

第 9 章 CANopen 通讯

9.1 CANopen 通讯协议介绍

开放的现场总线标准中 CANopen 是最著名和成功的一种，已经在欧洲和美国获得广泛的认可和大量应用。1992 年在德国成立了“自动化 CAN 用户和制造商协会” (CiA, CANinAutomation)，开始着手制定自动化 CAN 的应用层协议 CANopen。此后，协会成员开发出一系列 CANopen 产品，在机械制造、制药、食品加工等领域获得大量应用。

FD1X5 和 iSMK 伺服是标准的 CAN 从站设备，严格遵循 CANopen2.0A/B 协议，任何支持该协议的上位机均可以与其进行通讯。FD1X5 和 iSMK 伺服内部使用了一种严格定义的对象列表，我们把它称作对象辞典，这种对象辞典的设计方式基于 CANopen 国际标准，所有的对象有明确的功能定义。这里说的对象（Objects）类似我们常说的内存地址，有些对象如速度和位置等可以由外部控制器修改，有些对象却只能由驱动器本身修改，如状态、错误信息。这些对象都为十六进制数，如工作模式的 CANopen 地址为 0x60400010，举例如表 9-1 所示。

表 9 - 1 对象辞典举例列表

完整的 CANopen 地址组成			属性	含义
Index	Subindex	Bits(数据长度)		
0x6040	00	0x10	RW	设备状态控制字
0x6060	00	0x08	RW	工作模式
0x607A	00	0x20	W	目标位置
0x6041	00	0x10	MW	设备状态字

对象的属性有下面几种：

- 1. RW(读写)：对象可以被读也可以被写入；
- 2. RO(只读)：对象只能被读；
- 3. WO（只写）：只能写入；
- 4. M（可映射）：对象可映射，类似间接寻址；
- 5. S（可存储）：对象可存储在 Flash - ROM 区，掉电不丢失。

9.2 硬件说明

FD1X5 系列伺服驱动器 CANopen 口硬件定义如下：

表 9-2 管脚名称及功能描述表

 RJ-45接头	针脚号	信号标识	信号名称
	1	CAN_H	CAN_H bus (high dominant)
	2	CAN_L	CAN_L bus (low dominant)
	3	CAN_GNDB	信号地

iSMK 系列伺服驱动器 CANopen 口硬件定义如下：

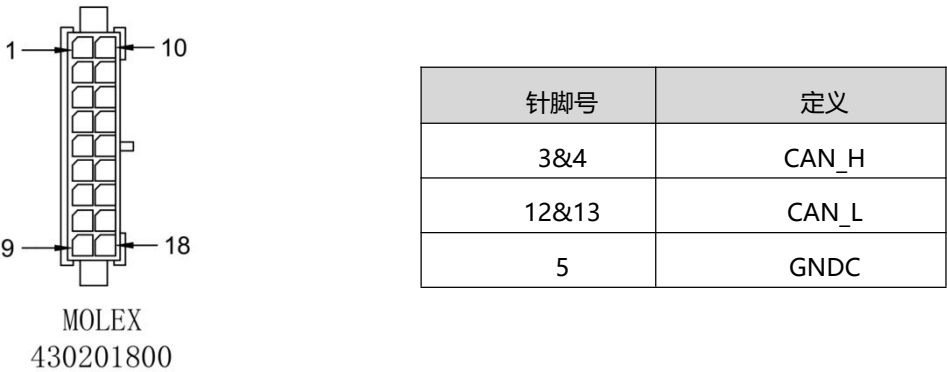


图 9-1 iSMK 驱动器 canopen 定义

CAN 通讯协议主要描述设备之间的信息传递方式，CAN 层的定义与开放系统互连模型 OSI 一致，每一层与另一设备上相同的那一层通讯，实际的通讯发生在每一设备上相邻的两层而设备只通过模型物理层的物理介质互连，CAN 的规范定义了模型的最下面两层数据链路层和物理层。CAN 总线物理层没有严格规定，能够使用多种物理介质例如双绞线光纤等，最常用的就是双绞线信号，使用差分电压传送（常用总线收发器），两条信号线被称为 CAN_H 和 CAN_L，静态时均是 2.5V 左右，此时状态表示为逻辑 1，也可以叫做隐位，用 CAN_H 比 CAN_L 高表示逻辑 0，称为显位，此时通常电压值为 CAN_H=3.5V 和 CAN_L=1.5V，竞争时显位优先。CAN 通讯接口管脚名称及功能如表 9-2 和图 9-1 所示。

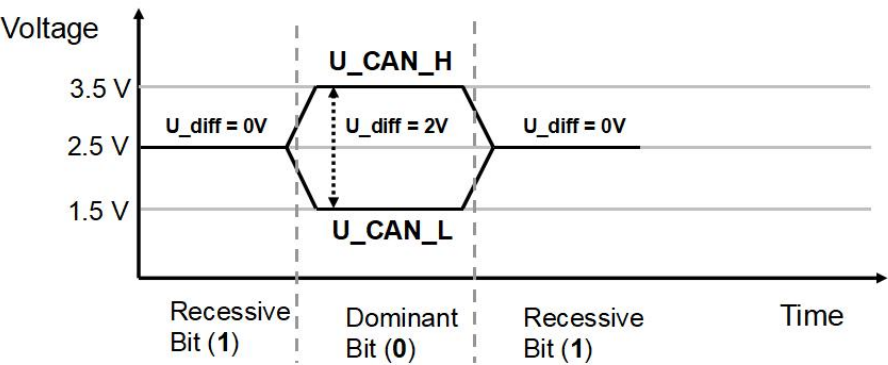


图 9-2 CAN 信号标识

注意：

- 1. 所有从站的 CAN_L、CAN_H 脚直接相接即可，采用串连的方式接线。
- 2. 主站端和最后一个从站端需要接 120 欧姆的终端电阻。FD1x5 驱动器自带终端电阻拨码 SW1，拨为 ON 开启终端电阻。更改终端电阻拨码状态后请重启驱动器。
- 3. 通讯电缆请采用屏蔽双绞线，并做好接地处理（短距离通讯时 3 脚地线可以不接，但是长距离、高波特率通讯时建议把 3 脚接地）；
- 4. 各种波特率所理论上能够通讯的最长距离如表 9-3 所示。
- 5. FD1X5 和 iSMK 系列伺服驱动器不需要连接外部 24V 电源给 CAN 供电。

表 9-3 各波特率理论上能够通讯的最长距离表

通讯速度 (bit/s)	通讯距离 (M)
1M	25
800K	50
500K	100
250K	250
125K	500
50K	600
25K	800
10K	1000

9.3 软件说明

9.3.1 EDS 说明

EDS (电子数据表格) 文件是 PLC 所连接从站的标识文件或者类似码，通过该文件来辨认从站所属的类型 (是 401、402、403 中的何种类别，或者属于 402 中的哪一种设备)。该文件包含包含了从站的所有信息，比如生产厂家、序列号、软件版本、支持波特率种类、可以映射的 OD 及各个 OD 的属性等等参数，类似于 Profibus 的 GSD 文件。因此在进行硬件配置前，我们首先需要把从站的 EDS 文件导入到上位组态软件中。

9.3.2 SDO 说明

SDO 主要用来在设备之间传输低优先级的对象，典型是用来对从设备进行配置、管理,比如用来修改电流环、速度环、位置环的 PID 参数，PDO 配置参数等，这种数据传输跟 MODBUS 的方式一样，即主站发出后，需要从站返回数据响应。这种通讯方式只适合对参数的设置，不适合于对实时性要求较高的数据传输。

SDO 的通讯方式分为上传和下载，上位机可以根据专用的 SDO 读写指令来读写伺服内部的 OD 即可。在 CANopen 协议中，对对象字典的内容进行修改可以通过 SDO (Service Data Object) 来完成，下面介绍 SDO 命令的结构和遵循的准则。

SDO 的基本结构如下：Client→Server/Server→Client

Byte0	Byte1-2	Byte3	Byte4-7
SDO Command specifier	对象索引	对象子索引	最大 4 字节数据

SDO 命令字包含如下信息：

- 下载/上传 (Download/upload)
- 请求/应答 (Request/response)
- 分段/加速传送 (Segmented/expedited transfer)
- CAN 帧数据字节长度，用于后续每个分段的交替清零和置位的触发位 (toggle bit)

SDO 中实现了 5 个请求/应答协议：

- 启动域下载 (Initiate Domain Download) ；
- 域分段下载 (Download Domain Segment) ；
- 启动域上传 (Initiate Domain Upload) ；

- 域分段上传 (Upload Domain Segment) ；
- 域传送中止 (Abort Domain Transfer) 。

其中，下载 (Download) 是指对对象字典进行写操作，上传 (Upload) 指对对象字典进行读操作；读取参数时,使用启动域上传(Initiate Domain Upload)协议;设置参数时,使用启动域下载(Initiate Domain Download)协议；协议的 SDO 命令字 (SDO CAN 报文的第一个字节) 语法在表 9-4 和表 9-5 中说明，其中 “-” 表示不相关，应为 0) 。

表 9-4 启动域下载

Bit	7	6	5	4	3	2	1	0
Client→	0	0	1	-	n		e	s
←Server	0	0	1	-	-	-	-	-

表 9-5 启动域上传

Bit	7	6	5	4	3	2	1	0
Client→	0	0	1	-	-	-	-	-
←Server	0	0	1	-	n		e	s

说明：

- n——表示报文数据中无意义数据的字节数【从(8 - n)字节到第 7 字节数据无意义】(当 e=1 且 s=1 时 n 有效，否则 n 为 0)；
- e——e=0 时正常传送，e=1 时加速传送；
- s——表示是否指明数据长度，0 为数据长度未指明，1 为数据长度指明。
- e=0, s=0——由 CiA 保留；
- e=0, s=1——数据字节为字节计数器，byte4 是数据低位部分 (LSB) ， byte7 是数据高位部分 (MSB) ；
- e=1——数据字节为将要下载 (download) 的数据。

SDO 报文格式说明

SDO 的传输为不高于 4 个字节的对象数据传输，报文由 COB-ID 和数据段组成。下表可以看出发送 SDO(RSD)和接收 SDO 的 COB-ID 不一致。

数据段采用“低位在前，高位在后”的排列方式。所有 SDO 报文数据段都必须是八个字节。SDO 传输报文格式如下表所示。

表 9-6 SDO 传输报文格式表

	COB-ID	DLC	Data							
			0	1	2	3	4	5	6	7
主站 →	0x600+Node_ID	8	发送命令字	对象索引	对象子索引	00				
← 从站	0x580+Node_ID	8	接收命令字	对象索引	对象子索引	最大 4 字节数据				

其中，命令代码指明了该段 SDO 的传输类型和传输数据长度，索引和子索引是传输的对象，数据是该对象的数值。

SDO 读报文

根据接收的数据长度，接收的 SDO 命令字会有所改变，报文格式如下：

表 9-7 SDO 读报文说明

		COB-ID	DLC	Data								
				0	1	2	3		4	5	6	7
➡ 主站		0x600+Node_ID	8	40	对象索引		对象子索引		-	-	-	-
⬅ 从站	正常	0x580+Node_ID	8	4F	对象索引		对象子索引		数据	-	-	-
	正常	0x580+Node_ID	8	4B	对象索引		对象子索引		数据		-	-
	正常	0x580+Node_ID	8	43	对象索引		对象子索引		数据			
	错误	0x580+Node_ID	8	80	对象索引		对象子索引		错误代码			

读取参数时 SDO 报文发送命令字均为 0x40，根据驱动器返回的报文数据长度，接收的命令字会有不同。



注意
“-”表示有数据但不予考虑，写数据时建议写 0，下同。

SDO 写报文

使用 SDO 对不高于 4 个字节的对象进行修改参数，按照读写方式及内容数据长度的不一致，传输报文各不相同。SDO 写报文如下表所示：

表 9-8 SDO 写报文说明

		COB-ID	DLC	Data							
				0	1	2	3	4	5	6	7
→ 主站		0x600+Node_ID	8	2F	对象索引		对象子索引	数据	-	-	-
		0x600+Node_ID	8	2B	对象索引		对象子索引	数据		-	-
		0x600+Node_ID	8	23	对象索引		对象子索引	数据			
← 从站	正常	0x580+Node_ID	8	60	对象索引		对象子索引	-	-	-	-
	错误	0x580+Node_ID	8	80	对象索引		对象子索引	错误代码			

写入参数时，SDO 报文发送时命令字与对象的数据长度有关，如果待发数据为 1 个字节，则发送命令字为 0x2F；如果待发数据为 2 个字节，则发送命令字为 0x2B；如果待发数据为 4 个字节，则发送命令字为 0x23；SDO 报文发送成功，接收命令字为 0x60；SDO 报文发送失败，接收命令字为 0x80。

举例：

从站站号为 1，用 SDO 写对象目标位置 607A.00，写入数值为 100000，即 0x186A0，主站发送报文如下表所示。（所有均为 16 进制）

主站发送 SDO 报文

COB-ID	DLC	Data									
		0	1	2	3	4	5	6	7		
0x601	8	23	7A	60	00	A0	86	01	00		

写入正常驱动器接收 SDO 报文

COB-ID	DLC	Data							
		0	1	2	3	4	5	6	7
0x601	8	60	7A	60	00	A0	86	01	00

若写入不正常，则返回错误命令字 0x80

COB-ID	DLC	Data							
		0	1	2	3	4	5	6	7
0x601	8	80	7A	60	00	01	00	01	06

表 9-9 SDO 报文错误代码

错误代码	代码功能描述
0x05040001	无效命令，未知或非法的 Client/Server 命令字
0x06010001	试图读只写对象参数
0x06010002	试图写只读对象参数
0x06020000	无效索引，对象字典中不存在该对象
0x06040041	无法映射，对象参数不支持映射到 PDO
0x06060000	驱动器处于报错故障状态导致对象参数访问失败
0x06070010	数据类型不匹配，服务参数长度不匹配
0x06070012	数据类型不匹配，服务参数长度太大
0x06070013	数据类型不匹配，服务参数长度太短
0x06090011	无效子索引
0x06090030	无效数据，超出对象参数设定范围
0x06090031	写入数据数值太大
0x06090032	写入数据数值太小
0x08000022	由于当前设备状态导致数据不能传送或保存到应用

表 9-10 通过 SDO 报文设定原点模式

参数地址	名称	数值	报文(ID=1)
60400010	控制字	F	发→601 2B 40 60 00 0F 00 00 00 收←581 60 40 60 00 0F 00 00 00
60600008	工作模式	6	发→601 2F 60 60 00 06 00 00 00 收←581 60 60 60 00 06 00 00 00
60980008	原点模式	33	发→601 2F 98 60 00 21 00 00 00 收←581 60 98 60 00 21 00 00 00
60990120	原点转折信号速度	200RPM	发→601 23 99 60 01 03 9D 36 00 收←581 60 99 60 01 03 9D 36 00
60990220	原点信号速度	100RPM	发→601 23 99 60 02 82 4E 1B 00 收←581 60 99 60 02 82 4E 1B 00
60400010	控制字	1F	发→601 2B 40 60 00 1F 00 00 00 收←581 60 40 60 00 1F 00 00 00
发送 601 40 41 60 00 00 00 00 00 读状态字，驱动器回复 581 4B 41 60 00 31 C0 FF FF，bit15=1 代表原点找到			

表 9-11 通过 SDO 报文设定位置模式

参数地址	名称	数值	报文(ID=1)
60400010	控制字	2F	发→601 2B 40 60 00 2F 00 00 00 收←581 60 40 60 00 2F 00 00 00
		4F	发→601 2B 40 60 00 4F 00 00 00 收←581 60 40 60 00 4F 00 00 00
60600008	工作模式	1	发→601 2F 60 60 00 01 00 00 00 收←581 60 60 60 00 01 00 00 00
607A0020	目标位置	50000inc	发→601 23 7A 60 00 50 C3 00 00 收←581 60 7A 60 00 50 C3 00 00
60810020	梯形速度	200RPM	发→601 23 81 60 00 03 9D 36 00 收←581 60 81 60 00 03 9D 36 00
60830020	梯形加速度	100rps/s	发→601 23 83 60 00 6E A3 01 00 收←581 60 83 60 00 6E A3 01 00
60840020	梯形减速度	100rps/s	发→601 23 84 60 00 6E A3 01 00 收←581 60 84 60 00 6E A3 01 00
60400010	控制字	3F	发→601 2B 40 60 00 3F 00 00 00 收←581 60 40 60 00 3F 00 00 00
		5F	发→601 2B 40 60 00 5F 00 00 00 收←581 60 40 60 00 5F 00 00 00
发送 601 40 41 60 00 00 00 00 00 读状态字, 驱动器回复 581 4B 41 60 00 37 C4 FF FF,bit10 代表目标位置到			

表 9-12 通过 SDO 报文设定速度模式

参数地址	名称	数值	报文(ID=1)
60600008	工作模式	3	发→601 2F 60 60 00 03 00 00 00 收←581 60 60 60 00 03 00 00 00
60FF0020	目标速度	-100RPM	发→601 23 FF 60 00 7E B1 E4 FF 收←581 60 FF 60 00 7E B1 E4 FF
60400010	控制字	2F	发→601 2B 40 60 00 2F 00 00 00 收←581 60 40 60 00 2F 00 00 00
60830020	梯形加速度	100rps/s	发→601 23 83 60 00 6E A3 01 00 收←581 60 83 60 00 6E A3 01 00
60840020	梯形减速度	100rps/s	发→601 23 84 60 00 6E A3 01 00 收←581 60 84 60 00 6E A3 01 00

注意: 报文以 16 进制表示, 本案例使用的电机分辨率为 65536

9.3.3 PDO 说明

PDO 一次性可传送 8 个字节的数据, 没有其它协议预设定 (意味着数据内容已预先定义), 主要用来传输需要高频率交换的数据。

PDO 的传输方式打破了现有的数据问答式传输理念, 采用全新的数据交换模式, 设备双方在传输前先在各个设备定义好数据接收和发送区域, 在数据交换时直接发送相关的数据到对方的数据接收区即可, 减少了问答式的询问时间, 从而极大的提高了总线通讯的效率, 从而得到了极高的总线利用率。

9.3.3.1 PDO COB-ID 说明

COB-ID 是 CANopen 通讯协议的特有方式，它的全称是 Communication Object Identifier-通讯对象-ID，这些 COB-ID 为 PDO 定义了相应的传输级别，有了这些传输级别后，控制器和伺服就能够在各自的软件里配置里定义相同的传输级别和其里面的传输内容，这样控制器和伺服都采用的同一个传输级别和传输内容后，数据的传输即透明化了，也就是双方都知道所要传输的数据内容了，也就不需要在传输数据时还需要对方回复数据是否传输成功。

缺省 ID 分配表是基于 CANopen 2.0A 定义的 11 位 CAN-ID (CANopen 2.0B 协议 COB-ID 是 29 位)，包含一个 4 位的功能码部分和一个 7 位的节点 ID(Node-ID)部分，如图 9-3 所示。

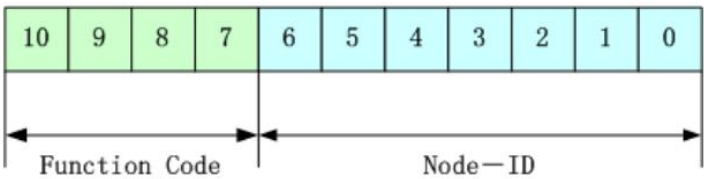


图 9-3 缺省 ID 说明图

➔

注意

Node-ID —— 即伺服的站号，Node-ID 范围是 1~127；

Function Code —— 数据传输的功能码，定义各种 PDO、SDO、管理报文的传输级别，功能码越小，优先级越高。

表 9-13 CANopen 预定义主/从连接集 CAN 标识符分配表

对象	COB-ID
NMT Module Control	000H
SYNC	080H
TIME SSTAMP	100H
对象	COB-ID
紧急	081H-0FFH
PDO1 (发送)	181H-1FFH
PDO1 (接收)	201H-27FH
PDO2 (发送)	281H-2FFH
PDO2 (接收)	301H-37FH
PDO3 (发送)	381H-3FFH
PDO3 (接收)	401H-47FH
PDO4 (发送)	481H-4FFH
PDO4 (接收)	501H-57FH
SDO (发送/服务器)	581H-5FFH
SDO (接收/客户)	601H-67FH
NMT Error Control	701H-77FH

➔

注意

1、COB-ID 越小，优先级越高；

2、每一个级别的 COB-ID 前面的功能码是固定格式；

3、COB-ID 为 00H、80H、100H、701H-77FH、081H-0FFH 均为系统管理格式。

9.3.3.2 COB-ID

- 发送 PDO 相对于伺服来说就是指伺服发送出去的数据，这些数据由 PLC 来接收。发送 PDO 的功能码 (COB-ID) 为：
 - 1、0x180+伺服站号
 - 2、0x280+伺服站号
 - 3、0x380+伺服站号
 - 4、0x480+伺服站号
- 接收 PDO 相对于伺服来说就是指伺服接收的数据，这些数据由 PLC 来发送，发送 PDO 的功能码 (COB-ID) 为：
 - 1、0x200+伺服站号
 - 2、0x300+伺服站号
 - 3、0x400+伺服站号
 - 4、0x500+伺服站号

➔

注意

由于 FD1X5 和 iSMK 系列伺服驱动器是按标准的 CANopen 2.0A 协议来设计的，但同时也支持 CANopen 2.0B 协议，也就是说，如果上面的 8 个 PDO 不够用的情况下，您还可以定义新的 PDO，比如用 0x43FH 来作为 1 号站的通讯 PDO，只要控制器和伺服都按照这个来定义即可。

9.3.3.3 PDO 传输类型

- 同步 (SYNC) ——由同步报文触发传输 (传输类型：0-240)

在该传输模式下，控制器必须具有发送同步报文的能力 (频率最高为 1KHZ 的周期发送的报文)，伺服在接收到该同步报文后在发送。

非周期——由远程帧预触发传送，或者由设备子协议中规定的对象特定事件预触发传送。该方式下伺服驱动器每接收到一个同步报文 PDO 里的数据即发送一次。

周期——传送在每 1 到 240 个 SYNC 消息后触发。该方式下伺服驱动器每接收到 n 个同步报文后，PDO 里的数据发送一次。

当 CANopen 通讯配置为同步传输模式时，驱动器在收到同步报文后，才会上传 TPDO 里映射的数据。

TPDO设置 (专家模式)							
TPDO1	TPDO2	TPDO3	TPDO4	TPDO5	TPDO6	TPDO7	TPDO8
Index	Name		Value		Unit		
1	1A0000	TPDO1映射组	2		DEC		
2	1A0001	TPDO1映射1	60630020		HEX		
3	1A0002	TPDO1映射2	60410010		HEX		
4	1A0003	TPDO1映射3			HEX		
5	1A0004	TPDO1映射4			HEX		
6	1A0005	TPDO1映射5			HEX		
7	1A0006	TPDO1映射6			HEX		
8	1A0007	TPDO1映射7			HEX		
9	1A0008	TPDO1映射8			HEX		
10	180001	TPDO1站号	00000181		HEX		
11	180002	TPDO1传输类型	1		DEC		
12	180003	TPDO1禁止时间	0		DEC		
13	180005	TPDO1事件时间			DEC		

图 9 - 4 同步模式下 TPDO 配置

表 9 - 14 同步模式下 TPDO 配置

名称	含义
TPDO1 映射组	2, 本 PDO 中所配对象的个数, TPDO1 中配了实际位置、状态字 2 个对象
映射 1-8	配置伺服 CANopen 控制对象
TPDO1 站号	180+驱动器 ID (TPDO2 站号应设置为: 280+驱动器 ID)
TPDO1 传输类型	同步传输模式, 驱动器收到同步报文后向控制器发送 TPDO
TPDO1 禁止时间	一定设置为 0
注: TPDO1 中所配对象实际位置、状态字长度之和为 4+2=6 个字节	

➔

注意

TPDO1 中所配对象实际位置、状态字长度之和为 4+2=6 个字节。

RPDO 传输模式默认为 254, 不需要设置, 接受数据后立即生效。

默认同步报文的为

COB-ID	DLC
0x80	0

- 异步(传输类型: 254/255)

从站报文数据改变后即发送, 不管主站是否询问, 而且可以定义同一个报文两次发送之间的时间间隔, 避免高优先级报文一直占据总线 (PDO 的数值越低优先级越高)。

对于 FD1X5 和 iSMK 系列伺服驱动器, 它支持所有 256 种传输方式, 用户只需要根据控制器所支持的传输方式来选择驱动器的传输方式即可。

在异步传输模式下, PDO 里映射的对象数据一旦发生变化就会传输。

TPDO设置 (专家模式)

TPDO1	TPDO2	TPDO3	TPDO4	TPDO5	TPDO6	TPDO7	TPDO8	
Index		Name			Value		Unit	
1	1A0000	TPDO1映射组			2		DEC	
2	1A0001	TPDO1映射1			60630020		HEX	
3	1A0002	TPDO1映射2			60410010		HEX	
4	1A0003	TPDO1映射3					HEX	
5	1A0004	TPDO1映射4					HEX	
6	1A0005	TPDO1映射5					HEX	
7	1A0006	TPDO1映射6					HEX	
8	1A0007	TPDO1映射7					HEX	
9	1A0008	TPDO1映射8					HEX	
10	180001	TPDO1站号			00000181		HEX	
11	180002	TPDO1传输类型			254		DEC	
12	180003	TPDO1禁止时间			10		DEC	
13	180005	TPDO1事件时间			0		DEC	

图 9 - 5 异步传输模式下 TPDO 配置

表 9 - 15 异步传输模式下 TPDO 配置

名称	含义
TPDO1 映射组	代表该 PDO 中所配对象个数，TPDO1 中配了实际位置、状态字 2 个对象
映射 1-8	配置伺服 CANopen 控制对象
TPDO1 站号	180+驱动器 ID（TPDO2 站号应设置为：280+驱动器 ID）
TPDO1 传输类型	254 或 255，异步传输模式
TPDO1 禁止时间	单位为 ms，防止伺服发送报文过于频繁堵塞网络，多轴异步传输模式下根据实际需要设置



注意

TPDO1 中所配对象实际位置、状态字长度之和为 4+2=6 个字节。

RPDO 传输模式默认为 254，不需要设置，接受数据后立即生效。

一个 PDO 可以指定一个禁止时间，即定义两个连续 PDO 传输的最小间隔时间，避免由于高优先级信息的数据量太大，始终占据总线，而使其它优先级较低的数据无力竞争总线的问题。禁止时间由 16 位无符号整数定义，单位 1ms。

● 事件时间定时上报功能

在异步传输模式中，除了逢变即发，如果需要驱动器向控制器周期性上传数据，可以设置事件时间。

TPDO设置 (专家模式)

TPDO1	TPDO2	TPDO3	TPDO4	TPDO5	TPDO6	TPDO7	TPDO8
Index	Name		Value		Unit		
1	1A0000	TPDO1映射组	2		DEC		
2	1A0001	TPDO1映射1	60630020		HEX		
3	1A0002	TPDO1映射2	60410010		HEX		
4	1A0003	TPDO1映射3			HEX		
5	1A0004	TPDO1映射4			HEX		
6	1A0005	TPDO1映射5			HEX		
7	1A0006	TPDO1映射6			HEX		
8	1A0007	TPDO1映射7			HEX		
9	1A0008	TPDO1映射8			HEX		
10	180001	TPDO1站号	00000181		HEX		
11	180002	TPDO1传输类型	254		DEC		
12	180003	TPDO1禁止时间	0		DEC		
13	180005	TPDO1事件时间	50		DEC		

图 9 - 6 异步模式下使用事件时间定时上传

表 9 - 16 异步模式下使用事件时间定时上传

名称	含义
TPDO1 映射组	代表该 PDO 中所配对象的个数，TPDO1 中配了实际位置，状态字 2 个对象
映射 1-8	配置伺服 CANopen 控制对象
TPDO1 站号	180+驱动器 ID（TPDO2 站号应设置为：280+驱动器 ID）
TPDO1 传输类型	254 或 255，异步传输模式
TPDO1 禁止时间	使用事件时间上传时，该对象设置为 0
TPDO1 事件时间	驱动器向控制器发送 PDO 的周期时间，单位 ms
注：TPDO1 中所配对象实际位置、状态字长度之和为 4+2=6 个字节	

9.3.3.4 保护方式/监督类型说明

监督类型是指在运行过程中主站选择何种检查方式检查从站，通过这两种方式来判断从站是否出现故障，并根据这些故障做出相应的处理！

1、主站心跳报文

从站以“监督时间”周期性的上传报文到主站，如果超过“心跳消费者时间”后主站还没有收到从站的下一个心跳报文，那么主站判断通讯出错，主站产生报警！

表 9-17 从站上传心跳报文格式

COB-ID	Byte 0
0x700+Node_ID	状态
案例报文(从站 ID=1): 701 05	

2、从站心跳报文

主站以“监督时间”周期性的发送报文到从站，如果超过“心跳生产者时间”从站还没有收到主站的下一个心跳报文，那么从站判断通讯出错！当通讯中断模式（0x600700 设置）为 1，CAN 通讯出错时驱动器报警停机。

表 9-18 主站下发的心跳报文格式

COB-ID	Byte 0
0x700+主站 ID	主站状态
案例报文(主站 ID=127): 77F 05	

表 9-19 状态值含义

状态值	含义
0x00	启动(boot-up)
0x04	停止(Stopped)
0x05	运行(Operational)
0x7f	预操作(Pre-operational)

当一个 Heartbeat 节点启动后它的 Boot-up 报文是其第一个 Heartbeat 报文。



注意

心跳报文产生时间以及从站心跳报文由主站上电配置，默认断电不保存。

3、节点保护

主站以“监督时间”周期性的发送远程请求报文到从站，从站接收到后即回应，如果超过“监督时间*寿命因子”时间后，主站还没有收到从站回应的报文，主站判断从站出错。同时，从站也可以监控主站的远程请求状态，从收到的第一条远程帧开始启动通讯保护，如果超过“节点保护时间*节点保护系数”时间没有收到主站远程帧，从站也会判断通讯出错。需设置通讯中断模式（0x600700）为 1，CAN 通讯出错时驱动器才会报警停机。

主站请求报文格式——（0x700+节点号）（该报文无数据）

从站响应报文格式——（0x700+节点号）+状态

表 9-20 从站应答报文

COB-ID	Byte 0
0x700+Node_ID	Bit7:触发位 Bit6-Bit0:状态

表 9-21 从站应答报文状态值含义

状态值	含义
0	初始化(Initializing)
1	未连接(Disconnected)
2	连接(Connecting)
3	准备(Preparing)
4	停止(Stopped)
5	运行(Operational)
127	预操作(Pre-operational)

状态——数据部分包括一个触发位（bit7），触发位必须在每次节点保护应答中交替置“0”或者“1”。触发位在第一次节点保护请求时置为“0”。位 0 到位 6（bit0～6）则用来表示节点状态，数值含义如表 9-20 所示。

标准的 CAN 从站一般都只支持一种节点保护方式，FD1X5 和 iSMK 系列伺服驱动器两种保护方式都支持。但一个节点不能够同时支持节点保护和心跳报文，只能选其中一种作为保护。

9.3.3.5 启动过程说明

在网络初始化过程中，CANopen 支持扩展的 boot-up，也支持最小化 boot-up 过程。可以用节点状态转换图表示这种初始化过程，如图 9-7 所示。

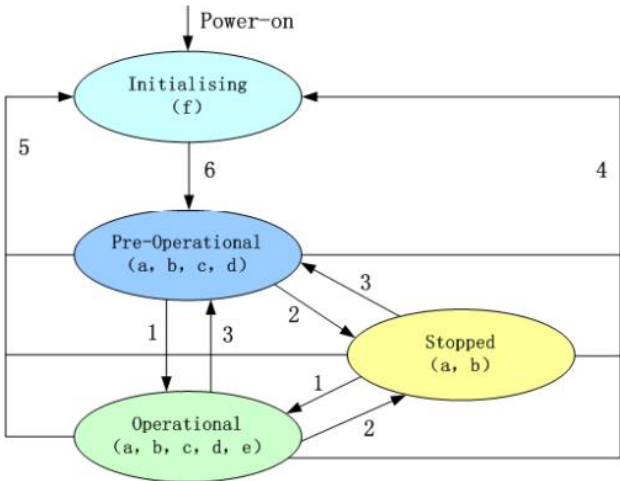


图 9-7 节点状态转换图

注意：图中括号内的字母表示处于不同状态时，可以使用的通讯对象。

其中：a——NMT d——Emergency b——Node Guard
 c——PDO c——SDO f——Boot-up

可以通过 NMT 管理报文来实现在各种模式之间切换，只有 NMT-Master 节点能够传送 NMT Module Control 报文，所有从设备都必须支持 NMT 模块控制服务，同时 NMT Module Control 消息不需要应答。设备在初始化结束后，自动进入 Pre_Operational 状态，发送 Boot-up 消息。NMT 消息格式如下：NMT-Master →NMT Slave(s)

表 9-22 NMT 管理报文格式

COB-ID	Byte0	Byte1
0x000	CS	Node-ID

当 Node-ID=0 时，表示所有的 NMT 从设备被寻址。CS 是命令字，其取值如表 9-23 所示。

表 9-23 CS 取值表

命令字	NMT 服务
0x01	开启远程节点
0x02	关闭远程节点
0x80	进入预操作状态
0x81	复位节点
0x82	复位通讯

9.3.3.6 应急报文说明

当设备内部出现致命错误将触发应急报文，由应用设备以最高优先级发送到其他设备。一条应急报文由 8 字节组成。

表 9-24 应急报文格式

COB-ID	Byte 0-1	Byte2	Byte4-5	Byte6-7
紧急报文站号 0x101400	应急错误代码 0x603F00	错误寄存器(0x100100)	错误状态 0x260100	错误状态 0x260200

表 9-25 应急错误代码 0x603F00

报警内容	应急错误代码(Hex)	报警内容	应急错误代码(Hex)
通讯式编码器没有连接	0x7331	电流传感器故障	0x5210
通讯式编码器多圈错误	0x7320	软件看门狗复位	0x6010
通讯式编码器校验错误	0x7330	异常中断	0x6011
驱动器温度过高	0x4210	MCU 故障	0x7400
驱动器总线电压过高	0x3210	电机型号配置错误	0x6320
驱动器总线电压过低	0x3220	电机动力线缺相	0x6321
驱动器功率部分短路或电机短路	0x2320	预使能报警	0x5443
电流采样饱和	0x2321	正限位报错	0x5442
驱动器制动电阻异常	0x7110	负限位报错	0x5441
实际跟随误差超过允许	0x8611	SPI 故障	0x6012
逻辑电压低	0x5112	总线通讯错误	0x8100
电机或驱动器过载	0x2350	总线通讯超时	0x81FF
输入脉冲频率过高	0x8A80	全闭环检查错误	0x8A81
电机温度过高	0x4310	主编码器 ABZ 故障	0x7382
通讯式编码器没有回应	0x7331	主编码器计数错误	0x7306
EEPROM 数据错误	0x6310		

表 9-26 错误寄存器

Bit	错误类型
0	一般错误
1	电流
2	电压
3	温度
4	通讯错误
5	设备配置文件特定
6	编码器
7	保留

9.4 CANopen 总线通信设置

本章节将介绍 CAN 总线通讯参数的设置，在上位机软件界面中点击**专家模式->ECAN 配置->ECAN 设置**进入参数设置界面。具有网络管理功能的主站上电会通过发送 SDO 的方式来初始化从站的参数，一般情况下同步 ID、节点保护时间、节点保护时间系数、节点保护站号、紧急报文站号、心跳报文产生时间等参数不需要用户自己设定。

表 9 - 27 CANopen 相关参数

CANopen 地址	参数名称	含义	默认值
10050020	同步 ID	传输类型为 1-240,同步模式时有用，异步模式时不需要设置。	80
100C0010	节点保护时间	通过节点保护主站可以监视每个节点当前的状态，主站以节点保护时间为周期发送远程帧（默认 COBID 为 0x700+站号，不含内容的报文）询问从节点状态，从节点需要在一定时间范围内做出回应，否则主节点认为从节点掉线，驱动器进入报警状态。	1000
100D0008	节点保护时间系数		3
100E0020	节点保护 ID	700+驱动器 ID	
10140020	紧急报文站号	80+驱动器站号	
10170010	心跳报文产生时间	从节点周期发送报文给主节点，主节点超过一定的时间未收到该报文就认为从节点掉线	
2F810008	CAN 波特率	CAN 波特率设置 100: 1M 50: 500k 25: 250k 12: 125k 5: 50k 1: 10k	50
30110108	ECAN 同步周期	插补模式下根据主站同步报文周期设置，异步模式不需要设置。 0:1ms 1:2ms 2:4ms 3:8ms	2
30110208	ECAN 同步时钟模式	插补模式时设成 1 开启同步时钟，非插补模式时设成 0 关闭同步时钟	0

CANopen 地址	参数名称	含义	默认值
30110410	ECAN 同步 丢失计数	同步模式下监控通讯状态，数值不发送变化说明通讯状况良好，若数值不断变化说明有干扰或同步周期设置不正确。	
60070010	通讯中断模式	CAN 通讯中断模式，决定驱动器在超过节点保护时间*节点保护系数的时间后仍然没有收到节点保护报文的动作逻辑 0：不处理 1：报错	0

● 通过 PLC 初始化配置 PDO 参数

对于可以导入 EDS 文件的 CANopen 主站来说不需要在伺服内部进行 PDO 设置,可直接在主站配置 PDO 信息，上电后 PLC 初始化会发送 SDO 报文来配置伺服的 PDO，配置完成后主站会发送启动报文启动从站，然后就可以进行 PDO 通讯了，多数 PLC 都可以采用这种方式，例如：施耐德 PLC，西门子 S7-1200+CM CANOPEN 模块，步科 F1 等。

➔

Note

EDS 文件下载地址：http://download.kinco.cn/D_Software/Servo/EDS.zip

9.5 基于 CANopen 的插补模式

插补模式适用于单轴运算控制以及多轴的同步控制，由主站进行运动轮廓规划，目标位置在同步模式下周期性更新。在上位机软件界面中点击**专家模式**->**ECAN 配置**->**ECAN 设置**进入插补模式参数设置界面。

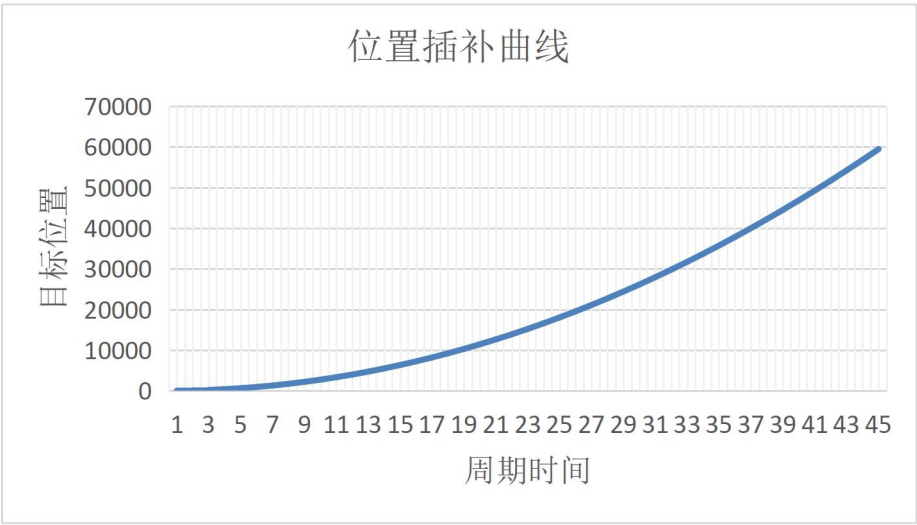


图 9-8 位置插补曲线

表 9-28 插补控制相关参数

内部地址	参数名称	含义	设置值
60600008	工作模式	设定工作模式为插补模式	7
60400010	控制字	0x1F 使能驱动器, 运行插补模式 0x06 松轴, 关闭驱动器使能 0x86 复位驱动器故障	1F 6 86
607A0020	目标位置	目标绝对/相对位置	用户定义
30110108	ECAN 同步周期	插补模式下根据主站同步报文周期设置 <div> <div>设置值</div> <div>同步周期</div> <div>0 1ms</div> <div>1 2ms</div> <div>2 4ms</div> <div>3 8ms</div> </div>	用户定义
30110208	ECAN 同步时钟模式	插补模式时设成 1 开启时钟同步, 非插补模式时设成 0 关闭时钟同步。	0
30110410	ECAN 同步丢失计数	插补模式下监控同步通讯状态, 数值不断变化说明通讯有干扰或同步周期与主站设置不一致	/

● CAN 通讯中断报警功能

关于通讯中断报警功能, 需要设置以下参数

表 9-29 通讯中断报警功能设置

CANopen 地址	参数名称	含义	默认值
100C0010	节点保护时间	通过节点保护主站可以监视每个节点当前的状态, 主站以节点保护时间为周期发送远程帧(默认 COBID 为 0x700+站号, 不含内容的报文) 询问从节点状态, 从节点需要在一定时间范围内做出回应, 否则主节点认为从节点掉线, 驱动器进入报警状态。	1000
100D0008	节点保护时间系数		3
100E0020	节点保护 ID	700+驱动器 ID	
10140020	紧急报文站号	80+驱动器站号	
60070010	通讯中断模式	CAN 通讯中断模式, 决定驱动器在超过节点保护时间*节点保护系数的时间后仍然没有收到节点保护报文的动作逻辑 0: 不处理 1: 报错	0

第 10 章 报警排除

10.1 错误和历史错误

错误：点击“驱动器”→“故障显示”，或点击按钮（错误发生时变为），错误窗口会弹出，并显示最近一次的错误信息。可根据第 10 章 10.2 章节查看出现故障的原因进行故障排除。

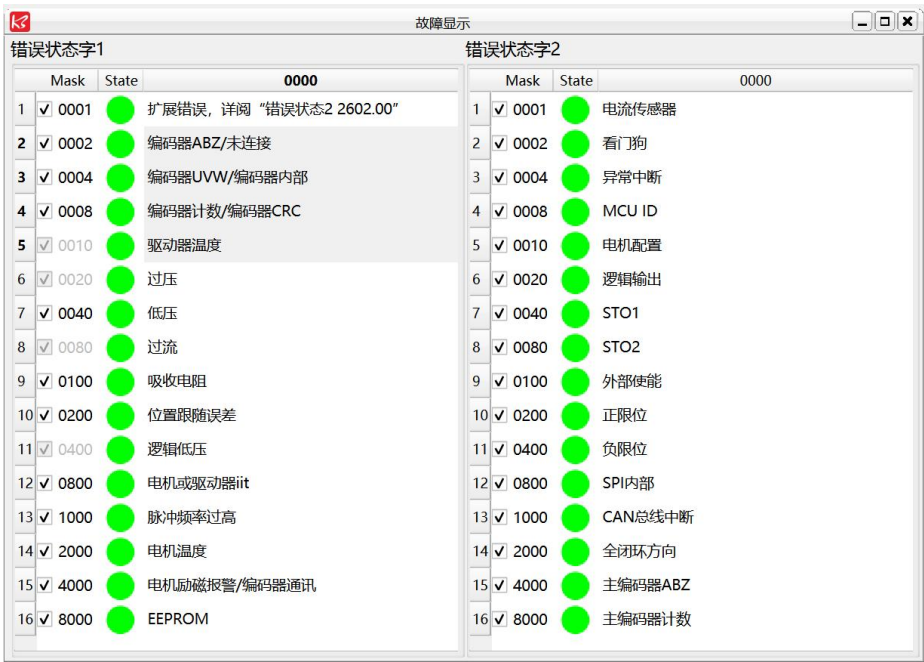


图 10-1 错误显示界面

历史错误：点击菜单栏“驱动器”→“历史故障”，历史错误窗口会弹出，并显示最近 8 次错误信息，包括错误字、总线电压、速度、电流、温度、工作模式、功率管状态。最新的历史故障显示在第一行。

历史故障								
	Code	DC V	RPM	Ap	°C	OperationMode	PWM State	Time Min
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0

图 10-2 历史错误显示界面

表 10-1 错误状态（2601.00）信息

位	错误名称	错误码	描述
0	扩展错误		参考错误状态 2 定义（2602.00）
1	编码器通信错误	0x7331	通讯式编码器未连接
2	编码器内部故障	0x7320	编码器内部故障
3	编码器 CRC 错误	0x7330	编码器通讯受到干扰
4	驱动器温度过高	0x4210	散热器温度过高
5	驱动器总线电压过高	0x3210	总线电压过高
6	驱动器总线电压过低	0x3220	总线电压过低
7	驱动器输出短路	0x2320	驱动器功率管或电机短路
8	驱动器制动电阻异常	0x7110	制动电阻过载
9	实际跟随误差超过允许	0x8611	实际跟随误差超过设置的最大跟随误差
10	逻辑电压低	0x5112	逻辑电源电压过低
11	电机或驱动器 Ilt 故障	0x2350	电机或驱动器功率管 Ilt 故障
12	输入脉冲频率过高	0x8A80	脉冲输入频率过高
13	电机温度过高	0x4310	电机温度传感器报警
14	编码器信息错误	0x7331	未连接编码器或编码器通讯超时
15	EEPROM 数据错误	0x6310	EEPROM 数据校验错误

表 10-2 错误状态 2（2602.00）信息

位	错误名称	错误码	描述
0	电流传感器故障	0x5210	电流传感器信号偏移或波纹过大
1	看门狗报错	0x6010	软件看门狗异常
2	异常中断	0x6011	异常中断
3	MCU 故障	0x7400	MCU 型号错误
4	电机配置错误	0x6320	EEPROM 中没有电机配置信息/电机从未配置
5~7	保留		
8	预使能报警	0x5443	输入口定义预使能，在驱动器使能或将要使能时，预使能输入口无有效输入
9	正限位报错	0x5442	正限位（找原点后有效），正限位报错仅在限位功能定义（2010.19）设为 0 时才会产生故障报警
10	负限位报错	0x5441	负限位（找原点后有效），负限位报错仅在限位功能定义（2010.19）设为 0 时才会产生故障报警
11	SPI 故障	0x6012	内部固件错误或 SPI 处理错误
12	CAN 总线中断	0x8100	通讯中断模式(6007.00)设置为 1 时才会产生故障报警
13	全闭环故障	0x8A81	全闭环工作状态下，主编码器计数方向与电机编码器计数方向相反
14	主编码器 ABZ 故障	0x7382	主编码器连接错误
15	主编码器计数错误	0x7306	主编码器索引信号异常



提醒

在每个错误项旁边都有一个屏蔽（掩码）选择框，默认为全部勾选，☒ 表示可以被取消，☐ 表示不可被取消。未被勾选 ☐ 表示对应的错误项将会被忽略。错误掩码也可以通过对象错误掩码（2605.01）和错误掩码 2（2605.04）进行设置（见表 10-4）。

表 10-3 错误扩展（2605.07）信息

位	错误名称	错误码	描述
0	原点记录错误	0x5210	电流传感器信号偏移或波纹过大
1	内部制动电阻过温	0x7111	内部制动电阻实际功率过大
2	内部制动电阻短路	0x7112	内部制动单元损坏，制动电路短路
3	电机缺相	0x6321	电动力线 UVW 中某相未连接
4	ADC 采样饱和	0x2321	电流采样 ADC 达到极限，电流失控
12	服务超时	0x81FF	通讯总线错误扩展

表 10-4 错误掩码

内部地址	数据类型	名称	描述	默认值
2605.01	Uint16	错误掩码	错误状态（2601.00）的掩码。位 = 0 代表相应的错误项将会被忽略。	0xFFFF
2605.02	Uint16	错误保存掩码-使能	驱动器处于使能状态时，错误状态（2601.00）在故障历史中的掩码。位=0 代表相应的错误项不会被保存在故障历史中。	0xFBFF
2605.03	Uint16	错误保存掩码-未使能	驱动器处于使能关闭状态时，错误状态（2601.00）在故障历史中的掩码。位=0 代表相应的错误项不会被保存在故障历史中。	0x0000
2605.04	Uint16	错误掩码 2	错误状态 2（2602.00）的掩码。位= 0 代表相应的错误项将会被忽略。	0xFFFF
2605.05	Uint16	错误保存掩码-使能 2	驱动器处于使能状态时，错误状态 2（2602.00）在故障历史中的掩码。位=0 代表相应的错误项不会被保存在故障历史中。	0xF1FF
2605.06	Uint16	错误保存掩码-未使能 2	驱动器处于使能关闭状态时，错误状态 2（2602.00）在故障历史中的掩码。位=0 代表相应的错误项不会被保存在故障历史中。	0x003F

10.2 报警原因及处理措施

当驱动器报警时，驱动器红色 ERR 灯将会常亮。

若想查询更详细的错误信息和错误历史记录，请用 RS485 串口通讯线（或 usb 调试线）将驱动器连接到电脑上位机查看。

表 10 - 5 错误状态字 1 报警代码

报警代码	报警信息	报警原因	处理措施
FFF.F	未配置电机/电机配置不正确	1、驱动器未配置电机。 2、当前电机型号与驱动器保存的电机型号不同	1、驱动器出厂参数中不含电机配置，连接电机编码器线后即可自动识别电机参数。 2、连接上位机软件检查当前电机型号与铭牌中的 motor code 是否匹配；当前电机型号和实际连接电机不同时，请重新自识别电机参数。
000.1	扩展错误	错误状态字 2 报警	查看错误状态字 2，根据报警代码检查错误含义。
000.2	编码器通讯故障	编码器线出错或未连接；与 400.0 报警同时出现表示通讯式编码器无回应	步骤 1：确认驱动器与电机间编码器接线正确，线缆连接牢固。 步骤 2：通过上位机软件检查当前电机型号与铭牌中的 motor code 是否匹配。当前电机型号和实际连接电机不同时，请参考 4.2 章节操作重新自识别电机参数。

000.4	编码器内部故障	多圈绝对值编码器的多圈数据无效，需复位	<p>步骤 1:通过上位机软件修改通讯式编码器数据复位 (0x269000)数据为 10。</p> <p>步骤 2: 复位故障或重启驱动器。如果错误仍然存在，请检查电池线是否连接牢固，可尝试更换电池线或电机比对。</p>
000.8	编码器 CRC 错误	电机型号设置错误； 编码器接线错误或外部干扰造成；	<p>1、电机配置是否正确</p> <p>步骤 1: 通过上位机软件检查当前电机代码与所连的电机铭牌中的 motor code 是否一致。</p> <p>步骤 2: 确认驱动器与电机间编码器接线正确，线缆连接牢固。</p> <p>2、检查驱动器接地线是否连接良好（不同于电机 PE 线）</p> <p>3、检查整个设备的接地线是否连接良好。</p> <p>4、使用独立的电源给驱动器供电。</p>
001.0	驱动器温度过高	驱动功率模块的温度到达报警值	<p>步骤 1: 检查电机以及驱动器功率是否满足要求。</p> <p>步骤 2: 重启驱动器，检查上电过程中散热风扇是否能正常开启。检查驱动器外壳中的散热孔是否堵塞。</p> <p>步骤 3: 环境温度超过 40°工况下，电气柜内需做好散热措施或降额使用。</p> <p>步骤 4: 适当增加驱动器安装距离</p> <p>步骤 5: 驱动器内部功率电路损坏，更换驱动器。</p>
002.0	总线电压过高	直流总线电压超过过压报警点	<p>步骤 1: 检查动力电源电压是否在驱动器输入电压范围内，电源电压是否稳定。</p> <p>步骤 2: 高速制动场合产生的能量将反馈到驱动器母线电容中，可考虑连接合适的制动电阻，将多余的能量通过制动电阻来吸收消耗。</p> <p>步骤 3: 制动电阻阻值太大会导致母线能量泄放不及时，需降低制动电阻阻值。</p> <p>注: iSMK 系列不支持外接制动电阻，FD1X5 系列制动电阻推荐范围请查看 3.2.2.1 章节</p>
004.0	总线电压过低	直流总线电压低于低压报警点	<p>步骤 1: 检查动力电源电压是否在驱动器输入电压范围内，电源电压是否稳定。</p> <p>步骤 2: 先上动力电源再上逻辑电源，确保使能前驱动器已经通上动力电源。</p>
008.0	驱动器输出短路	瞬时电流超过了过流保护值	<p>步骤 1: 检查电机配置参数与电机图纸是否一致。</p> <p>步骤 2: 检查动力线 UVW 间是否存在短路，可尝试更换动力线缆/电机/驱动器比对。</p> <p>步骤 3: 干扰导致过流报警，参考 3.4 设置 EMC 条件。</p>
010.0	制动电阻异常	外部制动电阻过载；	<p>步骤 1: 通过上位机检查外部制动电阻阻值和制动电阻功率是否设置正确。</p> <p>步骤 2: 制动电阻功率不足，更换更大功率的制动电阻。</p>
		内部制动电阻过温；	驱动器内部制动电阻的功率不足，请断开 RB+与 RB-端的短接线，在 RB+与 RB-端连接合适的外部制动电阻。
		内部制动单元损坏，制动电路短路；	驱动器断电后用万用表测量 DC-与 RB-端，如果导通，则需更换驱动器。

020.0	实际跟随误差超过允许	实际跟随误差超过了设定的最大跟随误差值, 可能的原因有: 1.电机线接线不正确 2.控制环刚性太小 3.最大跟随误差设置值太小 4.目标速度超过最大速度限制 5.目标扭矩限制值过小 6.抱闸未打开 7.机械装置卡塞/摩擦力	步骤 1: 确保正确连接动力线 UVW 步骤 2: 重新调整增益, 加大刚性, 刚性调整方法请参考第五章 步骤 3: 重新调整最大跟随误差(0x606500) 步骤 4: 重新调整最大速度限制(0x607f00) 步骤 5: 重新调整目标电流限制(0x607300), 目标电流限制默认值等于电机最大电流(0x64100B) 步骤 6: 测量抱闸线连接是否正常, 测量抱闸电压, 更换电机比对测试 步骤 7: 消除机械卡塞问题, 加涂润滑剂
040.0	逻辑电源过低	逻辑电压低于报警下限	检查逻辑电压是否在驱动器输入电压范围内, 电源电压是否稳定
080.0	电机或驱动器 IIT 错误	电机或驱动器长时间过载运行, 可能的原因有: 1.电机动力线以及编码器接线错误 2.电机轴旋转时, 抱闸未松放 3.驱动器控制环参数设置不当 4.机械装置被卡住或摩擦力过大 5.驱动器/电机选型错误无法满足应用要求	步骤 1: 检查确认编码器/动力线连接正确 步骤 2: 测量抱闸线连接是否正常, 测量抱闸电压, 更换电机比对测试 步骤 3: 重新调整控制环参数 步骤 4: 消除机械卡塞问题, 加涂润滑剂 步骤 5: 减小负载运行或更换更大功率的产品
100.0	频率过高	外部脉冲输入频率过高	1、降低脉冲频率 2、使用超过 600KHz 脉冲时, 需要增加脉冲频率控制 0x250808 的值。
200.0	电机温度过高	输入口定义并触发电机故障	用户自定义报警
400.0	编码器信息错误	编码器线出错或未连接; 与 000.2 报警同时出现表示通讯式编码器无回应	步骤 1: 确认驱动器与电机间编码器接线正确, 线缆连接牢固。 步骤 2: 通过上位机软件检查当前电机型号与铭牌中的 motor code 是否匹配。当前电机型号和实际连接电机不同时, 请设置正确的电机参数。
800.0	EEPROM 错误	EEPROM 读写数据时, 数据损坏	步骤 1: 初始化控制环参数后存储控制参数, 重启驱动器 步骤 2: 通过上位机软件导入 cdi 文件

表 10-6 错误状态字 2 报警代码

报警代码	报警信息	报警原因	处理措施
000.1	电流传感器故障	电流传感器偏移或纹波过大	步骤 1: 电流传感器受外部干扰导致报警, 请参考 3.4 设置 EMC 条件。 步骤 2: 电流传感器电路损坏, 更换驱动器比对
000.2	看门狗报错	软件看门狗异常	初始化控制环参数后存储控制参数重启驱动器
000.4	异常中断	中断异常或中断无效	初始化控制环参数后存储控制参数重启驱动器
000.8	MCU 故障	1.软件程序与硬件不匹配 2.检测到 MCU 错误	步骤 1: 检查驱动器属性中的软件版本, 更新正确的软件 步骤 2: 更换驱动器比对

001.0	电机配置错误	1.无法自动识别电机型号 EEPROM 无电机数据或者 电机没有被正确配置 2.电机线缺失,动力线 UVW 某相未连接	步骤 1: 连接上位机软件检查当前电机型号与铭牌中的 motor code 是否匹配; 当前电机型号和实际连接电机不 同时, 请重新设置正确的电机参数。 步骤 2: 确认电机动力线 UVW 连接正确且连接可靠 步骤 3: 更换动力线或驱动器比对测试。
010.0	预使能报警	DIN 配置了预使能功能, 需 确认预使能信号有效输入再 使能驱动器	用户自定义报警
020.0	正限位报错	正限位信号被触发, 仅在限 位功能定义 (0x201019) 设 为 0 时才会产生故障报警	用户自定义报警
040.0	负限位报错	负限位信号被触发, 仅在限 位功能定义 (0x201019) 设 为 0 时才会产生故障报警	用户自定义报警
080.0	SPI 故障	内部固件在处理 SPI 时出错	请联系供货商
100.0	CAN 总线故障	通讯中断模式 (0x600700) 设为 1 时才会开启	用户自定义报警
200.0	全闭环故障	全闭环模式下, 主编码器计 数方向与电机编码器计数方 向相反	步骤 1: 检查主编码器计数方向与电机编码器计数方向, 计数方向不一致请更改主编码器计数方向(0x250A03) 步骤 2: 主编码器速度与电机编码器速度比例关系不一致。 检查机械安装, 排除机械打滑或卡塞, 检查全闭环主轴检 查(0x250A05)和全闭环从轴检查(0x250A06)以及全闭环 比例检测(0x250A09)设置值。
400.0	主编码器 ABZ 故障	主编码器连接错误	检查主编码器信号线连接是否正确
800.0	主编码器计数错误	主编码器索引信号异常	步骤 1: 正确填写主编码器周期(0x250A01), 设置为 0 不 开启检查。 步骤 2: 排查干扰

表 10-7 性能异常及解决方案

异常现象	报警原因	处理措施
电机空载运行出现 异响或震荡	1、接线错误 2、控制环参数设置不当 3、电机油封安装不当或电机故障	使用标准的动力线, 编码器线以及控制线, 确保线缆接线正确且无 松动现象, 线缆无破损。 初始化控制参数—存储控制参数重启后再尝试运行。 手扭动电机轴承检查电机是否异常, 尝试重新安装电机油封。
定位不准	1、控制环参数不当 2、齿轮前脉冲数据与控制器下发 的脉冲个数不一致 3、电子齿轮比设置过大 4、往复运动累积误差 5、机械系统因素	根据第五章节调整驱动器控制环参数 脉冲线使用带屏蔽的双绞线缆, 线缆较长时需适当加大控制线的线 径。脉冲线尽可能远离电源线, 驱动器与电机接地良好。 恢复出厂齿轮比运行。 在工艺允许的条件下将机械回原, 在累积误差超过允许前搜索原点。 检查联轴设备是否安装牢固, 滑轮或齿轮咬合良好, 负载惯量是否 过大, 尝试减轻负载或更换更大功率的电机等。

附录一 配置第三方电机的方法

名称	参数	备注
电机极对数	必填	请参考电机图纸
电机最大电流	必填	请参考电机图纸
相电感	必填	请参考电机图纸
相电阻	必填	请参考电机图纸
反向电动势	必填	请参考电机图纸
扭矩系数	必填	请参考电机图纸
转子惯量	必填	请参考电机图纸
抱闸占空比	参考备注	90%
抱闸延时	参考备注	150ms
电机型号	参考备注	FD1x5 驱动器目前只支持多摩川编码器协议，设置如下： 多摩川：VX
使用内部电机库	参考备注	设置为 1 或者 2 皆可
反馈类型	参考备注	根据编码器反馈类型，设置对应值。 Bit0: UVW 接线检查 Bit2: 通讯式接线检查 Bit4: ABZ 接线检查 Bit5: 省线式编码器 多摩川编码器设置为：04
反馈精度	参考备注	编码器反馈精度 增量式：编码器的线数×4 通讯式：单圈分辨率低于 16 位设置为实际的编码器分辨率； 单圈分辨率高于 16 位一般设置为 65536，防止位置溢出。
反馈周期	参考备注	增量式：电机转一圈的脉冲数 通讯式：定义编码器类型，例如：多摩川编码器的多圈 16 位， 单圈 17 位，反馈周期设置为 0x1617
励磁模式	参考备注	模式 0 或者模式 10
励磁电流	参考备注	比额定电流小，一般为电机额定电流的 0.75 倍
励磁时间	参考备注	默认值 2000ms
电机 iit 电流	参考备注	一般设定为额定电流
电机 iit 时间	参考备注	一般为 60 秒
电机电流环带宽	参考备注	默认值 2000Hz



注意

- 在填完以上的表格后，进入我们的上位机软件，找到电机配置选型，把信息依次填入，最后保存电机参数，初始化控制环参数，重启驱动器。
- 上电发现电机没有任何报警，即可开始尝试运行其他控制模式。
- 如果有报警，即对照对应驱动器手册，逐步排查。
- 电机试运行前，需先限制目标电流，防止驱动器输出电流过大。
- 如果发现电机没有任何报警，电机也不转，先检查分辨率有没有设置正确，其次检查 UVW 线有没有接反，有时由于各厂家规范的不同，可能需要对调 UV 线，最后检查励磁模式是否设置正确。

附录二 常用公式

小车行走电机的选型方式，适用于电机+减速机+轮子的机构

公式: $T \cdot n = \mu \cdot m \cdot g \cdot d / 2$	
轮子的直径 d	m
减速机的减速比 n	1: n
电机的扭矩 T	Nm, kgm ² /s ²
整车载重能力 m	kg
摩擦系数 μ	无单位
重力加速度 g	m/s ²

脉冲模式下，脉冲数与机械位移之间的关系

公式: $N \cdot A / B = s \cdot n \cdot r / P$	
齿轮比分子 A	无单位
齿轮比分母 B	无单位
丝杆螺距 P	mm
电机单圈脉冲数 r	无单位
减速比 1: n	无单位
机械位移 s	mm
脉冲数 N	无单位

转速和线速度的关系

公式: $n = v \div r \div \pi$	
转速 n	rpm
线速度 v	mm/s
半径 r	mm

参数名称	工程单位	内部单位	换算关系
速度	rpm	DEC	$DEC = [(RPM \cdot 512 \cdot \text{编码器分辨率}) / 1875]$
加速度		DEC	$DEC = [(RPS / S \cdot 65536 \cdot \text{编码器分辨率}) / 4000000]$
电流	A	DEC	$1Arms = [2048 / (\text{驱动器峰值电流 } I_{peak} / 1.414)]dec$

例如:

速度工程单位是 rpm，内部单位是 dec，两者关系是 1RPM 约等于 2730dec（编码器分辨率 10000）！假设需要速度为 10rpm，那么用通讯控制时需写入速度为 27300dec，16 进制为 6AA4。

电流工程单位是 Arms，内部单位是 dec，假设使用的驱动器为 FD125（驱动器峰值电流 I_{peak} 为 48A），那么 1Arms 约等于 60dec，若需要电流为 10Arms，则用通讯控制时需写入电流为 600d。