POS0 为: 0;

如果需要使用内部位置指令 5 则 POS2 为: 1; POS1 为: 0; POS0 为: 1;

如果需要使用内部位置指令 6 则 POS2 为: 1; POS1 为: 1; POS0 为: 0;

如果需要使用内部位置指令 7 则 POS2 为: 1; POS1 为: 1; POS0 为: 1;

3.1.5.1.3、POSx 设定 (绝对式位置环)

“位置给定方式选择”写 4 表示激活了该种给定

具体应用操作方式与 3.1.5.1.2 相同

3.1.5.1.4、关于位置误差清除方式

(1)、使用命令 0 的 bit1 来实现

参数名称	地址(H)	数据类型
脉冲清除方式	10D9	INT16

部分描述:

脉冲清除模式

0:清除误差值(命令 0 的 bit1 为 1: 持续把位置反馈赋值给位置给定, 为 0 将停止)

1:绝对位置清除(命令 0 的 bit1 实现一次上升沿触发: 把回零偏移赋值给位置反馈)

2:位置给定、反馈均清零(命令 0 的 bit1 为 1: 位置反馈和位置给定将持续清零, 为 0 将停止)

3:命令终止

(2)、使用快捷指令 1 来实现

参数名称	地址(H)	数据类型
快捷指令 1	1024	UINT16

部分描述:

每次写入均自动清零

- 1: 位置误差清除 (位置反馈、位置给定均不清零)
- 2: 位置误差清除 (位置反馈、位置给定均清零)
- 3: 位置反馈清零

3.1.5.2、速度环

以 RTU 模式应用举例 (假设节点 ID 是 0x08)

(1)、关使能

2、点击每一行都会在下方面显示厂家注释，可以通过点击上方“注释”，或者点击右下角的“添加注释”可以为该行添加自定义注释；
3、双击参数值单元格，可以修改数值，修改完后，按回车或点击“设定值变更”按钮可修改完成

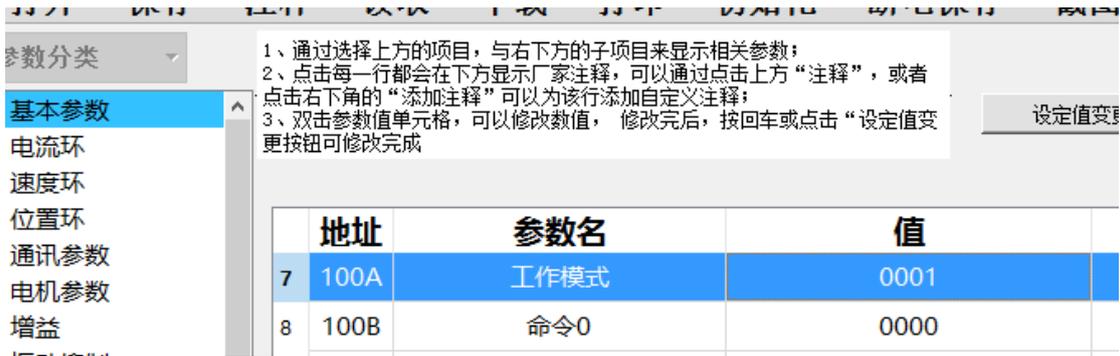
地址	参数名	值
1 1000	启动方式	1

注意：实际应用中，并不需要先关使能，再切换工作模式，这里只是举例。

控制端发送 (“06” 功能码) :

08 06 10 00 00 01 4c 53

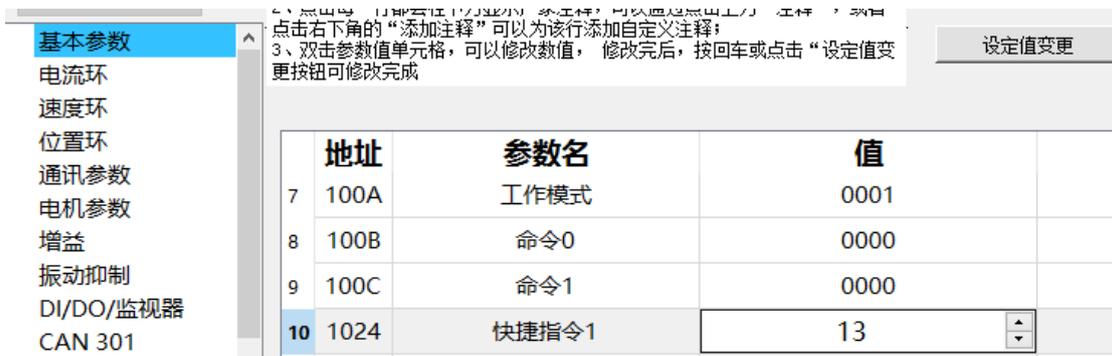
(2)、进入速度环模式



控制端发送（“06”功能码）：

08 06 10 0a 00 01 6c 51

（3）激活串口控制速度环接口



注意：该步骤写入完成后，驱动器将自动切换：

地址	名称	当前值（十进制）	描述
1065	电流环限制选择	15	不限制电流输出 （以最大电流输出）
108C	速度给定方式选择	2	2:USB/RS485/RS422 给定
108D	转速限制	15	不限制

以上参数均可以在调试软件里默认保存，因此，如果始终运行在速度环模式下，则不需要实时修改。

控制端发送（“06”功能码）：

08 06 10 24 00 0d 0c 5d

（4）、修改加速度(50000 RPM/Sec)

控制端发送（“10”功能码）：

08 10 10 04 00 02 04 c3 50 00 00 2d 55

(5)、修改减速度(50000 RPM/Sec)

控制端发送 (“10” 功能码) :

08 10 10 06 00 02 04 c3 50 00 00 ac 8c

(6)、开使能

控制端发送 (“06” 功能码) :

08 06 10 00 00 0a 0d 94

(7)、速度给定

给定 100 RPM: 正转

控制端发送 (“10” 功能码) :

08 10 10 02 00 02 04 00 64 00 00 d0 f5

给定 -100 RPM: 反转

控制端发送 (“10” 功能码) :

08 10 10 02 00 02 04 ff 9c ff ff 60 a0

注意: 每次开使能之前或者检测到有故障, 速度给定会自动清零、电流给定会自动清零、位置给定和位置给定的误差自动清除。

3.1.5.3、电流环

电流环工作模式, 也称为力矩环模式, 目前, 使用串口 RS422/485 控制驱动器, 电流控制单位: 0.1 安培。

以 RTU 模式应用举例 (假设节点 ID 是 0x08)

(1)、关使能

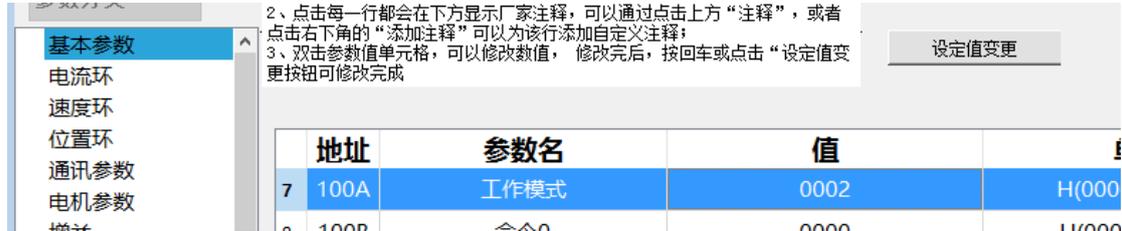
地址	参数名	值
1 1000	启动方式	1

注意: 实际应用中, 并不需要先关使能, 再切换工作模式, 这里只是举例。

控制端发送 (“06” 功能码) :

08 06 10 00 00 01 4c 53

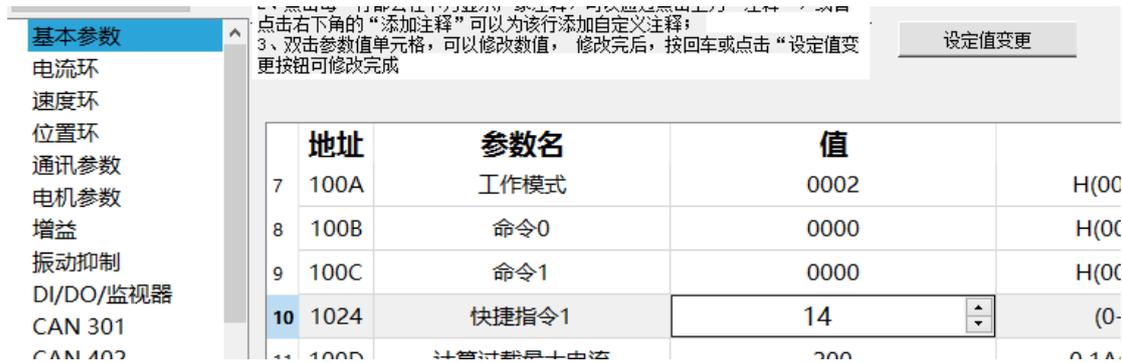
(2)、进入电流模式



控制端发送（“06”功能码）：

08 06 10 0a 00 02 2c 50

(3) 激活串口控制电流环接口



注意：该步骤写入完成后，驱动器将自动切换：

地址	名称	当前值（十进制）	描述
1062	电流给定方式选择	2	2:USB/RS485/RS422 给定
1065	电流环限制选择	15	不限制电流输出(以最大电流输出)
108C	速度给定方式选择	2	2:USB/RS485/RS422 给定
108D	转速限制	15	不限制

以上参数均可以在调试软件里默认保存，因此，如果始终运行在速度环模式下，则不需要实时修改。

控制端发送（“06”功能码）：

08 06 10 24 00 0e 4c 5c

(4)、修改加速度(50000 RPM/Sec)

控制端发送 (“10” 功能码) :

08 10 10 04 00 02 04 c3 50 00 00 2d 55

(5)、修改减速度(50000 RPM/Sec)

控制端发送 (“10” 功能码) :

08 10 10 06 00 02 04 c3 50 00 00 ac 8c

(6)、开使能

控制端发送 (“06” 功能码) :

08 06 10 00 00 0a 0d 94

(7)、电流给定

给定 1 安培：正向力矩

控制端发送 (“06” 功能码) :

08 06 10 01 00 0a 5c 54

给定 -1 安培：反向力矩

控制端发送 (“06” 功能码) :

08 06 10 01 ff f6 1d e5

注意：给定电流的力矩输出必须要大于负载的力矩，否则，电机无法转动。

注意：每次开使能之前或者检测到有故障，速度给定会自动清零、电流给定会自动清零、位置给定和位置给定的误差自动清除。

3.2、CANOPEN 通讯控制

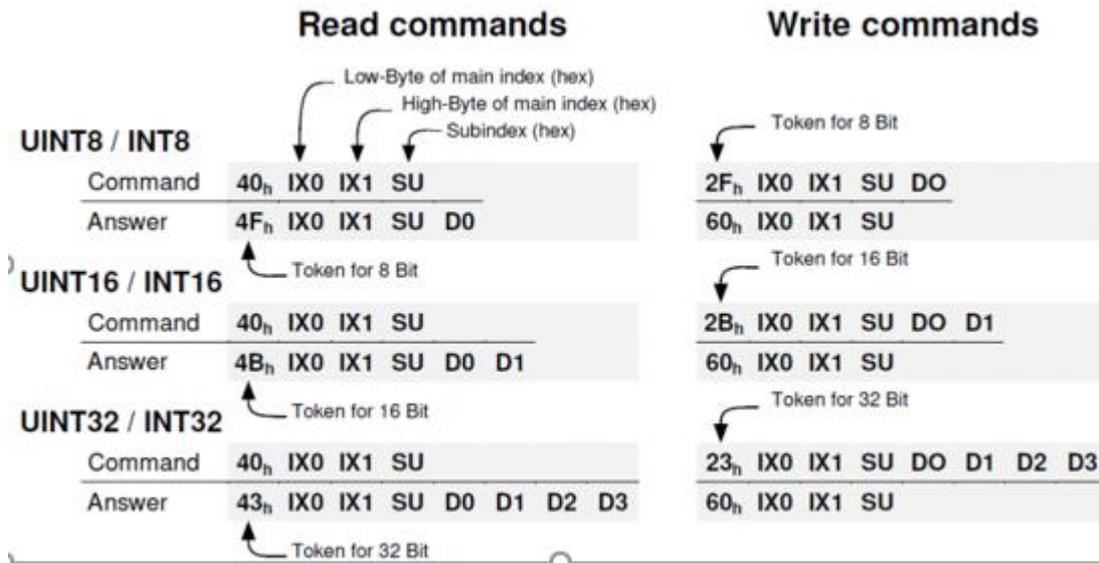
基本参数介绍：

目前支持标准 Cia301+DSP402 运动控制协议

节点 ID 范围是 1 - 127，驱动器的默认节点 ID 是 0x08。

Object	COD-Ids	Index of communication parameters in OD
NMT	0	-
SYNC	80h	1005h
Emergency	80h+ node ID	1014h
Tx-PDO1	180h+ node ID	1800h
Rx-PDO1	200h+ node ID	1400h
Tx-PDO2	280h+ node ID	1801h
Rx-PDO2	300h+ node ID	1401h
Tx-PDO3	380h+ node ID	1802h
Rx-PDO3	400h+ node ID	1402h
Tx-PDO4	480h+ node ID	1803h
Rx-PDO4	500h+ node ID	1403h
Tx-SDO	580h+ node ID	1200h
Rx-SDO	600h+ node ID	1200h
NMT Node Guarding	0x700 + node ID	

服务数据对象 (SDO) 通信协议
SDO 消息读/写格式, 如下图所示:



3.2.1、激活控制

地址	参数名	值	单位
1	1000	启动方式	3
2	1001	电机电流给定值	0
3	1002	电机速度设定值	0
4	1004	速度环控制加速度	50000
5	1006	速度环控制减速度	50000
6	1008	电机位置设定值	0
7	100A	工作模式	0000

对“启动方式”写 3，然后进行“断电保存”操作，重启后即可激活 CANOPEN 通讯，重启后即可按照标准 Can402 运动控制协议进行电机控制

3.2.2、使能控制

按照 Can402 协议，在无故障的情况下，开使能需要对控制字的低 4 位分别写 0, 6, 7, 0x0F，才能开使能成功，但是为了方便客户使用，客户可以对控制字的低 4 位由 0 写 0x0F，即可开使能成功。

注意：由于伺服驱动器在上电后会处于“电源欠压的故障状态”，因此首先需要进行故障复位操作（即控制字的 bit7 实现一次上升沿触发“故障复位”）

3.2.3、工作模式

伺服驱动器在 Can402 应用上支持多种工作模式：

工作模式 (6060_00)	模式名称
1	行规位置环 (PP)
3	行规速度环 (PV)
4	行规力矩环 (TQ)
6	回零模式 (HOME)
8	周期同步位置环 (CSP)

3.2.3.1、行规位置环 (PP) 应用举例 1

Object 0x6060: Modes of operation : 0x01

Object 0x6061: Modes of operation display: 0x01

Controlword(6040h) of the profiled position mode:

Bit	Function
0...3	Described in Device Control
4	Set new point
5	Change set immediately
6	Absolute/relative movement
7	Described in Device Control
8	Halt
9..12	Described in Device Control
13	Manufacturer Definition
14..15	Described in Device Control

Name	Value	Description
New set-point	0	不假定目标位置。
	1	假定目标位置。
Change set immediately	0	完成实际定位，然后开始下一个定位。
	1	中断实际定位并开始下一个定位。
abs\rel	0	目标位置是一个绝对值。
	1	目标位置是一个相对值。

Halt	0	执行定位。
	1	以 <i>profile acceleration</i> 停止轴。
Clear buffered position	0	不执行指令
	1	把位置给定和位置反馈都清零

Statusword(6041h) of the profiled position mode:

Bit	Function
0...9	Described in Device Control
10	Target reached
11	Described in Device Control
12	Set new point acknowledge
13	Following error
14	Clear buffered position acknowledge
12...15	Described in Device Control

Name	Value	Description
Target reached	0	Halt = 0: Target position not reached. Halt = 1: Axle decelerates.
	1	Halt = 0: Target position reached. Halt = 1: Velocity of axle is 0.
Set new point acknowledge	0	Trajectory generator has not assumed the positioning values (yet).
	1	Trajectory generator has assumed the positioning values.
Following error	0	
	1	Following error.
Clear buffered position acknowledge	0	Clear buffered position = 0
	1	Clear buffered position = 1

3. 2. 3. 1. 1、修改工作模式

(1) 写 Object 0x6060: Modes of operation , 修改为 profiled position(0x01) 。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	60	60	00	01	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	60	60	00	----	----	----	----

(2) 读取 Object 0x6061: Modes of operation display, 确保该参数的值是 0x01。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	40	61	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4F	61	60	00	01	----	----	----

3.2.3.1.2、开使能

(1) 复位错误

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	80	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(2)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	00	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(3)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	06	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(4)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7

608	2B	40	60	00	07	00	----	----
------------	----	----	----	----	----	----	------	------

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(5)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	0F	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

3. 2. 3. 1. 3、修改加速度和减速度

(1) 修改 profiled acceleration:10000 (RPM/(秒)) , 即每秒能够加速到 10000RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	83	60	00	27	10	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	83	60	00	----	----	----	----

(2) 修改 profiled deceleration: 10000 (RPM/(秒)) , 即每秒能够减速到 10000RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	84	60	00	27	10	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	84	60	00	----	----	----	----

3. 2. 3. 1. 4、位置控制

(1) 修改 Object 0x6081: Profile velocity:2000 (RPM) ,

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	81	60	00	D0	07	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	81	60	00	----	----	----	----

(2) 修改 Object 0x607A: Target position:512pulse

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	7A	60	00	00	02	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	7A	60	00	----	----	----	----

(3) Set new Point.

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	1F	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(4) 读取 Object 0x6041: Statusword bit10 位置到达。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	10	41	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4B	41	60	00	*	*	----	----

请等待 Statusword bit10 被置 1.

(5) Clear new Point.

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	0F	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

3.2.3.2、行规速度环(PV)应用举例 1

Object 0x6060: Modes of operation : 0x03

Object 0x6061: Modes of operation display: 0x03

Controlword(6040h) of the profiled velocity mode:

Bit	Function
0...3	Described in Device Control
7	Described in Device Control
8	Halt
9..12	Described in Device Control
13	Manufacturer Definition
14..15	Described in Device Control

Name	Value	Description
Halt	0	执行速度控制。
	1	以 <i>profile acceleration</i> 停止轴。
Clear buffered position	0	不执行指令
	1	把位置给定和位置反馈都清零

Statusword(6041h) of the profiled velocity mode:

Bit	Function
0...9	Described in Device Control
10	Target reached
11	Described in Device Control
13	Following error
14	Clear buffered position acknowledge
12...15	Described in Device Control

Name	Value	Description
Target reached	0	Halt = 0: Target velocity not reached. Halt = 1: Axle decelerates.
	1	Halt = 0: Target velocity reached. Halt = 1: Velocity of axle is 0.
Following error	0	
	1	Following error.
Clear buffered position acknowledge	0	Clear buffered position = 0
	1	Clear buffered position = 1

3.2.3.2.1、修改工作模式

(1) 写 Object 0x6060: Modes of operation , 修改为 profile velocity mode (0x03) 。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	60	60	00	03	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	60	60	00	----	----	----	----

(2) 读取 Object 0x6061: Modes of operation display, 确保该参数的值是 0x03。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	40	61	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4F	61	60	00	03	----	----	----

3.2.3.2.2、开使能

(1) 复位错误

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	80	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(2)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	00	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(3)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	06	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(4)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	07	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(5)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	0F	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

3.2.3.2.3、修改加速度和减速度

(1) 修改 profiled acceleration:10000 (RPM/(秒))，即每秒能够加速到 10000 RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	83	60	00	27	10	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	83	60	00	----	----	----	----

(2) 修改 profiled deceleration: 10000 (RPM/(秒))，即每秒能够减速到 10000 RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	84	60	00	27	10	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	84	60	00	----	----	----	----

3.2.3.2.4、速度控制

(1) 修改 Object 0x6080: max motor speed:2000 (RPM) ,电机最高转速限制, 该参数不需要实时修改.

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	80	60	00	D0	07	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	80	60	00	----	----	----	----

(2) 修改 Object 0x60FF: Target velocity:100,发送成功后, 电机会自动根据配置的电子齿轮比计算执行到电机轴的速度给定值。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	FF	60	00	64	00	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	FF	60	00	----	----	----	----

(3) 读取 Object 0x6041: Statusword bit10 速度给定到达, 或者直接读取速度反馈值进行比较。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	10	41	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4B	41	60	00	*	*	----	----

请等待 Statusword bit10 被置 1.

3.2.3.3、行规力矩环(TQ)应用举例 1

Object 0x6060: Modes of operation : 0x04

Object 0x6061: Modes of operation display: 0x04

Controlword(6040h) of the profiled torque mode:

Bit	Function
0...3	Described in Device Control
7	Described in Device Control
8	Halt
9..12	Described in Device Control
13	Manufacturer Definition
14..15	Described in Device Control

Name	Value	Description
Halt	0	执行力矩控制控制。
	1	以 <i>profile acceleration</i> 停止轴。
Clear buffered position	0	不执行指令
	1	把位置给定和位置反馈都清零

Statusword(6041h) of the profiled velocity mode:

Bit	Function
0...9	Described in Device Control
10	Target reached
11	Described in Device Control
13	Following error
14	Clear buffered position acknowledge
12...15	Described in Device Control

Name	Value	Description
Target reached	0	Halt = 0: Target torque not reached. Halt = 1: Axle decelerates.
	1	Halt = 0: Target torque reached. Halt = 1: Torque of axle is 0.
Following error	0	
	1	Following error.
Clear buffered position acknowledge	0	Clear buffered position = 0
	1	Clear buffered position = 1

3. 2. 3. 3. 1、修改工作模式

(1) 写 Object 0x6060: Modes of operation , 修改为 profile torque mode (0x04) 。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	60	60	00	04	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	60	60	00	----	----	----	----

(2) 读取 Object 0x6061: Modes of operation display, 确保该参数的值是 0x04。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	40	61	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4F	61	60	00	04	----	----	----

3. 2. 3. 3. 2、开使能

(1) 复位错误

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	80	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(2)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	00	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(3)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	06	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(4)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	07	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(5)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	0F	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

3.2.3.3.3、修改加速度和减速度

(1) 修改 profiled acceleration:10000 (RPM/(秒)) , 即每秒能够加速到 10000 RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	83	60	00	27	10	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	83	60	00	----	----	----	----

(2) 修改 profiled deceleration: 10000 (RPM/(秒)) , 即每秒能够减速到 10000 RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	84	60	00	27	10	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	84	60	00	----	----	----	----

3.2.3.3.4、力矩控制

(1) 修改 Object 0x6080: max motor speed:2000 (RPM) ,电机最高转速限制, 该参数不需要

实时修改.

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	80	60	00	D0	07	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	80	60	00	----	----	----	----

(2) 修改 Object 0x6071: Target torque:10,单位: 0.1 安培。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	71	60	00	0A	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	71	60	00	----	----	----	----

(3) 读取 Object 0x6041: Statusword bit10 力矩给定到达, 或者直接读取力矩反馈值来监测当前电机输出力矩大小。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	10	41	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4B	41	60	00	*	*	----	----

请等待 Statusword bit10 被置 1.

3.2.3.4、周期同步位置环(CSP)应用举例 1

此模式是依据 EtherCAT 总线上修改而来, 舍弃了行规位置环(PP)控制需要握手的繁杂应答机制, 在开启电机功率电输出后, 只要电机位置反馈(0x6064_00)与电机给定位置(0x607A_00), 注意: 该模式是绝对位置方式控制。

Object 0x6060: Modes of operation : 0x08

Object 0x6061: Modes of operation display: 0x08

Controlword(6040h) of the cyclic sync position mode:

Bit	Function
0...3	Described in Device Control
4	Set new point
5	Change set immediately
6	Absolute/relative movement
7	Described in Device Control
8	Halt
9..12	Described in Device Control
13	Manufacturer Definition
14..15	Described in Device Control

Name	Value	Description
abs/rel	0	目标位置是一个绝对值。
	1	目标位置是一个相对值。
Halt	0	执行定位。
	1	以 <i>profile acceleration</i> 停止轴。
Clear buffered position	0	不执行指令
	1	把位置给定和位置反馈都清零

Statusword(6041h) of the cyclic sync position mode:

Bit	Function
0...9	Described in Device Control
10	Target reached
11	Described in Device Control
12	Set new point acknowledge
13	Following error
14	Clear buffered position acknowledge
12...15	Described in Device Control

Name	Value	Description
Target reached	0	Halt = 0: Target position not reached. Halt = 1: Axle decelerates.
	1	Halt = 0: Target position reached. Halt = 1: Velocity of axle is 0.
Following error	0	
	1	Following error.
Clear buffered	0	Clear buffered position = 0
	1	Clear buffered position = 1

position acknowledge		
-------------------------	--	--

3.2.3.4.1、修改工作模式

(1) 写 Object 0x6060: Modes of operation , 修改为 cyclic sync position mode (0x08) 。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	60	60	00	08	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	60	60	00	----	----	----	----

(2) 读取 Object 0x6061: Modes of operation display, 确保该参数的值是 0x08。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	40	61	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4F	61	60	00	08	----	----	----

3.2.3.4.2、开使能

(1) 复位错误

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	80	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(2)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	00	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(3)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	06	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(4)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	07	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(5)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	0F	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

3.2.3.4.3、修改加速度和减速度

(1) 修改 profiled acceleration:10000 (RPM/(秒))，即每秒能够加速到 10000 RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	83	60	00	27	10	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	83	60	00	----	----	----	----

(2) 修改 profiled deceleration: 10000 (RPM/(秒))，即每秒能够减速到 10000 RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
-------	--------	--------	--------	--------	--------	--------	--------	--------

608	23	84	60		00	27	10	00	00
------------	----	----	----	--	----	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	84	60	00	----	----	----	----

3.2.3.4.1、位置控制

(1) 修改 Object 0x6080: max motor speed:2000 (RPM) ,电机最高转速限制.

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	80	60	00	D0	07	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	80	60	00	----	----	----	----

(2) 修改 Object 0x607A: Target position:512pulse,发送成功后, 电机会自动计算当前位置反馈和 位置给定的误差, 并跟随到位置给定值。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	7A	60	00	00	02	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	7A	60	00	----	----	----	----

(3) 读取 Object 0x6041: Statusword bit10 位置到达, 或者直接读取位置反馈值进行比较。

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	10	41	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4B	41	60	00	*	*	----	----

请等待 Statusword bit10 被置 1.

3.2.3.5、回零模式(HOME)

Object 0x6060: Modes of operation : 0x06.

Object 0x6061: Modes of operation display: 0x06.

回零模式需要根据实际应用来选择合适的回零方法，对于正限位、负限位、回零开关的 数字 IO 输入接口的选择，需要在调试软件进行配置

回零模式由控制字和状态字的位控制。

Homing Mode Controlword(6040h)

Bit	Function
0...3	Described in Device Control
4	Home operation start
5..6	Reserved
7	Described in Device Control
8	Halt
9..12	Described in Device Control
13..15	Described in Device Control

Name	Value	Description
Homing operation start	0	Homing mode 不激活
	0→1	开启 homing mode.
	1	Homing mode 激活
	1→0	中断 homing mode.
Halt	0	执行 the instruction of bit 4.
	1	以 homing deceleration 停止轴转动

Notes:

如果通过将位 4 从“1”设置为“0”来中断回原点，则运动电机不中断；也就是说，电动机也保持其当前状态是否移动。 归属序列被中断而归属目标不被中断实现。 通过将该位设置回“1”，再次开始归位模式。

如果发生停止，驱动器将停止归位方法并停止电机根据 object 609Ah。当该位设置为“0”且位 4 保持为“1”时，家庭方法再次开始

Statusword of Homing Mode

Bit	Function
0...9	Described in Device Control
10	Target reached
11	Described in Device Control
12	Homing attained
13	Homing error
14...15	Described in Device Control

Name	Value	Description
Target reached	0	Halt = 0: Homing position 未到达 Halt = 1: Axle decelerates
	1	Halt = 0: Homing 位置到达 Halt = 1: Velocity of axle is 0
Homing attained	0	Homing mode 尚未完成
	1	Homing mode 成功执行
Homing error	0	No homing error.
	1	Homing error occurred. Homing mode carried out unsuccessfully. Error cause found in error code.

在归位过程中，没有参考位置限制激活。如果没有物理限制存在 - 例如限位开关 - 负载可能无限期地移动。

3.2.3.5.1、修改回零模式

(1) 写 Object 0x6060: Modes of operation ,进入 Homing(0x06) .

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	60	60	00	06	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	60	60	00	----	----	----	----

(2) 读取 Object 0x6061: Modes of operation display is not 0x06?

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	40	61	60	00	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4F	61	60	00	06	----	----	----

3.2.3.5.2、开使能

(1) 复位错误

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	80	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(2)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	00	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(3)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	06	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(4)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	07	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

588	60	40	60	00	----	----	----	----
------------	----	----	----	----	------	------	------	------

(5)

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	0F	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

3.2.3.5.3、修改 Object 0x6099: Homing speeds

(1) 修改 Speed during search for switch: 500RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	99	60	01	F4	01	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	99	60	01	----	----	----	----

(2) 修改 Speed during search for zero: 50 RPM

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	99	60	02	32	00	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	99	60	02	----	----	----	----

(3) Modify Object 0x6098: Homing method: 17.

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	98	60	00	11	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	81	60	00	----	----	----	----

(4) 修改 Object 0x607C: Home offset:0 pulse

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
-------	--------	--------	--------	--------	--------	--------	--------	--------

608	23	7A	60	00	00	02	00	00
------------	----	----	----	----	----	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	7A	60	00	----	----	----	----

3.2.3.5.4、Homing 控制

(1) 开启 homing mode.

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	1F	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

(2) 读取 Object 0x6041: Statusword bit10 位置到达 and bit12 回零完成

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	10	41	60	00	----	----	----	----

Drive response

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	4B	41	60	00	*	*	----	----

请等待 Statusword bit10 被置 1 and bit12 被置 1.

(3) 停止 homing mode.

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	40	60	00	0F	00	----	----

Drive response

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	40	60	00	----	----	----	----

3.2.4、电子齿轮比

默认电子齿轮比为 1，如果需要改动，可以通过 can 来配置或者通过驱动器自带调试软件改动，如果当前电子齿轮比为 2，位置给定（607A_00）给定 1000 个脉冲，在运行后，电机实际会转到 2000 个脉冲位置。

电子齿轮比的配置将会影响到位置给定、位置反馈、速度给定、速度反馈等关键控制参数。

驱动器计算齿轮比是通过这些对象索引： 608F、6091、6092

$$\text{电子齿轮比} = \frac{\frac{\text{Encoder_increments}(608F_01)}{\text{Motor_revolutions}(608F_02)} \times \frac{\text{Gear_ratio}(6091_01)}{\text{Gear_ratio}(6091_02)}}{\frac{\text{Feed_constant}(6092_01)}{\text{Feed_constant}(6092_02)}}$$

电子齿轮比影响到的参数：

- Position actual value(6064_00),
- TargetPosition(607A_00),
- VelocityDemandValue(606B_00),
- VelocityActualValue(606C_00),
- ProfileVelocity(6081_00),
- TargetVelocity(60FF_00) 等参数。

3.2.5、Receive PDO 映射配置

允许在预操作模式下动态实时修改 RXPDO 映射。

3.2.5.1、默认的映射表

RXPDO 1 :

Index	Sub-index	Name	Default Value
1400h	0h	Number of entries	3
	1h	COB-ID used by RXPDO 1	208h
	2h	Transmission type	FFh
	3h	Inhibit time	3h

Index	Sub-index	Name	Default Value
1600h	0h	Number of mapped entries	2
	1h	Controlword	6040 00 10h
	2h	OperationMode	6060 00 08h
	3h		0000 00 00 h
	4h		0000 00 00 h

RXPDO 2 :

Index	Sub-index	Name	Default Value
1401h	0h	Number of entries	3
	1h	COB-ID used by RXPDO 2	308h
	2h	Transmission type	FFh
	3h	Inhibit time	3h

Index	Sub-index	Name	Default Value
1601h	0h	Number of mapped entries	2
	1h	TargetPosition	607A 00 20h
	2h	ProfileVelocity	6081 00 20h
	3h		0000 00 00 h
	4h		0000 00 00 h

RXPDO 3 :

Index	Sub-index	Name	Default Value
1402h	0h	Number of entries	3
	1h	COB-ID used by RXPDO 3	408h
	2h	Transmission type	FFh
	3h	Inhibit time	3h

Index	Sub-index	Name	Default Value
-------	-----------	------	---------------

1602h	0h	Number of mapped entries	0
	1h		0000 00 00h
	2h		0000 00 00h
	3h		0000 00 00 h
	4h		0000 00 00 h

RXPDO 4 :

Index	Sub-index	Name	Default Value
1403h	0h	Number of entries	3
	1h	COB-ID used by RXPDO 4	508h
	2h	Transmission type	FFh
	3h	Inhibit time	3h

Index	Sub-index	Name	Default Value
1603h	0h	Number of mapped entries	0
	1h		0000 00 00h
	2h		0000 00 00h
	3h		0000 00 00 h
	4h		0000 00 00 h

3.2.5.2、配置 RXPDO1 映射应用举例 1

必须在预操作模式下配置映射。

(1) 禁止启用 RXPDO 1 接收。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	14	01	08	02	00	80

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	14	01	----	----	----	----

(2) 清除 RXPDO 1 映射

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	00	16	00	00	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	16	00	----	----	----	----

(3) RXPDO 1 的子索引 01 的映射是 60400010 (控制字)。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	16	01	10	00	40	60

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	16	01	----	----	----	----

(4) 配置 RXPDO 1 的子索引 02 到 60600008 的映射 (工作模式给定)。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	16	02	08	00	60	60

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	16	02	----	----	----	----

(5) 配置 RXPDO 1 异步触发模式。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	00	14	02	FF	00	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	14	02	----	----	----	----

(6) 将 RXPDO 1 映射的对象数设置为 2。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	00	16	00	02	00	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	16	00	----	----	----	----

(7) 启用 RXPDO 1 接收。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	14	01	08	02	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	14	01	----	----	----	----

3.2.6、Transmit PDOs

允许在预操作模式下动态实时修改 TXPDO 映射。

3.2.6.1、默认映射表

TXPDO 1 :

Index	Sub-index	Name	Default Value
1800h	0h	Number of entries	5
	1h	COB-ID used by TXPDO 1	00000188h
	2h	Transmission type	FFh
	3h	Inhibit time	3h
	4h	Reserved	
	5h	EventTime	64h

Index	Sub-index	Name	Default Value
1A00h	0h	Number of mapped entries	3
	1h	position_actual_value	6064 00 20h
	2h	StatusWord	6041 00 10h
	3h	DisplayMode	6061 00 08 h
	4h		0000 00 00 h

TXPDO 2 :

Index	Sub-index	Name	Default Value
1801h	0h	Number of entries	5
	1h	COB-ID used by TXPDO 2	00000288h
	2h	Transmission type	FFh
	3h	Inhibit time	0h
	4h	Reserved	
	5h	EventTime	0h

Index	Sub-index	Name	Default Value
1A01h	0h	Number of mapped entries	1
	1h	position_actual_value	6064 00 20h
	2h		0000 00 00h
	3h		0000 00 00 h
	4h		0000 00 00 h

TXPDO 3 :

Index	Sub-index	Name	Default Value
1802h	0h	Number of entries	5
	1h	COB-ID used by TXPDO 3	00000388h
	2h	Transmission type	FFh
	3h	Inhibit time	0h
	4h	Reserved	
	5h	EventTime	0h

Index	Sub-index	Name	Default Value
1A02h	0h	Number of mapped entries	0
	1h	position_actual_value	0000 00 00h
	2h		0000 00 00h
	3h		0000 00 00 h
	4h		0000 00 00 h

TXPDO 4 :

Index	Sub-index	Name	Default Value
1803h	0h	Number of entries	5
	1h	COB-ID used by TXPDO 4	00000488h
	2h	Transmission type	FFh
	3h	Inhibit time	0h
	4h	Reserved	
	5h	EventTime	0h

Index	Sub-index	Name	Default Value
1A03h	0h	Number of mapped entries	0
	1h		0000 00 00h
	2h		0000 00 00h
	3h		0000 00 00 h
	4h		0000 00 00 h

3.2.6.2、配置 TXPDO1 映射应用举例 1

必须先进入操作前模式，然后才能进行配置。

(1) 禁止启用 TXPDO 1 接收。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	18	01	88	01	00	80

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	18	01	----	----	----	----

(2) 清除 TXPDO 1 映射

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	00	1A	00	00	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	1A	00	----	----	----	----

(3) TXPDO 1 的子索引 01 的映射是 60640020 (position actual value) 。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	1A	01	20	00	64	60

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	1A	01	----	----	----	----

(4) 配置 TXPDO 1 子索引 02 的映射是 60410010 (StatusWord) 。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	1A	02	10	00	41	60

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	1A	02	----	----	----	----

(5) 配置 TXPDO 1 子索引 03 的映射是 60610008 (DisplayMode) 。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	1A	03	08	00	61	60

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	1A	03	----	----	----	----

(6) 将 TXPDO 1 配置为异步触发模式。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	00	1A	02	FF	00	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	1A	02	----	----	----	----

(7) 配置 TXPDO 1 抑制时间为 10ms。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2B	00	18	03	64	00	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	18	03	----	----	----	----

(8) 配置 TXPDO 1 触发时间为 100ms。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
601	2B	00	1A	05	64	00	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	1A	05	----	----	----	----

(9) 将 TXPDO 1 映射的对象数设置为 3。

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	2F	00	1A	00	03	----	----	----

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	1A	00	----	----	----	----

(10) 启动 TXPDO 1 接收

主机设备发送

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
608	23	00	18	01	88	01	00	00

驱动器响应

COBID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
588	60	00	18	01	----	----	----	----

3.2.7、 position actual value(6064h_00h) 清零

使用回零方法 35 可以把位置反馈值清零，

如果未使用回零方法，也可以清除位置实际值（6064h_00h）。

- (1) 写 Object 0x6060: Modes of operation ,进入 profiled position(0x01) 模式。
 - (2) 写 Object 0x6040: Controlword's bit13 置 1. Object 0x6041: Statusword's bit 14 也被置 1.
- If bit13 被置 1, 则 position actual value(6064h_00h) 和 position actual internal value (6063h_00h), position demandl value(6062h_00h) ,position demand internal value (60FCh_00h) 都能够被清零。

3.2.8、详细手册资料

[详细手册资料、例子请查看本公司对应的 Can402 应用手册](#)

3.3、模拟量输入控制

硬件参数配置，如下图：

地址	参数名	值	单位
1	模拟输入0增益补偿	1.503	(-3.5----3.5
2	模拟输入0偏补偿	0	(-32768----32
3	模拟输入0低通滤波	0	0.1ms(0----32
4	模拟输入0电压死区	50	mV(0----327
5	模拟输入1增益补偿	1.503	(-3.5----3.5
6	模拟输入1偏补偿	0	(-32768----32
7	模拟输入1低通滤波	0	0.1ms(0----32
8	模拟输入1电压死区	50	mV(0----327

参数出厂已经配置好，用户可以根据需要 修改 电压死区（保证控制的稳定性、避免零速抖动） 即可，其他参数配置保持不变

3.3.1、速度环配置

这里仅使用模拟输入 0 通道

(1)、

地址	参数名	值
1 108C	速度给定方式选择	0003
2 108D	转速限制	000F
3 108E	速度给定平滑滤波时间常数	0
4 108F	速度反馈平滑滤波时间常数	20

*Bit0~Bit3:
0: CAN402 PP模式给定
1: CAN402 PV模式给定
2: USB/RS485/RS422给定
3: 模拟量0通道给定
4: 模拟量1通道给定

将“速度给定方式选择”写“0003”，表示使用模拟量输入 0 通道做为速度给定源，

将“转速限制”写“000F”，表示不限制速度最大（防止受其他影响）

(2)、出厂默认 1

参数名	值	单位
10 速度设定比例	1	(-3.5----3.5)
11 速度反馈比例	0	(-3.5----3.5)
12 保留	0	(-32768----32767)
13 速度给定/限制0	0	rpm或0.1rpm(-32768----32...
14 速度给定/限制1	0	rpm或0.1rpm(-32768----32...

“速度设定比例” 写 1，表示：当模拟量输入是+12v 的时候，对应正向最高转速；

当模拟量输入是-12v 的时候，对应反向最高转速；

3.3.2、力矩环配置

这里仅使用模拟输入 0 通道

3、双击参数值单元格，可以修改数值，修改完后，按回车或点击“设定值变更按钮”可修改完成

地址	参数名	值	单位	
1	1062	电流给定方式选择	3	(0---65535)
2	1063	电流环给定比例	1	(-3.5---3.5)
3	1064	电流环给定滤波常数	0	ms(0---65535)
4	1065	电流环限制选择	000F	H(0000---65535)

自定义注释

*电流环给定
0: CAN402PT模式给定
1: 用户自定义给定
2: USB/RS485/RS422给定
3: 模拟量0通道给定
4: 模拟量1通道给定

添加注释 添加完成 清空当前文本

将“电流给定方式选择”写“0003”，表示使用模拟量输入 0 通道做为电流给定源，

将“电流环限制选择”写“000F”，表示不限制给定电流（防止受其他影响）

“电流环给定比例”写 1，表示：当模拟量输入是+12v 的时候，对应正向电流最大输出；

当模拟量输入是-12v 的时候，对应反向电流最大输出；

3.4、脉冲输入控制

3.4.1、方向+脉冲配置（位置环）

地址	参数名	值
1 10D0	位置环给定方式选择	1
2 10D1	电子齿轮分子0	1
3 10D2	电子齿轮分子1	1
4 10D3	电子齿轮分子2	1
5 10D4	电子齿轮分母	1

自定义注释

添加注释 | 添加完成

*位置环给定
 0: A/B脉冲 (增量式位置环)
 1: 方向+脉冲 (增量式位置环)
 2: USB/RS485/RS422绝对给定
 3: POSx设定 (增量式位置环)
 4: POSx设定 (绝对位置环)

将“位置环给定方式选择”写1，表示选择方向+脉冲控制（首次修改，需要“断电保存”，重新上电生效）

电子齿轮比：电子齿轮分子0 / 电子齿轮分母

3.4.2、AB 脉冲配置（位置环）

地址	参数名	值	单位
1 10D0	位置环给定方式选择	0	(0----65535)
2 10D1	电子齿轮分子0	1	(1----65535)
3 10D2	电子齿轮分子1	1	(1----65535)
4 10D3	电子齿轮分子2	1	(1----65535)
5 10D4	电子齿轮分母	1	(0----65535)

自定义注释

添加注释 | 添加完成 | 清空当前文本

*位置环给定
 0: A/B脉冲 (增量式位置环)
 1: 方向+脉冲 (增量式位置环)
 2: USB/RS485/RS422绝对给定
 3: POSx设定 (增量式位置环)
 4: POSx设定 (绝对位置环)

将“位置环给定方式选择”写0，表示选择 AB 脉冲控制（首次修改，需要“断电保存”，重新上电生效）

电子齿轮比：电子齿轮分子0 / 电子齿轮分母

3.5、占空比输入控制

物理接线:

占空比输入：差分接入 (PULS+、PULS-)，或者 OPC 接法 (OPC1、PULS-)

方向输入：差分接入 (SIGN+、SIGN-)，或者 OPC 接法 (OPC2、SIGN-)

3.5.1、速度环配置

将“速度给定方式选择”写“0005”，表示配置 PWM 输入通道给定（首次修改，需要“断电保存”，重新上电生效）

占空比为 0%，即低电平，表示速度给定 0；

占空比为 100%，即高电平，表示速度给定为最高速度限制值；

3.5.2、电流环配置

更按钮可修改完成

地址	参数名	值	单位
1 1062	电流给定方式选择	5	(0----65535)
2 1063	电流环给定比例	1	(-3.5----3.5)
3 1064	电流环给定滤波常数	0	ms(0----65535)
4 1065	电流环限制选择	000F	H(0000----FF)
5 1066	电流给定/限制0	0	0.1A(-32768----32768)

自定义注释

只读参数

- 2: USB/RS485/RS422给定
- 3: 模拟量0通道给定
- 4: 模拟量1通道给定
- 5: PWM输入通道给定
- 15: 最大电流

将“电流给定方式选择”写 5，表示配置 PWM 输入通道给定（首次修改，需要“断电保存”，重新上电生效）

占空比为 0%，即低电平，表示电流给定 0；

占空比为 100%，即高电平，表示电流给定为电流最大输出值；