



试用水印

AIMOTOR CANOPEN 总线型驱动器使用说明

简易版

2025 年 5 月 9 日

V3.2 版

目录

AIMOTOR CANOPEN 总线型驱动器使用说明	1
一、产品简介	1
1.1 产品概述	1
1.2 产品特性	1
1.3 组网方案	2
二、对象字典的参数说明与设置	3
2.1 对象字典	3
2.1.1 CIA301 子协议通讯参数	4
2.1.2 厂家自定义参数	6
2.1.3 Cia402 轴控参数	8
2.2 各控制模式以及关联对象字典	11
2.3 Cia402 对象字典位置单位计算	13
2.4 6040h 控制字位定义	14
2.5 6041h 状态字位定义	14
2.6 控制指令与状态跃迁	15
2.7 6040h 控制命令步骤	16
2.8 6098h 原点搜索方式	17
2.9 607Eh 指令方向极性	18
2.10 60FDh 数字量输入状态监视	18
2.11 60FEh 数字量输出状态监视与强制	18
三、SDO 控制报文参考	19
3.1 速度模式控制	19
3.2 位置模式控制	20
3.3 回原模式控制	21
3.3 转矩模式控制	22
四、PDO 控制模式	23
4.1 PDO 映射配置流程图	23
4.2 PDO 通讯和映射对象	24
4.3 PDO 配置过程参考例	24
4.4 同步报文配置描述例	27
4.5 映射对象长度对应表	27
五、故障查询	28
5.1 故障码表	28
5.2 故障查询方法	29

一、产品简介

1.1 产品概述

AIMOTOR 驱动器采用 CANOPEN 总线通讯接口，将 CANOPEN 从站技术、矢量控制技术、内置微细分技术、自适应滤波技术、闭环控制技术融为一体，实现了系统的实时控制与实时数据传输，优化了电机的性能：具有极佳的平稳性和超低噪声；平滑、精确的纯正弦电流矢量控制技术有效的减小了电机发热。

1.2 产品特性

- 新一代 32 位 ARM 技术，性价比超高、平稳性佳、噪声、振动性能优越
- 采用 CANOPEN 从站技术，支持 CIA301 以及 CIA402 子协议，支持 CSP、PV、PP、PT、HM 模式
- 用户可以通过 CANOPEN 总线或者 RS485 串口通讯的方式设置系统参数。

CANOpen 协议的特性:

开放性: CANOpen 是一个公开的标准，任何公司都可以使用它来开发产品，无需支付许可费用。

网络管理: CANOpen 协议定义了网络管理功能，如节点自诊断、网络监控等，使得网络维护更加方便。

对象字典: CANOpen 使用对象字典来描述设备的功能和状态，使得设备之间的通信更加标准化。

服务: CANOpen 定义了一系列的服务，如 NMT (Network Management)、TPDO (Time-Triggered Protocol Data Object)、RPDO (Real-Time Protocol Data Object) 等，以满足不同的应用需求。

PDO (Process Data Object) : PDO 是 CANOpen 中用于快速传输过程数据的数据对象，可以实时地传输控制和状态信息。

SDO (Service Data Object) : SDO 用于访问设备的对象字典，可以读取或写入设备的状态和配置信息。

紧急消息: CANOpen 还定义了紧急消息，用于在发生严重故障时立即通知网络中的其他设备。

心跳消息: 心跳消息用于监测网络中设备的运行状态，如果一段时间内没有收到某个设备的心跳消息，网络管理系统会认为该设备已经离线。



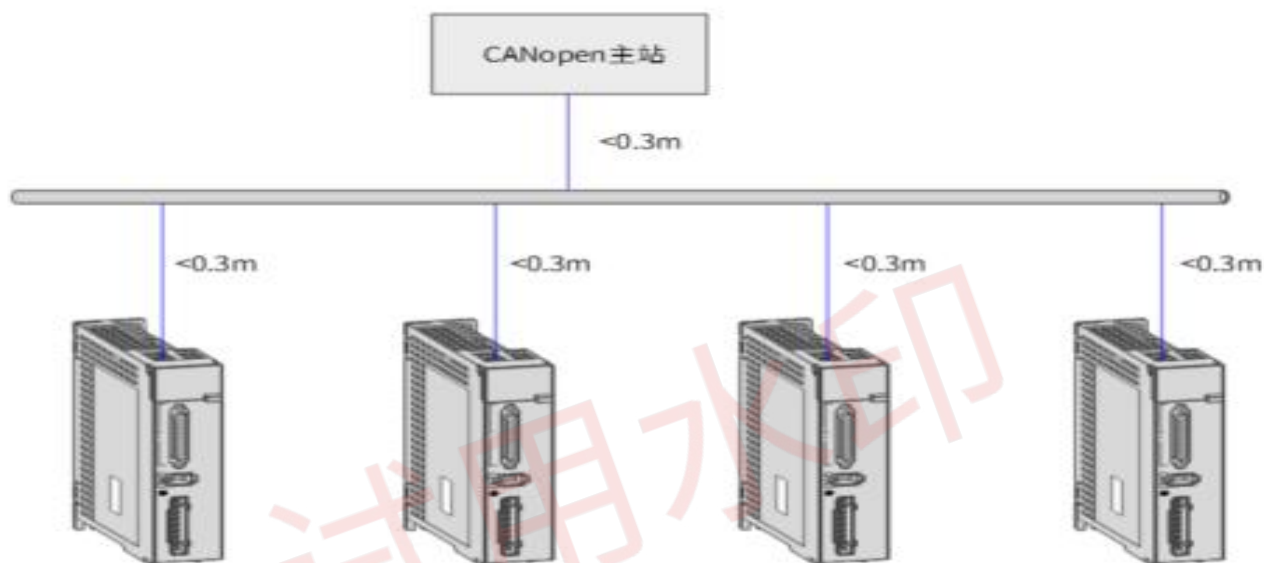
图<canopen 设备模型>

1.3 组网方案

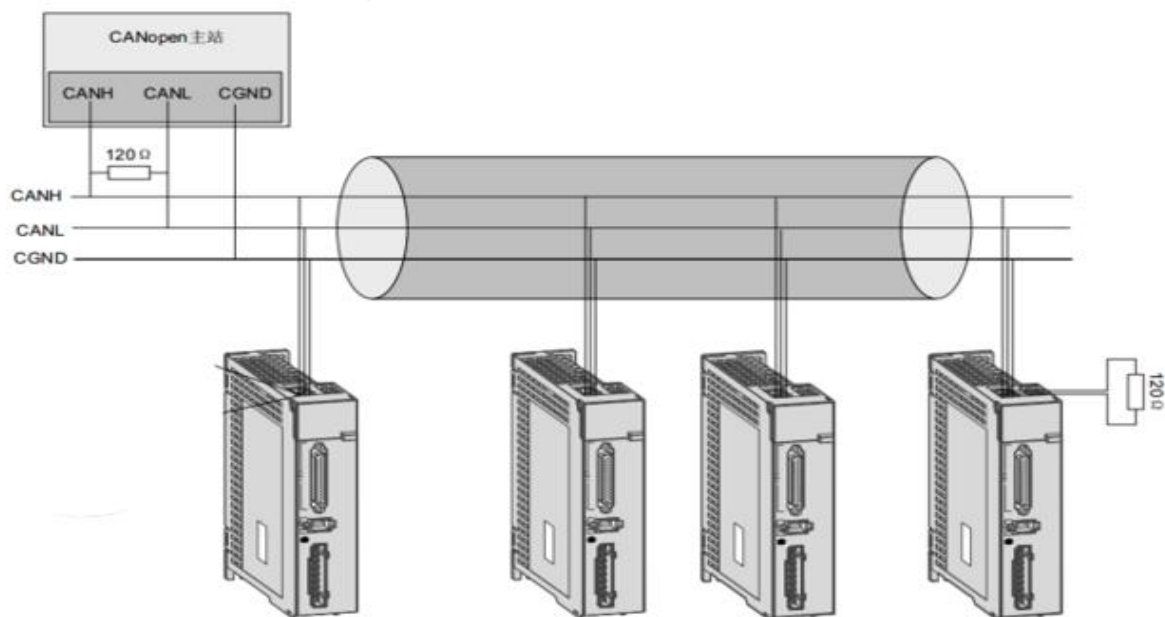
CAN 总线的传输距离与波特率、通信线缆有直接关系，最大总线线路长度与波特率关系如下表所示：

本机出厂默认 NodeId=1，速率=500kbps，如果想更改，请使用 AIMotor 系列 RS485 调试上位机更改 H0C_00 与 H0C-08 参数：

传输距离	通讯速率	节点数	双绞线线径
25m	1Mbps	64	0.2mm ²
95m	500kbps	64	0.3mm ²
560m	100kbps	64	0.5mm ²
1100m	50kbps	64	0.75mm ²



图<组网方案 1>



图<组网方案 2>

二、对象字典的参数说明与设置

2.1 对象字典

对象字典是设备规范中最重要的部分。它是一组参数和变量的有序集合，包含了设备描述及设备网络状态的所有参数。通过网络可以采用有序的预定义的方式来访问的一组对象。

CANopen 协议采用了带有 16 位索引和 8 位子索引的对象字典，对象字典的结构如右图所示。

0000h-0FFFh 用户无访问意义；

1000h-1FFFh 按照 CIA301 协议定义通信对象参数及设备类型等；

2000h-5FFFh 制造商定义对象区，我司将该区域定义了 AIMOTOR 驱动器所有参数功能码对象；

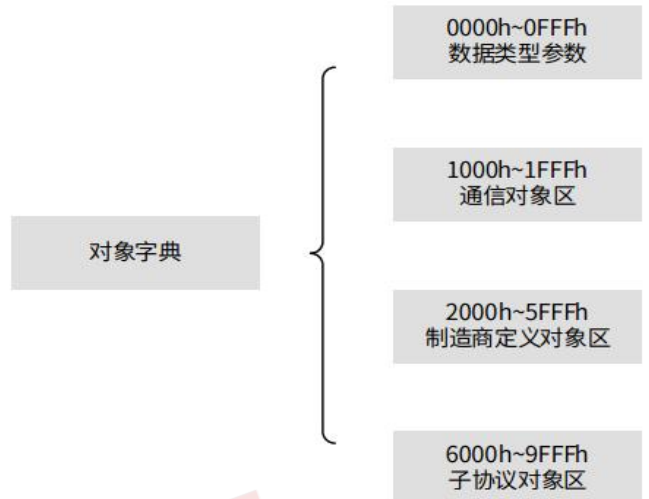
6000h-9FFFh 按照 CIA402 协议定义的一些常用 CIA402 控制对象；

AIMOTOR 驱动器参数功能码与对象字典的对应关系如下：

对象字典索引 = $0x2000 + \text{功能码组号}$ ；对象字典子索引 = $\text{功能码组内偏置的十六进制} + 1$

驱动器参数功能码与对象字典的映射关系举例：

参数功能码			对应的对象字典地址	
参数功能码名	功能码组号(16 进制)	组内偏置 (10 进制)	索引(16 进制)	子索引(16 进制)
H02-00	0x02	00	0x2002	0x01
H03-09	0x03	09	0x2003	0x0A
H0B-26	0x0B	26	0x200B	0x1B



2.1.1 CIA301 子协议通讯参数

索引	子索引	名称	说明	类型	属性	默认值	范围
1000	0	设备类型	与 CIA 规则一致	Uint32	RO	-	-
1001	0	错误寄存器	按位来表示错误类型信息;				
1003	0	当前存在的报警数量	1、00h 表示当前记录报警数量, 只可写 0 清除记录;	Uint8	RW	0	0~9
	1-0A	历史报警缓存条目	2、01h-0Ah 历史报警缓存条目数据: bit00-bit15, 标准错误码 bit16-bit23 厂家自定义错误码	Uint32	RO	-	0~0xFFFFFFFF
1005	0	同步报文 COB-ID	周期同步功能下使用	Uint32	RW	0x80	0~0xFFFFFFFF
1006	0	同步循环周期 us		Uint32	RW	0	0~0xFFFFFFFF
100A	0	软件版本	AIMOTOR_CIA402	str	RO	-	-
100C	0	节点守护时间	节点守护功能下使用	Uint16	RW	0	0~32767
100D	0	寿命因子		Uint8	RW	0	0~255
1010	00	子索引个数	暂无作用, 保留;	Uint8	RO	4	-
	01-04	保存全部参数	用户期望保存设置的厂家参数可操作 H0C_13 参数 (索引 200Ch-子索引 0E) 写 1 触发; CIA301, CIA402 参数不支持保存, 需用户按需配置	Uint32	RW	0	0~0xFFFFFFFF
1011	00	子索引个数	暂无作用, 保留;	Uint8	RO	4	-
	01-04	恢复全部参数	用户期望恢复厂家参数出厂值可操作 H02_31 参数 (索引 2002h-子索引 20) 写 1 触发; 站号默认 1, CAN 速率默认 500K	Uint32	RW	0	0~0xFFFFFFFF
1014	0	紧急报文 COB-ID	紧急报文标识符	Uint32	RW	0x80	0~0xFFFFFFFF
1016	0	子索引个数	设置消费者心跳时间	Uint8	RO	1	-
	1	消费者心跳时间	bit0-bit15, 监视心跳超时时间 ms bit16-bit23 监视的节点 id	Uint32	RW	0	0~0xFFFFFFFF
1017	0	生产者心跳时间	设置生产者心跳时间 (本机产生心跳的间隔时间) ms	Uint32	RW	0	0~0xFFFFFFFF
1200	0	子索引个数	设置 SDO 通讯 COB-ID	Uint8	RO	2	-
	1	客户端到服务器 COB-ID		Uint32	RW	0x600+NodeID	0~0xFFFFFFFF
	2	服务器到客户端 COB-ID		Uint32	RW	0x580+NodeID	0~0xFFFFFFFF

索引	子索引	名称	说明	类型	属性	默认值	范围
1400	0	子索引个数	设置 RPDO1 通讯参数	Uint8	RO	6	-
	1	RPDO1 的 COB-ID		Uint32	RW	0x200	0~0xFFFFFFFF
	2	RPDO1 的传输类型		Uint8	RW	1	0~255
1401	0	子索引个数	设置 RPDO2 通讯参数	Uint8	RO	6	-
	1	RPDO2 的 COB-ID		Uint32	RW	0x300	0~0xFFFFFFFF
	2	RPDO2 的传输类型		Uint8	RW	1	0~255
1402	0	子索引个数	设置 RPDO3 通讯参数	Uint8	RO	6	-
	1	RPDO3 的 COB-ID		Uint32	RW	0x400	0~0xFFFFFFFF
	2	RPDO3 的传输类型		Uint8	RW	1	0~255
1403	0	子索引个数	设置 RPDO4 通讯参数	Uint8	RO	6	-
	1	RPDO4 的 COB-ID		Uint32	RW	0x500	0~0xFFFFFFFF
	2	RPDO 4 的传输类型		Uint8	RW	1	0~255
1600	0	RPDO1 映射的个数	设置 RPDO 组 1 映射对象; 出厂默认映射如下: 子索引 1 映射 6040 控制字; 子索引 2 映射 6060 控制模式; 子索引 3 映射 607A 目标位置; 子索引 4-8 未映射可按需分配;	Uint8	RO	8	-
	1	RPDO1 映射对象 1		Uint32	RW	0x60400010	0~0xFFFFFFFF
	2	RPDO1 映射对象 2		Uint32	RW	0x60600008	0~0xFFFFFFFF
	3	RPDO1 映射对象 3		Uint32	RW	0x607A0020	0~0xFFFFFFFF
	4-8	RPDO1 映射对象 4-8		Uint32	RW	0x00000000	0~0xFFFFFFFF
1601	0	RPDO2 映射的个数	设置 RPDO 组 2 映射对象; 出厂默认映射如下: 子索引 1 映射 60FF 目标速度; 子索引 2 映射 6071 目标转矩; 子索引 3-8 未映射可按需分配;	Uint8	RO	8	-
	1	RPDO2 映射对象 1		Uint32	RW	60FF0020	0~0xFFFFFFFF
	2	RPDO2 映射对象 2		Uint32	RW	60710010	0~0xFFFFFFFF
	3-8	RPDO2 映射对象 3-8		Uint32	RW	0x00000000	0~0xFFFFFFFF
1602	00	RPDO3 映射的个数	设置 RPDO 组 3 映射对象; 子索引 1-8 未映射可按需分配;	Uint8	RO	8	-
	01-08	RXPDO3 映射对象 1-8		Uint32	RW	0x00000000	0~0xFFFFFFFF
1603	0	RPDO4 映射的个数	设置 RPDO 组 3 映射对象; 子索引 1-8 未映射可按需分配;	Uint8	RO	8	-
	01-08	RXPDO 映射对象 1-8		Uint32	RW	0x00000000	0~0xFFFFFFFF

索引	子索引	名称	说明	类型	属性	默认值	范围
1800	0	子索引个数	设置 TPDO1 通讯参数	Uint8	RO	6	-
	1	TPDO1 的 COB-ID		Uint32	RW	0x180	0~0xFFFFFFFF
	2	TPDO1 的传输类型		Uint8	RW	1	0~255
1801	0	子索引个数	设置 TPDO2 通讯参数	Uint8	RO	6	-
	1	TPDO2 的 COB-ID		Uint32	RW	0x280	0~0xFFFFFFFF
	2	TPDO2 的传输类型		Uint8	RW	1	0~255
1802	0	子索引个数	设置 TPDO3 通讯参数	Uint8	RO	6	-
	1	TPDO3 的 COB-ID		Uint32	RW	0x380	0~0xFFFFFFFF
	2	TPDO3 的传输类型		Uint8	RW	1	0~255
1803	0	子索引个数	设置 TPDO4 通讯参数	Uint8	RO	6	-
	1	TPDO4 的 COB-ID		Uint32	RW	0x480	0~0xFFFFFFFF
	2	TPDO4 的传输类型		Uint8	RW	1	0~255
1A00	0	TPDO1 映射的个数	设置 TPDO 组 1 映射对象; 出厂默认映射如下: 子索引 1 映射 6041 状态字; 子索引 2 映射 6061 显示模式; 子索引 3 映射 6064 实际位置; 子索引 4-8 未映射可按需分配;	Uint8	RO	8	-
	1	TPDO1 映射对象 1		Uint32	RW	0x60410010	0~0xFFFFFFFF
	2	TPDO1 映射对象 2		Uint32	RW	0x60610008	0~0xFFFFFFFF
	3	TPDO1 映射对象 3		Uint32	RW	0x60640020	0~0xFFFFFFFF
	4-8	TPDO1 映射对象 4-8		Uint32	RW	0x00000000	0~0xFFFFFFFF
1A01	0	TPDO2 映射的个数	设置 TPDO 组 2 映射对象; 出厂默认映射如下: 子索引 1 映射 606C 实际速度; 子索引 2 映射 6077 实际转矩; 子索引 3-8 未映射可按需分配;	Uint8	RO	8	-
	1	TPDO2 映射对象 1		Uint32	RW	606C0020	0~0xFFFFFFFF
	2	TPDO2 映射对象 2		Uint32	RW	60770010	0~0xFFFFFFFF
	3-8	TPDO2 映射对象 3-8		Uint32	RW	0x00000000	0~0xFFFFFFFF
1A02	00	TPDO3 映射的个数	设置 TPDO 组 3 映射对象; 子索引 1-8 未映射可按需分配;	Uint8	RO	8	-
	01-08	TPDO3 映射对象 1-8		Uint32	RW	0x00000000	0~0xFFFFFFFF
1A03	0	TPDO4 映射的个数	设置 TPDO 组 3 映射对象; 子索引 1-8 未映射可按需分配;	Uint8	RO	8	-
	01-08	TPDO4 映射对象 1-8		Uint32	RW	0x00000000	0~0xFFFFFFFF

2.1.2 厂家自定义参数

2000h-5FFFh 制造商定义对象区, 我司将该区域定义了 AIMOTOR 驱动器所有参数功能码对象;

每一个参数都有对应的字典对象索引与子索引, AIMOTOR 驱动器参数功能码与对象字典的对应关系如下:

对象字典索引= 0x2000 + 功能码组号; 对象字典子索引= 功能码组内偏置的十六进制+ 1;

举例: 驱动器参数功能码与对象字典的映射关系

参数功能码			对应的对象字典地址	
参数功能码名	功能码组号(16 进制)	组内偏置 (10 进制)	索引(16 进制)	子索引(16 进制)
H02-00	0x02	00	0x2002	0x01
H03-09	0x03	09	0x2003	0x0A
H0B-26	0x0B	26	0x200B	0x1B
H12-20	0x12	20	0x2012	0x15

由于参数功能码过多，所以不在 CANOPEN 说明书列举，请参阅同系列标准脉冲产品版说明书参数说明。

- ◆ 使用 SDO 通讯对厂家自定义参数对象进行读写操作时，请注意参数的属性；
 - 1、 如果修改方式属性为<使能断开>参数，则需要先断开电机使能然后再写入的参数对象值；
 - 2、 如果出厂值属性为<注册值>参数，用户只能读取不可写入；
- ◆ H00 组、H01 组、H0B 组参数，用户只能读取不可写入；

◆ SDO 通讯错误中止码

中止码	标识符	说明
0x00000000	OD_SUCCESSFUL	SDO 通讯成功
0x06010001	OD_READ_NOT_ALLOWED	试图读取只写对象
0x06010002	OD_WRITE_NOT_ALLOWED	试图写入只读对象
0x06020000	OD_NO_SUCH_OBJECT	对象在字典中不存在
0x06040041	OD_NOT_MAPPABLE	对象不能映射在 PDO
0x06060000	OD_ACCESS_FAILED	由于硬件错误，访问失败
0x06070010	OD_LENGTH_DATA_INVALID	数据类型不匹配、服务参数长度不匹配
0x06090011	OD_NO_SUCH_SUBINDEX	子索引不存在
0x06090030	OD_VALUE_RANGE_EXCEEDED	超出参数值范围（仅限写访问）
0x06090031	OD_VALUE_TOO_LOW	超出参数值太高
0x06090032	OD_VALUE_TOO_HIGH	超出参数值太低
0x05030000	SDOABT_TOGGLE_NOT_ALTERNED	切换位不交替
0x05040000	SDOABT_TIMED_OUT	SDO 协议通讯超时
0x05040001	SDOABT_CS_NOT_VALID	命令说明符无效
0x05040002	SDOABT_INVALID_BLOCK_SIZE	块大小无效（仅块模式）
0x05040005	SDOABT_OUT_OF_MEMORY	内存溢出
0x08000000	SDOABT_GENERAL_ERROR	SDO 报文大小错误
0x08000021	SDOABT_LOCAL_CTRL_ERROR	由于本地控制，数据不能传输或存储到应用程序中

2.1.3 Cia402 轴控参数

索引	子索引	名称	说明			类型	属性	默认值	范围
603F	0	故障码	驱动器当前产生的故障码, 详见章节‘5.0 故障码信息表’描述;			Uint16	RO	0	0~65535
6040	0	控制字	详见章节‘2.4 6040 控制字’描述;			Uint16	RW	0	0~65535
6041	0	状态字	详见章节‘2.5 6041 状态字’描述;			Uint16	RO	-	0~65535
605A	0	快速停机代码	值	减速度来源	减到 0 后	Uint16	RW	1	0~65535
			0	自由停机	解除使能				
			1	0x6084	解除使能				
			2	0x6085	解除使能				
			3	0x6084	保持锁轴				
			4	0x6085	保持锁轴				
6060	0	运行模式设置	0: 无作用 1: PP(轮廓位置模式) 3: PV(轮廓速度模式) 4: PT(轮廓转矩模式) 6: HM(回原点模式) 8: CSP(循环同步位置模式)			Int8	RW	1	0~8
6061	0	运行模式状态	查询 6060H 的状态; 0: 无作用 1: PP(位置模式) 3: PV(速度模式) 4: PT(轮廓转矩模式) 6: HM(回原点模式) 8: CSP(循环同步位置模式)			Int8	RO	-	0~8
60FC	0	内部指令位置	驱动器内部生成的位置指令; 单位: Enc			Int32	RO	-	-2147483647~ 2147483647
6062	0	内部指令位置	驱动器内部生成的位置指令; 单位: Pul			Int32	RO	-	-2147483647~ 2147483647
6063	0	实际位置	电机实际位置, 单位: Enc			Int32	RO	-	-2147483647~ 2147483647
6064	0	实际位置	电机实际位置, 单位: Pul			Int32	RO	-	-2147483647~ 2147483647
606B	0	内部指令速度	驱动器内部生成的速度指令; 单位: Pul/s			Int32	RO	-	-2147483647~ 2147483647
606C	0	实际速度	电机实际速度, 单位: Pul/s			Int32	RO	-	-2147483647~ 2147483647
6071	0	目标转矩	设置 PT(轮廓转矩模式)运行转矩; (1000 表示额定转矩) 单位: 0.1%			Int16	RW	0	-3000~3000

索引	子索引	名称	说明	类型	属性	默认值	范围
6074	0	内部指令转矩	驱动器内部生成的转矩指令; 单位: 0.1%	Int16	RO	-	-3000~3000
6077	0	实际转矩	电机实际转矩, 单位: 0.1%	Int16	RO	-	-3000~3000
6078	0	实际电流	电机实际电流, 单位: 0.01A	Int16	RO	-	-32767~32767
607A	0	目标位置	设置 PP(轮廓位置模式)运行位置; 单位: Pul	Int32	RW	0	-2147483647~ 2147483647
607C	0	原点偏移	设置 HM(回原点模式)原点偏移 量; 单位: Pul	Int32	RW	0	-2147483647~ 2147483647
607E	0	指令极性	设置 PP/PV/PT 轮廓模式下指令方 向逻辑;	Uint8	RW	0	0~255
607F	0	最大轮廓速度	设置 PT(轮廓转矩模式)运行最大 速度限制; 单位: Pul/s	Uin32	RW	0	-2147483647~ 2147483647
6081	0	轮廓速度	设置 PP 轮廓位置模式最大速度; 单位: Pul/s	Int32	RW	0	-2147483647~ 2147483647
6083	0	轮廓加速度	设置 PP、PV 轮廓模式加速度; 单位: Pul/s ²	Int32	RW	100000	-2147483647~ 2147483647
6084	0	轮廓减速度	设置 PP、PV 轮廓模式减速度; 单位: Pul/s ²	Int32	RW	100000	-2147483647~ 2147483647
6085	0	快速停机减速度	设置 PP/PV/PT 轮廓模式下 快速停机减速度; 单位: Pul/s ²	Int32	RW	500000	-2147483647~ 2147483647
6098	0	回原点方式	目前可设置回原点方式的值为: 01/02/04/06/17/18/24/28/33/34/35 /37/38; 详见章节‘2.8 回原点模式 方法’内的描述;	Int8	RW	35	0~38
6099	0	子索引个数	-	Uint8	RO	2	0~255
	1	寻原点速度 1	HM(回原点模式)接近原点速度:高 速找原点; 单位: Pul/s	Int32	RW	10000	-2147483647~ 2147483647
	2	寻原点速度 2	HM(回原点模式)确定原点速度:低 速找原点; 单位: Pul/s	Int32	RW	1000	-2147483647~ 2147483647
609A	0	回原点加减速速度	HM(回原点模式)的回原点加减速 度; 单位: Pul/s ²	Int32	RW	10000	-2147483647~ 2147483647
60FD	0	数字量信号输入状态	DI1-DI7 输入端口状态反馈 2 进制 bit15-bit21 位表示	Uint32	RO	-	0~127

索引	子索引	名称	说明	类型	属性	默认值	范围
60FE	0	子索引个数	-	Uint8	RO	2	0~255
	1	数字量信号输出	详见章节‘2.10 60FEh’内的描述；	Uint32	RW	-	-2147483647~ 2147483647
	2	数字量信号强制掩码		Uint32	RW	0	-2147483647~ 2147483647
60FF	0	目标速度	设置 PV(轮廓速度模式) 运行速度；单位：Pul/s	Int32	RW	0	-2147483647~ 2147483647
6502	0	本驱动支持的运行模式	用于查询产品支持的 CIA402 运 动控制模式	Uint32	RO	0x00AD	0~0xFFFFFFFF

试用水印

2.2 各控制模式以及关联对象字典

控制模式	索引+子索引	名称	数据类型	读写权限	单位	是否必须
同步位置模式(CSP)	6060	控制模式设置=8	Int8	RW	-	√
	6040	控制字	Uint16	RW	-	√
	607A	目标位置	Int32	RW	Pul	√
	6061	控制模式反馈	Int8	RO	-	
	6041	状态字	Uint16	RO	-	√
	6064	实际位置	Int32	RO	Pul	√
	606C	实际速度	Int32	RO	Pul/s	
	6077	实际转矩	Int16	RO	0.1%	
轮廓位置模式(PP)	6060	控制模式设置=1	Int8	RW	-	√
	6040	控制字	Uint16	RW	-	√
	607A	目标位置	Int32	RW	Pul	√
	6081	最大速度	Int32	RW	Pul/s	√
	6083	加速度	DINT	RW	Pul/s ²	√
	6084	减速度	DINT	RW	Pul/s ²	√
	6061	控制模式反馈	Int8	RO	-	
	6041	状态字	Uint16	RO	-	√
	6064	实际位置	Int32	RO	Pul	
	606C	实际速度	Int32	RO	Pul/s	
	6077	实际转矩	Int16	RO	0.1%	
轮廓速度模式(PV)	6060	控制模式设置=3	Int8	RW	-	√
	6040	控制字	Uint16	RW	-	√
	60FF	目标速度	DINT	RW	Pul/s	√
	6083	加速度	DINT	RW	Pul/s ²	√
	6084	减速度	DINT	RW	Pul/s ²	
	6061	控制模式反馈	Int8	RO	-	
	6041	状态字	Uint16	RO	-	
	6064	实际位置	Int32	RO	Pul	
	606C	实际速度	Int32	RO	Pul/s	
	6077	实际转矩	Int16	RO	0.1%	

控制模式	索引+子索引	名称	数据类型	读写权限	单位	是否必须
轮廓转矩模式(PT)	6060	控制模式设置=4	Int8	RW	-	√
	6040	控制字	UInt16	RW	-	√
	6071	目标转矩	Int16	RW	0.1%	√
	607F	最大速度限制	UInt32	RW	Pul/s	√
	6061	控制模式反馈	Int8	RO	-	
	6041	状态字	UInt16	RO	-	
	6064	实际位置	Int32	RO	Pul	
	606C	实际速度	Int32	RO	Pul/s	
	6077	实际转矩	Int16	RO	0.1%	
回原点模式(HM)	6060	控制模式设置=6	Int8	RW	-	√
	6040	控制字	UInt16	RW	-	√
	6098	回零方法	Int8	RW	-	√
	6099:01	寻原点高速 1	Int32	RW	Pul/s	√
	6099:02	寻原点低速 2	Int32	RW	Pul/s	√
	609A	原点加速度	Int32	RW	Pul/s ²	√
	607C	原点偏移	Int32	RW	Pul	
	6061	控制模式反馈	Int8	RO	-	
	6041	状态字	UInt16	RO	-	
	6064	实际位置	Int32	RO	Pul	
	606C	实际速度	Int32	RO	Pul/s	
	6077	实际转矩	Int16	RO	0.1%	
其他辅助关联参数	605A	快速停机代码	UInt16	RW	-	
	6085	急停减速度	UInt32	RW	Pul/s ²	
	607E	指令极性	UInt8	RW	-	
	60FD	数字量输入状态	UInt32	RO	-	
	60FE	数字量输出状态	UInt32	RO	-	

- ◆ 无论采用哪种控制模式对从站进行控制，都离不开对 6040H(控制字)和 6041H(状态字)两个对象字典的读写操作，主从站以这两个对象字典作为传输媒介实现指令下发和状态监视。2.4-2.5 小节重点介绍这两个对象字典各个位的定义和含义。

2.3 Cia402 对象字典位置单位计算

CIA402 协议中速度值不是我们理解的表示每分钟转速单位 r/min 而是每秒指令脉冲数单位 Pul/s; 加减速单位是 Pul/s^2 ;

如果我想要设置每分钟的转速, 应该先将分钟转为秒, 再乘以电机 1 圈脉冲数单位即可;

AIMotor 系列驱动出厂默认 1000 脉冲电机旋转 1 圈; 如果更改 1 圈脉冲数请使用调试软件或 SDO 通讯方式设置 H05-07 电子齿轮分子和 H05-09 电子齿轮分母, 如设置 2000 脉冲电机旋转一圈, 则设置 H05-09 参数值 2000 即可;

单位转换示意表

1 圈脉冲数	用户理解的单位				转换成遵循 CIA402 协议规定的单位			
H05-07 分子 H05-09 分母	目标位置 r	目标速度 r/min	加速到目标的 加速时间	减速到 0 的减 速时间	目标位置 Pul	目标速度 Pul/s	加速度 Pul/s^2	减速度 Pul/s^2
出厂默认 1000	运行到 100.5 转	每分钟 2000 转	2 秒加速到每 分钟 2000 转	3 秒减速到完 全停止	100.5×1000 =100500	$2000/60 \times 1000$ =33333	$33333/2$ =16666	$33333/3$ =11111

举例: 假设控制模式为 PP 轮廓位置模式下, 如上表, 根据 1 圈脉冲数和用户期望的目标位置、目标转速和加减速时间, 计算 CIA402 里的位置值、速度值和加减速值, 然后使用 SDO 通讯控制电机绝对定位运行的具体步骤;

Step1: 使用上位机将驱动器控制模式参数 H02-00 设为 8 (CanOpen 控制模式);

Step2: 使用 SDO 通讯将 CIA402 对象字典里的 6060h 运动模式写入 1 (轮廓位置模式);

Step3: 使用 SDO 通讯将 CIA402 对象字典里的轮廓位置模式相关对象-目标位置 607Ah 写入 100500; 目标速度 6081h 写入 33333; 加速度 6083h 写入 16666; 减速度 6084h 写入 11111;

Step4: 使用 SDO 通讯将 CIA402 对象字典里的 6040h 控制字顺序依次写入 00h→06h→07h→0Fh→1F 电机启动运行, 当运行到 100500 位置电机自动停止。

2.4 6040h 控制字位定义

位	0	1	2	3	4-6	7	8	9-15
功能	进入伺服准备好	进入接通主回路	执行快速停止	执行使能运行	运行模式相关	故障复位	暂停	NA

其他位的补充说明：

位 2：快速停止触发逻辑是 0 有效，**注意与其他触发的逻辑区分开**；

位 4：在 pp 和 hm 模式下触发逻辑是上升沿有效，pp 为触发更新后位置运行，hm 为触发回原点运行；

位 7：从机故障复位触发逻辑是上升沿有效；

位 8：从机运行暂停触发逻辑是上升沿有效；

◆ **控制字的每一个 bit 位单独赋值无意义，必须与其他位共同构成某一控制指令（参考 2.6 控制指令与状态跃迁）。**

◆ bit0~bit3 和 bit7 在各伺服模式下意义相同，必须按顺序发送命令，才可将伺服驱动器按照 CiA402 状态机切换流程引导入预计的状态，每一命令对应唯一——确定的状态。

◆ bit4~bit6 与各伺服模式相关（请查看不同模式下的控制指令状态跃迁）。

2.5 6041h 状态字位定义

位	0	1	2	3	4	5	6	7/8	9	10	11	12-13	14	15
功能	已准备好	等待使能	已使能运行	发生故障	已上电	快速停止激活	不可运行	N A	远程控制	目标到达	N A	运行模式相关	N A	原点已找到

其他位的补充说明：

当驱动器投入电源后位 4 将置位。

◆ **状态字的每一个 bit 位单独读取无意义，必须与其他位共同组成，反馈伺服当前状态（参考 2.6 控制指令与状态跃迁）。**

◆ bit0~bit9 在各伺服模式下意义相同，控制字 6040h 按顺序发送命令后，伺服反馈唯一确定的状态。

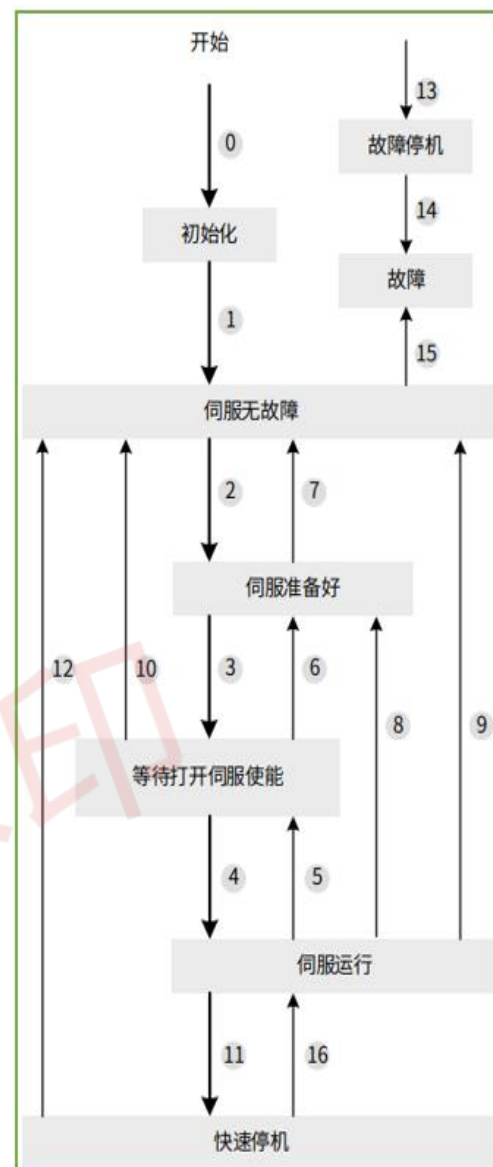
◆ bit12~bit13 与各伺服模式相关（请查看不同模式下的控制指令）。

◆ bit10、bit15 在各伺服模式下意义相同，反馈伺服执行某伺服模式后的状态。

2.6 控制指令与状态跃迁

CIA402 状态跃迁图

CIA402 状态切换		控制字 6040	状态字 6041
标号	过程迁移	控制命令	反馈 bit0-bit9
0	上电→初始化	自动过渡, 无需命令	0x0000
1	初始化→伺服无故障	1、如果初始化结束检测到故障直接进入 13, 2、如果初始化结束无故障, 自动过渡, 无需命令	0x0250
2	伺服无故障→伺服准备好	0x06	0x0231
3	伺服准备好→等待打开伺服使能	0x07	0x0233
4	等待打开伺服使能→伺服运行	0x0F	0x0237
5	伺服运行→等待打开伺服使能	0x07	0x0233
6	等待打开伺服使能→伺服准备好	0x06	0x0231
7	伺服准备好→伺服无故障	0x00	0x0250
8	伺服运行→伺服准备好	0x06	0x0231
9	伺服运行→伺服无故障	0x00	0x0250
10	等待打开伺服使能→伺服无故障	0x00	0x0250
11	伺服运行→快速停机	0x02	0x0217
12	快速停机→伺服无故障	快速停机完成后自动过渡, 无需命令	0x0250
13	→故障停机	除故障状态外, 其它状态下, 驱动一旦检测到报警, 自动切换到故障停机状态, 无需命令	0x021F
14	故障停机→故障	故障停机完成后自动过渡, 无需命令	0x0218
15	故障→伺服无故障	0x80	0x0250
16	快速停机→伺服运行	快速停机完成后, 发送命令 0x0F	0x0237



- ◆ 用户必须遵循状态跃迁表对 CANOPEN 电机进行控制;
- ◆ 控制字 6040h 每一个 bit 位单独赋值无意义, 必须与其他位共同构成某一控制指令。
- ◆ 状态字 6041h 的 bit0~bit9 在伺服任何模式下意义相同, bit10~bit15(除 bit11 bit14 无意义) 与各伺服模式运行状态有关, 在上表中均以 “0” 表示;

2.7 6040h 控制命令步骤

注意：使用 CanOpen 控制需要使用我司调试 AIMOTOR 软件将 H02-00 控制模式参数设 8（CanOpen 控制），一旦设为 CanOpen 控制必须遵循 CanOpen 控制要求,其对应的对象字典生效；其它控制模式无效；

顺序控制步骤		步 0	步 1	步 2	步 3	步 4	步 5	步 6	步 7
模式	控制字	预备工作	初始	准备好	等使能	使能	启动运行	变位/速	正常停止
CSP 循环同步位置模式	6040	建立通讯 分配 PDO 映射 激活主站循环同步	00h	06h	07h	0Fh	主站发送位置下发指令	主站控制	主站位置指令 停止下发
PP 轮廓位置模式	6040	建立通讯状态 设置 PP 模式相关的运动参数	00h	06h	07h	0Fh	绝对定位非立即更新 0Fh→1Fh		位置到达自动停止 无需控制命令
						2Fh	绝对定位立即更新 2Fh→3Fh		
						4Fh	相对定位非立即更新 4Fh→5Fh		
						6Fh	相对定位立即更新 6Fh→7Fh		
PV 轮廓速度模式	6040	建立通讯状态 设置 PV 模式相关的运动参数	00h	06h	07h	0Fh	使能后即启动运行	变更速度即可	目标速度对象 索引 60FFh 写 0
PT 轮廓转矩模式	6040	建立通讯状态 设置 PT 模式相关的运动参数	00h	06h	07h	0Fh	使能后即启动运行	变更转矩即可	最高速度对象 索引 607Fh 写 0
HM 回原点模式	6040	建立通讯状态 设置 HM 模式相关的运动参数	00h	06h	07h	0Fh	0Fh→1Fh 启动原点搜索	-	1:原点找到自动停止 2:1Fh→0Fh 停止

- ◆ CSP 循环同步位置模式模式，由主控单元按约定的同步周期进行位置下发，电机根据下发的位置点和同步帧进行执行位移，电机运行的轮廓曲线（如位置、速度等）由主控单元事先规划计算；
- ◆ PP/PV/PT 轮廓模式，主控单元可以使用 SDO 通讯方式设定好模式必须的参数，根据模式控制命令步骤控制电机运行，电机运行的轮廓曲线由电机根据设定的参数值（如加减速速度，运行速度等）进行规划计算。

◆ 注 1：运行过程中非正常停止方式

切换状态方式：将控制字 6040h 写 0h 或 6h 或 7h，CIA402 控制状态不再处于运行状态，电机将解除使能并按照惯性自由停止；

快速停机方式：将控制字 6040h 写 2h，电机将按照 605Ah 对象索引里选择的减速度方式和停止后状态；

减速暂停方式：将控制字 6040h 写 10Fh，电机将按照 605Ah 对象索引里选择的减速度方式和停止后状态；

2.8 6098h 原点搜索方式

设定值	初始方向	减速点	原点	过程步骤
24	正向	原点开关	原点开关	电机首先以设定的方向高速搜索原点开关，遇到原点开关信号上升沿开始低速运行脱离，脱离到原点开关信号下降沿电机换向并以继续低速搜索原点开关信号上升沿，遇到原点开关信号上升沿立即停止，回零成功。
28	反向	原点开关	原点开关	
34	正向	Z 信号	电机 Z 信号	电机首先以设定的方向高速搜索 Z 相，遇到电机 Z 相信号上升沿开始低速换向运行，遇到 Z 相另一侧上升沿信号立即停止，回零成功。
33	反向	Z 信号	电机 Z 信号	
4	正向	原点开关	电机 Z 信号	电机首先以设定的方向高速搜索原点开关，遇到原点开关信号上升沿开始低速换向运行脱离，脱离到原点开关信号下降沿再次换向低速搜索原点开关信号上升沿，遇到原点开关信号上升沿后运行到 Z 相信号立即停止，回零成功。
6	反向	原点开关	电机 Z 信号	
18	正向	正向超程	正向超程	电机首先以设定的方向高速搜索原点开关，遇到正向超程开关信号上升沿开始低速换向脱离，脱离到正向超程开关信号下降沿电机再次换向低速搜索正向超程开关信号上升沿，遇到正向超程开关信号上升沿立即停止，回零成功。
17	反向	反向超程	反向超程	
2	正向	正向超程	Z 信号	电机首先以设定的方向高速搜索正向超程开关，遇到正向超程开关信号上升沿开始减速换向运行脱离，脱离到正向超程开关信号下降沿继续运行到电机 Z 相信号立即停止，回零成功。
1	反向	反向超程	Z 信号	
37	正向	机械极限位	机械极限位	电机首先以设定的方向和力矩（H05_58 设定的力矩）低速搜索机械极限位置，遇到机械极限位置抵住堵转且转矩到达触停回零转矩限制上限且保持默认时间电机立即停止，回零成功。
38	反向	机械极限位	机械极限位	
35	-	-	-	将当前位置设为原点

注意：

- ◆ 原点搜索方式中有用到的外部开关信号，请关联选择的对应的 DI 功能选项，否则将发生报警 ER.601 回零失败；
- ◆ 更换原点搜索方式请在解除电机使能的情况下修改回原点模式 6098h 对象参数；
- ◆ 选择的模式中如果减速点为原点开关，并且关联正反超程开关 DI，电机在搜索路径中碰到超程开关电机自动换向继续搜索；
- ◆ 过程步骤中高速搜索速度 6099 索引的子索引 1 对象设置，低速搜索速度由 6099 索引的子索引 2 设置；
- ◆ 如果在 H05_35 参数时间内仍未找到原点，将发生报警 ER.601 回零超时；
- ◆ 原点复归成功后，DO 功能（OutFun16-HomeAttain）原点回零完成输出有效，使能 OFF 后原点回零完成输出无效；

2.9 607Eh 指令方向极性

位	0	1	2	3	4	5	6	7
功能	NA	NA	NA	NA	NA	转矩指令极性	速度指令极性	位置指令极性

Bit7=1, 表示 PP 轮廓位置模式下, 将位置指令 607Ah \times (-1), 电机转向反向。

Bit6=1, 表示 PV 轮廓速度模式下, 将速度指令 60FFh \times (-1), 电机转向反向。

Bit5=1, 表示 PT 轮廓转矩模式下, 将转矩指令 6071h \times (-1), 电机转向反向。

NA: 表示无定义

2.10 60FDh 数字量输入状态监视

位	0-15	16	17	18	19	20	21	22	23-31
物理输入端子监控	保留	DI1	DI2	DI3	DI4	DI5	DI6	DI7	保留

2.11 60FEh 数字量输出状态监视与强制

位	0	1-15	16	17	18	19	20	21-31
物理输出端子	刹车抱闸	保留	DO1	DO2	DO3	DO4	DO5	保留
60FE-01h 状态监控&位强制控制	1、当 60FE-02h 对应的位设为 0 时, 60FE-01h 的该位状态有内部状态决定, 用户可监控该位当前状态; 2、当 60FE-02h 对应的位设为 1 时, 60FE-01h 的该位保持当前输出状态不再由内部状态决定并且, 用户可强制对 60FE-01h 的该位进行写操作, 写 1 该位物理端口输出强制 ON, 写 0 该位物理端口输出强制 OFF; 3、bit0 位, 是刹车抱闸电机专用控制位, 电机使能运行时此位用户强制关闭无效。只有解除使能运行后可进行强制写 1 开启或写 0 关闭;							
60FE-02h 强制控制位选使能								

三、SDO 控制报文参考

以下 SDO 控制报文均以控制节点 1 电机为例：

3.1 速度模式控制

描述	帧 ID	帧类型	帧格式	DLC	数据段 DATA[0]~DATA[7] 16 进制 大端模式存放			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
写运行模式=3	601	数据帧	标准帧	8	2F	60 60	00	03 00 00 00
从机应答	581	数据帧	标准帧	8	60	60 60	00	00 00 00 00
写目标速度	601	数据帧	标准帧	8	23	FF 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	FF 60	00	00 00 00 00
写加速度	601	数据帧	标准帧	8	23	83 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	83 60	00	00 00 00 00
写减速度	601	数据帧	标准帧	8	23	84 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	84 60	00	00 00 00 00
[注 1]写控制字=6	601	数据帧	标准帧	8	2B	40 60	00	06 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 1]写控制字=7	601	数据帧	标准帧	8	2B	40 60	00	07 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 1]写控制字=0F	601	数据帧	标准帧	8	2B	40 60	00	0F 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
读状态字	601	数据帧	标准帧	8	40	41 60	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	4B	41 60	00	xx xx xx xx 返回值
读当前速度	601	数据帧	标准帧	8	40	6C 60	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	43	6C 60	00	xx xx xx xx 返回值

[注 1] 写 6040h 控制字 06,电机进入准备好状态；写 07, 电机等待使能状态；

07→0F,电机将使能并按照速度配置的参数进行速度运行；

0F→07,电机将解除使能，自由停止；

3.2 位置模式控制

描述	帧 ID	帧类型	帧格式	DLC	数据段 DATA[0]~DATA[7] 16 进制 大端模式存放			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
写运行模式=1	601	数据帧	标准帧	8	2F	60 60	00	01 00 00 00
从机应答	581	数据帧	标准帧	8	60	60 60	00	00 00 00 00
写目标位置	601	数据帧	标准帧	8	23	7A 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	7A 60	00	00 00 00 00
写运行速度	601	数据帧	标准帧	8	23	81 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	81 60	00	00 00 00 00
写加速度	601	数据帧	标准帧	8	23	83 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	83 60	00	00 00 00 00
写减速度	601	数据帧	标准帧	8	23	84 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	84 60	00	00 00 00 00
[注 1]写控制字=6	601	数据帧	标准帧	8	2B	40 60	00	06 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 1]写控制字=7	601	数据帧	标准帧	8	2B	40 60	00	07 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 1]写控制字=0F	601	数据帧	标准帧	8	2B	40 60	00	0F 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 1]写控制字=1F	601	数据帧	标准帧	8	2B	40 60	00	1F 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
读状态字	601	数据帧	标准帧	8	40	41 60	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	4B	41 60	00	xx xx xx xx 返回值
读当前位置	601	数据帧	标准帧	8	40	64 60	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	43	64 60	00	xx xx xx xx 返回值

[注 1] 写 6040h 控制字 06,电机进入准备好状态; 写 07, 电机等待使能状态; 然后写 0F 或 2F 或 4F 或 6F,电机将使能; 最后写 1F 或 3F 或 5F 或 7F, 电机将按照命令运行;

0F→1F,电机将执行绝对定位非立即更新; 2F→3F,电机将执行绝对定位立即更新; 定位完成后电机自动停止;

4F→5F,电机将执行相对定位非立即更新; 6F→7F,电机将执行相对定位立即更新; 定位完成后电机自动停止;

3.3 回原模式控制

描述	帧 ID	帧类型	帧格式	DLC	数据段 DATA[0]~DATA[7] 16 进制 大端模式存放			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
写运行模式=6	601	数据帧	标准帧	8	2F	60 60	00	06 00 00 00
从机应答	581	数据帧	标准帧	8	60	60 60	00	00 00 00 00
[注 1]写回原方式	601	数据帧	标准帧	8	2F	98 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	98 60	00	00 00 00 00
写回原高速	601	数据帧	标准帧	8	23	99 60	01	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	99 60	01	00 00 00 00
写回原低速	601	数据帧	标准帧	8	23	99 60	02	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	99 60	02	00 00 00 00
写加速度	601	数据帧	标准帧	8	23	9A 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	9A 60	00	00 00 00 00
[注 2]写控制字=6	601	数据帧	标准帧	8	2B	40 60	00	06 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 2]写控制字=7	601	数据帧	标准帧	8	2B	40 60	00	07 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 2]写控制字=0F	601	数据帧	标准帧	8	2B	40 60	00	0F 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 2]写控制字=1F	601	数据帧	标准帧	8	2B	40 60	00	1F 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
读状态字	601	数据帧	标准帧	8	40	41 60	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	4B	41 60	00	xx xx xx xx 返回值
读当前位置	601	数据帧	标准帧	8	40	64 60	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	43	64 60	00	xx xx xx xx 返回值

[注 1]: 本机支持的原点复归方式请查看 2.8 章节[6098h 原点搜索方式];

[注 2]: 写 6040h 控制字 06,电机进入准备好状态; 写 07, 电机等待使能状态; 然后写 0F 电机将使能;

0F→1F,电机将按照回原点的参数进行原点回归, 原点回归成功后电机自动停止, 当前位置清零;

1F→0F,电机将停止当前的原点回归动作;

3.3 转矩模式控制

描述	帧 ID	帧类型	帧格式	DLC	数据段 DATA[0]~DATA[7] 16 进制 大端模式存放			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
写运行模式=4	601	数据帧	标准帧	8	2F	60 60	00	04 00 00 00
从机应答	581	数据帧	标准帧	8	60	60 60	00	00 00 00 00
写目标转矩	601	数据帧	标准帧	8	2B	71 60	00	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	98 60	00	00 00 00 00
写速度限制	601	数据帧	标准帧	8	23	7F 60	01	xx xx xx xx 写入值
从机应答	581	数据帧	标准帧	8	60	99 60	01	00 00 00 00
[注 1]写控制字=6	601	数据帧	标准帧	8	2B	40 60	00	06 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 1]写控制字=7	601	数据帧	标准帧	8	2B	40 60	00	07 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
[注 1]写控制字=0F	601	数据帧	标准帧	8	2B	40 60	00	0F 00 00 00
从机应答	581	数据帧	标准帧	8	60	40 60	00	00 00 00 00
读状态字	601	数据帧	标准帧	8	40	41 60	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	4B	41 60	00	xx xx xx xx 返回值
读当前转矩	601	数据帧	标准帧	8	40	77 60	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	4B	77 60	00	xx xx xx xx 返回值

[注 1]:

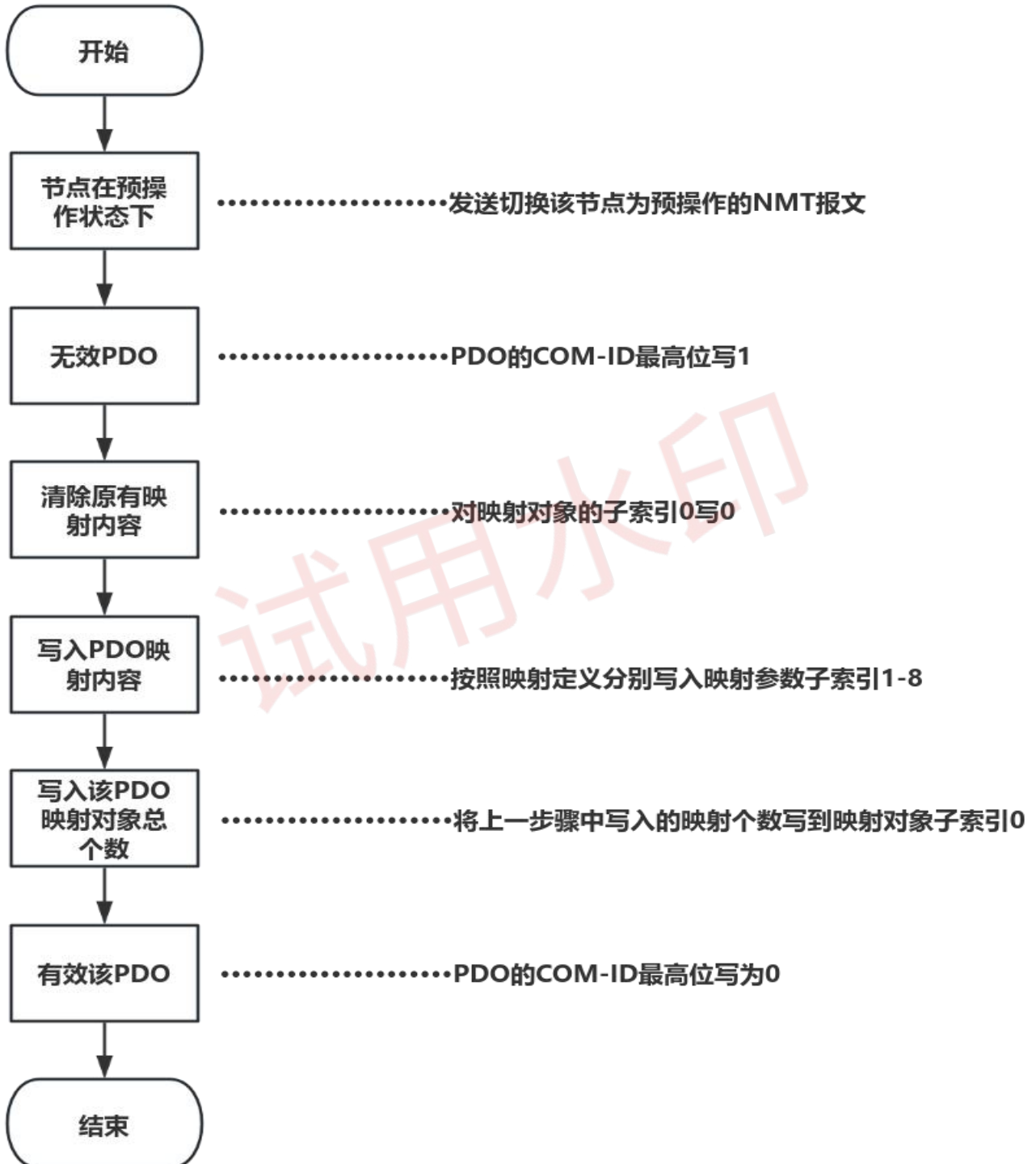
写 6040h 控制字 06,电机进入准备好状态; 写 07, 电机等待使能状态;

07→0F,电机将使能并按照转矩配置的参数进行转矩运行;

0F→07,电机将解除使能, 自由停止;

四、PDO 控制模式

4.1 PDO 映射配置流程图



4.2 PDO 通讯和映射对象

按照接收与发送的不同, PDO 可分为 RPDO 和 TPDO。PDO 由通信参数和映射参数共同决定最终传输的方式以及内容。AIMOTOR 伺服驱动器使用了四个 TPDO 和四个 RPDO 来实现 PDO 的传输, 相关对象列表如下。

名称		COB-ID	通信对象	映射对象
RPDO	1	200h+NODE_ID	1400h	1600h
	2	300h+NODE_ID	1401h	1601h
	3	400h+NODE_ID	1402h	1602h
	4	500h+NODE_ID	1403h	1603h
TPDO	1	180h+NODE_ID	1800h	1A00h
	2	280h+NODE_ID	1801h	1A02h
	3	380h+NODE_ID	1802h	1A03h
	4	480h+NODE_ID	1803h	1A04h

4.3 PDO 配置过程参考例

以下 PDO 控制报文均以控制节点 1 电机为例:

其中 TPDO1 映射对象为 1: 6041_00h (状态字) 2: 606C_00h (实际转速)。

其中 RPDO1 映射对象为 1: 6040_00h (控制字) 2: 6060_00h (控制模式) 3: 60FF_00h (目标转速)。

描述	帧 ID	帧类型	帧格式	DLC	数据段 DATA[0]~DATA[7] 16 进制 大端模式存放			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
将节点设为预操作状态	000	数据帧	标准帧	2	80	01	-	-
无从机应答	-	-	-	-	-	-	-	-
设置 RPDO1 的 COB_ID 且失效 RPDO1	601	数据帧	标准帧	8	23	00 14	01	01 02 00 80
从机应答	581	数据帧	标准帧	8	60	00 14	01	00 00 00 00
写 RPDO1 的传输类型	601	数据帧	标准帧	8	2F	00 14	02	01 00 00 00
从机应答	581	数据帧	标准帧	8	60	00 14	02	00 00 00 00
RPDO1 映射个数设 0	601	数据帧	标准帧	8	2F	00 16	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	60	00 16	00	00 00 00 00
写 RPDO1 的映射对象 1	601	数据帧	标准帧	8	23	00 16	01	10 00 40 60
从机应答	581	数据帧	标准帧	8	60	00 16	01	00 00 00 00

写 RPDO1 的映射对象 2	601	数据帧	标准帧	8	23	00 16	02	08 00 60 60
从机应答	581	数据帧	标准帧	8	60	00 16	02	00 00 00 00
写 RPDO1 的映射对象 3	601	数据帧	标准帧	8	23	00 16	03	20 00 FF 60
从机应答	581	数据帧	标准帧	8	60	00 16	03	00 00 00 00
RPDO1 映射个数设 3	601	数据帧	标准帧	8	2F	00 16	00	03 00 00 00
从机应答	581	数据帧	标准帧	8	60	00 16	00	00 00 00 00
设置 RPDO1 的 COB_ID 且生效 RPDO1	601	数据帧	标准帧	8	23	00 14	01	01 02 00 00
从机应答	581	数据帧	标准帧	8	60	00 14	01	00 00 00 00
设置 RPDO2 的 COB_ID 且失效 RPDO2	601	数据帧	标准帧	8	23	01 14	01	01 03 00 80
从机应答	581	数据帧	标准帧	8	60	01 14	01	00 00 00 00
设置 RPDO3 的 COB_ID 且失效 RPDO3	601	数据帧	标准帧	8	23	02 14	01	01 04 00 80
从机应答	581	数据帧	标准帧	8	60	02 14	01	00 00 00 00
设置 RPDO4 的 COB_ID 且失效 RPDO4	601	数据帧	标准帧	8	23	03 14	01	01 05 00 80
从机应答	581	数据帧	标准帧	8	60	03 14	01	00 00 00 00
设置 TPDO1 的 COB_ID 且失效 TPDO1	601	数据帧	标准帧	8	23	00 18	01	81 01 00 80
从机应答	581	数据帧	标准帧	8	60	00 18	01	00 00 00 00
设置 TPDO1 的传输类型	601	数据帧	标准帧	8	2F	00 18	02	01 00 00 00
从机应答	581	数据帧	标准帧	8	60	00 18	02	00 00 00 00
TPDO1 映射个数设 0	601	数据帧	标准帧	8	2F	00 1A	00	00 00 00 00
从机应答	581	数据帧	标准帧	8	60	00 1A	00	00 00 00 00

写 TPDO1 的映射 对象 1	601	数据帧	标准帧	8	23	00 1A	01	10 00 41 60
从机应答	581	数据帧	标准帧	8	60	00 1A	01	00 00 00 00
写 TPDO1 的映射 对象 2	601	数据帧	标准帧	8	23	00 1A	02	20 00 6C 60
从机应答	581	数据帧	标准帧	8	60	00 1A	02	00 00 00 00
TPDO1 映射个数设 2	601	数据帧	标准帧	8	2F	00 1A	00	02 00 00 00
从机应答	581	数据帧	标准帧	8	60	00 1A	00	00 00 00 00
设置 TPDO1 的 COB_ID 且生效 TPDO1	601	数据帧	标准帧	8	23	00 18	01	81 01 00 00
从机应答	581	数据帧	标准帧	8	60	00 18	01	00 00 00 00
设置 TPDO2 的 COB_ID 且失效 TPDO2	601	数据帧	标准帧	8	23	01 18	01	81 02 00 80
从机应答	581	数据帧	标准帧	8	60	01 18	01	00 00 00 00
设置 TPDO3 的 COB_ID 且失效 TPDO3	601	数据帧	标准帧	8	23	02 18	01	81 03 00 80
从机应答	581	数据帧	标准帧	8	60	02 18	01	00 00 00 00
设置 TPDO4 的 COB_ID 且失效 TPDO4	601	数据帧	标准帧	8	23	03 18	01	81 04 00 80
从机应答	581	数据帧	标准帧	8	60	03 18	01	00 00 00 00
将节点设为操作状 态	000	数据帧	标准帧	2	01	01	-	-
无从机应答	-	-	-	-	-	-	-	-

4.4 同步报文配置描述例

描述	帧 ID	帧类型	帧格式	DLC	数据段 DATA[0]~DATA[7] 16 进制 大端模式存放			
					DATA[0]	DATA[1]~ DATA[2]	DATA[3]	DATA[4]~ DATA[7]
[注 1]同步帧	080	数据帧	标准帧	0				
[注 2]TPDO 返回应答	181	数据帧	标准帧	6	37	02 E8	03	00 00
[注 3]RPDO 填充报文	201	数据帧	标准帧	7	0F	00 03	E8	03 00 00

[注 1]: 在 TPDO 传输类型为 1-240 时, 接收到响应个数的同步帧时, 发送该 TPDO 应答报文。当 RPDO 传输类型为 0-240 时, 只要接收到一个同步帧则将该 RPDO 最新的数据更新到应用。

[注 2]: 以上例报文内容为 37 02 E8 03 00 00 其中 TPDO1 映射对象 1 为 6041_00h 其数据宽度为两个字节, 在 TPDO1 返回的报文中 37 02 即为 6041_00h 的值。其中 TPDO1 映射对象 2 为 606C_00h 其数据宽度为四个字节, 在 TPDO1 返回的报文中 E8 03 00 00 即为 606C_00h 的值。

[注 3]: 以上例报文内容为 0F 00 03 E8 03 00 00 其中 RPDO1 映射对象 1 为 6040_00h 其数据宽度为两个字节, 则 0F 00 即为 6040_00h 的设置值。其中 RPDO1 映射对象 2 为 6060_00h 其数据宽度为一个字节, 则 03 即为 6060_00h 的设置值。其中 RPDO1 映射对象 3 为 60FF_00h 其数据宽度为四个字节, 则 E8 03 00 00 即为 60FF_00h 的设置值。

4.5 映射对象长度对应表

对象长度	位长
08h	8 位
10h	16 位
20h	32 位

例如: 表示 16 位命令字 6040_00h 的映射参数为 60400010h。

五、故障查询

5.1 故障码表

描述	厂家自定义故障码 H0B_34 参数	Cia402 标准故障码 603Fh 对象	硬件报警灯 闪烁次数	报警类型
无故障	0x0000	0x0000	绿	-
参数 id 超范围	0x0101	0x6320	10 红 1 绿	NO.1
参数 CRC 错误	0x0102	0x6320	12 红 1 绿	NO.1
注册参数 CRC 错误	0x0104	0x6320	12 红 1 绿	NO.1
内部程序异常触发看门狗	0x0105	0x6320	11 红 1 绿	NO.1
DI 功能重复分配	0x0130	0x6320	12 红 1 绿	NO.1
硬件过流	0x0201	0x2312	4 红 1 绿	NO.1
ia/ic 静态电流偏置校准失败	0x0208	0xFF00	12 红 1 绿	NO.1
软件过流	0x0207	0x2311	4 红 1 绿	NO.1
飞车	0x0234	0xFF00	12 红 1 绿	NO.1
编码器数据异常	0x0A33	0x7306	9 红 1 绿	NO.1
主回路电过压	0x0400	0x3210	3 红 1 绿	NO.1
主回路电欠压	0x0410	0x3220	3 红 1 绿	NO.1
电机过载	0x0620	0x3230	4 红 1 绿	NO.1
散热器过热	0x0650	0x4210	6 红 1 绿	NO.1
位置偏差过大	0x0B00	0x8611	2 红 1 绿	NO.1
回零方式不匹配	0x0668	0xFF00	8 红 1 绿	NO.2
回原点超时	0x0601	0x8610	8 红 1 绿	NO.2
紧急停机	0x0900	0x5442	12 红 1 绿	NO.3
正向超程警告	0x0950	0x5443	12 红 1 绿	NO.3
负向超程警告	0x0952	0x5444	12 红 1 绿	NO.3
编码器电池失效	0x0731	0x7306	12 红 1 绿	NO.2
编码器多圈计数错误	0x0733	0x7306	12 红 1 绿	NO.2
编码器多圈计数溢出	0x0735	0x7306	12 红 1 绿	NO.2
编码器电池警告	0x0730	0x7307	12 红 1 绿	NO.3
CAN 通信连接中断	0x0D03	0x8130	12 红 1 绿	NO.2
变更参数需重新上电生效	0x0941	0xFF00	7 红 1 绿	NO.3
参数存储频繁	0x0942	0x7600	12 红 1 绿	NO.3

- ◆ 厂家自定义故障码，一个故障信息对应一个故障码，具有唯一性；
- ◆ Cia402 标准故障码，参考 cia402 规定，当超出 cia402 规定外的故障信息以 0xFF00 故障码表示，不具有唯一性；
- ◆ 一体集成式电机因未有显示面板，在不连接调试软件下，可参考指示灯闪烁变化判断报警信息，不具有唯一性；

5.2 故障查询方法

1、使用对象字典厂家自定义参数查询

参数号：H0B-33，选择查询的故障记录； 对象字典索引：200Bh，子索引 22h，读写

参数号：H0B-34，所选次数故障码； 对象字典索引：200Bh，子索引 23h，仅读

使用：H0B-33(200B-22h)用于选择查看的故障记录，H0B-34(200B-23h)显示所选择的历史故障

举例：当 H0B-33(200B-22h)设为 0 时，一旦发生故障 H0B-34(200B-23h)将显示当前厂家自定义故障码；

当 H0B-33(200B-22h)设为 1 时系统将查询历史前 1 次产生的厂家自定义故障码，并在 H0B-34(200B-23h)显示；

当 H0B-33(200B-22h)设为 2 时系统将查询历史前 2 次产生的厂家自定义故障码，并在 H0B-34(200B-23h)显示；

↓

↓

当 H0B-33(200B-22h)设为 9 时系统将查询历史前 9 次产生的厂家自定义故障码，并在 H0B-34(200B-23h)显示；

2、使用对象字典 CIA402+CIA301 参数查询

当前故障码； 对象字典索引：603Fh，子索引 00h，只读

历史故障数量；对象字典索引：1003h，子索引 00h，读写

历史故障条目；对象字典索引：1003h，子索引 01h~0Ah，仅读

使用：一旦发生故障，603F-00h 将产生当前故障对应的 Cia402 标准故障码,同时相应的厂家自定义故障码和 Cia402 标准故障码组合存储在对 1003h 的 01h~0Ah 中；

对象 1003h 的 00h 为当前记录的故障数量。将 1003-00h 写 0，表示清除历史故障记录，不允许写入非 0 值；

对象 1003h 的 01h~0Ah 为 10 个故障记录条目队列，采用堆栈方式存储，遵循先进先出规则。在历史报警存储满的情况下，产生新的报警，会删除最早出现的错误，之前的错误依次向下移动；每个条目的储存规则是 bit0-bit15 储存 Cia402 标准故障码，bit16-bit31 储存厂家自定义故障码；

举例：用户可先读取 603F-00h 里的 Cia402 标准故障码值（非 0 值时表示有故障）判断当前节点是否故障。一旦产生了故障，可参考“3.1 章节故障表”初步确定故障信息。因为 Cia402 标准故障码在本系统中不具有唯一性，可继续读取 1003-01h 里的值（最近 1 次故障记录），值的低 16 位与当前的 603F-00h 里的 Cia402 标准故障码值一致，高 16 位表示对应的厂家定义故障码，根据获取厂家定义故障码来确定具体的故障信息。