

# NDAM-9020

## CANopen 通讯模块

UM05050106 V1.01 Date: 2013/09/02

产品用户手册

类别	内容
关键词	NDAM-9020 CANopen 从站
摘要	NDAM-9020 使用指南

## 修订历史

版本	日期	原因
V1.00	2013/04/05	创建文档
V1.01	2013/09/01	第一次发布

## 目 录

1. NDAM-9020 简介 .....	1
1.1 主要技术指标 .....	2
1.1.1 通讯参数 .....	2
1.1.2 系统参数 .....	2
1.1.3 电源参数 .....	2
1.2 原理框图 .....	2
1.3 引脚信息 .....	3
1.3.1 硬件说明 .....	3
1.3.2 引脚排列 .....	3
1.3.3 端子描述 .....	4
1.4 电气参数 .....	4
1.5 机械规格 .....	5
1.5.1 机械尺寸 .....	5
2. NDAM-9020 硬件接线和设置 .....	7
2.1 电源接线 .....	7
2.2 CAN通讯接线 .....	7
2.3 通讯参数配置 .....	8
2.3.1 CAN波特率设置 .....	8
2.3.2 节点ID设置 .....	8
3. NDAM-9020 的通讯端口 .....	10
3.1 NDAM-9020 的CANopen协议 (DS301) .....	10
3.1.1 NDAM-9020 的预定义连接集 .....	10
3.1.2 网络管理服务 (NMT) .....	11
3.1.3 同步报文对象(SYNC) .....	13
3.1.4 紧急报文对象(EMCY) .....	13
3.1.5 服务数据对象(SDO) .....	15
3.1.6 过程数据对象 (PDO) .....	18
4. NDAM-9020 的配置与使用实例 .....	23
4.1 搜索设备 .....	23
4.2 查看设备信息 .....	23
4.3 用USB调试设备 .....	24
4.4 PDO映射关系配置和EDS文件生成 .....	25
4.4.1 RPDO映射配置 .....	25
4.4.2 TPDO映射配置 .....	26
4.4.3 下载配置并生成EDS文件 .....	27
4.5 CANopen主站卡操作NDAM-9020 模块 .....	28
4.6 普通CAN卡操作NDAM-9020 .....	30
4.6.1 配置PDO映射关系 .....	30
4.6.2 通过CAN卡启动从站 .....	31
4.6.3 设置TPDO传输参数 .....	32
4.6.4 设置RPDO数据 .....	33

附录 1	NDAM-9020 通用对象字典 .....	35
附录 2	RPDO回读区对象字典 .....	42
附录 3	NDAM模块通道定义 .....	44
附录 4	NDAM模块配置区对象字典 .....	46
	NDAM-1804 配置区对象字典 .....	46
	NDAM-2808 配置区对象字典 .....	46
	NDAM-4055 配置区对象字典 .....	47
	NDAM-3402 配置区对象字典 .....	48
	NDAM-3412 配置区对象字典 .....	49
	NDAM-3800 配置区对象字典 .....	50
	NDAM-4017 配置区对象字典 .....	50
	NDAM-4400 配置区对象字典 .....	51
	NDAM-4410 配置区对象字典 .....	52
	NDAM-5508 配置区对象字典 .....	53
	NDAM-6608 配置区对象字典 .....	55
	NDAM-7404 配置区对象字典 .....	57
5.	免责声明.....	60

## 1. NDAM-9020 简介

现场总线 CAN-bus 作为现场总线的一员，凭借其可靠、实时、灵活、经济的特点，目前在自动化，汽车，船舶，工程车辆等诸多行业被广泛使用。CANopen 协议是在现场总线 CAN-bus 之上定义的一套应用层协议。从 CANopen 推出到现在，已经成为电梯网络、运动控制、工程机械等行业的通用标准。

NDAM-9020 CANopen 从站模块主要用于分布式数据采集系统中。作为 CANopen 通讯模块与现有的 NDAM 数据采集模块一起组成基于 CANopen 的工业控制现场数据采集终端。该终端可兼容几乎所有的 PLC 和 PAC 供应商提供的 CANopen 接口产品。

NDAM-9020 外观结构图如下图 1.1 所示：



图 1.1 NDAM-9020 外观示意图

## 1.1 主要技术指标

### 1.1.1 通讯参数

- ◆ CAN 通讯接口
- ◆ 通讯协议：CANopen
- ◆ 通讯速率：支持 10K~1M 标准波特率
- ◆ 节点有效地址：1~127
- ◆ 免驱的 USB 配置接口，用户可通过 USB 口配置 NDAM 模块的 IO 参数，并设置 PDO 与 NDAM 采集模块的通道映射动态生成 EDS 文件
- ◆ 隔离耐压：2000VDC

### 1.1.2 系统参数

- ◆ CPU：32 位 RISC ARM
- ◆ 操作系统：实时操作系统
- ◆ 供电电压：+10~+30V<sub>DC</sub>，电源反接保护
- ◆ 工作温度范围：-40℃~+85℃
- ◆ 塑料外壳，标准 DIN 导轨安装

### 1.1.3 电源参数

- ◆ 输入电压范围：10~30VDC
- ◆ 保护：过压保护、过流保护、防反接保护
- ◆ 功率：<2W

## 1.2 原理框图

NDAM-9020 CANopen 通讯模块采用 32 位 ARM 处理器，使用实时操作系统实现软件控制，具有非常快速的数据处理能力。能够实时的响应外部通讯命令。

NDAM-9020 模块硬件电路包含 NDAM 通讯接口驱动电路、CAN 通讯电路，电源、CPU 最小系统等几部分。模块内部结构图如下图 1.2 所示。

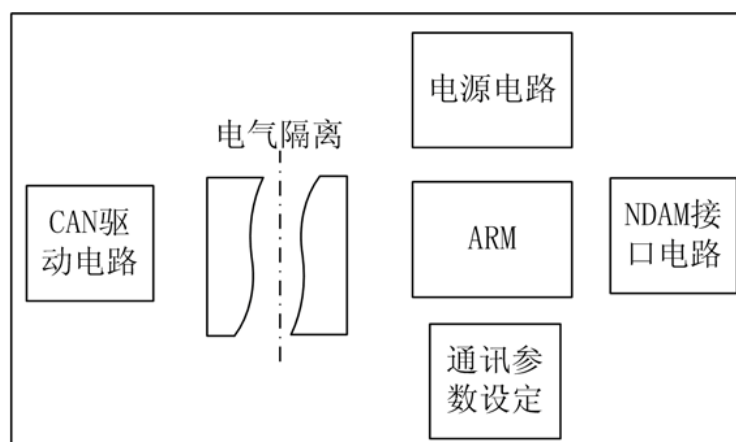


图 1.2 NDAM-9020 原理框图

## 1.3 引脚信息

### 1.3.1 硬件说明

NDAM-9020 CANopen 通讯模块正面视图如图 1.3 所示。该部分包括 LED 电源和状态指示，CANopen 接口等。



图 1.3 NDAM-9020 面板

工作状态显示部分由 2 个 LED 组成，分别是 State 和 Power。

按照 CANopen 协议规范文档 DS303-3 的定义，在 NDAM-9020 模块中使用红绿双色 LED 指示灯来指示当前模块所处的状态即 State 指示灯，指示灯具体含义可参见 CANopen 协议规范中关于指示灯的定义。

波特率和节点地址设置使用编码开关实现，具体含义参加 2.3.2。

### 1.3.2 引脚排列

NDAM-9020 只有电源端子引出，具体如下图 1.4 示。

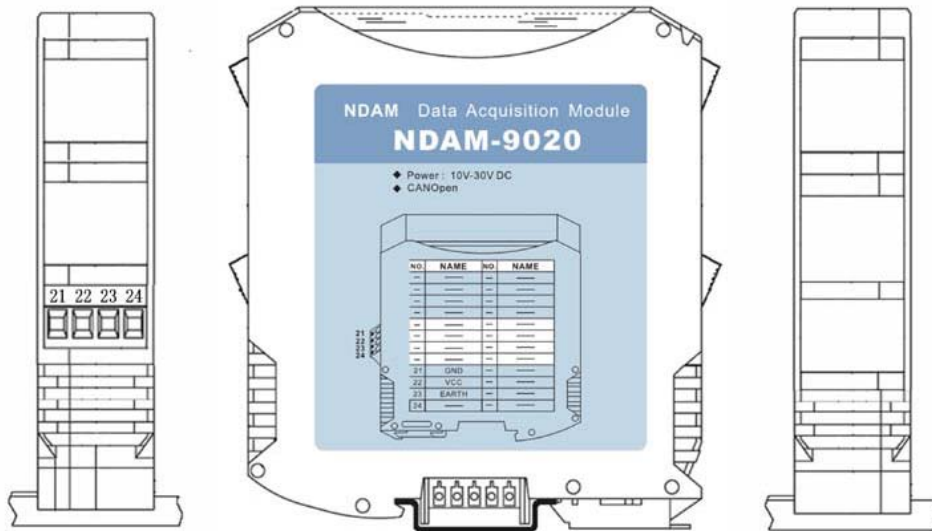


图 1.4 NDM-9020 端子排列

### 1.3.3 端子描述

NDAM-9020 共有三个输入输出端子，端子的编号和具体的名称与功能如表 1.1 所示。

表 1.1 端子定义

类型	端口号	引脚名	说明
电源输入	21	GND	电源地输入端
	22	VCC	电源正输入端
	23	EARTH	大地输入端

### 1.4 电气参数

除非特别说明，下表所示参数为温度  $T=25^{\circ}\text{C}$  时测得。

表 1.2 电气参数

参数	Parameter	最小值 Min	典型值 Typ	最大值 Max	单位 Unit
网络参数	CAN-bus				
带宽	Data transfer Rate	10		1000	kbps
隔离电压	Isolation Voltage		2000	2500	Vdc
电源输入	Power Input				
输入电压	Input Voltage	10	24	30	Vdc
电源纹波峰峰值	$V_{p-p}$ of Power supply Ripple			5	V
功耗	Power consumption		1.2	2	W



## 1.5 机械规格

### 1.5.1 机械尺寸

NDAM 系列产品均采用工业级塑料外壳，尺寸大小为  $114.5 \times 99 \times 22.5\text{mm}$ ，如图 1.5 所示。由于导轨端子为自堆叠形式，所以安装在导轨上以后会多占用  $7\text{mm}$  的导轨。

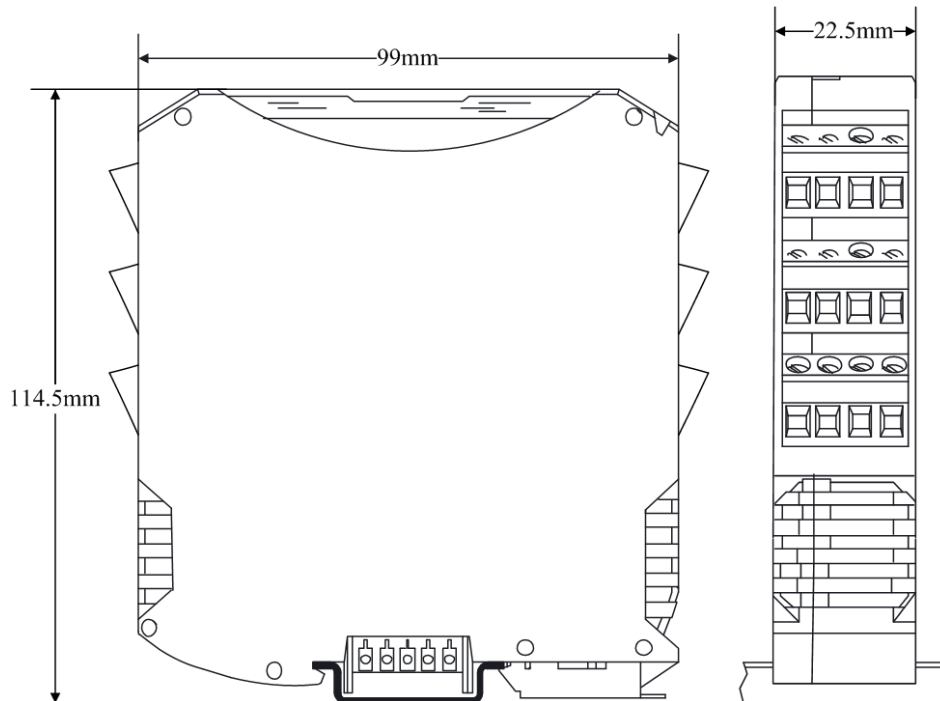


图 1.5 模块尺寸图

首先，将专用的导轨端子叠起来安装在标准 DIN 导轨（ $35\text{mm}$  宽 D 型导轨）的中间。辅助安装螺纹应该在下，如图 1.6 中红色框所示。

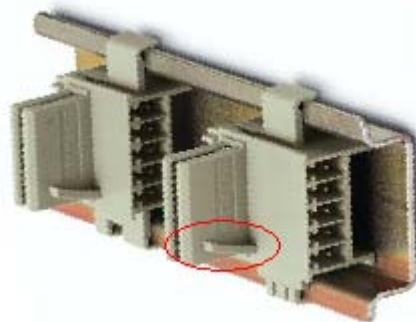


图 1.6 导轨端子的安装

然后，将 NDAM 模块卡到导轨端子上。需先用模块钩住导轨的上边沿，然后对准安装辅助螺纹，往下按即可把模块装在导轨上，图 1.7 为安装过程示意图。

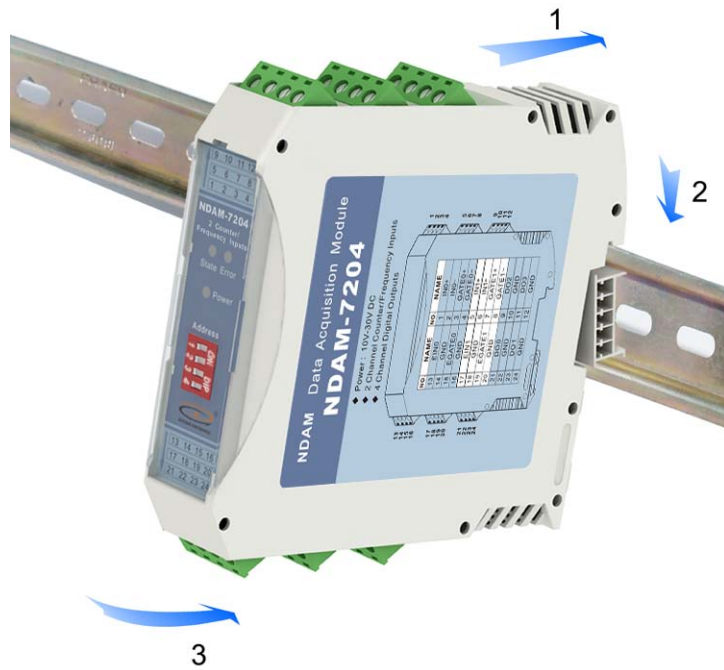


图 1.7 安装方法

最终，多个 NDM 模块组合安装如图 1.8 所示。



图 1.8 NDM 模块组合安装图

## 2. NDAM-9020 硬件接线和设置

### 2.1 电源接线

NDAM-9020 供电电源可选择 10~30V 的直流电源，电压接入到 VCC 和 GND 端子上，如图 2.1 所示：

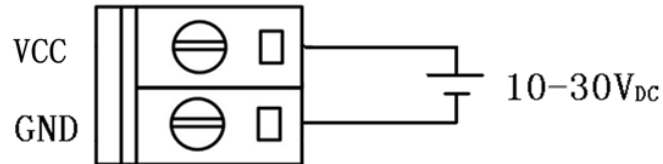


图 2.1 电源连接图

### 2.2 CAN 通讯接线

NDM-9020 模块的 CAN 接口采用标准的 9 针 D-sub 连接器输出，其引脚分配如下图 2.2 所示：

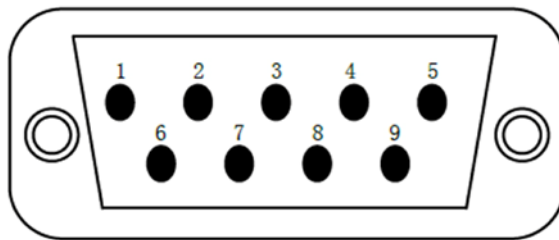


图 2.2 9 针 D-Sub 连接器示意图

9 针 D-Sub 引脚分配如下表 2.1 所示：

表 2.1 9 针 D-Sub 连接器引脚分配

引脚	信号	说明
1	--	保留
2	CAN_L	CAN 总线导线
3	CAN_GND	CAN 总线接地线
4	--	保留
5	CAN_SHLD	CAN 导线屏蔽层
6	--	保留
7	CAN_H	CAN 总线导线
8	--	保留
9	--	保留

## 2.3 通讯参数配置

NDAM-9020 模块的 CAN 通讯参数可以通过面板上的旋转编码开关进行设置，面板配置示意图如下图 2.3 示：

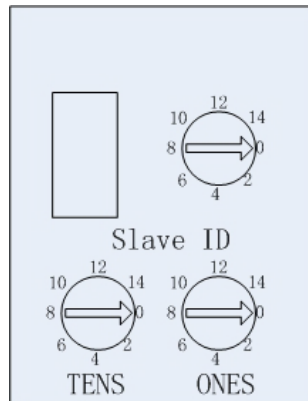


图 2.3 模块配置面板

### 2.3.1 CAN 波特率设置

通过旋转 BAUD RATE 下的编码开关，可设置 NDAM-9020 的 CAN 波特率。开关设置值与波特率对应关系如下表 2.2 所示：

表 2.2 波特率设置开关与标准波特率对应关系

波特率索引值	标准波特率
0	1Mbps
1	800Kbps
2	500Kbps
3	250Kbps
4	125Kbps
5	100Kbps
6	50Kbps
7	20Kbps
8	10Kbps
9~15	无效

当波特率设置开关设置值大于 8 时，波特率设置开关无效，将使用默认的波特率 (500Kbps)。

### 2.3.2 节点 ID 设置

通过旋转 Slave ID 下面的编码开关设置模块节点 ID，节点 ID 设置的有效地址为 1~127，其中 TENS 表示节点 ID 的十位，ONES 表示为节点 ID 的个位。如设置节点地址为 124 则编码开关设置如下图 2.4 所示，节点开关设置超过正常范围，模块会自动复位不能正常工作。请保证开关设置有效。

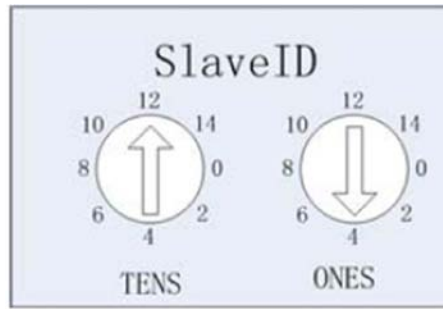


图 2.4 ID 设置示意图

### 3. NDAM-9020 的通讯端口

NDAM-9020 有两个通讯端口，与 NDAM 系列数据采集模块之间的 NDAM 通讯端口，以及 CANopen 通讯端口，其中与 NDAM 模块之间的通讯对用户来说完全透明的，用户无需关心。下面只对 CANopen 通讯端口做介绍。

#### 3.1 NDAM-9020 的 CANopen 协议 (DS301)

##### 3.1.1 NDAM-9020 的预定义连接集

在 NDAM-9020 中使用了对象字典的 0x1000~0x1FFF 和厂商自定义区 0x2000~0x5FFF 的区域。这些对象字典负责着 CANopen 与 CAN 网络上的其它应用数据的通信和数据交换，对象字典使用了索引与子索引来定义，每个对象字典项都有自己的数据长度（UIN8，UINT16，UIN32 等）和属性（RO、WO、RW、CONST、MAPPAL）。这些对象字典的数据可以通过 SDO 服务来修改，当然也只有这些项的属性必须是 WO 或 RW 才能修改。

通常情况下，在一个典型的 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。所谓的预定义连接是指与通信相关的 COB-ID 与节点 ID 相关联。

NDAM-9020 的预定义连接集如下表 3.1 所示，

表 3.1 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
TIME STAMP	0010	-	0x100	0x1012, 0x1013
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

注：NDAM-9020 通过 USB 配置 PDO 映射时，最多使能 4 个 TPDO 和 4 个 RPDO，配置的 PDO 条目超过此数目时，则超出的 PDO 条目禁能但映射关系会保存。在使用时需要 CANopen 主机使能相关 PDO 并分配 COB-ID。

### 3.1.2 网络管理服务 (NMT)

#### 1. 网络控制 (NMT Module Control)

NDAM-9020 支持 DS301 所定义的网络管理命令，这些网络管理命令可以是 CANopen 主站发出也可以是其它的从节点发出。其操作命令如下表 3.2 所示，其中当 Node\_ID=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 3.3 所示。

表 3.2 NMT 控制命令

COB-ID(CAN-ID)	DLC	BYTE0	BYTE1
0x000	2	CS(命令字)	NodeID(节点号)

表 3.3 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 3.4 所示的命令。

表 3.4 NMT 启动从节点

COB-ID(CAN-ID)	DLC	BYTE0	BYTE1
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 3.5 所示。

表 3.5 NMT 启动从节点

COB-ID(CAN-ID)	DLC	BYTE0	BYTE1
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输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 3.6 所示。

主节点→从节点(命令):

表 3.6 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 3.7 NMT 从节点应答帧

COB-ID(CAN-ID)	DLC	BYTE0
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 3.8 所示。

表 3.8 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 3.9 和表 3.10 所示。

主节点→从节点:

表 3.9 保护节点(远程帧)

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

从节点→主节点:

表 3.10 从节点(0x20)应答帧

COB-ID(CAN-ID)	DLC	BYTE0
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 Gardting)

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 3.6), 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 3.7 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 NDAM-9020 初始化完成(Boot-up), 就会发送一个标识报文, 其报文格式如表 3.11 所示。



表 3.11 初始化完成标识报文格式

COB-ID(CAN-ID)	DLC	BYTE0
0x700 +NodeID	1	0x00

例：假设 NDAM-9020 的节点号为 0x20，则发送的启动文如表 3.12 所示。

表 3.12 初始化完成标识报文

COB-ID(CAN-ID)	DLC	BYTE0
0x720	1	0x00

### 5. 心跳报文 (Heartbeat Producer)

心跳报文分为生产者和消费者，在 NDAM-9020 模块中只支持心跳报文生产，即 NDAM-9020 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 3.7 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 3.13 所示的报文。

表 3.13 从节点(0x20)心跳报文

COB-ID(CAN-ID)	DLC	BYTE0
0x720	1	0x05

### 3.1.3 同步报文对象(SYNC)

同步报文分为消费和生产，在 NDAM-9020 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 3.14 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 3.14 同步报文帧格式（远程帧）

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

### 3.1.4 紧急报文对象(EMCY)

在 NDAM-9020 中支持紧急报文，即在 NDAM-9020 内部出现错误时发送该报文，其报文格式如表 3.15 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 3.16 所示。

表 3.15 紧急报文帧格式

COB-ID(CAN-ID)	DLC	BYTE0	BYTE1	BYTE2	BYTE3 – BYTE7
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 3.16 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 3.17 所示。

表 3.17 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，则 NDAM-9020 发送紧急报文如表 3.18 所示。

表 3.18 紧急报文（总线错误）

COB-ID(CAN-ID)	DLC	BYTE0	BYTE1	BYTE2	BYTE3 – BYTE7
0xA0	8	0x20	0x81	0x11	0x00000000

注意：NDAM-9020 模块发生紧急情况，这些错误主动发出。

### 3.1.5 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。

NDAM-9020 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 NDAM-9020 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 3.19 和表 3.20 所示，通过 SDO 的命令字来区分该帧数据的类型。

表 3.19 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 3.20 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 3.21 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 3.22 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 36 相同。

例：假设当前 NDAM-9020 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 3.23 和表 3.24 所示。

表 3.23 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 3.24 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 3.25 和表 3.26 所示。

表 3.25 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 3.26 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 的客户端和服务器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 3.27 和表 3.28 所示。

表 3.27 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 3.28 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 3.29 和表 3.30 所示。

表 3.29 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 3.30 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

注意：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 3.31。

表 3.31 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)

### 3.1.6 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 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 NDAM-9020 中最多可支持 16 个 TPDO (索引范围从 0x1800~0x180F) 和 8 个 RPDO(0x1400~0x1407), 通过 USB 配置 PDO 映射参数时最多使能 4 个 TPDO(索引范围 0x1800~0x1803)和 4 个 RPDO(0x1400~0x1403)。超过部分需要主机使能。

#### 1. 过程数据接收 (RPDO)

在 NDAM-9020 出厂时已经为每个 PDO 预定义了映射对象, 如图 3.1 所示, 当模块收到 RPDO 数据后, 会根据 RPDO 与 NDAM 模块的通道的映射关系将 RPDO 的数据的发送到 NDAM 模块对应通道的寄存器中实现输出功能。整个输出过程对 CANopen 主机来说都是透明的。

NDAM 数据采集模块通道数据长度分为 16bit 和 32bit 两种, 为方便与 NDAM 模块实现通道映射 RPDO 的长度类型也分为 16bit 和 32bit 两种。其中 16bit 数据区和 32bit 数据区在物理上占用同一内存区。每次只能使用一种确定长度的数据, 如 0x2100 01 被映射到某个 NDAM 模块的 16bit 通道, 则 0x2101 01 这个 32bit 缓冲区将不可用。同理如 0x2101 01 被

映射到某个 NDAM 模块的 32bit 通道则 0x2100 01 和 0x2100 02 这两个 16bit 缓冲区将不能使用。

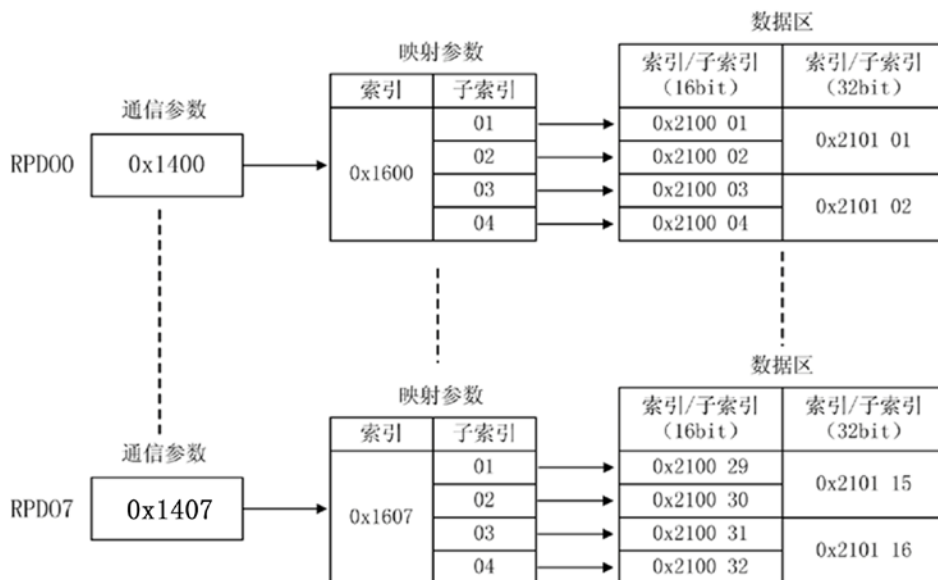


图 3.1 NDAM-9020 的 RPDO 映射关系

● 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 3.32 所示。

表 3.32 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		在接收到第 n 个编号的 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。传输类型与值 n 相对应。SYNC 报文的传输是循环的。
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 NDAM-9020 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-ID 也应为 0x220 如表 3.33 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 7 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 3.34 所示。

表 3.33 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 3.34 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. RPDO 数据的回读

CANopen 主机设置 RPDO 后，可以通过 RPDO 的回读功能来实现读取实际输出值。如果在 RPDO 混合映射的情况下，如果不希望修改 RPDO 中某子项的值，可以先回读 RPDO 的当前值，对不想修改的子项可以按原值赋值即可。例如：RPDO1 设置了四个子项，如下表 3.35 所示：

表 3.35 RPDO1 数据分配示例

RPDO1_1	RPDO1_2	RPDO1_3	RPDO1_4
DATA_A	DATA_B	DATA_C	DATA_D

如果用户只想修改 RPDO1\_1 的值，其它值需要保持不变，因为 CANopen 协议规定操作 RPDO1 是整体操作的，即写 RPDO1 时所有的子项都要赋值。所有用户必须先知道其它子项的当前值才能防止误操作。

RPDO 数据的回读是通过 SDO 读特定对象字典来实现。RPDO 回读区对象字典如附录 2 所示。

## 3. 过程数据发送 (TPDO)

在 NDAM-9020 中最多支持 16 个 TPDO，出厂时已经为模块预定义了映射关系，如下图 3.2 所示。NDAM-9020 定时(10ms)获取所带 NDAM 数据采集模块输入通道的数据，并根据 TPDO 的映射参数将数据上传到 CANopen 主机。

NDAM 数据采集模块通道数据长度分为 16bit 和 32bit 两种，为方便与 NDAM 模块实现通道映射。NDAM-9020 的 TPDO 的长度类型也分为 16bit 和 32bit 两种。其中 16bit 数据区和 32bit 数据区在物理上占用同一内存区。每次只能使用一种确定长度的数据，如 0x2000 01 被映射到某个 NDAM 模块的 16bit 通道，则 0x2001 01 这个 32bit 缓冲区将不可用。同理如 0x2001 01 被映射到某个 NDAM 模块的 32bit 通道则 0x2000 01 和 0x2000 02 这两个 16bit 缓冲区将不能使用。



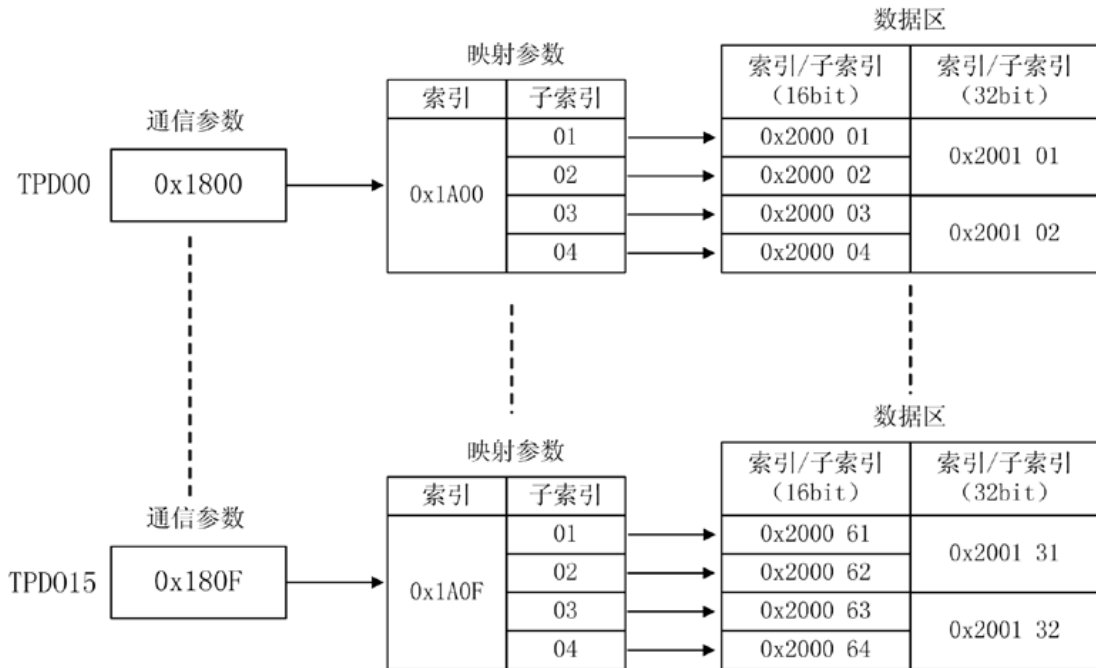


图 3.2 TPDO 映射关系

● 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 3.36 所示。NDAM-9020 出厂默认为 254。

表 3.36 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。NDAM-9020 出厂默认为 100。

➤ 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。NDAM-9020 出厂默认为 0。

## 4. NDAM-9020 的配置与使用实例

NDAM-9020 作为通用的 NDAM 模块转 CANopen 的协议转换器，根据用户使用的 NDAM 模块型号的不同需要用户通过 USB 配置所带 NDAM 模块通道与 PDO 的映射关系。并生成对应的 EDS 文件 CANopen 主机调用。配置使用 USBNDAMUtility 软件实现。下面介绍用户的配置与调试过程。

### 4.1 搜索设备

模块上电，插入 USB 连接线，打开软件，在主界面上点击“搜索设备”，在弹出的搜索框中会显示搜索进度和搜索到的 NDAM 采集模块。搜索到所有模块后点击“停止”按钮完成搜索。在软件左边的树形列表中会显示搜到的模块。

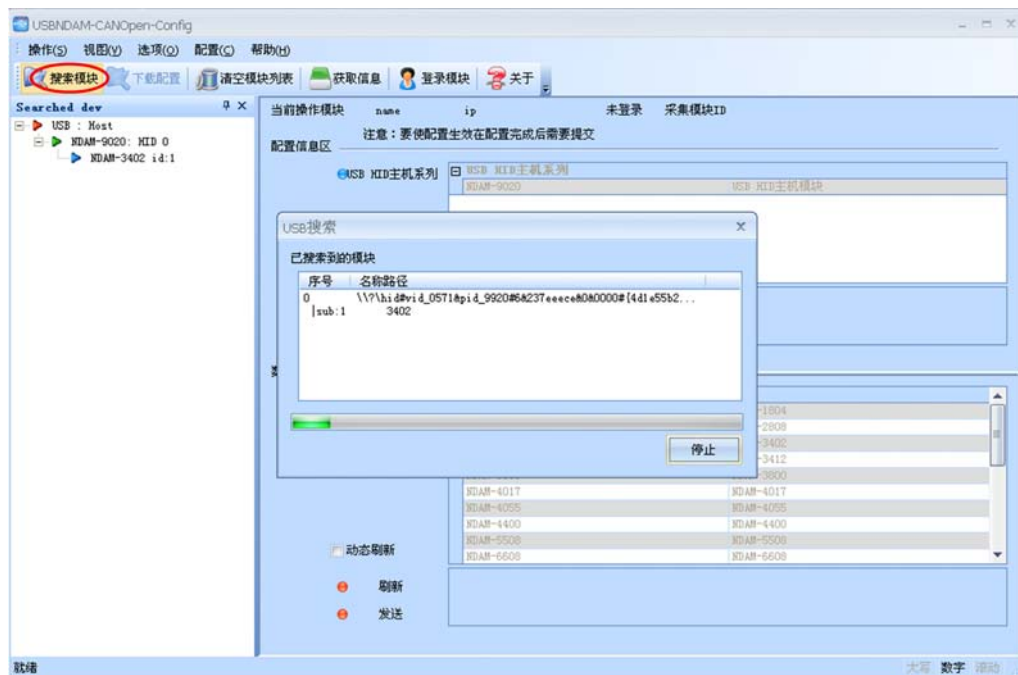


图 4.1 搜索设备

### 4.2 查看设备信息

在软件左边的树形下拉列表中点击选中“NDAM-9020: HID 0”会显示设备的基本信息如下图 4.2 所示，软件界面信息栏上半部分显示的为 NDAM-9020 的版本信息以及节点 ID 和波特率参数，下半部分显示的是所带 NDAM 数据采集模块的站地址和版本信息等。



图 4.2 查看设备信息

### 4.3 用 USB 调试设备

在设备的先期调试中,可以通过 USB 来对数据采集模块进行属性配置和 IO 数据的读写来确定现场连线及信号是否正常。下面以 NDAM-3402 为例介绍调试过程。

在模块树形下拉列表中单击“NDAM-3402N id:1”。打开 NDAM-3402 监测界面。如下图 4.3 所示。

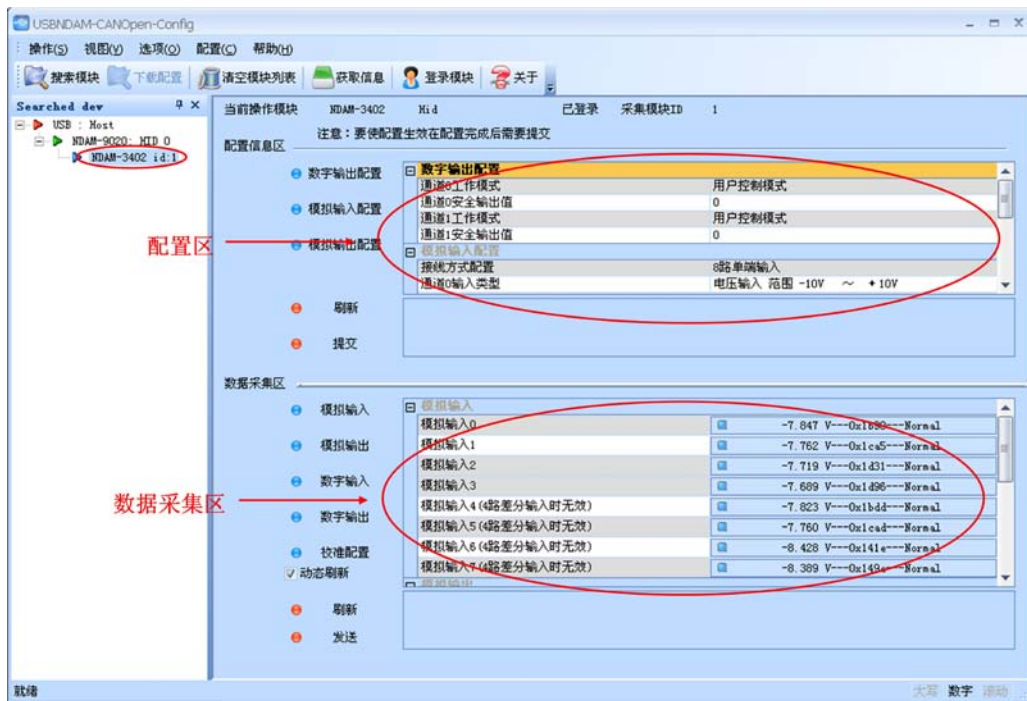


图 4.3 数据采集模块设备调试

监测界面上部分是模块的 IO 配置区,用户可以根据使用情况设置模块的 IO 属性,如通道量程,输出模式等。如果用户在使用过程中不在改变配置属性,那么可以在 USB 环境

产品用户手册 ©2013 Guangzhou ZHIYUAN Electronics Stock Co., Ltd.

下配置好属性参数，采集模块会自动保存且掉电不丢失。在 CANopen 通讯环境中当从站使用时则不需要再通过 SDO 修改 IO 属性从而简化 CANopen 主站的操作流程。

监测界面下部分为数据采集区，选中“动态刷新”会实时显示各个通道的采样数据。

不同模块的配置属性及数据区数据具体可参见各数据采集模块的手册。

#### 4.4 PDO 映射关系配置和 EDS 文件生成

在 CANopen 协议中，过程数据对象是通过 PDO 的读写来实现的。为了方便用户使用，用户只需要将 NDAM 模块的 IO 通道映射到 PDO 中，那么在 CANopen 主机下直接对相应的 PDO 操作即等同于对 NDAM 模块通道的操作。

PDO 映射关系配置如下图 4.4 所示。

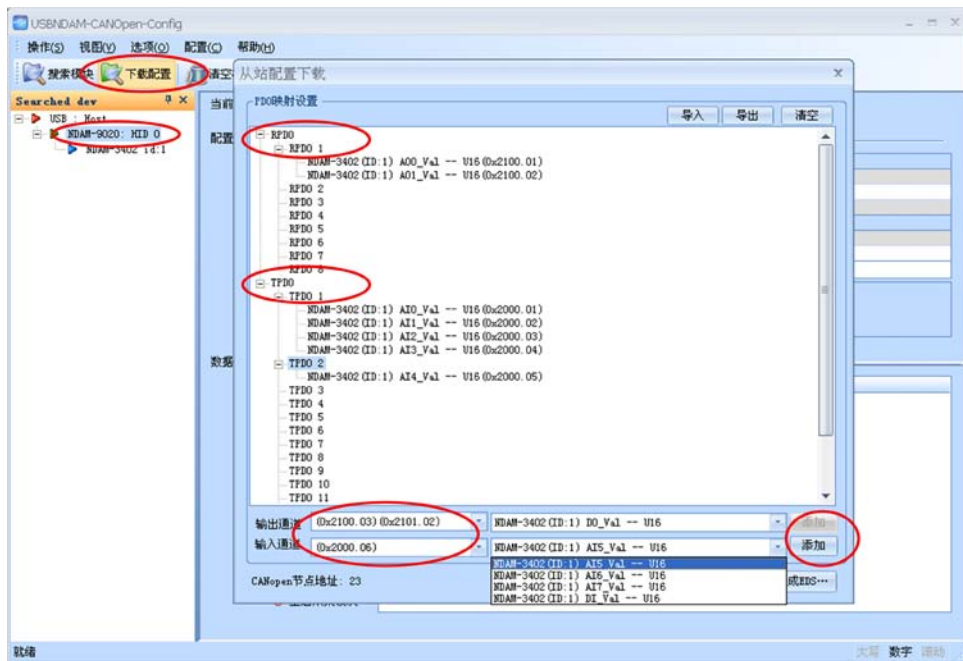


图 4.4 PDO 映射关系配置

在模块列表中选中 NDAM-9020，然后点击菜单栏中的“下载配置”按钮。弹出从“从站配置下载”界面，软件会自动将所带 NDAM 数据采集模块的 IO 通道添加到通道下拉表中(NDAM-9020 支持的模块通道数据附录 3 所示)。用户根据使用的情况添加通道映射到 PDO 中。下面介绍 TPDO 和 RPDO 的映射配置过程。

##### 4.4.1 RPDO 映射配置

RPDO 为 CANopen 中接收的过程数据。对应的是 NDAM 模块的输出通道。在使用中根据要使用的 NDAM 输出通道进行配置。配置流程如下：

- 1、选择 RPDO 根节点。如需要添加 IO 通道到 RPDO1 则单击选中 RPDO1 做为根节点。
- 2、选择缓冲区。在输出通道一行中选择数据缓冲区的对象字典索引和子索引无特殊要求一般为默认值即可。
- 3、选中 IO 通道。在输出通道一栏后面的 NDAM 模块通道下拉框中选中您要添加的 NDAM 模块通道。如 NDAM-3402 的 AO0 通道。
- 4、添加映射。点击“添加”按钮完成通道映射。

添加 RPDO 映射完成后如下图 4.5 所示：

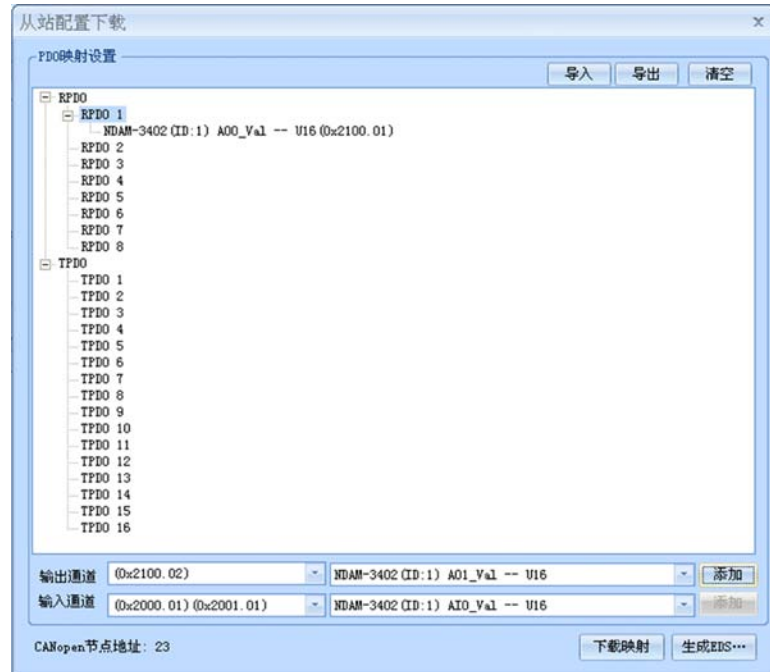


图 4.5 RPDO 通道映射

注：对于同一个型号的不同 ID 的模块，根据模块后面括号中的 ID 号来区分。如上图中添加的为 ID=1 的 NDAM-3402 的 A0 通道到 RPDO1。

#### 4.4.2 TPDO 映射配置

TPDO 为 CANopen 中发送的过程数据。对应的是 NDAM 模块的输入通道。在使用中根据要使用的 NDAM 的输入通道进行配置。配置流程如下：

- 1、选择 TPDO 根节点。如需要添加 IO 通道到 TPDO1 则单击选中 TPDO1 做为根节点。
- 2、选择缓冲区。在输入通道一行中选择数据缓冲区的对象字典索引和子索引无特殊要求一般为默认值即可。
- 3、选中 IO 通道。在输入通道一栏后面的 NDAM 模块通道下拉框中选中您要添加的 NDAM 模块通道。如 NDAM-3402 的 AI 通道 0。
- 4、添加映射。点击“添加”按钮完成通道映射。

添加 TPDO 映射完成后如下图 4.6 所示：

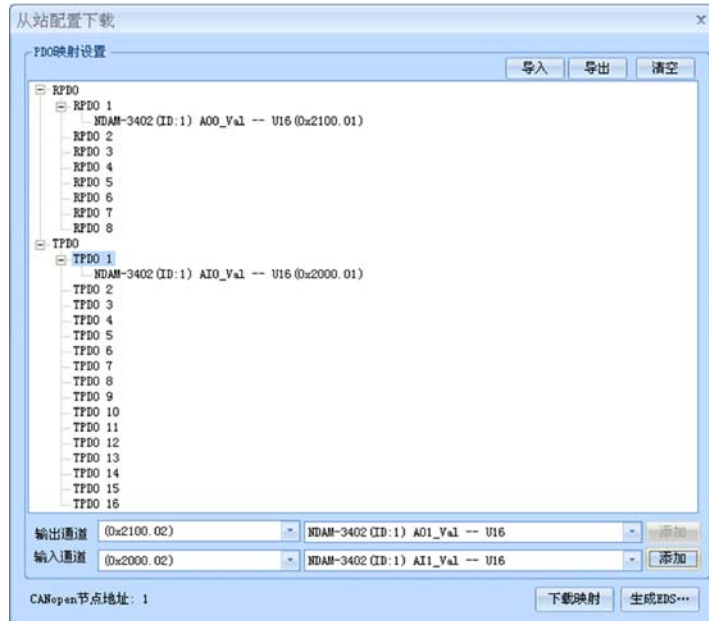


图 4.6 TPDO 通道映射

注：对于同一个型号的不同 ID 的模块，根据模块后面括号中的 ID 号来区分。如上图添加的为 ID=1 的 NDAM-3402 的 AI 通道 0 到 TPDO1。

#### 4.4.3 下载配置并生成 EDS 文件

按照上面介绍的方法将需要使用的 IO 通道映射到 PDO 条目中。如下图 4.7 所示，添加完毕后点击“下载映射”按钮。映射参数将下载到 NDAM-9020 中并自动保存，掉电不丢失。再点击“生成 EDS 文件”按钮，在弹出的文件保存框中，选择路径并保存，如下图 4.8 所示。我们将其保存为 NDAM\_9020\_1.eds。EDS 文件供 CANopen 主机使用。

配置完成后重启模块，配置生效。

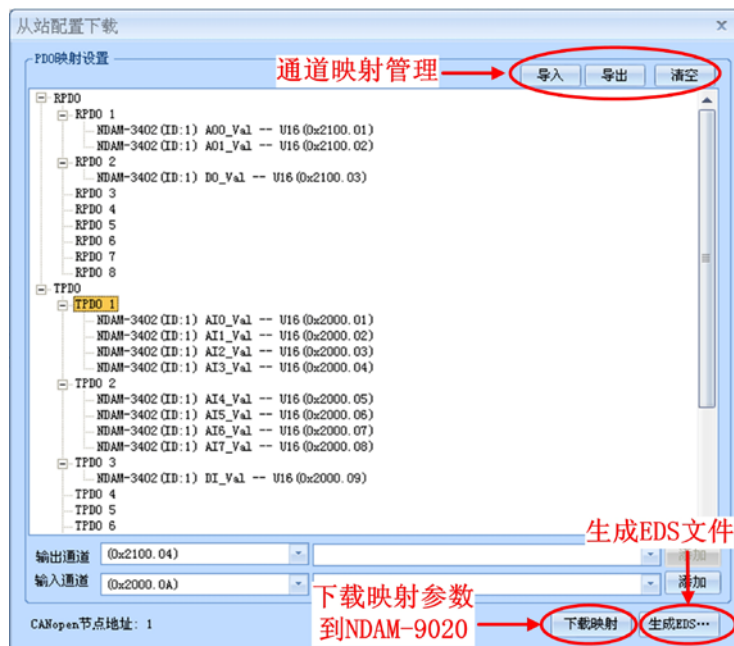


图 4.7 PDO 映射关系配置

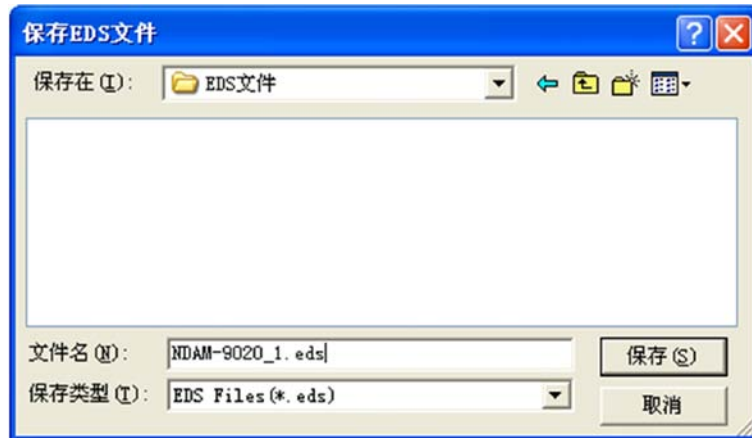


图 4.8 生成 EDS 文件

软件还提供简单的通道映射管理功能，用户在同样的通道使用情况下，可以配置好一次后将通道数据导出到 xml 文件中。下一次可以直接导入相应的 xml 文件，实现通道映射。

#### 4.5 CANopen 主站卡操作 NDAM-9020 模块

在 CANopen 主站卡中使用先前生成的 EDS 文件即可对 NDAM-9020 进行操作。下面以本公司的 USBCAN-E-P 主站卡为例介绍操作流程。

打开 CANmanage for CANopen 软件，点击界面中的“导入 EDS 文件”按钮，在弹出的文件选择框中选择刚生成的 NDAM-9020\_1.eds 文件。

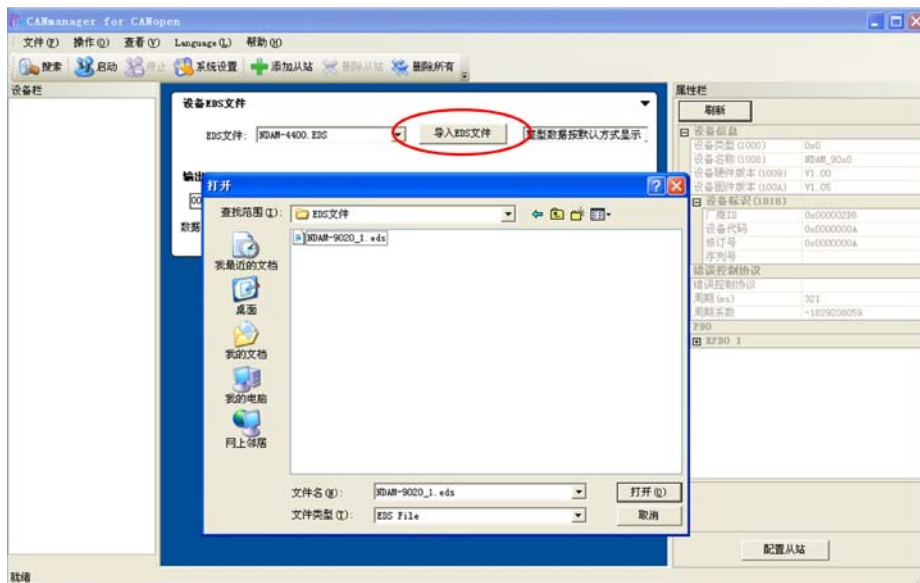


图 4.9 导入 EDS 文件

点击“系统设置”按钮，弹出如下图 4.10 的系统设置面板，根据通讯参数配置好主站卡的现相关参数然后点击“确定”按钮。



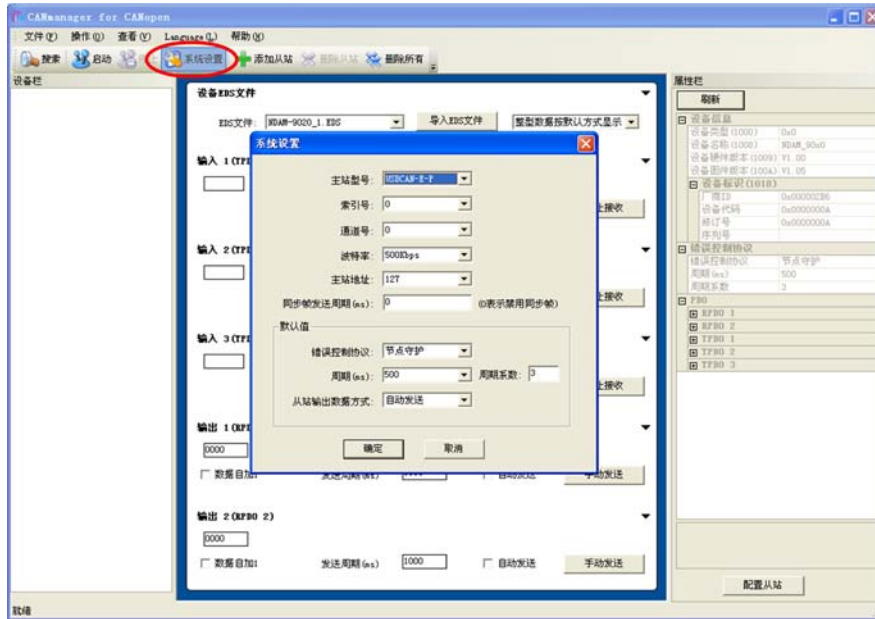


图 4.10 CANopen 主站卡设置

然后点击“搜索”按钮，搜索 CANopen 从站设备，如下图 4.11 所示。

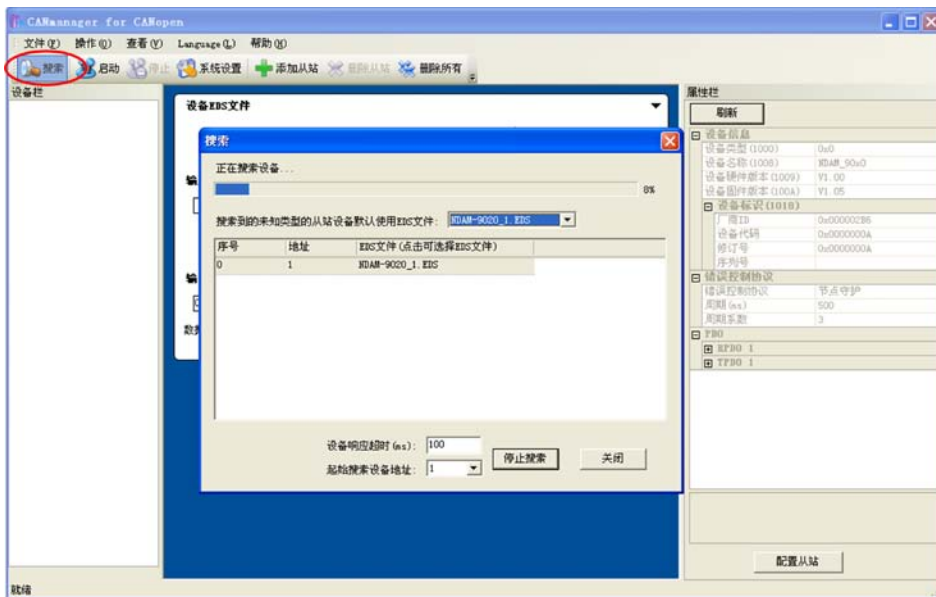


图 4.11 搜索 CANopen 从站

搜索到从站后点击“启动”按钮在弹出的系统设置中设置好参数后点击“确定”按钮，模块进入操作状态，如下图 4.12 所示。下面输入的 TPDO1, TPDO2 的显示框内显示的为 NDAM-3402 的 AI 通道数据，显示值会随 NDAM-3402 的采样数据变化而变化。TPDO3 的显示框显示的为 NDAM-3402 的 DI 通道的输入值。RPDO1 下的输入框为 NDAM-3402 的 AO 通道输出值，修改数字点击“手动发送”按钮，NDAM-3402 的 AO 输出值会相应改变。

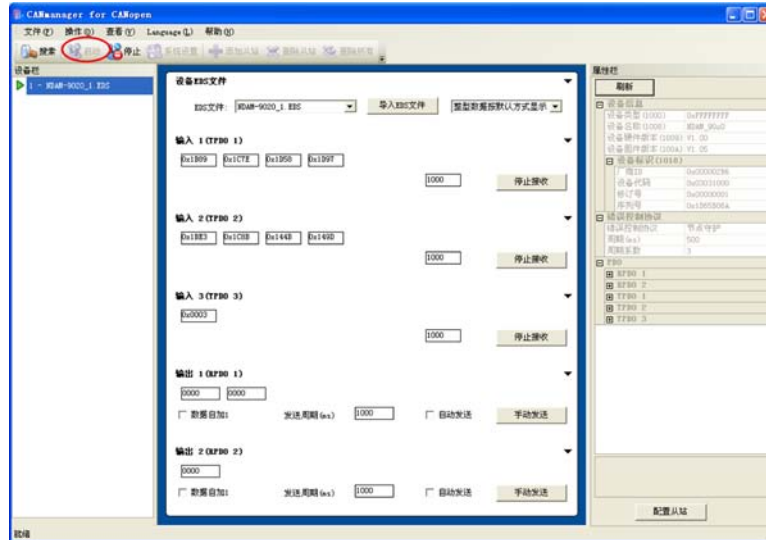


图 4.12 CANopen 从站操作

## 4.6 普通 CAN 卡操作 NDA M-9020

在简单的使用中，如果用户没有现成的 CANopen 主站卡，可以使用普通的 CAN 卡通过发送 CAN 帧来实现模块的操作。

下面以 NDA M-9020 带 NDA M-3402 数据采集模块为例介绍使用过程。其中 NDA M-9020 的节点地址设置为 0x01，波特率为 500K。

### 4.6.1 配置 PDO 映射关系

如上面 4.4 章节介绍，配置 NDA M-3402 的 AI 采样通道映射到 TPDO 中，配置 NDA M-3402 的 AO 输出通道到 RPDO 中。最终配置如下图 4.13 所示。点击“下载映射”将映射关系下载到 NDA M-9020。不使用 CANopen 主站卡可以不必生成 EDS 文件。重启模块映射关系生效。

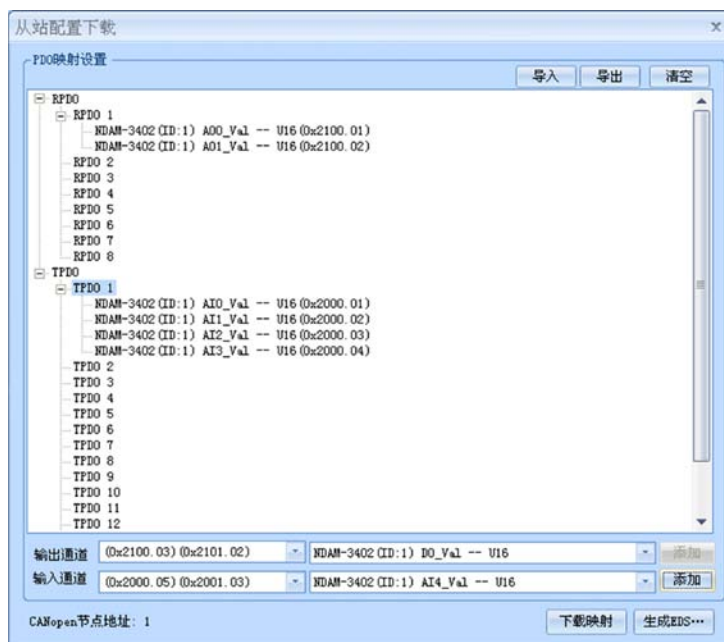


图 4.13 NDA M-3402 通道映射

#### 4.6.2 通过 CAN 卡启动从站

连接好 CAN 线启动 CAN 卡。给 NDAM-9020 上电，NDAM-9020 初始化完成后发送如下图 4.14 所示的启动报文。启动报文具体格式可参见 3.1.24 所示。

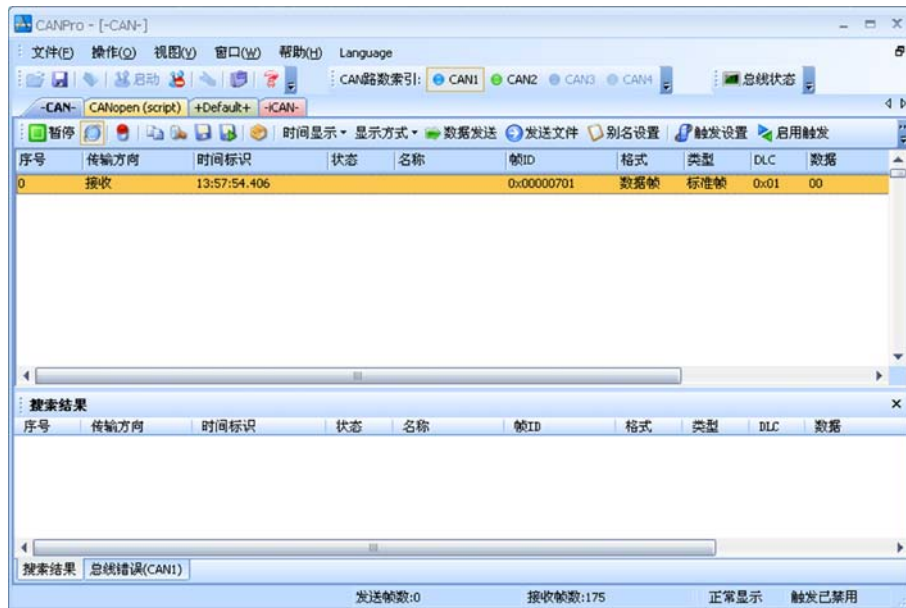


图 4.14 NDAM-9020 启动报文

NDAM-9020 发送完启动报文后，即可以进行操作。如果不需要进行 PDO 的重映射以及 COBID 的重分配。可以采用预定义连接集进行通讯。首先需要启动 NDAM-9020。通过 CAN 卡发送启动命令帧如下图所示。启动从站报文具体格式参见 3.1.21。

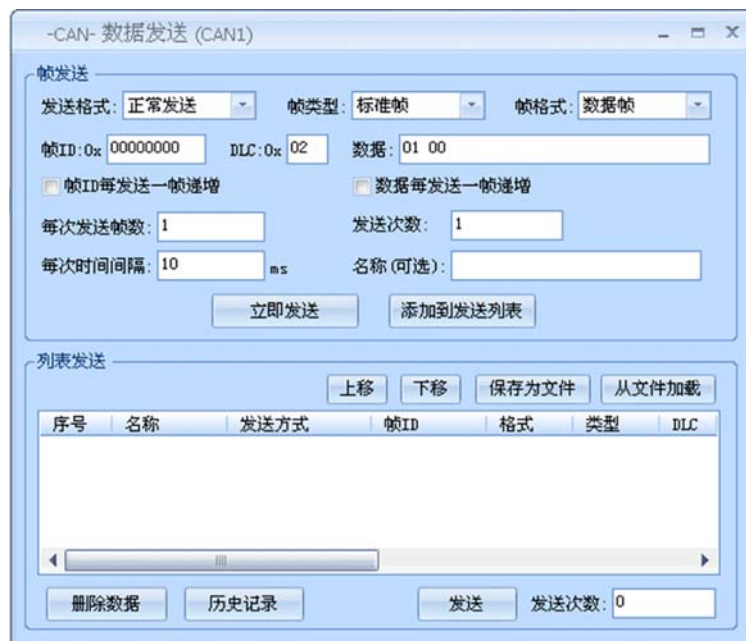


图 4.15 启动 NDAM-9020

### 4.6.3 设置 TPDO 传输参数

发送启动 NDAM-9020 的命令后，NDAM-9020 会根据传输参数上传 TPDO 数据。NDAM-9020 出厂时 TPDO 的默认传输特性值为 254，约束时间为 10ms，不使能时间触发上传功能。只有在 TPDO 数据变化后模块才会自动上传数据，在有些输入比较稳定的情况下可能数据上传很慢。用户可以通过设置定时上传功能实现 TPDO 的定时上传。所有设置都是通过 SDO 写相应对象字典来实现。

TPDO1 的定时上传时间设置值在对象字典索引 1800 子索引 05 处。如下图 4.16 为设置定时上传时间为 50ms。发送 CAN 帧如下图 4.16 所示。

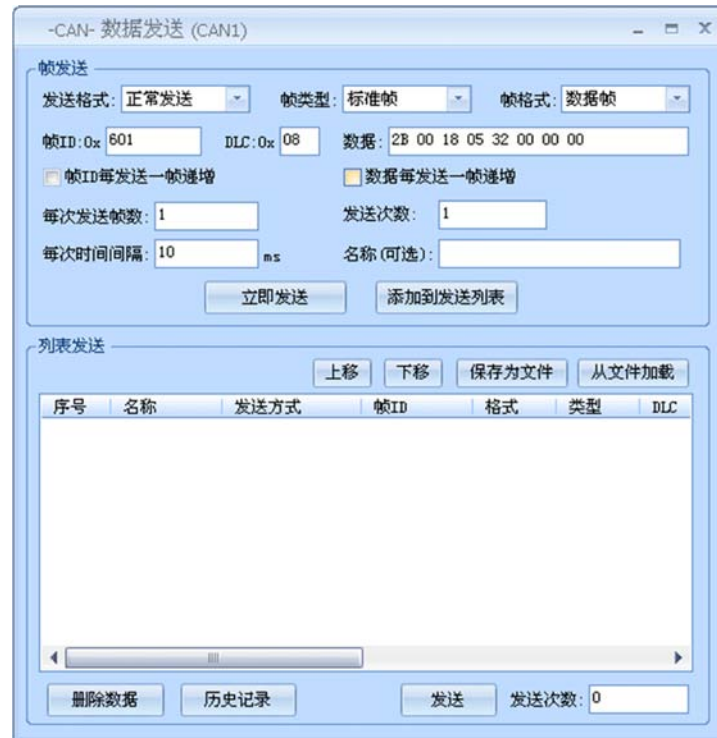


图 4.16 设置 TPDO 触发时间

CAN 格式组成如下：

CAN 帧 ID: 0x600+NodeID

DLC: 0x08

帧数据: 2B 00 18 05 32 00 00 00

帧 ID 根据 NodeID 的不同稍作修改，帧数据中 Byte1: 0x2B 为 SDO 命令字具体可参见 3.1.5 中的介绍，在设置约束时间时其值固定不变。Byte2-Byte3 为操作的对象字典索引值，为小端格式低字节在前，00 18 表示操作的对象索引为 1800，Byte4 为操作对象字典的子索引为 05，Byte5-Byte6 为设置的时间，也是低字节在前，本次写入的数据为 0x0032，表示自动上传时间为 50ms。Byte7-Byte8 为填充值，无意义一般为 0。

设置完定时上传时间 NDAM-9020 会自动上传 TPDO 数据给主机。如下图 4.17 所示，TPDO1 每隔 50ms 自动上传一次。

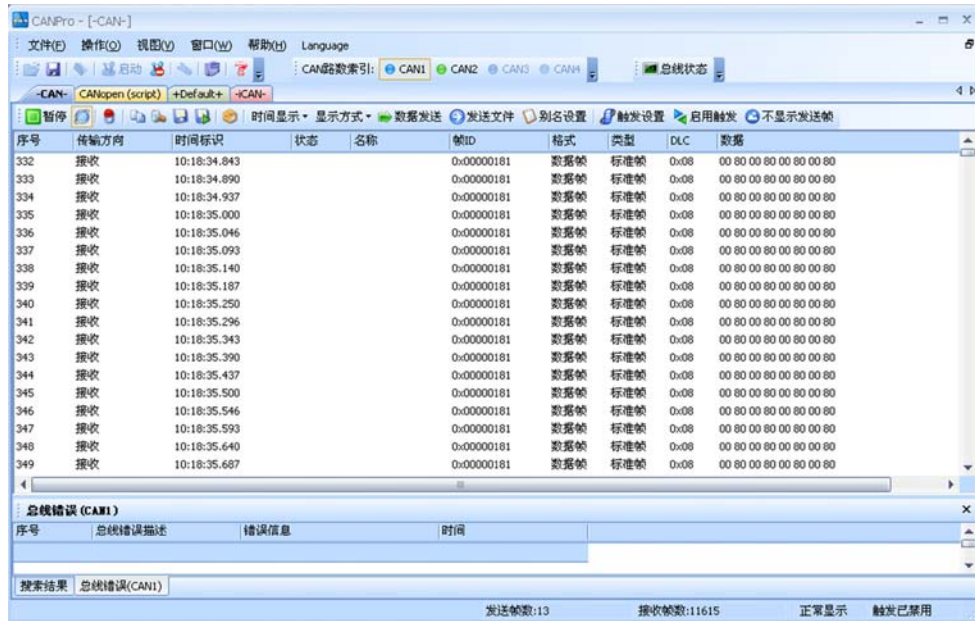


图 4.17 TPDO 定时上传

对于其它的 TPDO 设置定时上传参数时，主要需要对上面介绍的 CAN 帧修改一下对象字典索引和子索引值。各 TPDO 定时上传参数的对象字典分配如下表 4.1 所示。

表 4.1 TPDO 定时上传参数对象字典分配

TPDO 条目	对象字典
TPDO1	1800.05
TPDO2	1802.05
.	.
.	.
TPDO15	180E.05
TPDO16	180F.05

#### 4.6.4 设置 RPDO 数据

设置 RPDO 的值只需要按照预定义连接集里面的相应 RPDO 的 COBID 发送数据即可，设置 RPDO1 的 CAN 帧如下图 4.18 所示，发送的 CAN 帧格式如下：

帧 ID: 0x200+NodeID

DLC: 0x04

帧数据: 00 01 00 02

其中帧 ID 即为 RPDO 的 COBID，DLC 为帧数据长度，根据设置值的个数修改。帧数据为设置的 RPDO 输出值，低字节在前。



图 4.18 设置 RPDO1 数据

### 附录 1 NDAM-9020 通用对象字典

索引 (Index)	子索引 (Subindex)	名称 (Name)	类型(Type)	属性 (Attr.)	默认值 (Def.)	描述 (Desc.)
<b>通信参数区</b>						
0x1000	-	Device Type	UINT32	RO	0x00000000	NDAM-9020 内部 CANopen 协议栈 设备 XGate 类型
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	-
0x1007		Sync Windows Length	UINT32	RW	0	-
0x1008		XGate name	STRING	Const	XGate-COP10	NDAM-9020 内部 CANopen 协议栈 设备 XGate 设备名称
0x1009		XGate hardware version	STRING	Const	V1.01	NDAM-9020 内部 CANopen 协议栈 设备 XGate 硬件版本
0x100A		XGate software version	STRING	Const	V1.00	NDAM-9020 内部 CANopen 协议栈 设备 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	广州致远电子有限公司在 CiA 组织的厂商代码(可更改)
	2	Product code	UINT32	RO	-	NDAM-9020 内部 CANopen 协议栈设备 XGate 产品代码
	3	Revision number	UINT32	RO	-	NDAM-9020 内部 CANopen 协议栈设备 XGate 修订码
	4	Serial number	UINT32	RO	-	NDAM-9020 内部 CANopen 协议栈设备 XGate 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	0x80000000	
	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	0x80000000	



	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	0x80000000	
	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	0x80000000	
	2	transmission type	UINT8	RW	0xFE	-
RPDO 映射参数						
0x1600 ~ 0x1607	0	number of mapped objects	UINT8	RO	动态生成	RPDO 映射参数数量
	1	PDO mapping 1. app. object	UINT32	RW	动态生成	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+0x480	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	0x80000000	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	0x80000000	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	0x80000000	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	0x80000000	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	0x80000000	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	0x80000000	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	0x80000000	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 supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	0x80000000	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
0x180C	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	0x80000000	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
0x180D	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	0x80000000	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
0x180E	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	0x80000000	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
0x180F	0	largest subindex supported	UINT8	RO	0x05	-
	1	COB-ID used	UINT32	RW	0x80000000	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 ~ 0x1A0F	0	number of mapped objects	UINT8	RO	动态生成	TPDO 映射参数数量
	1~N	PDO mapping 1. app. object	UINT32	RW	动态生成	映射参数
数据输入区 (TPDO 发送数据映射区)						
0x2000	0	Number Of Entries	UINT8	RO	64	输入数据(2 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
	3	Application data Input 16bit #2	UINT16	RW	-	编号为#2, 长度为 2 字节
	4~64	#3~#63	UINT16	RW	-	编号为#n, 长度为 2 字节
0x2001	0	Number Of Entries	UINT8	RO	32	输入数据(4 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
	3	Application data Input 32bit #2	UINT32	RW	-	编号为#2, 长度为 4 字节
	4~32	#3~#23	UINT32	RW	-	编号为#n, 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	32	输出数据(2 字节)区的长度
	1	Application data Output 16bit #0	UINT16	RW	-	编号为#0, 长度为 2 字节
	2	Application data Output 16bit #1	UINT16	RW	-	编号为#1, 长度为 2 字节

	3	Application data Output 16bit #2	UINT16	RW	-	编号为#2, 长度为 2 字节
	4~32	#3~#47	UINT16	RW	-	编号为#n, 长度为 4 字节
0x2101	0	Number Of Entries	UINT8	RO	16	输出数据(4 字节)区的长度
	1	Application data Output 16bit #0	UINT32	RW	-	编号为#0, 长度为 4 字节
	2	Application data Output 32bit #1	UINT32	RW	-	编号为#1, 长度为 4 字节
	3	Application data Output 32bit #2	UINT32	RW	-	编号为#2, 长度为 4 字节
	4~16	#3~#23	UINT32	RW	-	编号为#n, 长度为 4 字节
<b>设备状态</b>						
0x2400		Actual NodeId	UINT8		-	模块当前的 NodeID
0x2401		Actual BaudRate	UINT8		-	模块当前的波特率索引值
0x2402		ModleStatus	UINT8		-	模块当前的模块状态
0x2403	0	Entries Number	UINT8	CONST	3	-
	1	Set NodeId	UINT8	RO	-	用户通过 uart 口设置的 NodeID, 可能与当前 NodeID 不同
	2	Set Index of Baudrate	UINT8	RO	-	用户通过 uart 口设置的波特率索引值, 可能与当前波特率索引不同
	3	Set Baudrate Value	UINT32	RO	-	用户通过 uart 口设置的 CAN 定时参数, 当该值有效, 模块就会使用该值初始化 CAN 控制器
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 Num..	UINT32	RO	-	用设备序列号
	6	Device Name	STRING	RO	-	用户设备名称
0x2405	0	UartComm. Baudrate	UINT8	RO	0x07	通信串口波特率

## 附录 2 RPDO 回读区对象字典

RPDO 回读数据都是以 16bit 为单位的，对于通道数据超过 16bit 的用户需要根据自己的实际使用情况进行整合。如 RPDO1 的第一个数据长度 32bit。那么实际的长度需要将 RPDO1\_0 和 RPDO1\_1 两个 16bit 数据组合成一个 32bit 的数据作为实际的通道数据。

索引	子索引	说明	值
30F0	0x00	子索引数目	0x04
	0x01	RPDO1_0 缓冲区数据	
	0x02	RPDO1_1 缓冲区数据	
	0x03	RPDO1_2 缓冲区数据	
	0x04	RPDO1_3 缓冲区数据	
30F1	0x00	子索引数目	0x04
	0x01	RPDO2_0 缓冲区数据	
	0x02	RPDO2_1 缓冲区数据	
	0x03	RPDO2_2 缓冲区数据	
	0x04	RPDO2_3 缓冲区数据	
30F2	0x00	子索引数目	0x04
	0x01	RPDO3_0 缓冲区数据	
	0x02	RPDO3_1 缓冲区数据	
	0x03	RPDO3_2 缓冲区数据	
	0x04	RPDO3_3 缓冲区数据	
30F3	0x00	子索引数目	0x04
	0x01	RPDO4_0 缓冲区数据	
	0x02	RPDO4_1 缓冲区数据	
	0x03	RPDO4_2 缓冲区数据	
	0x04	RPDO4_3 缓冲区数据	
30F4	0x00	子索引数目	0x04
	0x01	RPDO5_0 缓冲区数据	
	0x02	RPDO5_1 缓冲区数据	
	0x03	RPDO5_2 缓冲区数据	
	0x04	RPDO5_3 缓冲区数据	

30F5	0x00	子索引数目	0x04
	0x01	RPDO6_0 缓冲区数据	
	0x02	RPDO6_1 缓冲区数据	
	0x03	RPDO6_2 缓冲区数据	
	0x04	RPDO6_3 缓冲区数据	
30F6	0x00	子索引数目	0x04
	0x01	RPDO7_0 缓冲区数据	
	0x02	RPDO7_1 缓冲区数据	
	0x03	RPDO7_2 缓冲区数据	
	0x04	RPDO7_3 缓冲区数据	
30F7	0x00	子索引数目	0x04
	0x01	RPDO8_0 缓冲区数据	
	0x02	RPDO8_1 缓冲区数据	
	0x03	RPDO8_2 缓冲区数据	
	0x04	RPDO8_3 缓冲区数据	

### 附录 3 NDAM 模块通道定义

模块型号	通道类型	配置名称	备注
NDAM-1804	输入	NDAM-1804(ID:n)DI_Val—U16	数字量输入通道输入值
	输入	NDAM-1804(ID:n)DI0_Counter Val—U32	数字量输入通道 0 计数值
		NDAM-1804(ID:n)DI11_Counter Val—U32	数字量输入通道 11 计数值
NDAM-2808	输出	NDAM-2808(ID:n) DO_Val—U16	继电器输出通道
NDAM-4055	输入	NDAM-4055(ID:n) DI_Val—U16	数字量输入通道输入值
	输入	NDAM-4055(ID:n)DI0_Counter Val—U32	数字量输入通道 0 计数值
		NDAM-4055(ID:n)DI7_Counter Val—U32	数字量输入通道 7 计数值
输出	NDAM-4055(ID:n) DO_Val—U16	数字量输出通道输出值	
NDAM-3402	输入	NDAM-3402(ID:n) DI_Val—U16	数字量输入通道输入值
	输出	NDAM-3402(ID:n) DO_Val—U16	数字量输出通道输出值
	输入	NDAM-3402(ID:n) AI0_Val—U16	模拟量输入通道 0 采样值
		NDAM-3402(ID:n) AI7_Val—U16	模拟量输入通道 7 采样值
输出	NDAM-3402(ID:n) AO0_Val—U16 NDAM-3402(ID:n) AO1_Val—U16	模拟量输出通道 0 输出值 模拟量输出通道 1 输出值	
NDAM-3412	输入	NDAM-3412(ID:n) DI_Val—U16	数字量输入通道输入值
	输入	NDAM-3412(ID:n) DI_Counter Val—U32	数字量输入通道计数值
	输出	NDAM-3412(ID:n) DO_Val—U16	数字量输出通道输出值
	输入	NDAM-3412(ID:n) AI0_Val—U16	模拟量输入通道 0 采样值
		NDAM-3412(ID:n) AI3_Val—U16	模拟量输入通道 3 采样值
输出	NDAM-3412(ID:n) AO_Val—U16	模拟量输出通道输出值	
NDAM-3800	输入	NDAM-3800(ID:n) AI0_Val—U16 NDAM-3800(ID:n) AI7_Val—U16	模拟量输入通道 0 采样值 模拟量输入通道 7 采样值
NDAM-4017	输入	NDAM-4017(ID:n) AI0_Val—U16 NDAM-4017(ID:n) AI7_Val—U16	模拟量输入通道 0 采样值 模拟量输入通道 7 采样值



NDAM-4400	输出	NDAM-4400(ID:n) AO0_Val—U16 .	模拟量输出通道 0 输出值 .
		NDAM-4400(ID:n) AO3_Val—U16	模拟量输出通道 3 输出值
NDAM-4410	输出	NDAM-4400(ID:n) AO0_Val—U16 .	模拟量输出通道 0 输出值 .
		NDAM-4400(ID:n) AO3_Val—U16	模拟量输出通道 3 输出值
NDAM-5508	输出	NDAM-5508(ID:n) DO_Val—U16	数字量输出通道输出值
	输入	NDAM-5508(ID:n) AI0_Val—U16 .	模拟量输入通道 0 采样值 .
		NDAM-5508(ID:n) AI4_Val—U16	模拟量输入通道 4 采样值
NDAM-6608	输出	NDAM-6608(ID:n) DO_Val—U16	数字量输出通道输出值
	输入	NDAM-6608(ID:n) AI0_Val—U16 .	模拟量输入通道 0 采样值 .
		NDAM-6608(ID:n) AI5_Val—U16	模拟量输入通道 5 采样值
NDAM-7404	输入	NDAM-7404(ID:n) DI_Val—U16	数字量输入通道输入值
	输入	NDAM-7404 (ID:n) CF0 CaptureVal—U32 .	CF0 通道捕获值 .
		NDAM-7404 (ID:n) CF3 CaptureVal—U32	CF3 通道捕获值
	输入	NDAM-7404 (ID:n) CF0 SampVal—U32 .	CF0 通道采样值 .
		NDAM-7404 (ID:n) CF3 SampVal—U32	CF3 通道采样值
输出	NDAM-7404(ID:n) DO_Val—U16	数字量输出通道输出值	

## 附录 4 NDAM 模块配置区对象字典

模块使用过程中如果不需要修改配置参数则用户可以通过 USB 配置好参数即可。如果用户需要在使用过程中通过 CANopen 主机修改配置参数。则要通过 SDO 操作相应的对象字典。各模块配置参数对象字典分配如下：

### NDAM-1804 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明
0x3000 (ID=1)	0x00	RO	子索引数目	0x01
	0x01	RW	DI 通道计数使能配置	一个 bit 对应一个通道 bit 为 0 对应通道计数禁止 1 则使能
0x3001 (ID=2)	同上	同上	同上	同上
0x3002 (ID=3)	同上	同上	同上	同上
0x3003 (ID=4)	同上	同上	同上	同上
0x3004 (ID=5)	同上	同上	同上	同上
0x3005 (ID=6)	同上	同上	同上	同上
0x3006 (ID=7)	同上	同上	同上	同上
0x3007 (ID=8)	同上	同上	同上	同上

### NDAM-2808 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明
0x3008 (ID=1)	0x00	RO	子索引数目	0x02
	0x01	RW	DO 通道使能配置	一个 bit 对应一个通道 bit 为 0 对应通道关闭 1 则使能
	0x02	RW	DO 通道安全值配置	一个 bit 对应一个通道的安全值
0x3009 (ID=2)	同上	同上	同上	同上

0x300A (ID=3)	同上	同上	同上	同上
0x300B (ID=4)	同上	同上	同上	同上
0x300C (ID=5)	同上	同上	同上	同上
0x300D (ID=6)	同上	同上	同上	同上
0x300E (ID=7)	同上	同上	同上	同上
0x300F (ID=8)	同上	同上	同上	同上

## NDAM-4055 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明
0x3010 (ID=1)	0x00	RO	子索引数目	0x03
	0x01	RW	DO 通道使能配置	一个 bit 对应一个通道 bit 为 0 对应通道关闭 1 则使能
	0x02	RW	DO 通道安全值配置	一个 bit 对应一个通道的安全值
	0x03	RW	DI 计数使能配置	一个 bit 对应一个通道 bit 为 0 对应通道计数禁止 1 则使能
0x3011 (ID=2)	同上	同上	同上	同上
0x3012 (ID=3)	同上	同上	同上	同上
0x3013 (ID=4)	同上	同上	同上	同上
0x3014 (ID=5)	同上	同上	同上	同上
0x3015 (ID=6)	同上	同上	同上	同上
0x3016 (ID=7)	同上	同上	同上	同上

0x3017 (ID=8)	同上	同上	同上	同上
------------------	----	----	----	----

## NDAM-3402 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明
0x3018 (ID=1)	0x00	RO	子索引数目	0x0E
	0x01	RW	DO 通道使能配置	一个 bit 对应一个通道 bit 为 0 对应通道关闭 1 则使能
	0x02	RW	DO 通道安全值配置	一个 bit 对应一个通道的安全值
	0x03	RW	AI 通道输入模式	0: 4 路差分输入 1: 8 路单端输入
	0x04 ~ 0x0B	RW	AI0-AI7 通道输入量程	0: 电压输入 -10V~+10V 1: 电流输入 0~+20mA 2: 通道关闭
	0x0C	RW	AO 通道输出量程	0: 0~+10V 1: 0~+20mA 2: 4~20mA
	0x0D	RW	AO0 通道安全值	
	0x0E	RW	AO1 通道安全值	
0x3019 (ID=2)	同上	同上	同上	同上
0x301A (ID=3)	同上	同上	同上	同上
0x301B (ID=4)	同上	同上	同上	同上
0x301C (ID=5)	同上	同上	同上	同上
0x301D (ID=6)	同上	同上	同上	同上
0x301E (ID=7)	同上	同上	同上	同上
0x301F (ID=8)	同上	同上	同上	同上

NDAM-3412 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明
S0x3020 (ID=1)	0x00	RO	子索引数目	0x11
	0x01 ~ 0x04	RW	AI0~AI3 通道量程配置	0: 15mV      1: 50mV 2: 100mV     3: 500mV 4: 1000mV    5: 2500mV 6: ±20mA
	0x05	RW	AI0 上限值	
	0x06	RW	AI0 下限值	
	0x07	RW	AI1 上限值	
	0x08	RW	AI1 下限值	
	0x09	RW	AI2 上限值	
	0x0A	RW	AI2 下限值	
	0x0B	RW	AI3 上限值	
	0x0C	RW	AI3 下限值	
	0x0D	RW	DO 输出模式	0: 状态指示模式 1: 用户控制模式
	0x0E	RW	DO 安全值	一个 bit 对应一个通道的安全值
	0x0F	RW	DI 工作模式	0: 普通 DI 模式 1: 事件计数模式
	0x10	RW	DI0 事件触发方式	0: 上升沿递增 1: 下降沿递增 2: 上升沿和下降沿递增
	0x11	RW	AO 输出模式	0: 电压输出模式 1: 电流输出模式
	0x3021 (ID=2)	同上	同上	同上
0x3022 (ID=3)	同上	同上	同上	同上
0x3023 (ID=4)	同上	同上	同上	同上
0x3024	同上	同上	同上	同上

(ID=5)				
0x3025 (ID=6)	同上	同上	同上	同上
0x3026 (ID=7)	同上	同上	同上	同上
0x3027 (ID=8)	同上	同上	同上	同上

**NDAM-3800 配置区对象字典**

索引	子索引	操作属性	配置项名称	配置值说明
0x3028 (ID=1)	0x00	RO	子索引数目	0x08
	0x01 ~ 0x08	RW	AI0~AI7 输入量程配置	0: ±5V      1: ±10V 2: 0~20mA    3:0~40mA 4: 通道关闭
0x3029 (ID=2)	同上	同上	同上	同上
0x302A (ID=3)	同上	同上	同上	同上
0x302B (ID=4)	同上	同上	同上	同上
0x302C (ID=5)	同上	同上	同上	同上
0x302D (ID=6)	同上	同上	同上	同上
0x302E (ID=7)	同上	同上	同上	同上
0x302F (ID=8)	同上	同上	同上	同上

**NDAM-4017 配置区对象字典**

索引	子索引	操作属性	配置项名称	配置值说明
	0x00	RO	子索引数目	0x0B
	0x01	RW	AI0~AI7 输入量程配置	0: ±10V      1: ±5V

0x3030 (ID=1)	~ 0x08			2: $\pm 2.5V$ 4: $\pm 500mV$ 6: $\pm 150mV$ 17: 0~20mA 15: 通道关闭
	0x09	RW	AI 采样频率配置	0: 75Hz      1: 50Hz 2: 25Hz      3: 10Hz 4: 5Hz      5: 2.5Hz
	0x0A	WO	AI 校准开始	1: 开始校准      0: 无效
	0x0B	RO	AI 校准结束标志查询	0: 校准未结束      1: 校准结束
0x3031 (ID=2)	同上	同上	同上	同上
0x3032 (ID=3)	同上	同上	同上	同上
0x3033 (ID=4)	同上	同上	同上	同上
0x3034 (ID=5)	同上	同上	同上	同上
0x3035 (ID=6)	同上	同上	同上	同上
0x3036 (ID=7)	同上	同上	同上	同上
0x3037 (ID=8)	同上	同上	同上	同上

## NDAM-4400 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明
0x3038 (ID=1)	0x00	RO	子索引数目	0x09
	0x01	RW	AO0 安全值配置	
	0x02	RW	AO0 输出类型配置	0: 0~10V      1: 4~20mA 2: 0~20mA
	0x03	RW	AO1 安全值配置	1: 开始校准      0: 无效
	0x04	RW	AO1 输出类型配置	0: 0~10V      1: 4~20mA 2: 0~20mA
	0x05	RW	AO2 安全值配置	

	0x06	RW	AO2 输出类型配置	0: 0~10V 2: 0~20mA	1: 4~20mA
	0x07	RW	AO3 安全值配置		
	0x08	RW	AO3 输出类型配置	0: 0~10V 2: 0~20mA	1: 4~20mA
	0x09	RW	AO 同步输出配置	0: 同步输出禁止 1: 同步输出使能	
0x3039 (ID=2)	同上	同上	同上		同上
0x303A (ID=3)	同上	同上	同上		同上
0x303B (ID=4)	同上	同上	同上		同上
0x303C (ID=5)	同上	同上	同上		同上
0x303D (ID=6)	同上	同上	同上		同上
0x303E (ID=7)	同上	同上	同上		同上
0x303F (ID=8)	同上	同上	同上		同上

## NDAM-4410 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明	
0x3040 (ID=1)	0x00	RO	子索引数目	0x0B	
	0x01	RW	AO 输出类型	0: 立即输出 1: 同步输出	
	0x02 ~ 0x05	RW	AO0~AO3 量程配置	0: 0~5V 2: ±5V 4: 0~20mA(灌) 6: 0~20mA(拉)	1: 0~10V 3: ±10V 5: 4~20mA(灌) 7: 4~20mA(拉)
	0x06 ~ 0x09	RW	AO0~AO3 安全值配置		



	0x0A	RW	AO 通道使能	每 bit 对应一个通道 bit 为 0 通道禁止 bit 为 1 通道使能
	0x0B	RW	AO 同步输出触发	0: 无效 1: 触发 AO 同步输出
0x3041 (ID=2)	同上	同上	同上	同上
0x3042 (ID=3)	同上	同上	同上	同上
0x3043 (ID=4)	同上	同上	同上	同上
0x3044 (ID=5)	同上	同上	同上	同上
0x3045 (ID=6)	同上	同上	同上	同上
0x3046 (ID=7)	同上	同上	同上	同上
0x3047 (ID=8)	同上	同上	同上	同上

**NDAM-5508 配置区对象字典**

索引	子索引	操作属性	配置项名称	配置值说明
0x3048 (ID=1)	0x00	RO	子索引数目	0x12
	0x01	RW	AI0 输入类型	0: 关闭通道      20: Pt10 热电阻 21: Pt100 热电阻    22: Pt200 热电阻 23: Pt500 热电阻    24: Pt1000 热电阻 25: Cu50 热电阻      26: Cu100 热电阻
	0x02	RW	AI0 输入上限值	
	0x03	RW	AI0 输入下限值	
	0x04	RW	AI1 输入类型	0: 关闭通道      20: Pt10 热电阻 21: Pt100 热电阻    22: Pt200 热电阻 23: Pt500 热电阻    24: Pt1000 热电阻 25: Cu50 热电阻      26: Cu100 热电阻
	0x05	RW	AI1 输入上限值	

	0x06	RW	AI1 输入下限值	
	0x07	RW	AI2 输入类型	0: 关闭通道      20: Pt10 热电阻 21: Pt100 热电阻      22: Pt200 热电阻 23: Pt500 热电阻      24: Pt1000 热电阻 25: Cu50 热电阻      26: Cu100 热电阻
	0x08	RW	AI2 输入上限值	
	0x09	RW	AI2 输入下限值	
	0x0A	RW	AI3 输入类型	0: 关闭通道      20: Pt10 热电阻 21: Pt100 热电阻      22: Pt200 热电阻 23: Pt500 热电阻      24: Pt1000 热电阻 25: Cu50 热电阻      26: Cu100 热电阻
	0x0B	RW	AI3 输入上限值	
	0x0C	RW	AI3 输入下限值	
	0x0D	RW	AI4 输入类型	0: 关闭通道      20: Pt10 热电阻 21: Pt100 热电阻      22: Pt200 热电阻 23: Pt500 热电阻      24: Pt1000 热电阻 25: Cu50 热电阻      26: Cu100 热电阻
	0x0E	RW	AI4 输入上限值	
	0x0F	RW	AI4 输入下限值	
	0x10	RW	DO 输出模式配置	0: 模拟输入指示      1: 用户控制
	0x11	RW	DO 输出安全值	
	0x12	RW	校准配置	0: 通道 0 校准      1: 通道 1 校准 2: 通道 2 校准      3: 通道 3 校准 4: 通道 4 校准      5: 5 通道同时校准
0x3049 (ID=2)	同上	同上	同上	同上
0x304A (ID=3)	同上	同上	同上	同上
0x304B (ID=4)	同上	同上	同上	同上
0x304C (ID=5)	同上	同上	同上	同上

0x304D (ID=6)	同上	同上	同上	同上
0x304E (ID=7)	同上	同上	同上	同上
0x304F (ID=8)	同上	同上	同上	同上

## NDAM-6608 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明
0x3050 (ID=1)	0x00	RO	子索引数目	0x15
	0x01	RW	AI0 输入类型	0: 关闭通道      1: J 型热电偶 2: K 型热电偶    3: E 型热电偶 4: T 型热电偶    5: N 型热电偶 6: B 型热电偶    7: R 型热电偶 8: S 型热电偶    16: 电压/电流
	0x02	RW	AI0 输入上限值	
	0x03	RW	AI0 输入下限值	
	0x04	RW	AI1 输入类型	0: 关闭通道      1: J 型热电偶 2: K 型热电偶    3: E 型热电偶 4: T 型热电偶    5: N 型热电偶 6: B 型热电偶    7: R 型热电偶 8: S 型热电偶    16: 电压/电流
	0x05	RW	AI1 输入上限值	
	0x06	RW	AI1 输入下限值	
	0x07	RW	AI2 输入类型	0: 关闭通道      1: J 型热电偶 2: K 型热电偶    3: E 型热电偶 4: T 型热电偶    5: N 型热电偶 6: B 型热电偶    7: R 型热电偶 8: S 型热电偶    16: 电压/电流
	0x08	RW	AI2 输入上限值	
	0x09	RW	AI2 输入下限值	
	0x0A	RW	AI3 输入类型	0: 关闭通道      1: J 型热电偶 2: K 型热电偶    3: E 型热电偶 4: T 型热电偶    5: N 型热电偶

				6: B 型热电偶 8: S 型热电偶	7: R 型热电偶 16: 电压/电流
	0x0B	RW	AI3 输入上限值		
	0x0C	RW	AI3 输入下限值		
	0x0D	RW	AI4 输入类型	0: 关闭通道 2: K 型热电偶 4: T 型热电偶 6: B 型热电偶 8: S 型热电偶	1: J 型热电偶 3: E 型热电偶 5: N 型热电偶 7: R 型热电偶 16: 电压/电流
	0x0E	RW	AI4 输入上限值		
	0x0F	RW	AI4 输入下限值		
	0x10	RW	AI5 输入类型	0: 关闭通道 2: K 型热电偶 4: T 型热电偶 6: B 型热电偶 8: S 型热电偶	1: J 型热电偶 3: E 型热电偶 5: N 型热电偶 7: R 型热电偶 16: 电压/电流
	0x11	RW	AI5 输入上限值		
	0x12	RW	AI5 输入下限值		
	0x13	RW	DO 输出模式配置	0: 模拟输入指示	1: 用户控制
	0x14	RW	DO 输出安全值		
	0x15	RW	温度误差配置		
0x3051 (ID=2)	同上	同上	同上		同上
0x3052 (ID=3)	同上	同上	同上		同上
0x3053 (ID=4)	同上	同上	同上		同上
0x3054 (ID=5)	同上	同上	同上		同上
0x3055 (ID=6)	同上	同上	同上		同上
0x3056 (ID=7)	同上	同上	同上		同上

0x3057 (ID=8)	同上	同上	同上	同上
------------------	----	----	----	----

## NDAM-7404 配置区对象字典

索引	子索引	操作属性	配置项名称	配置值说明
0x3058 (ID=1)	0x00	RO	子索引数目	0x2E
	0x01 ~ 0x04	RW	CF0~CF3 工作模式	0: 技术模式 1: 捕获模式 2: 测频模式
	0x05 ~ 0x08	RW	CF0~CF3 门控信号工作模式	0: 普通 DI 模式 0 1: 普通 DI 模式 1 2: 高电平模式 3: 低电平模式
	0x09	RW	CF 通道复位使能	0: 禁止复位 1: 使能复位
	0x0A	RW	CF0 高电平门限配置	以 0.1V 为单位 写入 25 则门限电平为 2.5V
	0x0B	RW	CF0 低电平门限配置	以 0.1V 为单位 写入 10 则门限电平为 1.0V
	0x0C	RW	CF1 高电平门限配置	以 0.1V 为单位 写入 25 则门限电平为 2.5V
	0x0D	RW	CF1 低电平门限配置	以 0.1V 为单位 写入 10 则门限电平为 1.0V
	0x0E ~ 0x11	RW	CF0~CF3 滤波时间配置	以 us 为单位 写入 20 则滤波时间为 20us
	0x12	RW	CF 计数方向配置	bit0 对应 CF0, bit1 对应 CF1 依次类推 0: 递增计数 1: 递减计数
	0x13	RW	通道测频时间配置	每 4 个 bit 对应一个 CF 通道, bit3-bit0 对应 CF0 通道, bit7-bit4 对应 CF1 通道依次类推 测频时间为(设置值+1)*100ms
	0x14 ~ 0x17	RW	DO0~DO3 工作模式配置	0: 普通 DO 模式 1: PWM 输出模式 2: 匹配(超限) 低电平输出模式 3: 匹配(超限) 高电平输出模式 4: 匹配(超限)输出翻转模式
	0x18	RW	DO 安全值配置	一个 bit 对应一个通道的安全值
	0x19	RW	编码器使能控制	bit0 对应编码器 0 bit1 对应编码器 1 0: 编码器禁止 1: 编码器使能
	0x1A	RW	编码器计数复位模式	bit0 对应编码器 0 bit1 对应编码器 1

				0: 最大值复位 1: 索引复位
0x1B	RW	编码器计数模式		bit0 对应编码器 0 bit1 对应编码器 1 0: A 边沿计数 1: AB 边沿计数
0x1C ~ 0x1D	RW	编码器 0/1 定时器配置		以 1ms 为单位 写入 50 则编码器定时值设置为 50ms
0x1E ~ 0x1F	RW	编码器 0/1 最大计数值配置		
0x20	RW	编码器 0 转速上限		
0x21	RW	编码器 0 转速下限		
0x22	RW	编码器 1 转速上限		
0x23	RW	编码器 1 转速下限		
0x24 ~ 0x25	RW	编码器 0/1 解析度配置		
0x26	RW	CF0 加载寄存器 0		
0x27	RW	CF0 加载寄存器 1		
0x28	RW	CF1 加载寄存器 0		
0x29	RW	CF1 加载寄存器 1		
0x2A	RW	CF2 加载寄存器 0		
0x2B	RW	CF2 加载寄存器 1		
0x2C	RW	CF3 加载寄存器 0		
0x2D	RW	CF3 加载寄存器 1		
0x2E	RW	计数清零寄存器		bit0 对应 CF0, bit1 对应 CF1 依次类推 写入 1 对应通道请 0 写入 0 无效
0x3059 (ID=2)	同上	同上	同上	同上
0x305A (ID=3)	同上	同上	同上	同上
0x305B (ID=4)	同上	同上	同上	同上
0x305C	同上	同上	同上	同上

(ID=5)				
0x305D (ID=6)	同上	同上	同上	同上
0x305E (ID=7)	同上	同上	同上	同上
0x305F (ID=8)	同上	同上	同上	同上

## 5. 免责声明

### *版权*

本手册所陈述的产品文本及相关软件版权均属广州致远电子股份有限公司所有，其产权受国家法律绝对保护，未经本公司授权，其它公司、单位、代理商及个人不得非法使用和拷贝，否则将受到国家法律的严厉制裁。

### *修改文档的权利*

广州致远电子股份有限公司保留任何时候在不事先声明的情况下对本手册的修改的权力。