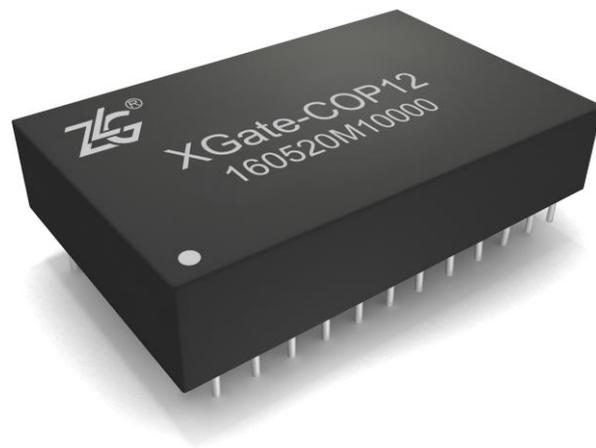


XGate-COP12

嵌入式 CANopen 通信模块

UM01010101 V1.00 Date: 2017/03/15

产品用户手册



类别	内容
关键词	CANopen 通信模块 XGate XGate-COP12 串口 UART SPI 从站 CAN-bus
摘要	



修订历史

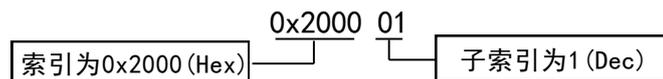
版本	日期	原因
V1.00	2017/03/15	创建文档

名词解释:

缩写	全拼	中文
PDO	Process Data Object	过程数据对象
TPDO	Transmit Process Data Object	发送过程数据对象
RPDO	Receive Process Data Object	接收过程数据对象
SDO	Service Data Object	服务数据对象
NMT	Network Management	网络管理
SYNC	Synchronization Objects	同步报文对象
EMCY	Emergency Objects	紧急对象报文
CAN-ID	Controller Area Network-Identify	控制器局域网标识符
COB-ID	Communication Object-Identify	通信对象标识符
SSDO	Servers Service Data Object	服务数据服务器
DS	Draft Standard	标准草案

书写规则:

本手册中，对象字典索引与子索引的书写遵循如下图所示的规则，其中索引为十六进制表示，子索引为十进制表示，索引与子索引中间用空格或“/”隔开。



目 录

1. CANopen 协议简介	1
1.1 CANopen 对象字典.....	1
1.2 CANopen 通讯	2
2. XGate-COP12 特点	4
2.1 XGate-COP12 状态指示灯 (DS303-3)	5
3. XGate-COP12 节点 ID 与 CAN 总线波特率	7
3.1 CANopen 节点地址和波特率设置.....	7
4. XGate-COP12 中使用 CANopen 协议 (DS301)	8
4.1 XGate-COP12 的对象字典	8
4.2 XGate-COP12 预定义连接	8
4.3 XGate-COP12 操作	9
4.3.1 网络管理服务 (NMT)	9
4.3.2 同步报文对象 (SYNC)	11
4.3.3 紧急报文对象 (EMCY)	12
4.3.4 服务数据对象 (SDO)	13
4.3.5 过程数据对象 (PDO)	16
5. 输入输出数据缓冲区.....	20
6. XGate-COP12 串口/SPI 操作	21
6.1 通信协议.....	21
6.1.1 数据帧格式.....	22
6.2 XGate-COP12 操作命令	23
6.2.1 读设备信息 (命令码: 0x01)	23
6.2.2 写设备信息 (命令码: 0x02)	24
6.2.3 写 XGate-COP12 输入缓冲区数据 (命令码: 0x10)	24
6.2.4 读 XGate-COP12 输出缓冲区数据 (命令码: 0x11)	25
6.2.5 读写 XGate-COP12 的 NodeID 和 CAN 波特率索引 (命令码: 0x12) ..	25
6.2.6 发送紧急代码 (命令码: 0x15)	26
6.2.7 读取模块状态 (命令码: 0x16)	27
6.2.8 启动节点进入操作状态 (命令码: 0x17)	27
6.2.9 复位模块 (命令码: 0x24)	28
6.2.10 读取 XGate-COP12 软硬件版本 (命令码: 0x25)	28
6.2.11 重置内部 E ² PROM (命令码: 0xED)	29
6.3 XGate-COP12 串口/SPI 操作错误响应	29
附录 A 串口可操作对象字典列表.....	31
附录 B XGate-COP12 缺省对象字典	32
附录 C XGate-COP12 对象字典样例	36

1. CANopen 协议简介

CANopen协议是在20世纪90年代末，由CiA组织（CAN-in-Automation）在CAL（CAN Application Layer）的基础上发展而来，一经推出便在欧洲得到了广泛的认可与应用。经过对CANopen协议规范文本的多次修改，使得CANopen协议的稳定性、实时性、抗干扰性都得到了进一步的提高。并且CiA在各个行业不断推出设备子协议，使CANopen协议在各个行业得到更快的发展与推广。目前CANopen协议已经在运动控制、车辆工业、电机驱动、工程机械、船舶海运等行业得到广泛的应用。

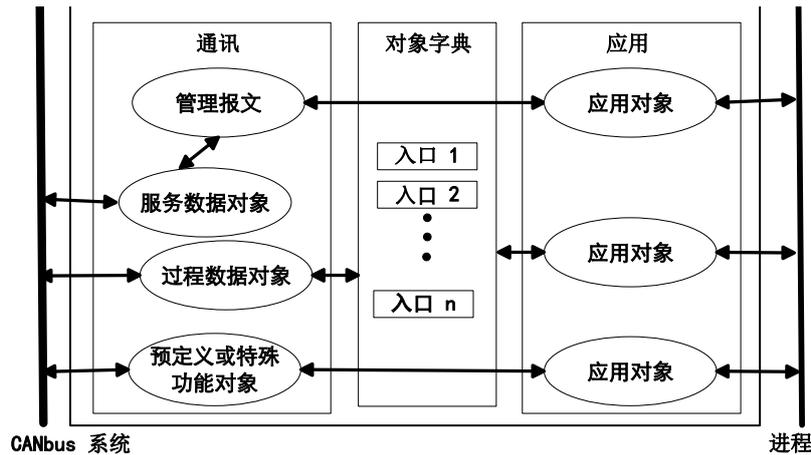


图 1.1 CANopen 设备结构

图1.1描述了CANopen设备的结构，CANopen协议通常分为用户应用层、对象字典和通讯三个部分。

1.1 CANopen 对象字典

CANopen对象字典（OD: Object Dictionary）是CANopen协议最为核心的概念。对象字典是一个有序的对象组，每个对象使用一个16位的编号寻址，这个索引值通常被称为索引，其有效范围在0x1000到0x9FFF之间。为了访问数据结构中的子元素，还定义了一个8位的二级编号，通常称为子索引。

每个CANopen设备都有一个对象字典，对象字典包含了描述这个设备和它的网络行为的所有参数，对象字典通常用电子数据文档（EDS: Electronic Data Sheet）来记录这些参数。CANopen网络中的主节点可以根据实际需要访问CANopen从节点的相关对象字典项。

CANopen对象字典中的对象由一系列子协议来描述。子协议为对象字典中的每个对象都描述了它的功能、名字、索引、子索引、数据类型，以及这个对象是否必需、读写属性等等，这样可保证不同厂商的同类型设备兼容。

CANopen协议的核心描述子协议是DS301，其包括了CANopen协议应用层及通信结构描述，其他的子协议都是对DS301协议描述文本的补充与扩展。在不同的应用行业都会起草一份CANopen设备子协议，子协议编号一般是DS4xx或DSP4xx。

CANopen协议包含了许多子协议，其主要划分为以下三类。

1. 通讯子协议（Communication Profile）

通讯子协议，描述对象字典的主要形式和对象字典中的通讯对象以及参数。这个子协议

适用所有的CANopen设备，其索引值范围从0x1000~0x1FFF。

2. 制造商自定义子协议 (Manufacturer-specific Profile)

制造商自定义子协议，对于在设备子协议中未定义的特殊功能，制造商可以在此区域根据需求定义对象字典对象。因此这个区域对于不同的厂商来说，相同的索引的对象字典项定义不一定相同，其索引值范围为0x2000~0x5FFF。

3. 设备子协议(Device Profile)

设备子协议，为各种不同类型的设备定义对象字典中的对象。目前已有十几种为不同类型的设备定义的子协议，例如DS401、DS402、DS406等，其索引值范围为0x6000~0x9FFF。

1.2 CANopen 通讯

在CANopen协议中主要定义了管理报文对象NMT (Network Management)、服务数据对象SDO (Service Data Object)、过程数据对象PDO (Process Data Object)、预定义报文或特殊功能对象等四种对象。

1. 网络管理 NMT (Network Management)

管理报文负责层管理、网络管理和ID分配服务，例如，初始化、配置和网络管理（其中包括节点保护）。网络管理中，同一个网络中只允许有一个主节点、一个或多个从节点，并遵循主从模式。通过NMT服务，我们可以对节点进行初始化、运行、监控、复位和停止。所有节点都被认为是NMT从站。

2. 服务数据对象 SDO (Service Data Object)

SDO主要用于主节点对从节点的参数配置。服务确认是SDO的最大的特点，为每个消息都生成一个应答，确保数据传输的准确性。在一个CANopen系统中，通常CANopen从节点作为SDO服务器，CANopen主节点作为客户端。客户端通过索引和子索引，能够访问数据服务器上的对象字典。这样CANopen主节点可以访问从节点的任意对象字典项的参数，并且SDO也可以传输任何长度的数据（当数据长度超过4个字节时就拆分成多个报文来传输）。

3. 过程数据对象 PDO (Process Data Object)

PDO用来传输实时数据，其传输模型为生产者消费者模型如图1.2所示。数据长度被限制为1~8字节。PDO通信对象具有如下的特点：

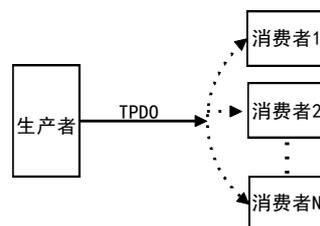


图 1.2 生产者消费者模型

- PDO 通讯没有协议规定，PDO 数据含义由它的 CAN-ID（也可称为 COB-ID）标示；
- 每个 PDO 在对象字典中用 2 个对象描述：
 - PDO通讯参数，该通讯参数定义了设备所使用的COB-ID、传输类型、定时周期；
 - PDO映射参数，映射参数包含了一个对象字典中的对象列表，这些对象映射

到相应的PDO，其中包括数据的长度（单位：位），只有生产者和消费者都了解这个映射参数，才能够正确地传递该PDO内容。

- PDO 消息内容是预定义的，如果 PDO 支持可变 PDO 映射，那么该 PDO 是通过 SDO 进行配置；
- PDO 可以有多种的传输方式：
 - 同步传输（通过接收同步对象实现同步），同步传输又可分为周期传输和非周期传输。非周期传输是由远程帧预触发或者由设备子协议中规定的对象特定事件预触发传送。周期传输则是通过接收同步对象（SYNC）来实现，可以设置 1~240 个同步对象触发；
 - 异步传输（由特定事件触发），其触发方式可有两种，第一种是通过发送与 PDO 的 COB-ID 相同的远程帧来触发 PDO 的发送，第二种是由设备子协议中规定的对象特定事件来触发（例如，定时传输，数据状态变化传输等）。

4. 预定义报文或特殊功能对象

预定义报文或特殊功能对象为 CANopen 设备提供特定的功能，方便 CANopen 主站对从站管理。在 CANopen 协议中，已经为特殊的功能预定义了 COB-ID，其主要有以下几种特殊报文：

- 同步（SYNC），该报文对象主要实现整个网络的同步传输，每个节点都以该同步报文作为 PDO 同步触发参数，因此该同步报文的 COB-ID 具有较高的优先级以及最短的传输时间；
- 时间标记对象（Time Stamp），为各个节点提供公共的时间参考；
- 紧急事件对象（Emergency），当设备内部发生错误触发该对象，即发送设备内部错误代码；
- 节点 / 寿命保护（Node / Life Guarding），主节点可通过节点保护方式获取从节点的状态。从节点可通过寿命保护方式获取主节点的状态；
- 启动报文对象（Boot-up），从节点初始化完成后向网络中发送该对象，并进入到预操作状态。

2. XGate-COP12 特点

XGate-COP12是CANopen从站通信模块，内部集成了CANopen从站协议栈代码，不需要进行二次开发。协议栈遵循CANopen协议描述文档DS301、DS302、DS303以及DS305标准。在默认情况下，CANopen从站启用预定义连接报文，该模块具有如下特点。

- 支持用户自定义对象词典配置；
- 网络管理服务对象（NMT：Boot up, Node Guarding / Life guarding, Heartbeat Producer）；
- 可自定义过程数据对象（TPDO 与 RPDO）；
- 服务数据对象（SDO 服务器 / SDDO）；
- 紧急报文对象（Emergency）；
- 同步报文对象（Sync）；
- 串口通信能力（UART，9600~460800 bps）；
- SPI 通信能力（最高 2Mbit/s）；
- 96 字节的输入输出数据缓冲（I/O）；
- CAN 总线支持 10Kbps~1Mbps 波特率；
- 小体积，31.8mm×20.3mm（DIP24 封装）。

XGate-COP12从站具有高实时性，可以支持多达12个RPDO和12个TPDO过程数据传输，适用于各种干扰强、实时性要求高的场合，体积小适用于嵌入到各种电路板中。

图2.1为XGate-COP12系统结构图，模块负责将用户CPU发送过来的数据以TPDO报文方式发送到CAN总线，以及读取来自CAN总线的RPDO数据。

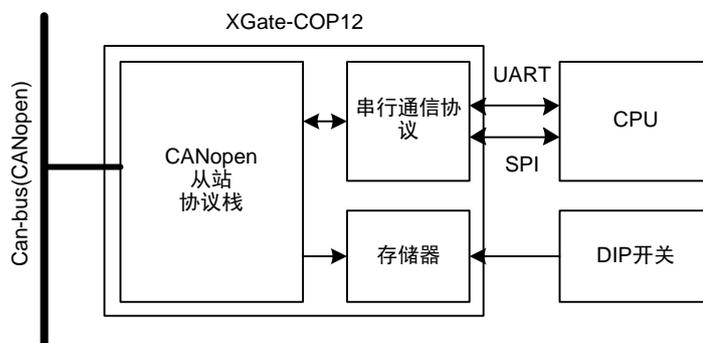


图 2.1 XGate-COP12 系统结构

图2.2为XGate-COP12和状态之间有的转换图，图中各字母所表示的各种状态下可进行的的操作，字母表达的操作为：

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

图中箭头所表示各个状态之间的转换关系，数字表示这种转换所需要进行的操作，数字表达的操作为：

- 1: Start Remote Node (0x01)
- 2: Stop Remote Node (0x02)
- 3: Enter Pre-Operational State (0x80)
- 4: Reset Node (0x81)
- 5: Reset Communication (0x82)
- 6: 设备初始化结束，自动进入 Pre Operational 状态，发送 Boot-up 消息

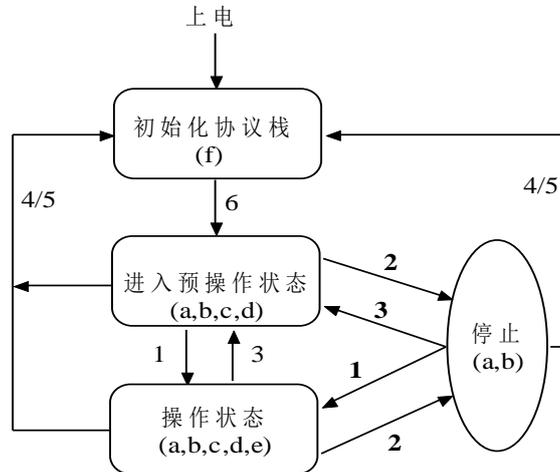


图 2.2 XGate-COP12 状态转换图

2.1 XGate-COP12 状态指示灯 (DS303-3)

按照CANopen协议规范文档DS303-3的定义，XGate-COP12模块使用两个LED指示灯指示当前模块所处的状态，如表2.1所示。

表 2.1 状态指示灯

指示灯名称	颜色	管脚
运行指示灯 (RUN)	绿色	13
错误指示灯 (ERR)	红色	14

其状态指示灯所指的各种状态所指示的含义见表2.2、表2.3和表2.4。

表 2.2 LED 状态说明

指示灯状态	现象描述
亮 (LED on)	常亮
暗 (LED off)	常暗
快闪 (LED flickering)	亮和暗的时间等长，频率约是 10Hz: 亮约 50ms, 暗约 50ms
慢闪 (LED blinking)	亮和暗的时间等长，频率约是 2.5Hz: 亮约 200ms, 暗约 200ms
闪一下 (LED single flash)	亮约 200ms 接着暗约 1000ms
闪两下 (LED double flash)	亮约 200ms 暗约 200ms, 再亮 200ms 最后暗约 1000ms
闪三下 (LED triple flash)	亮约 200ms 暗约 200ms, 再亮 200ms 暗约 200ms, 再亮 200ms 最后暗约 1000ms

表 2.3 错误状态指示灯 (ERR) 描述

编号	ERROR LED	状态	描述	种类
1	暗	没有错误	器件处于工作状态	强制
2	闪一下	到达警戒值	CAN 控制器的至少一个错误计数器到达或超出了警戒值（错误帧太多）	强制
3	快闪	自动波特率 / LSS	正在进行自动波特率检测或进行 LSS 服务(和 RUN LED 交替闪烁 (flickering))	可选
4	闪两下	错误控制事件	发生保护事件 (NMT 从机或 NMT 主机) 或心跳事件 (心跳使用者)	强制
5	闪三下	Sync 错误	SYNC 报文超出配置的通讯循环间隔仍未收到 (见对象字典条目 0x1006)	有条件, 如果支持对象 0x1006 则强制
6	亮	总线关闭	CAN 控制器总线关闭	强制

表 2.4 运行状态指示灯 (RNU) 描述

编号	RUN LED	状态	描述	种类
1	快闪	自动波特率 / LSS	正在进行 LSS 服务	可选
2	闪一下	停止	器件处于停止状态	强制
3	慢闪	预操作	器件处于预操作状态	强制
4	亮	工作	器件处于工作状态	强制
5	暗	故障	请检查模块复位引脚以及电源是否连接正确	



3. XGate-COP12 节点 ID 与 CAN 总线波特率

3.1 CANopen 节点地址和波特率设置

XGate-COP12支持通过串口或SPI接口设置节点ID和节点波特率，详见6.2.5。

4. XGate-COP12 中使用 CANopen 协议 (DS301)

4.1 XGate-COP12 的对象字典

XGate-COP12中使用的对象字典可由配套的PC配置工具定制，未使用该工具配置或恢复出厂设置则使用缺省的词典。附录B描述了缺省的对象词典，附录C描述了一个包含12个RPDO和12个TPDO的对象词典样例。

配置工具可用于增加、删除或修改过程变量及PDO映射内容从而适配不同应用。词典中0x1000~0x1018、0x2404、0x2405等必要对象（索引）不可删除。XGate-COP12可提供的PDO数量受数据缓冲区长度的限制，最多可设置12个RPDO和12个TPDO，建议使用配套提供的模版修改。

注：使用若使用了PC配置工具的更改NodeID功能，还需另行设置NodeID才会生效，否则会使用默认的NodeID，参见6.2.5节。

4.2 XGate-COP12 预定义连接

XGate-COP12的对象分布在对象字典的0x1000~0x1FFF、厂商自定义区0x2000~0x5FFF和0xA000~0xAFFF的网络变量区域。这些对象负责CANopen与CAN网络上的其它应用数据的通信和数据交换，每个对象都有自己的数据长度（UINT8，UINT16或UINT32）和属性（RO、WO、RW、CONST、MAPPABLE），属性是WO或RW的对象可以使用SDO服务修改。

表4.1列举了XGate-COP12所使用的部分对象及其属性，完整的对象字典请参考附录。

通常情况下，一个典型的CANopen网络中有一个CANopen主站和若干个CANopen从站，这种情况下通常使用CANopen预定义连接。预定义连接指与通信相关的COB-ID与节点ID相关联。

例：假设当前节点ID为0x20，则TPDO1的COB-ID为0x180+0x20(0x180+NodeID)。具体的预定义连接集如表4.2所示。

表 4.1 部分对象字典定义

索引	子索引	名称	类型	属性	是否可映射	存储	单位
0x1000	0x00	DeviceType	UINT32	RO	NO	NO	-
0x1008	0x00	DeviceName	String	RO	NO	NO	-
0x100C	0x00	Guard Time	UINT16	RW	NO	YES	ms
0x1400	0x00	Subindex num	UINT8	RO	NO	NO	-
	0x01	COB-ID	UINT32	RW	NO	YES	-
	0x02	Trans type	UINT8U	RW	NO	YES	-
.....							
0x2000	0x00	Subindex num	UINT8	RO	NO	NO	-
	0x01	Input buffer #0	UINT8	RW	YES	NO	-

注:有些对象的值可以储存在存储器中，对于对象0x1000~0x1FFF中的参数，如果选择使用预定义连接方式则存储的参数无效（节点地址改变之后，存储的参数无效，节点将使用预定义连接集），这些参数将使用预定义连接集定义的值。预定义连接集定义见表4.2。

表 4.2 预定义的主机/从机连接集

对象	功能码	节点地址	COB-ID	对象字典索引
广播报文				
NMT	0000	-	0	-
SYNC	0001	-	0x80	0x1005, 0x1006, 0x1007
点对点报文				
紧急报文	0001	1-127	0x81-0xFF	0x1014, 0x1015
TPDO1	0011	1-127	0x181-0x1FF	0x1800
RPDO1	0100	1-127	0x201-0x27F	0x1400
TPDO2	0101	1-127	0x281-0x2FF	0x1801
RPDO2	0110	1-127	0x301-0x37F	0x1401
TPDO3	0111	1-127	0x381-0x3FF	0x1802
RPDO3	1000	1-127	0x401-0x47F	0x1402
TPDO4	1001	1-127	0x481-0x4FF	0x1803
RPDO4	1010	1-127	0x501-0x57F	0x1403
默认 SDO (tx)	1011	1-127	0x581-0x5FF	0x1200
默认 SDO (rx)	1100	1-127	0x601-0x67F	0x1200
NMT 错误控制	1110	1-127	0x701-0x77F	0x1016, 0x1017

4.3 XGate-COP12 操作

4.3.1 网络管理服务 (NMT)

1. 网络控制 (NMT Module Control)

XGate-COP12支持DS301所定义的网络管理命令, 这些网络管理命令可以由CANopen主站发出也可以由其它的从节点发出。其操作命令如下表4.3所示, 其中当NodeID = 0时, 则所有的从站设备被控制 (广播方式), CS为命令字对应着不同的控制动作如表4.4所示。

表 4.3 NMT 控制命令

COB-ID(CAN-ID)	DLC	Byte 0	Byte 1
0x000	2	CS (命令字)	NodeID (节点号)

表 4.4 NMT 命令字及相应功能服务

CS (命令字)	NMT 服务 (控制动作)
0x01	启动从站设备
0x02	停止从节点设备
0x80	使从站进入预操作
0x81	复位从节点
0x82	复位节点通信

例: 需要启动CANopen网络中所有的节点, 可使用如下表4.5所示的命令。

表 4.5 NMT 启动从节点

COB-ID(CAN-ID)	DLC	Byte 0	Byte 1
0x000	2	0x01	0x00

如果需要控制网络中某个具体设备，使其进入到预操作状态，假设节点地址为0x20，则命令如表4.6所示。

表 4.6 NMT 启动从节点

COB-ID(CAN-ID)	DLC	Byte 0	Byte 1
0x000	2	0x80	0x20

2. 节点保护 (NMT Node Guarding)

通过节点保护服务，NMT主节点可以检查每个节点的当前状态，当这些节点没有数据传送时这种服务尤其有意义。主节点通过发送远程帧来触发相应从节点的节点保护，其命令格式如表4.7所示。

主节点→从节点(命令):

表 4.7 NMT 主节点保护命令帧(远程帧)

COB-ID(CAN-ID)	DLC
0x700 + NodeID	1

从节点→主节点(响应):

表 4.8 NMT 从节点应答帧

COB-ID(CAN-ID)	DLC	Byte 0
0x700 + NodeID	1	Bit7:触发位, Bit0~Bit6 状态

其中Byte0中的最高位(bit7)为触发位，即从站每发一帧应答就会交替变化(0、1)，以示帧与帧之间的区别，其中Bit0~Bit6为从节点的状态，该值所表达从站所处的状态如表4.9所示。

表 4.9 节点保护状态值

值 (Value)	所处状态
0x00	初始化 (Initialising)
0x04	停止状态 (Stopped)
0x05	操作状态 (Operational)
0x7F	预操作状态 (Pre-operational)

例：假设主节点需要对节点号为0x20从节点进行节点保护，其命令如表4.10和表4.11所示。

主节点→从节点:

表 4.10 保护节点(远程帧)

COB-ID(CAN-ID)	DLC
0x720	1

从节点→主节点:

表 4.11 从节点(0x20)应答帧

COB-ID(CAN-ID)	DLC	Byte 0
0x720	1	0x85

其中BYTE0的Bit7 = 1，状态=0x05，表示节点号为0x20的从站正处于操作状态。

3. 寿命保护 (NMT Life Garding)

其中节点保护主要针对的是NMT主节点获取从节点的状态，而寿命保护主要是节点对另一节点的监控。寿命保护包括两个参数，即保护时间和生命因子，启用寿命保护的节点接收来自另一节点的远程帧（远程帧格式与节点保护帧格式相同如表4.7），启用寿命保护的节点接收到该远程帧则应答该节点的状态（应答帧格式如表4.8所示）。

寿命保护的两个参数：保护时间和生命因子（分别位于对象字典的0x100C和0x100D）构成了节点的寿命时间（即寿命时间=保护时间x 生命因子），保护时间的单位为毫秒，如果两个参数中有一个为0则表示寿命保护未启用。如果在保护时间内未接收到远程帧则会出现” Message Lost”的提示信息，在寿命时间内未接收到远程帧则会出现” Connection Lost”信息，这些信息均在调试串口中打印出来，同时错误指示灯出现“闪烁两下”以示当前的寿命保护丢失。

4. 启动报文(NMT Boot-up)

当XGate-COP12初始化完成(Boot-up)，就会发送一个标识报文，其报文格式如表4.12所示。

表 4.12 初始化完成标识报文格式

COB-ID(CAN-ID)	DLC	Byte 0
0x700 +NodeID	1	0x00

例：假设XGate-COP12的节点号为0x20，则发送的启动文如表4.13所示。

表 4.13 初始化完成标识报文

COB-ID(CAN-ID)	DLC	Byte 0
0x720	1	0x00

5. 心跳报文 (Heartbeat Producer)

心跳报文分为生产者和消费者，在XGate-COP12模块中只支持心跳报文生产，即XGate-COP12可以生产心跳报文。该参数在对象字典0x1017中定义（数据长度16位，单位：毫秒），其心跳报文如表4.8所示，与节点保护和寿命保护的应答帧相同。

例：假设节点地址为0x20，处于操作状态，0x1017中的参数设置为100，则该从节点每隔100毫秒发送一帧如表4.14所示的报文。

表 4.14 从节点(0x20)心跳报文

COB-ID(CAN-ID)	DLC	Byte 0
0x720	1	0x05

注：在同一个 XGate-COP12 模块中同一时间寿命保护和心跳报文不能同时使用。

4.3.2 同步报文对象 (SYNC)

同步报文分为消费和生产，在XGate-COP12中只支持同步报文的消费，即接收来自主节点或其它节点的同步报文，同步报文的帧结构如表4.15所示。对象字典的0x1005定义了

接收同步报文的COB-ID, 在CANopen预定义连接集里定义其值为0x80, 对象字典的0x1007定义了同步的时间窗口(在接收到同步报文后要求更新相应数据的最长时间间隔)。同步报文主要应用在PDO接收和发送的过程中, 其使用方法在以下PDO数据发送和接收过程中详细介绍。

表 4.15 同步报文帧格式(远程帧)

COB-ID(CAN-ID)	DLC
0x80	0

4.3.3 紧急报文对象(EMCY)

在XGate-COP12中支持紧急报文, 即在XGate-COP12内部出现错误时发送该报文, 其报文格式如表4.16所示。其中紧急错误码指定当前出现的错误的具体类型。错误寄存器存放当前错误类型, 根据该值可以判断出当前紧急报文所代表的错误类型, 其值定义如表4.17所示。

表 4.16 紧急报文帧格式

COB-ID(CAN-ID)	DLC	Byte 0	Byte 1	Byte 2	Byte 3-7
0x80 + NodeID	8	紧急错误码	错误寄存器		生产厂商指定的信息
		索引 0x1003	索引 0x1001		

表 4.17 错误寄存器

位(Bit)	错误类型
0	普通错误(Generic)
1	电流错误(Current)
2	电压错误(Voltage)
3	温度错误(Temperature)
4	通信错误(Communication)
5	设备描述错误(Device profile specific)
6	Reserved(=0)
7	生产厂商定义的错误(Manufacturer specific)

紧急错误代码含义如表4.18所示。

表 4.18 紧急错误代码

应急错误代码	代码功能描述
00xx	Error Reset 或 No Error
10xx	Generic Error
20xx	Current
21xx	Current, device input side
22xx	Current, inside the device
23xx	Current, device output side
30xx	Voltage
31xx	Mains voltage
32xx	Voltage inside the device

33xx	Output voltage
40xx	Temperature
41xx	Ambient temperature
42xx	Device emperature
50xx	Device hardware
60xx	Device software
61xx	Internal software
62xx	User software
63xx	Data set
70xx	Additional modules
80xx	Monitoring
81xx	communication
8110	CAN overrun
8120	Error Passive
8130	Life Guard Error 或 Heartbeat Error
8140	Recovered from Bus-Off
82xx	Protocol Error
8210	PDO no processed Due to length error
8220	Length exceedd
90xx	External error
F0xx	Additional functions
FFxx	Device specific

例：假设节点地址为0x20，CAN总线错误超过警戒值则出现“Error Passive”的警告，XGate-COP12发送的紧急报文如表4.19所示。

表 4.19 紧急报文（总线错误）

COB-ID(CAN-ID)	DLC	Byte 0	Byte 1	Byte 2	Byte 3-7
0xA0	8	0x20	0x81	0x11	0x00000000

注：XGate-COP12 模块发生紧急情况，这些错误将会主动发出。

4.3.4 服务数据对象（SDO）

对象字典充当应用层和通信层之间的主要数据交换媒介。一个CANopen设备的所有数据项可以在对象字典中被管理。每个对象字典项可以使用索引和子索引来定位。CANopen定义通常所说的服务数据对象（SDO）来访问这些项。

XGate-COP12支持1个SDO服务器，即可以提供SDO服务，且SDO使用预定义连接的发送和接收COB-ID，0x580 + NodeID(发送)和0x600 + NodeID(接收)。SDO分为加速传输、段传输、和块传输。因为在XGate-COP12中SDO的加速传输经常会使用，所以本说明文档重点介绍加速传输，其它的传输类型可查阅CANopen DS301及相关的协议文档。

1. SDO 数据传输

加速传输一帧最多只能传输4个字节的数据，报文基本结构如表4.20和表4.21所示，通过SDO的命令字来区分该帧数据的类型。

表 4.20 SDO 报文格式 (客户端→服务器)

COB-ID(CAN-ID)	DLC	Byte 0	Byte 1-2	Byte 3	Byte 4-7
0x600 + NodeID	8	CMD(SDO 命令字)	对象索引	对象子索引	*

表 4.21 SDO 应答格式 (服务器→客户端)

COB-ID(CAN-ID)	DLC	Byte 0	Byte 1-2	Byte 3	Byte 4-7
0x580 + NodeID	8	CMD(SDO 命令字)	对象索引	对象子索引	*

下载 (Download) 是指客户端对服务器 (从站) 对象字典进行写操作, 上传 (Upload) 指客户端对服务器对象字典进行读操作。

表 4.22 启动域下载 (Initiate Domain Download)

CMD(SDO 命令字)								
Bit	7	6	5	4	3	2	1	0
Client→Server	0	0	1	-	n		e	s
Client←Server	0	1	1	-	-	-	-	-

说明: 其中“-”项为不相关项, 通常设置为0

- n : 如果 e=1, 且 s=1, 则有效, 否则为 0; 表示数据部分中无意义数据的字节数 (字节 [8 - n] 到 7 数据无意义);
- e : 0 = 正常传送, 1 = 加速传送;
- s : 是否指明数据长度, 0 = 数据长度未指明, 1 = 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, byte 4 是数据低位部分 (LSB), byte 7 是数据高位部分 (MSB);
- e = 1 : 数据字节为将要下载 (download) 的数据。

表 4.23 启动域上传 Initiate Domain Download)

CMD(SDO 命令字节)								
Bit	7	6	5	4	3	2	1	0
Client→Server	0	1	0	-	-	-	-	-
Server→Client	0	1	0	-	n		e	s

说明: 命令字与表4.22相同。

例: 假设当前XGate-COP12的地址为0x20, 对对象字典进行0x1800 03进行读写操作。

- 向 0x1800 03 写入 0x3E8, 操作过程如表 4.24 和表 4.25 所示。

表 4.24 向对象字典 0x1800 03 写入 0x3E8 命令

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte4	Byte 5	Byte 6	Byte 7
0x620	8	0x2B	0x00	0x18	0x03	0xE8	0x03	0x00	0x00

表 4.25 向对象字典 0x1800 03 写入 0x3E8 后的应答

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte4	Byte 5	Byte 6	Byte 7
0x5A0	8	0x60	0x00	0x18	0x03	0x00	0x00	0x00	0x00

- 读对象字典 0x1800 03 中的数据, 其命令与应答如表 4.26 和表 4.27 所示。

表 4.26 读取对象字典 0x1800 03 数据命令

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte4	Byte 5	Byte 6	Byte 7
0x620	8	0x40	0x00	0x18	0x03	0x00	0x00	0x00	0x00

表 4.27 读对象字典 0x1800 03 后的应答帧

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte4	Byte 5	Byte 6	Byte 7
0x5A0	8	0x4B	0x00	0x18	0x03	0xE8	0x03	0x00	0x00

2. SDO 中止服务

在SDO的传输过程中当出现错误，SDO的客户端和服务端都可以发送这个报文来通知对方中止当前的操作，中止报文的格式如表4.28和表4.29所示。

表 4.28 中止报文格式

COB-ID(CAN-ID)	DLC	Byte 0	Byte 1-2	Byte 3	Byte 4-7
0x600 + NodeID/0x580+NodeID	8	CMD(SDO 命令字)	索引	子索引	*

表 4.29 中止报文命令字定义

CMD(SDO 命令字节)								
Bit	7	6	5	4	3	2	1	0
Client→Server / Client→Server	1	0	0	-	-	-	-	-

例：假设当前节点的 NodeID 为 0x20，在读取数据过程中出现错误中止，则会返回相应的错误代码，如下表 4.30 和表 4.31 所示。

表 4.30 读取不存在的对象字典 0x6000 00

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte4	Byte 5	Byte 6	Byte 7
0x620	8	0x40	0x00	0x60	0x00	0x00	0x00	0x00	0x00

表 4.31 读取不存在的对象字典 0x6000 00 返回数据

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte4	Byte 5	Byte 6	Byte 7
0x5A0	8	0x80	0x00	0x60	0x00	0x00	0x00	0x02	0x06

注：返回数据为一中止传送的错误代码，SDO 错误代码为 32 位长度的数据，该帧数据应答的错误代码为 0x06020000，代表的含义为“对象字典不存在”。具体的中止错误代码含义可参照表 4.32。

表 4.32 中止错误代码表

中止代码	代码功能描述
0503 0000	触发位没有交替改变
0504 0000	SDO 协议超时
0504 0001	非法或未知的 Client/Server 命令字
0504 0002	无效的块大小（仅 Block Transfer 模式）
0504 0003	无效的序号（仅 Block Transfer 模式）
0503 0004	CRC 错误（仅 Block Transfer 模式）
0503 0005	内存溢出
0601 0000	对象不支持访问

0601 0001	试图读只写对象
0601 0002	试图写只读对象
0602 0000	对象字典中对象不存在
0604 0041	对象不能够映射到 PDO
0604 0042	映射的对象的数目和长度超出 PDO 长度
0604 0043	一般性参数不兼容
0604 0047	一般性设备内部不兼容
0606 0000	硬件错误导致对象访问失败
0606 0010	数据类型不匹配, 服务参数长度不匹配
0606 0012	数据类型不匹配, 服务参数长度太大
0606 0013	数据类型不匹配, 服务参数长度太短
0609 0011	子索引不存在
0609 0030	超出参数的值范围(写访问时)
0609 0031	写入参数数值太大
0609 0032	写入参数数值太小
0609 0036	最大值小于最小值
0800 0000	一般性错误
0800 0020	数据不能传送或保存到应用
0800 0021	由于本地控制导致数据不能传送或保存到应用
0800 0022	由于当前设备状态导致数据不能传送或保存到应用
0800 0023	对象字典动态产生错误或对象字典不存在 (例如, 通过文件生成对象字典, 但由于文件损坏导致错误产生)

4.3.5 过程数据对象 (PDO)

过程数据对象 (PDO) 用作传输实时数据, 传输模型采用生产者—消费者模型如图1.2所示, PDO的接收者可以是主节点也可以是其它的从节点, 且不需要应答。

在XGate-COP12中最多可支持12个TPDO (索引范围从0x1800~0x180B) 和12个RPDO(0x1400~0x140B), 在出厂时预定义连接集所定义的4个TPDO(索引范围0x1800~0x1803)和4个RPDO(0x1400~0x1403)可用。

1. 过程数据接收 (RPDO)

在XGate-COP12出厂时已经为每个PDO预定义了映射对象, 如图4.1所示, 全部RPDO的数据映射项都默认连接到XGate-COP12 8bit输出区。即当RPDO接收到来自网络的数据之后, 把数据更新到所对应的输出数据区, 当数据更新完成之后, XGate-COP12会给出一个中断信号 (高电平->低电平), 当数据未被读取时中断信号引脚将一址保持低电平, 数据被读出之后中断引脚将保持高电平。

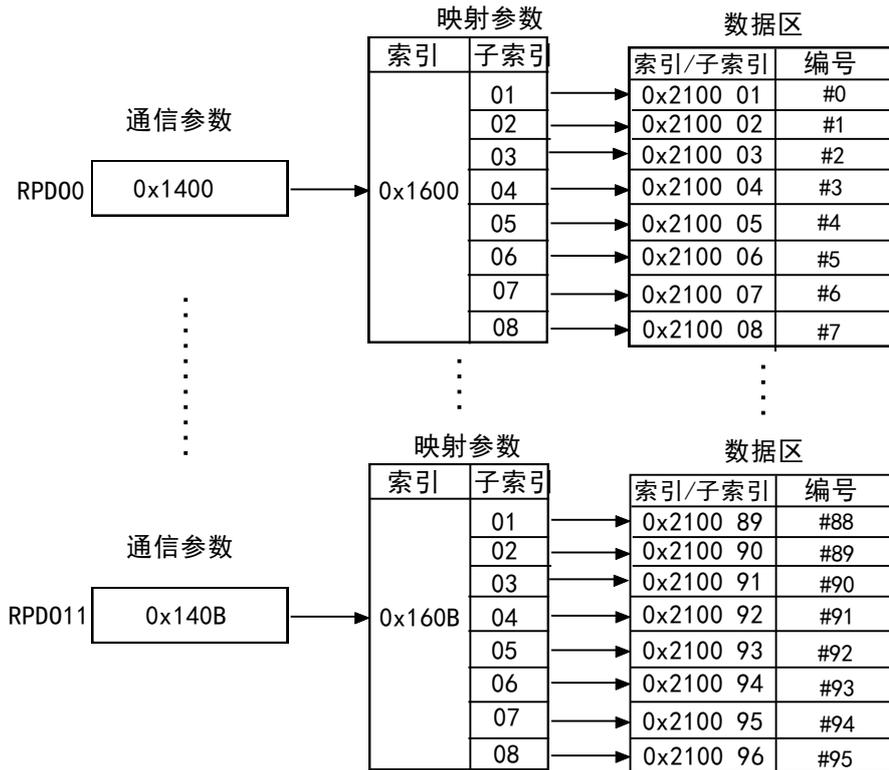


图 4.1 RPDO 映射关系

● 通信参数

RPDO的通信参数相比TPDO来说比较少，RPDO只有传输类型（transmission type）一项对应通信。其值定义如表4.33所示。

表 4.33 RPDO 的传输类型

传输类型	PDO 接收	数据更新
0	PDO 将一直被接收，分析。如果需要，在接收到下个有效的 SYNC 报文时对数据进行更新。	在接收到一个 SYNC 报文时对数据进行分析。如果与之前的 RPDO 相比数据已经更改了，那么数据将在输出上被更新。SYNC 报文的传输是非循环的。
1-240		
241-251	保留	
252	保留	
253		
254	PDO 将一直被接收到。	应用定义更新输出数据的事件。
255	PDO 将一直被接收到。	设备子协议定义更新输出数据的事件。

例：假设XGate-COP12节点为0x20，采用预定义连接，则RPDO0的COB-ID为0x220。则其接收的TPDO COB-D也应为0x220如表4.34所示，该TPDO正好与节点NodeID为0x20的RPDO1的COB-ID相同，则该RPDO接收这帧PDO数据，并且按照图4.1所示的映射关系图把数据更新到数据输出区，最后输出缓冲区对应的数据如表4.35所示。

表 4.34 RPDO0 接收其它节点的 TPDO

COB-ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte4	Byte 5	Byte 6	Byte 7
--------	-----	--------	--------	--------	--------	-------	--------	--------	--------

0x220	8	0x11	0x22	0x33	0x44	0x55	0x66	0x77	0x88
-------	---	------	------	------	------	------	------	------	------

表 4.35 数据区数据

数据区编号	#0	#1	#2	#3	#4	#5	#6	#7
数据	0x11	0x22	0x33	0x44	0x55	0x66	0x77	0x88

2. 过程数据发送 (TPDO)

在XGate-COP12中最多支持12个TPDO，在出厂时预定义的4个PDO可用，即TPDO0~TPDO3，其中TPDO4~TPDO11不可用。预定义的TPDO都已经在出厂时已经预定义了映射参数，分别映射到数据输入区0x2000 01 ~0x2000 96，如图4.2所示。

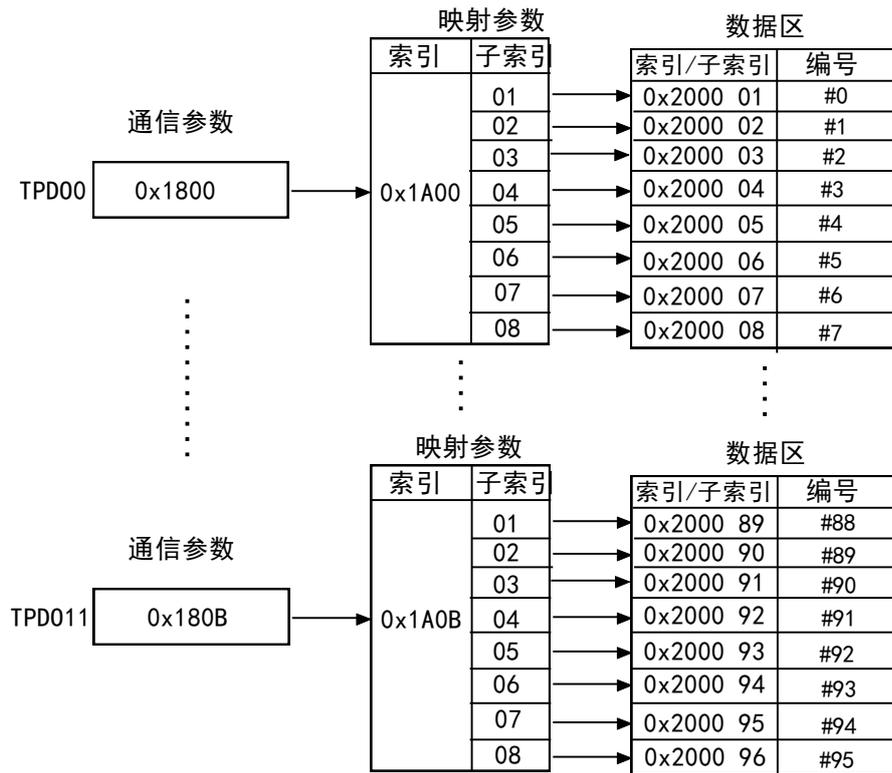


图 4.2 TPDO 映射关系

● 通信参数

每个TPDO都包含有相应的通信参数，这些通信参数决定着TPDO发送的类型以及发送的触发条件等。其中参数主要包含有三种，分别为传输类型、禁止时间以及事件定时。

➤ 传输类型 (Transmission type)

传输类型定义了该TPDO传输方式，通信参数的子索引2定义该对象，具体值定义如表4.36所示。

表 4.36 TPDO 传输类型

传输类型	数据请求	发送 PDO
0	数据（输入值）在接收到一个 SYNC 报文时被读取。	如果与之前的 PDO 内容相比 PDO 数据已经更改了，那么 PDO 将被发送。

1-240	数据接收第 n 个编号的 SYNC 报文时被收集和更新，然后在总线上发送。传输类型对应值 n。	
241-251	保留	
252	数据（输入值）在接收到一个 SYNC 报文时被读取。	PDO 在请求时通过一个远程帧被发送。
253	应用持续收集和更新输入数据。	
254	应用定义引发数据请求和 PDO 传输的事件。造成 PDO 传输的事件可以是事件定时器的时间已到。事件定时器周期用子索引 5 来配置。PDO 传输（与事件和事件定时器是否被配置都无关）总是启动一个新的事件定时器周期。	
255	设备子协议定义引发数据请求和 PDO 传输的事件。造成 PDO 传输的事件可以是事件定时器的时间已到。事件定时器周期用子索引 5 来配置。PDO 传输（与事件和事件定时器是否被配置都无关）总是启动一个新的事件定时器周期。	

➤ 禁止时间(Inhibit Time)

禁止时间的定义是为了防止TPDO发送过于频繁而占用大量的总线带宽，从而影响到总线通信。因而定义了同一个TPDO发送PDO的最短时间间隔（单位为毫秒），当该参数为0时无效，在通信参数子索引3中定义。

➤ 定时时间(Event Time)

定时时间参数定义了该PDO的发送循环时间（单位为毫秒），需要PDO的传输类型设置为254或255，当该参数为0时无效，在通信参数子索引5中定义。

例：假设当前的节点NodeID为0x20，TPDO0定时时间（Event time）参数为1000，传输类型（Transmission type）为254，数据输入区#0~#7号的数据为0x18，则TPDO0发送数据如图4.3所示。

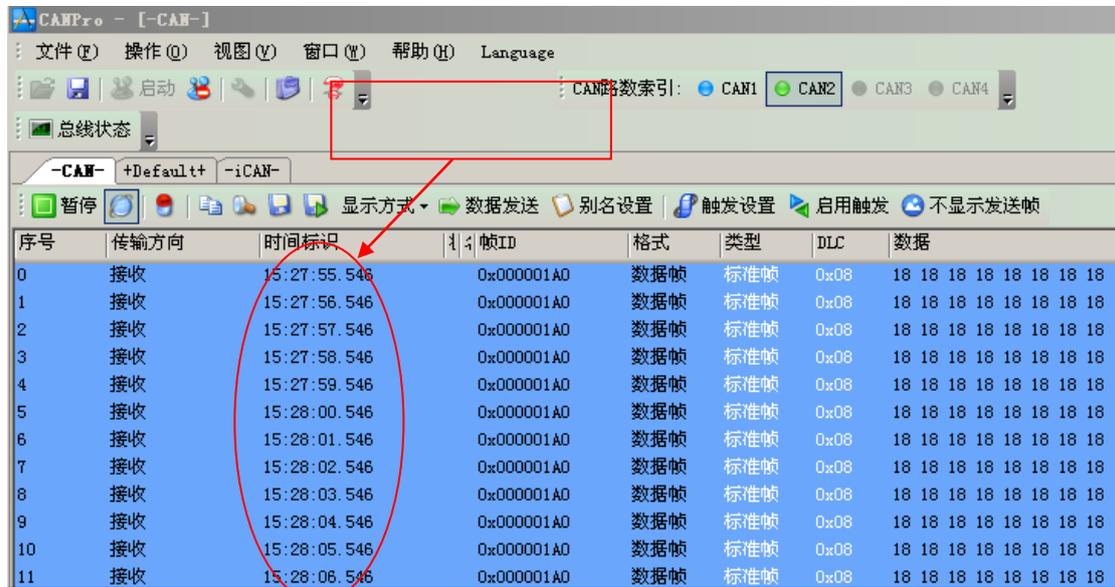


图 4.3 TPDO0 发送数据

5. 输入输出数据缓冲区

XGate-COP12拥有各96字节的输入和输出缓冲区，所有数据均使用小端存取，可按字节进行操作，见表5.1。

XGate-COP12出厂缺省把缓冲区部分内容映射到2个PDO中（详见附录B），用户可使用配套的PC工具重新配置。

注：加“#”号的数据为数据缓冲区的编号，括号内为该缓冲区中的数据。

表 5.1 输入输出数据区对应关系

数据输入区								
#0(0x12)	#1(0x34)	#2(0x56)	#3(0x78)	#4(0x89)	……	#93	#94	#95
数据输出区								
#0(0x12)	#1(0x34)	#2(0x56)	#3(0x78)	#4(0x89)	……	#93	#94	#95

需要注意的是：只有已被PDO映射的过程变量才会分配到这些缓冲区中，未被任何PDO映射的过程变量将不会分配数据缓冲区，也就不能在运行时由主站控制增加映射，需要使用配置工具重新配置。

6. XGate-COP12 串口/SPI 操作

6.1 通信协议

XGate-COP12与用户通信采用异步串口或SPI进行通信，通信模式为半双工，通信信号为TTL电平，通信协议采用广州致远电子股份有限公司自定义串行通信协议。

应答方式：用户设备主动询问（主），XGate-COP12被动回答（从）。

一次完整的数据通讯包含一对主/从应答帧，数据的通讯由主机发起，称为命令帧；从机接收到后进行应答，称为响应帧。

异步串口每1个字节用10bits传送，1个起始位、8个数据位、无奇偶校验位、1个停止位。波特率通过管脚配置，可选波特率见表6.1。

表 6.1 可选串口波特率

索引值	管脚 19	管脚 20	管脚 21	串口
0	0	0	0	9600 bps
1	0	0	1	19200 bps
2	0	1	0	38400 bps
3	0	1	1	57600 bps
4	1	0	0	115200 bps
5	1	0	1	230400 bps
6	1	1	0	460800 bps
7	1	1	1	115200 bps

SPI接口使用模式0（CPOL=0，CPHA=0）或模式3（CPOL=1，CPHA=1），最高速度2Mbit/s，传送宽度8bits，每一次传送都要重新拉起片选，见图6.1和图6.2。模块在复位运行后的前10ms会读取SCK管脚的电平，若为持续低电平则配置为SPI模式0，否则配置为SPI模式3。

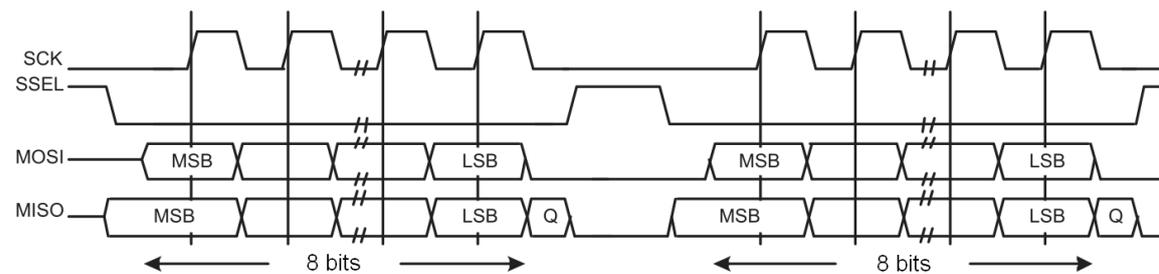


图 6.1 SPI 传输模式 0

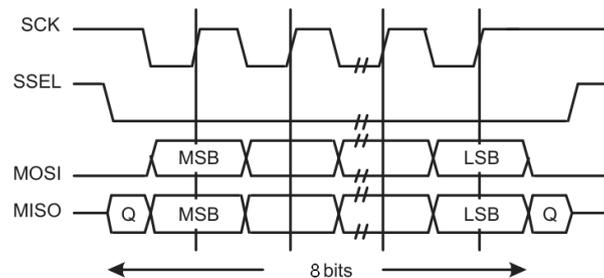


图 6.2 SPI 传输模式 3

SPI通信使用的帧格式定义与串口相同，但是每一帧均需要分为两段传送，帧开头的4字节为第一段，其余字节为第二段，两段之间使用SPI_SYN管脚同步。

模块空闲时SPI_SYN管脚为高电平，由SPI主机发起的一次完整通信流程为：

- 1) 确认模块SPI_SYN管脚为高电平；
- 2) 发送帧头部的4个字节；
- 3) 等待SPI_SYN管脚出现下降沿；
- 4) 发送剩余的字节；
- 5) 等待SPI_SYN管脚再次出现下降沿；
- 6) 读取4个字节；
- 7) 根据上一步读到的结果计算剩余的字节数量；
- 8) 读取剩余的字节。

图6.3展示了一次完整的SPI通信过程，图中的数据详见6.2.4的第二个示例。

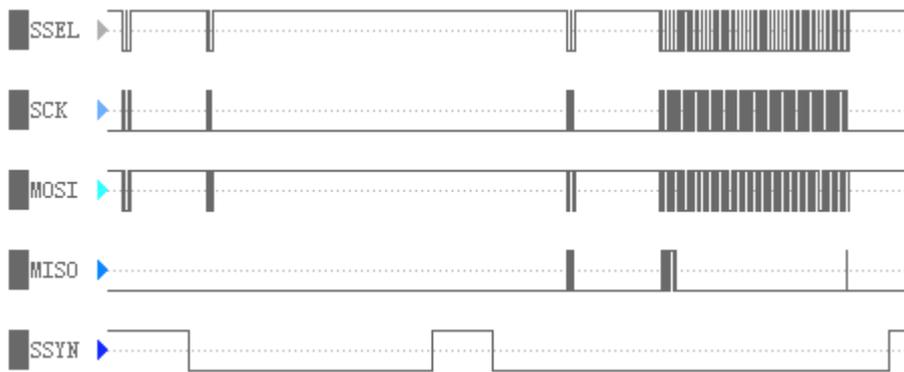


图 6.3 一次完整的 SPI 通信

注：若 SPI 通信失败，SPI 主机可通过连续读取 128 字节复位模块的 SPI 通信功能。

6.1.1 数据帧格式

表 6.2 命令帧格式

起始字	命令码	命令信息(长度)	特定参数	命令数据	校验码
1Byte	1Byte	1Byte	1Byte	nByte	1Byte
0x7E	CMD	CMDinfo	SpeByte	DATA	CRC

表 6.3 应答帧格式

起始字	响应码	响应信息(长度)	特定参数	响应数据	校验码
1Byte	1Byte	1Byte	1Byte	nByte	1Byte
0x7E	ACK	ACKinfo	SpeByte	DATA	CRC

下面按各命令排序，详细介绍命令帧、响应帧的规则。

命令帧、应答帧的总长度为：命令/响应信息的CMDinfo / ACKinfo（数据长度）+ 5Byte，各字段说明如下：

- 帧起始字符 SOF，固定为 0x7E，长度为一个字节。
- 命令 CMD / 响应码 ACK，通常 ACK 与 CMD 相同，长度为一个字节。

- 命令信息 CMDInfo / 响应信息 ACKInfo 指出命令信息/响应信息的长度(字节), 不包括本身。CMDInfo / ACKInfo = 0 表示没有数据, CMDInfo / ACKInfo = 1 表示该帧含有 1 字节数据。
- 特殊参数 SpeByte 包括 Error 和一些保留位, 具体含义见表 6.4。

在命令信息中, Error 为保留位, 通常 Error = 0; 在响应帧中, Error 为错误标识位, Error = 1 表示命令执行出错, DATA 区跟随错误代码, Error = 0, 表明请求成功, DATA 区跟随应答数据。

表 6.4 特定参数 (SpeByte) 定义

BIT.7	BIT.6	BIT.5	BIT.4	BIT.3	BIT.2	BIT.1	BIT.0
Error	0	0	1	0	0	0	1

- 命令 / 响应数据 DATA, 这部分与 CMD / ACK 相结合, 描述数据的具体含义。长度在 CMDInfo / ACKInfo 中说明, 最多为 97 个字节每帧。
- 校验和 CRC: 命令/响应数据的校验和, 长度为一个字节。校验和为前面所有数据的异或值。CRC 的计算公式如下:

$$\text{CRC} = 0x7E \wedge \text{CMD} \wedge \text{CMDInfo} \wedge \text{SpeByte} \wedge \text{DATA}[0] \wedge \text{DATA}[1] \wedge \dots \wedge \text{DATA}[n-1] \text{ 或}$$

$$\text{CRC} = 0x7E \wedge \text{ACK} \wedge \text{ACKInfo} \wedge \text{SpeByte} \wedge \text{DATA}[0] \wedge \text{DATA}[1] \wedge \dots \wedge \text{DATA}[n-1]$$

6.2 XGate-COP12 操作命令

用户通过 XGate-COP12 的通信接口对模块进行操作可使用的命令列举于 6.2.1 节, 本节所有的操作命令都假设被正确执行并返回。如果出现错误, 其错误响应帧和错误代码将在 6.2.11 节中介绍。

6.2.1 读设备信息 (命令码: 0x01)

设备信息默认由 PC 配置工具下载的对象词典设置, 用户可通过串口/SPI 设置覆盖之。相应信息位于对象字典的 0x1000/00 和 0x2404。其中 0x1000/00 为设备类型, 因为该设备是通用设备, 没有使用标准设备描述, 所以该参数按照 CiA 定义应该为 0x00000000, 且不建议更改。操作命令及其响应见表 6.5 和表 6.6。响应帧的数据共 35 字节, 其含义见表 6.7。

注: 该命令读取的内容不是 XGate-COP12 的信息, 而是用于描述用户设备。

表 6.5 读取设备信息命令

起始字节	命令码	命令信息	特定参数	命令数据(1 Byte)	CRC
0x7E	0x01	0x01	0x11	0x00	0x6F

表 6.6 读取设备信息响应

起始字节	命令码	命令信息	特定参数	命令数据(35 Byte)	CRC
0x7E	0x01	0x01	0x11/0x91	0x00 byte[34]	校验码

表 6.7 设备信息数据含义

偏移	数据长度(字节)	对对象词典索引	数据类型	说明
0	1	-	uint8_t	输入的命令数据(0x00)
1	4	0x1000/00	uint32_t	高 2 字节为 XGate-COP12 设备类型, 低 2 字节为用户设备类型代码
5	4	0x2404/01	uint32_t	硬件版本

注：模块的默认 NodeID 为 0x40，CAN 波特率索引为 0。使用 PC 配置工具下载对象词典后均会重置为该值。

表 6.14 CAN 波特率索引对应表

索引	CAN 波特率
0	1 MBit/sec (默认)
1	800 kBit/sec
2	500 kBit/sec
3	250 kBit/sec
4	125 kBit/sec
5	100 kBit/sec
6	50 kBit/sec
7	20 kBit/sec
8	10 kBit/sec

1. 读取 (操作模式: 0x01)

例：读取设置的值 (假设读回的NodeID值为0x40，CAN波特率为1 MBit/sec)。

命令：7E 12 01 11 01 7D

响应：7E 12 03 11 01 40 00 3F

表 6.15 读命令

起始字节	命令码	命令信息	特定参数	命令数据(1 Byte)	CRC
0x7E	0x12	0x01	0x11	操作模式(0x01)	0x7D

表 6.16 读命令响应

起始字节	命令码	命令信息	特定参数	命令数据(2 Byte)		CRC
0x7E	0x12	2	0x11	操作模式(0x00)	DAT(NodeID)	校验码

2. 写入 (操作模式: 0x00)

表 6.17 写命令

起始字节	命令码	命令信息	特定参数	命令数据(2 Byte)			CRC
0x7E	0x12	3	0x11	0x00	NodeID	CAN Baud	校验码

表 6.18 写命令响应

起始字节	命令码	命令信息	特定参数	命令数据(n Byte)	CRC
0x7E	0x12	n	0x11	DAT	校验码

例：设置XGate-COP12模块的NodeID为0x40，CAN波特率为125 kBit/sec。

命令：7E 12 03 11 00 40 04 3A

响应：7E 12 01 11 00 7C

6.2.6 发送紧急代码 (命令码: 0x15)

当用户设备出现某种错误之后，可通过CANopen发送到CAN总线上，通知CANopen主站设备当前设备发生了错误。错误代码由5个字节组成，由用户自定义。其格式见表6.19和表6.20。

表 6.19 发送紧急错误代码命令

起始字节	命令码	命令信息	特定参数	命令数据(6 Byte)		CRC
0x7E	0x15	0x06	0x11	操作模式 (0x00)	byte[5]	校验码

表 6.20 发送紧急错误代码命令响应

起始字节	命令码	命令信息	特定参数	命令数据(nByte)	CRC
0x7E	0x15	n	0x11	DAT	校验码

例：假设当前设备发生错误，错误代码定义为0x0504030201，则其命令格式如下所示。

命令：7E 15 06 00 00 01 02 03 04 05 6C

响应：7E 15 01 11 00 7B

6.2.7 读取模块状态（命令码：0x16）

用户通过该命令可以读取模块所处的状态和对象词典的状态。读取回来的节点值与当前状态的对应关系见表4.9。模块在复位后会根据PC配置工具下载的配置生成对象词典，若无异常则词典状态为0x0000；0x0070~0x007F表示PDO相关参数有误。其命令格式如表6.21和表6.22所示。

表 6.21 获取当前 XGate-COP12 状态

起始字节	命令码	命令信息	特定参数	命令数据(1 Byte)	CRC
0x7E	0x16	0x01	0x11	操作码(0x01)	0x79

表 6.22 获取当前 XGate-COP12 状态响应

起始字节	命令码	命令信息	特定参数	命令数据(4 Byte)			CRC
0x7E	0x16	2	0x11	操作码(0x01)	节点状态	词典状态	校验码

例：假设当前XGate-COP12模块处于操作状态，读取的节点状态为0x05，词典状态0x0000表示无异常。

命令：7E 16 01 11 01 79

响应：7E 16 04 11 01 05 00 00 79

6.2.8 启动节点进入操作状态（命令码：0x17）

用户可通过该条命令使CANopen网络中的所有从站设备进入到操作状态，其中包括模块本身也会进入到操作状态。其命令格式如表6.23和表6.24所示。

注：在网络中有主站管理的情况下，谨慎使用此条命令。

表 6.23 使能从站进入操作状态

起始字节	命令码	命令信息	特定参数	命令数据(1 Byte)	CRC
0x7E	0x17	0x01	0x11	操作码(0x00)	0x79

表 6.24 使能从站进入操作状态响应

起始字节	命令码	命令信息	特定参数	命令数据(1Byte)	CRC
0x7E	0x17	0x01	0x11	操作码(0x00)	0x79

例：使能当前CANopen网络进入到操作状态。

命令：7E 17 01 11 00 79

响应：7E 17 01 11 00 79

6.2.9 复位模块（命令码：0x24）

发送该条命令将强制模块执行复位操作。其命令格式如表6.23和表6.24所示。

表 6.25 复位命令

起始字节	命令码	命令信息	特定参数	命令数据(1 Byte)	CRC
0x7E	0x24	0x01	0x11	操作码	校验码

表 6.26 复位命令响应

起始字节	命令码	命令信息	特定参数	命令数据(1Byte)	CRC
0x7E	0x24	0x01	0x11	操作码	校验码

例：利用看门狗在1秒后模块。

命令：7E 24 01 11 00 4A

响应：7E 24 01 11 00 4A

例：仅复位CANopen协议栈。

命令：7E 24 01 11 01 4B

响应：7E 24 01 11 01 4B

6.2.10 读取 XGate-COP12 软硬件版本（命令码：0x25）

通过串口用户可以读取XGate-COP12的硬件版本和软件版本。

表 6.27 读取设备信息命令

起始字节	命令码	命令信息	特定参数	命令数据(1 Byte)	CRC
0x7E	0x25	0x01	0x11	操作码	校验码

表 6.28 读取设备信息响应

起始字节	命令码	命令信息	特定参数	命令数据(1 Byte)	CRC
0x7E	0x25	0x08	0x11	操作码 char[7]	校验码

1. 读取硬件版本(操作模式：0x01)

XGate硬件版本位于对象字典的0x1009/00，数据长度7字节（包括字符串结束符），存储的是版本号的ASCII码值。

例：如下命令读回硬件版本为56 31 2E 30 2E 30 00，即V1.0.0。

命令：7E 25 01 11 01 4A

响应：7E 25 08 11 01 56 31 2E 30 2E 30 00 24

2. 读取软件版本(操作模式：0x02)

XGate软件版本位于对象字典的0x100A/00，数据长度7字节（包括字符串结束符），存储的是版本号的ASCII码值。

例：如下命令读回设备类型为56 31 2E 30 2E 30 00，即V1.0.0。

命令：7E 25 01 11 02 49

响应：7E 25 08 11 02 56 31 2E 30 2E 30 00 27

6.2.11 重置内部 E²PROM (命令码: 0xED)

XGate-COP12使用内部E²PROM保存设置设置参数，如对象词典、NodeID等。发送该条命令将使模块清除这些参数。其命令格式如表6.23和表6.24所示。

表 6.29 重置命令

起始字节	命令码	命令信息	特定参数	命令数据(1 Byte)	CRC
0x7E	0xED	0x01	0x11	操作码	校验码

表 6.30 重置命令响应

起始字节	命令码	命令信息	特定参数	命令数据(1Byte)	CRC
0x7E	0xED	0x01	0x11	操作码	校验码

例：清除E²PROM内容。

命令：7E ED 02 11 EE 00 6E

响应：7E ED 01 11 EE 6D

例：重建E²PROM存储器。

命令：7E ED 02 11 EE 01 6F

响应：7E ED 01 11 EE 6D

注：XGate-COP12内部无硬件E²PROM，该功能使用Flash模拟实现，一般使用第一条示例即可。若长期使用后发现存储功能异常可使用第二条示例尝试修复。

6.3 XGate-COP12 串口/SPI 操作错误响应

在所有的串口/SPI操作命令中，通信过程中当命令的参数不正确或其它错误发生时，XGate-COP12都会返回错误代码（特定参数的最高位为1，表示当前为错误应答帧），错误代码帧格式如表6.31所示。其中ACK与操作的命令码相同，应答操作模式码与当前命令操作模式码相同。错误代码表示当前的操作所出现错误类别，错误代码表如表6.32所示。

表 6.31 XGate-COP12 命令执行错误响应

起始字	响应码	响应信息(长度)	特定参数	响应数据 (2 字节)		校验码
0x7E	ACK=CMD	0x02	0x91	操作模式	错误代码	CRC

表 6.32 错误代码表

错误代码(数据)	说明	备注
0x01	命令错误	不支持该命令
0x02	数据长度错误	超出可写区域
0x03	地址错误	超出地址范围
0x04	在操作协议栈时出现错误	-
0x05	存储数据出错	-
0x06	数据值超出范围	-
0x07	操作模式不支持	-

例：假设现在写XGate-COP12的CAN波特率索引值为10。因为波特率索引值的范围为0~8，所以该值超出范围无效，必然在执行过程中出错，错误代码为0x06(数据值超出范围)。命令帧与响应帧如下所示。

命令：7E 12 03 11 00 40 0A 34



响应: 7E 12 02 91 00 06 F9

附录A 串口可操作对象字典列表

操作名称	对对象字典索引/子索引	备注
读写设备类型	0x1000/00	设备类型(不建议更改, 非标准设备)
设备名称	0x2404/06	XGate-COP12/用户可更改(字符串, 11 字节)
硬件版本	0x2404/01	长度 4 字节
软件版本	0x2404/02	长度 4 字节
产品代码	0x2404/03	长度 4 字节
修订码	0x2404/04	长度 4 字节
产品代码	0x2404/05	长度 4 字节
设备名称	0x2404/06	XGate-COP12/用户可更改(字符串, 11 字节)
数据输入缓冲区	0x2000~0xAFFF	该区域对应于 CANopen 的输入缓冲区
数据输出缓冲区	(视实际配置的词典而定)	该区域对应于 CANopen 的输出缓冲区
通信串口波特率索引值	0x2405/00	通信串口的当前波特率

附录B XGate-COP12 缺省对象字典

索引 Index	子索引 Subindex	名称 Name	类型 Type	属性 Attr.	默认值 Deaf.	描述 Desc.
通信参数区						
0x1000	-	Device Type	UINT32	RO	0x00000000	设备类型, 高2字节为致远电子专用, 低2字节供用户使用
0x1001		Error Register	UINT8	RO	0	当前错误类型
0x1003	0	number of errors	UINT8	RO	0	-
	1~4	standard error field	UINT32	RO	0	历史紧急错误代码
0x1005	-	COB-ID SYNC	UINT32	RW	0x80	-
0x1006	-	Communication cycle period	UINT32	RW	0x3E8	-
0x1007	-	Sync Windows Length	UINT32	RW	0x3E8	-
0x1008	-	XGate name	STRING	Const	XGate-COP12	XGate 设备名称
0x1009	-	XGate hardware version	STRING	Const	V1.00	XGate 硬件版本
0x100A	-	XGate software version	STRING	Const	V1.00	XGate 软件版本
0x100C	-	Guard Time	UINT16	RW	0	-
0x100D	-	Life Time Factor	UINT8	RW	0	-
0x1010	0	largest supported Sub-Index	UINT8	RO	1	-
	1	save all parameters	UINT32	RW	0	-
0x1011	0	largest supported Sub-Index	UINT8	RO	1	-
	1	restore all default para.	UINT32	RW	0	-
0x1014	-	COB-ID Emergency message	UINT32	RW	NodeID+0x80	-
0x1016	0	Number Of Entries	UINT8	RO	0x01	-
	1	Consumer Heartbeat Time #1	UINT32	RW	0x64	-
0x1017	-	Producer Heartbeat Time	UINT16	RW	0	-
0x1018	0	number of Entries	UINT8	RO	0x04	-

	1	Vendor ID	UINT32	RO	0x2B6	致远电子在 CiA 组织的厂商代码 (可更改)
	2	Product code	UINT32	RO	0xB	XGate-COP12 产品代码
	3	Revision number	UINT32	RO	0xA	XGate-COP12 修订码
	4	Serial number	UINT32	RO	-	XGate-COP12 序列码
RPDO 通信参数						
0x1400	0	largest subindex supported	UINT8	RO	2	-
	1	COB-Id used	UINT32	RW	NodeID+0x200	RPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	254	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	Eventtimer	UINT16	RW	0	传输 PDO 定时时间
RPDO 映射参数						
0x1600	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21000108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21000208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21000308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21000408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21000508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21000608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21000708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21000808	映射参数
TPDO 通信参数						
0x1800	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x180	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	254	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
TPDO 映射参数						
0x1A00	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20000108	映射参数

	2	PDO mapping 2. app. object	UINT32	RW	0x20000208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0xA0000210	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x50000220	映射参数
数据输入区 (TPDO 发送数据映射区)						
0x2000	0	Number Of Entries	UINT8	RO	8	输入数据(1 字节)区的长度
	1	Application data Input 8bit #0	UINT8	RW	-	编号为#0, 长度为 1 字节
	2	Application data Input 8bit #1	UINT8	RW	-	编号为#1, 长度为 1 字节
	3	Application data Input 8bit #2	UINT8	RW	-	编号为#2, 长度为 1 字节
	4~8	#3~#7	UINT8	RW	-	编号为#n, 长度为 1 字节
0xA000	0	Number Of Entries	UINT8	RO	2	输入数据(1 字节)区的长度
	1	Application data Input 16bit #0	UINT16	RW	-	编号为#0, 长度为 2 字节
	2	Application data Input 16bit #1	UINT16	RW	-	编号为#1, 长度为 2 字节
0x5000	0	Number Of Entries	UINT8	RO	2	输入数据(1 字节)区的长度
	1	Application data Input 32bit #0	UINT32	RW	-	编号为#0, 长度为 4 字节
	2	Application data Input 32bit #1	UINT32	RW	-	编号为#1, 长度为 4 字节
数据输出区 (RPDO 接收到的数据映射区)						
0x2100	0	Number Of Entries	UINT8	RO	8	输出数据(1 字节)区的长度
	1	Application data Output 8bit #0	UINT8	RW	-	编号为#0, 长度为 1 字节
	2	Application data Output 8bit #1	UINT8	RW	-	编号为#1, 长度为 1 字节
	3	Application data Output 8bit #2	UINT8	RW	-	编号为#2, 长度为 1 字节
	4~8	#3~#7	UINT8	RW	-	编号为#n, 长度为 1 字节
设备状态						
0x2404	0	Number Entries	UINT8	RO	6	-

	1	Device Hardware Version	UINT32	RO	-	用户设备的硬件版本
	2	Device Software Version	UINT32	RO	-	用户设备软件版本
	3	Device Product Code	UINT32	RO	-	用户设备产品代码
	4	Device Revision Num	UINT32	RO	-	用户设备修订码
	5	Device SN	UINT32	RO	-	用设备序列号
	6	Device Name	STRING	RO	-	用户设备名称()
0x2405	0	Uart Baudrate	UINT8	RO	0x07	通信串口波特率

附录C XGate-COP12 对象字典样例

索引 (Index)	子索引 (Subindex)	名称 (Name)	类型(Type)	属性 (Attr.)	默认值 (Deaf.)	描述 (Desc.)
通信参数区						
0x1000	-	Device Type	UINT32	RO	0x00000000	设备类型，高2字节为致远电子专用，低2字节供用户使用
0x1001	-	Error Register	UINT8	RO	0	当前错误类型
0x1003	0	number of errors	UINT8	RO	0	-
	1~4	standard error field	UINT32	RO	0	历史紧急错误代码
0x1005	-	COB-ID SYNC	UINT32	RW	0x80	-
0x1006	-	Communication cycle period	UINT32	RW	0x3E8	-
0x1007	-	Sync Windows Length	UINT32	RW	0	-
0x1008	-	XGate name	STRING	Const	XGate-COP12	XGate 设备名称
0x1009	-	XGate hardware version	STRING	Const	V1.00	XGate 硬件版本
0x100A	-	XGate software version	STRING	Const	V1.00	XGate 软件版本
0x100C	-	Guard Time	UINT16	RW	0	-
0x100D	-	Life Time Factor	UINT8	RW	0	-
0x1010	0	largest supported Sub-Index	UINT8	RO	1	-
	1	save all parameters	UINT32	RW	0	-
0x1011	0	largest supported Sub-Index	UINT8	RO	1	-
	1	restore all default para.	UINT32	RW	0	-
0x1014	-	COB-ID Emergency message	UINT32	RW	NodeID+0x80	-
0x1016	0	Number Of Entries	UINT8	RO	0x01	-
	1	Consumer Heartbeat Time #1	UINT32	RW	-	-
0x1017	-	Producer Heartbeat Time	UINT16	RW	0	-
0x1018	0	number of Entries	UINT8	RO	0x04	-
	1	Vendor ID	UINT32	RO	0x2B6	致远电子在 CiA 组织的厂商代码

						(可更改)
	2	Product code	UINT32	RO	-	XGate-COP12 产品代码
	3	Revision number	UINT32	RO	-	XGate-COP12 修订码
	4	Serial number	UINT32	RO	-	XGate-COP12 序列码
RPDO 通信参数						
0x1400	0	largest subindex supported	UINT8	RO	2	-
	1	COB-Id used	UINT32	RW	NodeID+0x200	RPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	-
0x1401	0	largest subindex supported	UINT8	RO	2	-
	1	COB-Id used	UINT32	RW	NodeID+0x300	RPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	-
0x1402	0	largest subindex supported	UINT8	RO	2	-
	1	COB-Id used	UINT32	RW	NodeID+0x400	RPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	-
0x1403	0	largest subindex supported	UINT8	RO	2	- RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
0x1404	0	largest subindex supported	UINT8	RO	2	- RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
0x1405	0	largest subindex supported	UINT8	RO	2	- RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
0x1406	0	largest subindex supported	UINT8	RO	2	- RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
0x1407	0	largest subindex supported	UINT8	RO	2	- RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
0x1408	0	largest subindex supported	UINT8	RO	2	- RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
0x1409	0	largest subindex	UINT8	RO	2	-

		supported				RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
0x140A	0	largest subindex supported	UINT8	RO	2	- RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
0x140B	0	largest subindex supported	UINT8	RO	2	- RPDO 所使用的 COB-ID
	1	COB-Id used	UINT32	RW	NodeID+0x500	
	2	transmission type	UINT8	RW	0xFE	-
RPDO 映射参数						
0x1600	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21000108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21000208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21000308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21000408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21000508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21000608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21000708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21000808	映射参数
0x1601	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21000908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21000A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21000B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21000C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21000D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21000E08	映射参数

		app. object				
	7	PDO mapping 7. app. object	UINT32	RW	0x21000F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21001008	映射参数
0x1602	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21001108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21001208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21001308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21001408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21001508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21001608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21001708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21001808	映射参数
0x1603	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21001908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21001A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21001B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21001C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21001D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21001E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21001F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21002008	映射参数
0x1604	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量

	1	PDO mapping 1. app. object	UINT32	RW	0x21002108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21002208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21002308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21002408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21002508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21002608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21002708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21002808	映射参数
0x1605	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21002908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21002A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21002B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21002C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21002D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21002E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21002F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21003008	映射参数
0x1606	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21003108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21003208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21003308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21003408	映射参数

	5	PDO mapping 5. app. object	UINT32	RW	0x21003508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21003608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21003708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21003808	映射参数
0x1607	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21003908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21003A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21003B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21003C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21003D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21003E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21003F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21004008	映射参数
0x1608	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21004108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21004208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21004308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21004408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21004508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21004608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21004708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21004808	映射参数

0x1609	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21004908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21004A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21004B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21004C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21004D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21004E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21004F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21005008	映射参数
0x160A	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21005108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21005208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21005308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x21005408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21005508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21005608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21005708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21005808	映射参数
0x160B	0	number of mapped objects	UINT8	RO	8	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	0x21005908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x21005A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x21005B08	映射参数

	4	PDO mapping 4. app. object	UINT32	RW	0x21005C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x21005D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x21005E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x21005F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x21006008	映射参数
TPDO 通信参数						
0x1800	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x180	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x1801	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x280	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x1802	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x380	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x1803	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x800004 80	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x1804	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x800001 C0	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间

0x1805	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x800002 C0	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x1806	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x800003 C0	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x1807	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x800004 C0	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x1808	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x800001 E0	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x1809	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x800002 E0	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x180A	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	NodeID+0x800003 E0	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
0x180B	0	largest subindex	UINT8	RO	0x05	-

		supported				
	1	COB-ID used	UINT32	RW	NodeID+0x800004 E0	TPDO 所使用的 COB-ID
	2	transmission type	UINT8	RW	0xFE	传输类型
	3	inhibit time	UINT16	RW	0	传输 PDO 禁止时间
	5	event timer	UINT16	RW	0	传输 PDO 定时时间
TPDO 映射参数						
0x1A00	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20000108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20000208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20000308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20000408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20000508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20000608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20000708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20000808	映射参数
0x1A01	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20000908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20000A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20000B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20000C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20000D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20000E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20000F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20001008	映射参数

0x1A02	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20001108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20001208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20001308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20001408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20001508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20001608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20001708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20001808	映射参数
0x1A03	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20001908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20001A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20001B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20001C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20001D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20001E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20001F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20002008	映射参数
0x1A04	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20002108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20002208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20002308	映射参数

	4	PDO mapping 4. app. object	UINT32	RW	0x20002408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20002508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20002608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20002708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20002808	映射参数
0x1A05	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量，最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20002908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20002A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20002B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20002C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20002D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20002E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20002F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20003008	映射参数
0x1A06	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量，最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20003108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20003208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20003308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20003408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20003508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20003608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20003708	映射参数

	8	PDO mapping 8. app. object	UINT32	RW	0x20003808	映射参数
0x1A07	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20003908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20003A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20003B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20003C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20003D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20003E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20003F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20004008	映射参数
0x1A08	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20004108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20004208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20004308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20004408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20004508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20004608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20004708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20004808	映射参数
0x1A09	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量, 最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20004908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20004A08	映射参数

	3	PDO mapping 3. app. object	UINT32	RW	0x20004B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20004C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20004D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20004E08	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20004F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20005008	映射参数
0x1A0A	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量，最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20005108	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20005208	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20005308	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20005408	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20005508	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20005608	映射参数
	7	PDO mapping 7. app. object	UINT32	RW	0x20005708	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20005808	映射参数
0x1A0B	0	number of mapped objects	UINT8	RO	8	TPDO 映射参数数量，最大为 8 个
	1	PDO mapping 1. app. object	UINT32	RW	0x20005908	映射参数
	2	PDO mapping 2. app. object	UINT32	RW	0x20005A08	映射参数
	3	PDO mapping 3. app. object	UINT32	RW	0x20005B08	映射参数
	4	PDO mapping 4. app. object	UINT32	RW	0x20005C08	映射参数
	5	PDO mapping 5. app. object	UINT32	RW	0x20005D08	映射参数
	6	PDO mapping 6. app. object	UINT32	RW	0x20005E08	映射参数

	7	PDO mapping 7. app. object	UINT32	RW	0x20005F08	映射参数
	8	PDO mapping 8. app. object	UINT32	RW	0x20006008	映射参数
数据输入区 (TPDO 发送数据映射区)						
0x2000	0	Number Of Entries	UINT8	RO	96	输入数据(1 字节)区的长度
	1	Application data Input 8bit #0	UINT8	RW	-	编号为#0, 长度为 1 字节
	2	Application data Input 8bit #1	UINT8	RW	-	编号为#1, 长度为 1 字节
	3	Application data Input 8bit #2	UINT8	RW	-	编号为#2, 长度为 1 字节
	4~96	#3~#95	UINT8	RW	-	编号为#n, 长度为 1 字节
数据输出区 (RPDO 接收到的数据映射区)						
0x2100	0	Number Of Entries	UINT8	RO	96	输出数据(1 字节)区的长度
	1	Application data Output 8bit #0	UINT8	RW	-	编号为#0, 长度为 1 字节
	2	Application data Output 8bit #1	UINT8	RW	-	编号为#1, 长度为 1 字节
	3	Application data Output 8bit #2	UINT8	RW	-	编号为#2, 长度为 1 字节
	4~96	#3~#95	UINT8	RW	-	编号为#n, 长度为 1 字节
设备状态						
0x2404	0	Number Entries	UINT8	RO	6	-
	1	Device Hardware Version	UINT32	RO	-	用户设备的硬件版本
	2	Device Software Version	UINT32	RO	-	用户设备软件版本
	3	Device Product Code	UINT32	RO	-	用户设备产品代码
	4	Device Revision Num	UINT32	RO	-	用户设备修订码
	5	Device SN	UINT32	RO	-	用设备序列号
	6	Device Name	STRING	RO	-	用户设备名称()
0x2405	0	Uart Baudrate	UINT8	RO	0x07	通信串口波特率